6

Consideration of Partial User Preferences in Evolutionary Multiobjective Optimization

Jürgen Branke

Institute AIFB, University of Karlsruhe, 76128 Karlsruhe, Germany
branke@aifb.uni-karlsruhe.de

Abstract. Evolutionary multiobjective optimization usually attempts to find a good approximation to the complete Pareto optimal front. However, often the user has at least a vague idea about what kind of solutions might be preferred. If such information is available, it can be used to focus the search, yielding a more fine-grained approximation of the most relevant (from a user's perspective) areas of the Pareto optimal front and/or reducing computation time. This chapter surveys the literature on incorporating partial user preference information in evolutionary multiobjective optimization.

6.1 Introduction

Most research in evolutionary multiobjective optimization (EMO) attempts to approximate the complete Pareto optimal front by a set of well-distributed representatives of Pareto optimal solutions. The underlying reasoning is that in the absence of any preference information, all Pareto optimal solutions have to be considered equivalent.

On the other hand, in most practical applications, the decision maker (DM) is eventually interested in only a single solution. In order to come up with a single solution, at some point during the optimization process, the DM has to reveal his/her preferences to choose between mutually non-dominating solutions. Following a classification by Horn (1997) and Veldhuizen and Lamont (2000), the articulation of preferences may be done either before (a priori), during (progressive), or after (a posteriori) the optimization process, see also Figure 6.1.

A priori approaches aggregate different objectives into a single auxilliary objective in one way or another, which allows to use standard optimization techniques (including single-objective evolutionary algorithms) and usually

Reviewed by: Carlos Coello Coello, CINEVESTAV-IPN, Mexico
Salvatore Greco, University of Catania, Italy
Kalyanmoy Deb, Indian Institute of Technology Kanpur, India

J. Branke et al. (Eds.): Multiobjective Optimization, LNCS 5252, pp. 157–178, 2008.

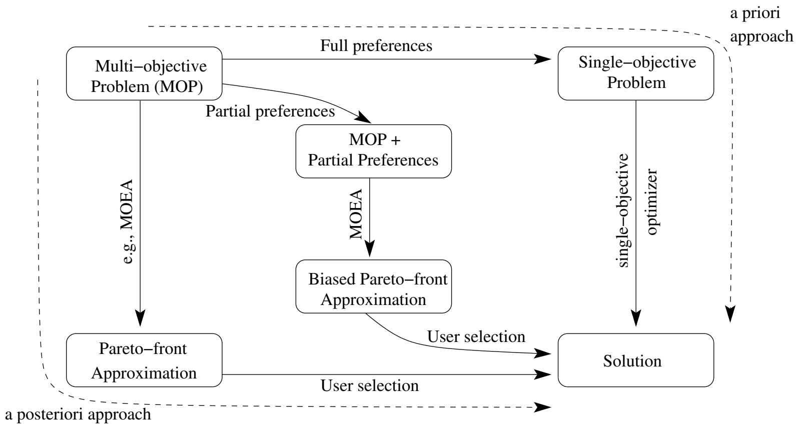

Fig. 6.1. Different ways to solve multiobjective problems.

results in a single solution. Many classical MCDM methodologies fall into this category. The most often used aggregation method is probably just a linear combination of the different objectives. Alternatives would be a lexicographic ordering of the objectives, or to use the distance from a specified target as objective. For an example of an approach based on fuzzy rules see Sait *et al.* (1999) or Sakawa and Yauchi (1999). As aggregation of objectives turn the multiobjective problem into a single objective problem, such evolutionary algorithms are actually out of scope of this chapter. A discussion of advantages and disadvantages of such aggregations can be found in Coello *et al.* (2002), Chapter 2.2. In any case, the aggregation of objectives into a single objective is usually not practical, because it basically requires to specify a ranking of alternatives before these alternatives are known. Classical MCDM techniques usually solve this predicament by repeatedly adjusting the auxilliary objective and re-solving the single objective problem until the DM is satisfied with the solution.

Most multiobjective evolutionary algorithms (MOEAs) can be classified as a posteriori. First, the EA generates a (potentially large) set of non-dominated solutions, then the DM can examine the possible trade-offs and choose according to his/her preferences. For an introduction to MOEAs, see Chapter 3. The most prominent MOEAs are the Non-Dominated Sorting Genetic Algorithm (NSGA-II, Deb *et al.*, 2002a) and the Strength-Pareto Evolutionary Algorithm (SPEA-II, Zitzler *et al.*, 2002).

Interactive approaches interleave the optimization with a progressive elicitation of user preferences. These approaches are discussed in detail in Chapter 7.

In the following, we consider an intermediate approach (middle path in Figure 6.1). Although we agree that it may be impractical for a DM to completely specify his or her preferences before any alternatives are known, we assume that the DM has at least a rough idea about what solutions might be preferred, and can specify partial preferences. The methods discussed here aim at integrating such imprecise knowledge into the EMO approach, biasing the search towards solutions that are considered as relevant by the DM. The goal is no longer to generate a good approximation to all Pareto optimal solutions, but a small set of solution that contains the DM's preferred solution with the highest probability. This may yield three important advantages:

1. **Focus:** Partial user preferences may be used to focus the search and generate a subset of all Pareto optimal alternatives that is particularly interesting to the DM. This avoids overwhelming the DM with a huge set of (mostly irrelevant) alternatives.
2. **Speed:** By focusing the search onto the relevant part of the search space, one may expect the optimization algorithm to find these solutions more quickly, not wasting computational effort to identify Pareto optimal but irrelevant solutions.
3. **Gradient:** MOEAs require some quality measure for solutions in order to identify the most promising search direction (gradient). The most important quality measure used in MOEA is Pareto dominance. However, with an increasing number of objectives, more and more solutions become incomparable, rendering Pareto dominance as fitness criterion less useful,resulting in a severe performance loss of MOEAs (e.g., Deb *et al.*, 2002b). Incorporating (partial) user preferences introduces additional preference relations, restoring the necessary fitness gradient information to some extend and ensuring MOEA's progress.

To reach these goals, the MOEA community can accomodate or be inspired by many of the classical MCDM methodologies covered in Chapters 1 and 2, as those generally integrate preference information into the optimization process. Thus, combining MOEAs, and their ability to generate multiple alternatives simultaneously in one run, and classical MCDM methodologies, and their ways to incorporate user preferences, holds great promise.

The literature contains quite a few techniques to incorporate full or partial preference information into MOEAs, and previous surveys on this topic include Coello (2000); Rachmawati and Srinivasan (2006), and Coello *et al.* (2002). In the following, we classify the different approaches based on the type of partial preference information they ask from the DM, namely objective scaling (Section 6.2), constraints (Section 6.3), a goal or reference point (Section 6.4), trade-off information (Section 6.5), or weighted performance measures (Section 6.6 on approaches based on marginal contribution). Some additional approaches are summarized in Section 6.7. The chapter concludes with a summary in Section 6.8.

6.2 Scaling

One of the often claimed advantages of MOEAs is that they do not require an a priori specification of user preferences because they generate a good approximation of the whole Pareto front, allowing the DM to pick his/her preferred solution afterwards. However, the whole Pareto optimal front may contain very many alternatives, in which case MOEAs can only hope to find a representative subset of all Pareto optimal solutions. Therefore, all basic EMO approaches attempt to generate a uniform distribution of representatives along the Pareto front. For this goal, they rely on distance information in the objective space, be it in the crowding distance of NSGA-II or in the clustering of SPEA-II[1]. Thus, what is considered uniform depends on the scaling of the objectives. This is illustrated in Figure 6.2. The left panel (a) shows an evenly distributed set of solutions along the Pareto front. Scaling the second objective by a factor of 100 (e.g., using centimeters instead of meters as unit), leads to a bias of the distribution and more solutions along the front parallel to the axis of the second objective (right panel). Note that depending on the shape of the front, this means that there is a bias towards objective 1 (as in the convex front in Figure 6.2), or objective 2 (if the front is concave). So, the user-defined scaling is actually a usually ignored form of user preference specification necessary also for MOEAs.

Many current implementations of MOEAs (e.g., NSGA-II and SPEA) scale objectives based on the solutions currently in the population (see, e.g., Deb (2001), S. 248). While this results in nice visualizations if the front is plotted with a 1:1 ratio, and relieves the DM from specifying a scaling, it assumes that ranges of values covered by the Pareto front in each objective are equally

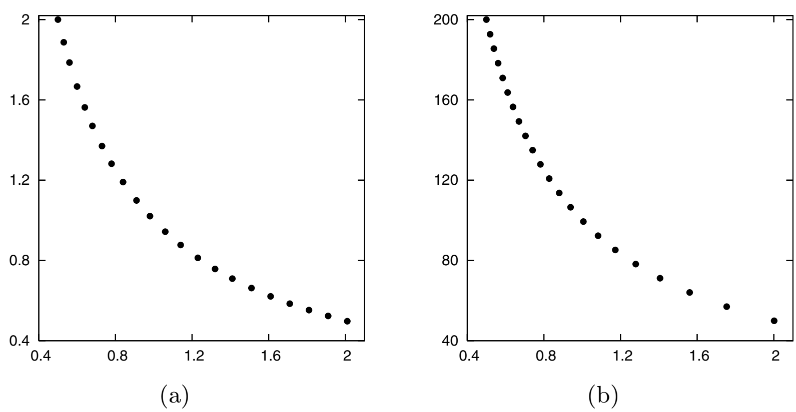

Fig. 6.2. Influence of scaling on the distribution of solutions along the Pareto front.

[1] The rest of this section assumes familiarity with the crowding distance concept. Readers unfamiliar with this concept are referred to, e.g., Deb (2001) or Chapter 3.

important. Whether this assumption is justified certainly depends strongly on the application and the DM's preferences.

In order to find a biased distribution anywhere on the Pareto optimal front, a previous study by Deb (2003) used a biased sharing[2] mechanism implemented on NSGA. In brief, the objectives are scaled according to preferences when calculating the distances. This allows to make distances in one objective appear larger than they are, with a corresponding change in the resulting distribution of individuals. Although this allows to focus on one objective or another, the approach does not allow to focus on a compromise region (for equal weighting of the objectives, the algorithm would produce no bias at all).

In Branke and Deb (2005), the biased sharing mechanism has been extended with a better control of the region of interest and a separate parameter controlling the strength of the bias. For a solution i on a particular front, the *biased crowding disctance measure* D_i is re-defined as follows. Let $\boldsymbol{\eta}$ be a user-specified direction vector indicating the most probable, or central linearly weighted utility function, and let α be a parameter controlling the bias intensity. Then,

$$D_i = d_i \left(\frac{d'_i}{d_i} \right)^{\alpha}, \tag{6.1}$$

where d_i and d'_i are the original crowding distance and the crowding distance calculated based on the locations of the individuals projected onto the (hyper)plane with direction vector $\boldsymbol{\eta}$. Figure 6.3 illustrates the concept.

As a result, for a solution in a region of the Pareto optimal front more or less parallel to the projected plane (such as solution 'a'), the original crowded distance d_a and projected crowding distance d'_a are more or less the same, thereby making the ratio d'_a/d_a close to one. On the other hand, for a solution in an area of the Pareto optimal front where the tangent has an orientation significantly different from the chosen plane (such as solution 'b'), the projected crowding distance d'_b is much smaller than the original crowding distance d_b. For such a solution, the biased crowding distance value D_i will be a small quantity, meaning that such a solution is assumed to be artificially crowded by neighboring solutions. A preference of solutions having a larger biased crowding distance D_i will then enable solutions closer to the tangent point to be found. The exponent α controls the extent of the bias, with larger α resulting in a stronger bias.

Note that biased crowding will focus on the area of the Pareto optimal front which is parallel to the iso-utility function defined by the provided direction vector η. For a *convex* Pareto optimal front, that is just the area around the optimal solution regarding a corresponding aggregate cost function. For a concave region, such an aggregate cost function would always prefer one of the edge points, while biased crowding may focus on the area in between.

[2] The sharing function in NSGA fulfills the same functionality as the crowding distance in NSGA-II, namely to ensure a diversity of solutions.

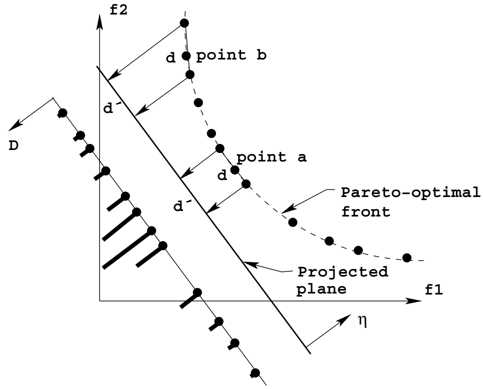

Fig. 6.3. The biased crowding approach is illustrated on a two-objective minimization problem (Branke and Deb, 2005).

Trautmann and Mehnen (2005) suggest an explicit incorporation of preferences into the scaling. They propose to map the objectives into the range $[0, 1]$ according to desirability functions. With one-sided sigmoid (monotone) desirability functions, the non-dominance relations are not changed. Therefore, the solutions found are always also non-dominated in the original objective space. What changes is the distribution along the front. Solutions that are in flat parts of the desirability function receive very similar desirability values and as MOEAs then attempt to spread solutions evenly in the desirability space, this will result in a more spread out distribution in the original objective space. However, in order to specify the desirability functions in a sensible manner, it is necessary to at least know the ranges of the Pareto front.

6.3 Constraints

Often, the DM can formulate preferences in the form of constraints, for example "Criterion 1 should be less than β". Handling constraints is a well-researched topic in evolutionary algorithms in general, and most of the techniques carry over to EMO in a straightforward manner. One of the simplest and most common techniques is probably to rank infeasible solutions according to their degree of infeasibility, and inferior to all feasible solutions (Deb, 2000; Jiménez and Verdegay, 1999). A detailed discussion of constraint handling techniques is out of the scope of this chapter. Instead, the interested reader is referred to Coello (2002) for a general survey on constraint handling techniques, and Deb (2001), Chapter 7, for a survey with focus on EMO techniques.

6.4 Providing a Reference Point

Perhaps the most important way to provide preference information is a reference point, a technique that has a long tradition in multicriteria decision making, see, e.g., Wierzbicki (1977, 1986) and also Chapter 2. A reference point consists of aspiration levels reflecting desirable values for the objective function, i.e., a target the user is hoping for. Such an information can then be used in different ways to focus the search. However, it should not lead to a dominated solution being preferred over the dominating solution.

The use of a reference point to guide the EMO algorithm has first been proposed in Fonseca and Fleming (1993). The basic idea there is to give a higher priority to objectives in which the goal is not fulfilled. Thus, when deciding whether a solution $\mathbf{x}$ is preferable to a solution $\mathbf{y}$ or not, first, only the objectives in which solution $\mathbf{x}$ does not satisfy the goal are considered, and $\mathbf{x}$ is preferred to $\mathbf{y}$ if it dominates $\mathbf{y}$ on these objectives. If $\mathbf{x}$ is equal to $\mathbf{y}$ in all these objectives, or if $\mathbf{x}$ satisfies the goal in all objectives, $\mathbf{x}$ is preferred over $\mathbf{y}$ either if $\mathbf{y}$ does not fulfill some of the objectives fulfilled by $\mathbf{x}$, or if $\mathbf{x}$ dominates $\mathbf{y}$ on the objectives fulfilled by $\mathbf{x}$. More formally, this can be stated as follows. Let $\mathbf{r}$ denote the reference point, and let there be m objectives without loss of generality sorted such that $\mathbf{x}$ fulfills objectives $k+1 \dots m$ but not objectives $1 \dots k$, i.e.

$$f_i(\mathbf{x}) > r_i \quad \forall i = 1 \dots k \tag{6.2}$$

$$f_i(\mathbf{x}) \leq r_i \quad \forall i = k+1 \dots m. \tag{6.3}$$

Then, $\mathbf{x}$ is preferred to $\mathbf{y}$ if and only if

$$\begin{aligned} &\mathbf{x} \succ_{1\dots k} \mathbf{y} \vee \\ &\quad \mathbf{x} =_{1\dots k} \mathbf{y} \wedge [(\exists l \in [k+1 \dots n] : f_l(\mathbf{y}) > r_k) \vee (\mathbf{x} \succ_{k+1 \dots n} \mathbf{y})] \end{aligned} \tag{6.4}$$

with $\mathbf{x} \succ_{i \dots j} \mathbf{y}$ meaning that solution $\mathbf{x}$ dominates solution $\mathbf{y}$ on objectives i to j (i.e., for minimization problems as considered here, $f_k(\mathbf{x} \leq f_k(\mathbf{y} \forall k = i \dots j$ with at least one strict inequality). A slightly extended version that allows the decision maker to additionally assign priorities to objectives has been published in Fonseca and Fleming (1998). This publication also contains the proof that the proposed preference relation is transitive. Figure 6.4 visualizes what part of the Pareto front remains preferred depending on whether the reference point is reachable (a) or not (b). If the goal has been set so ambitious that there is no solution which can reach the goal in even a single objective, the goal has no effect on search, and simply the whole Pareto front is returned.

In Deb (1999), a simpler variant has been proposed which simply ignores improvements over a goal value by replacing a solution's objective value $f_i(x)$ by $\max\{f_i(x), r_i\}$. If the goal vector $\mathbf{r}$ is outside the feasible range, the method is almost identical to the definition in Fonseca and Fleming (1993). However, if the goal can be reached, the approach from Deb (1999) will lose its selection

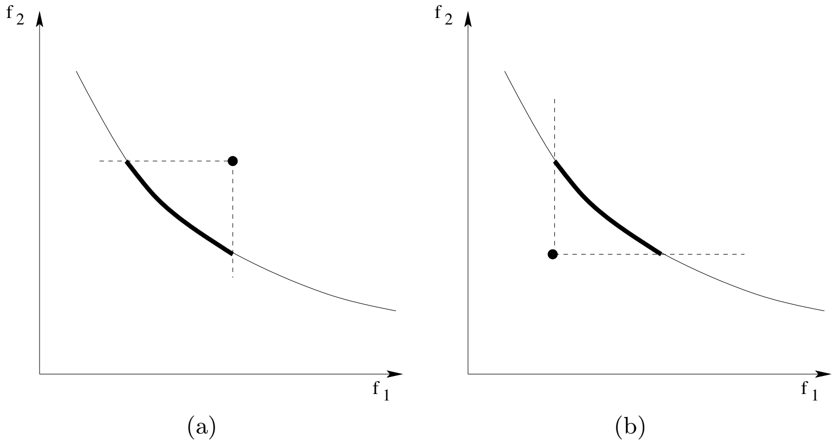

Fig. 6.4. Part of the Pareto optimal front that remains optimal with a given reference point **r** and the preference relation from Fonseca and Fleming (1993). The left panel (a) shows a reachable reference point, while the right panel (b) shows an unreachable one. Minimization of objectives is assumed.

pressure and basically stop search as soon as the reference point has been found, i.e., return a solution which is not Pareto optimal. On the other hand, the approach from Fonseca and Fleming (1993) keeps improving beyond the reference point. The goal-programming idea has been extended in Deb (2001) to allow for reference regions in addition to reference points.

Tan *et al.* (1999) proposed another ranking scheme which in a first stage prefers individuals fulfilling all criteria, and ranks those individuals according to standard non-dominance sorting. Among the remaining solutions, solution $\mathbf{x}$ dominates solution $\mathbf{y}$ if and only if $\mathbf{x}$ dominates $\mathbf{y}$ with respect to the objectives in which $\mathbf{x}$ does not fulfill the goal (as in Fonseca and Fleming (1993)), or if $|\mathbf{x} - \mathbf{r}| \succ |\mathbf{y} - \mathbf{r}|$. The latter corresponds to a "mirroring" of the objective vector along the axis of the fulfilled criteria. This may lead to some strange effects, such as non-transitivity of the preference relation (x is preferred to y, and y to z, but x and z are considered equal). Also, it seems odd to "penalize" solutions for largely exceeding a goal. What is more interesting in Tan *et al.* (1999) is the suggestion on how to account for multiple reference points, connected with AND and OR operations. The idea here is to rank the solutions independently with respect to all reference points. Then, rankings are combined as follows. If two reference points are connected by an AND operator, the rank of the solution is the maximum of the ranks according to the individual reference points. If the operator is an OR, the rank of the solution is the minimum of the ranks according to the individual reference points. This idea of combining the information of several reference points can naturally be combined with other preference relations using a reference point. The paper also presents a way to prioritize objectives by introducing additional goals. In effect, however, the priorization is equivalent to the one proposed in Fonseca and Fleming (1998).

In Deb and Sundar (2006); Deb *et al.* (2006), the crowding distance calculation in NSGA-II is replaced by the distance to the reference point, where solutions with a smaller distance are preferred. More specifically, solutions with the same non-dominated rank are sorted with respect to their distance to the reference point. Furthermore, to control the extent of obtained solutions, all solutions having a distance of ϵ or less between them are grouped. Only one randomly picked solution from each group is retained, while all other group members are assigned a large rank to discourage their use. As Fonseca and Fleming (1998) and Tan *et al.* (1999), this approach is able to improve beyond a reference point within the feasible region, because the non-dominated sorting keeps driving the population to the Pareto optimal front. Also, as Tan *et al.* (1999), it can handle multiple reference points simultaneously. With the parameter ϵ, it is possible to explicitly influence the diversity of solutions returned. Whether this extra parameter is an advantage or a burden may depend on the application.

Yet another dominance scheme was recently proposed in Molina *et al.* (2009), where solutions fulfilling all goals and solutions fulfilling none of the goals are preferred over solutions fulfilling only some of the goals. This, again, drives the search beyond the reference point if it is feasible, but it can obviously lead to situations where a solution which is dominated (fulfilling none of the goals) is actually preferred over the solution that dominates it (fulfilling some of the goals).

Thiele *et al.* (2007) integrate reference point information into the Indicator-Based Evolutionary Algorithm, see Section 6.6 for details.

The classical MCDM literature also includes some approaches where, in addition to a reference point, some further indicators are used to generate a set of alternative solutions. These include the reference direction method (Korhonen and Laakso, 1986) and light beam search (Jaszkiewicz and Slowinski, 1999). Recently, these methods have also been adopted into MOEAs.

In brief, the reference direction method allows the user to specify a starting point and a reference point, with the difference of the two defining the reference direction. Then, several points on this vector are used to define a set of achievement scalarizing functions, and each of these is used to search for a point on the Pareto optimal frontier. In Deb and Kumar (2007a), an MOEA is used to search for all these points simultaneously. For this purpose, the NSGA-II ranking mechanism has been modified to focus the search accordingly.

The light beam search also uses a reference direction, and additionally asks the user for some thresholds which are then used so find some possibly interesting neighboring solutions around the (according to the reference direction) most preferred solution. Deb and Kumar (2007b) use an MOEA to simultaneously search for a number of solutions in the neighborhood of the solution defined by the reference direction. This is achieved by first identifying the "most preferred" or "middle" solution using an achievement scalarizing function based on the reference point. Then, a modified crowding distance calculation is used to focus the search on those solutions which are not worse by more than the allowed threshold in all the objectives.

Summarizing, the first approach proposed in Fonseca and Fleming (1993) still seems to be a good way to include reference point information. While in most approaches the part of the Pareto optimal front considered as relevant depends on the reference point and the shape and location of the Pareto optimal front, in Deb and Sundar (2006) the desired spread of solutions in the vicinity of the Pareto optimal solution closest to the reference point is specified explicitly. The schemes proposed by Tan *et al.* (1999) and Deb and Sundar (2006) allow to consider several reference points simultaneously. The MOEAs based on the reference direction and light beam search (Deb and Kumar, 2007a,b) allow the user to specify additional information that influences the focus of the search and the set of solutions returned.

6.5 Limit Possible Trade-offs

If the user has no idea of what kind of solutions may be reachable, it may be easier to specify suitable trade-offs, i.e., how much gain in one objective is necessary to balance the loss in the other.

Greenwood *et al.* (1997) suggested a procedure which asks the user to rank a few alternatives, and from this derives constraints for linear weighting of the objectives consistent with the given ordering. Then, these are used to check whether there is a feasible linear weighting such that solution $\mathbf{x}$ is preferable to solution $\mathbf{y}$. More specifically, if the DM prefers a solution with objective values $\mathbf{f}(\mathbf{x})$ to a solution with objective values $\mathbf{f}(\mathbf{y})$, then, assuming linearly weighted additive utility functions and minimization of objectives, we know that

$$\sum_{k=1}^{n} w_k(f_k(\mathbf{x}) - f_k(\mathbf{y})) < 0 \tag{6.5}$$

$$\sum_{k=1}^{n} w_k = 1, \quad w_k \geq 0.$$

Let A denote the set of all pairs of solutions $(\mathbf{x}, \mathbf{y})$ ranked by the DM, and $\mathbf{x}$ preferred to $\mathbf{y}$. Then, to compare any two solutions $\mathbf{u}$ and $\mathbf{v}$, all linearly weighted additive utility functions are considered which are consistent with the ordering on the initially ranked solutions, i.e., consistent with Inequality 6.5 for all pairs of solutions $(\mathbf{x}, \mathbf{y}) \in A$. A preference of $\mathbf{u}$ over $\mathbf{v}$ is inferred if $\mathbf{u}$ is preferred to $\mathbf{v}$ for all such utility functions. A linear program (LP) is used to search for a utility function where $\mathbf{u}$ is not preferred to $\mathbf{v}$.

$$\min Z = \sum_{k=1}^{n} w_k(f_k(\mathbf{u}) - f_k(\mathbf{v})) \tag{6.6}$$

$$\sum_{k=1}^{n} w_k(f_k(\mathbf{x}) - f_k(\mathbf{y})) < 0 \qquad \forall(\mathbf{x}, \mathbf{y}) \in A \tag{6.7}$$

$$\sum_{k=1}^{n} w_k = 1, \quad w_k \geq 0.$$

If the LP returns a solution value $Z > 0$, we know there is no linear combination of objectives consistent with Inequality 6.7 such that $\mathbf{u}$ would be preferable, and we can conclude that $\mathbf{v}$ is preferred over $\mathbf{u}$. If the LP can find a linear combination with $Z < 0$, it only means that $\mathbf{v}$ is not preferred to $\mathbf{u}$. To test whether $\mathbf{u}$ is preferred to $\mathbf{v}$, one has to solve another LP and fail to find a linear combination of objectives such that $\mathbf{v}$ would be preferable. Overall, the method requires to solve 1 or 2 LPs for each pair of solutions in the population. Also, it needs special mechanisms to make sure that the allowed weight space does not become empty, i.e., that the user ranking is consistent with at least one possible linear weight assignment. The authors suggest to use a mechanism from White *et al.* (1984) which removes a minimal set of the DM's preference statements to make the weight space non-empty. Note that although linear combinations of objectives are assumed, it is possible to identify a concave part of the Pareto front, because the comparisons are only pair-wise. A more general framework for inferring preferences from examples (allowing for piecewise linear additive utility functions rather than linear additive utility functions) is discussed in Chapter 4.

In the guided MOEA proposed in Branke *et al.* (2001), the user is allowed to specify preferences in the form of maximally acceptable trade-offs like "one unit improvement in objective i is worth at most a_{ji} units in objective j". The basic idea is to modify the dominance criterion accordingly, so that it reflects the specified maximally acceptable trade-offs. A solution $\mathbf{x}$ is now preferred to a non-dominated solution $\mathbf{y}$ if the gain in the objective where $\mathbf{y}$ is better does not outweigh the loss in the other objective, see Figure 6.5 for an example. The region dominated by a solution is adjusted by changing the slope of the boundaries according to the specified maximal and minimal trade-offs. In this example, Solution A is now dominated by Solution B, because the loss in Objective 2 is too big to justify the improvement in Objective 1. On the other hand, Solutions D and C are still mutually non-dominated.

This idea can be implemented by a simple transformation of the objectives: It is sufficient to replace the original objectives with two auxiliary objectives Ω_1 and Ω_2 and use these together with the standard dominance principle, where

$$\Omega_1(x) = f_1(x) + \frac{1}{a_{21}} f_2(x)$$
$$\Omega_2(x) = \frac{1}{a_{12}} f_1(x) + f_2(x)$$

See Figures 6.7 and 6.6 for a visualization.

Because the transformation is so simple, the guided dominance scheme can be easily incorporated into standard MOEAs based on dominance, and it does not change the complexity nor the inner workings of the algorithm. However, an extension of this simple idea to more than two dimensions seems difficult.

Although developed independently and with a different motivation, the guided MOEA can lead to the same preference relation as the imprecise value function approach in Greenwood *et al.* (1997) discussed above. A maximally acceptable trade-off of the form "one unit improvement in objective i is worth at most a_{ji} units in objective j" could easily be transformed into the constraint

$$-w_i + a_{ji} \cdot w_j < 0 \qquad \text{or} \tag{6.8}$$
$$\frac{w_i}{w_j} > a_{ji} \tag{6.9}$$

The differences are in the way the maximally acceptable trade-offs are derived (specified directly by the DM in the guided MOEA, and inferred from a ranking of solutions in Greenwood *et al.* (1997)), and in the different implementation (a simple transformation of objectives in guided MOEA, and the solving of many LPs in the imprecise value function approach). While the guided MOEA is more elegant and computationally efficient for two objectives, the imprecise value function approach works independent of the number of objectives.

The idea proposed in Jin and Sendhoff (2002) is to aggregate the different objectives into one objective via weighted summation, but to vary the

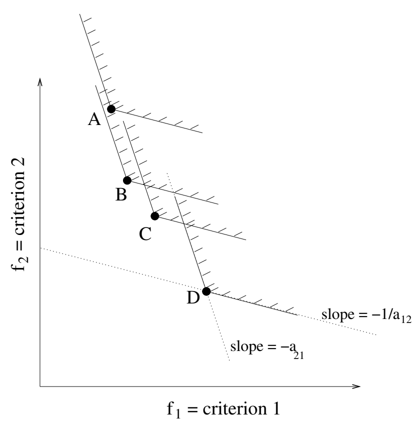

Fig. 6.5. Effect of the modified dominance scheme used by G-MOEA.

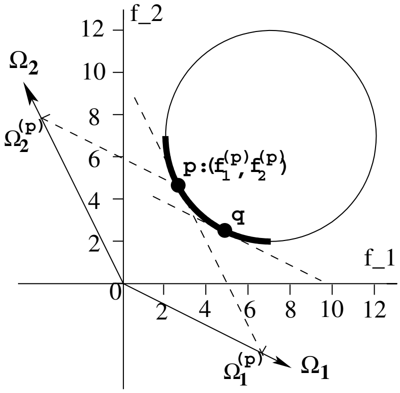

Fig. 6.6. When the guided dominance principle is used, non-dominated region of the Pareto optimal front is bounded by the two solutions p and q where the trade-off functions are tangent (Branke and Deb, 2005).

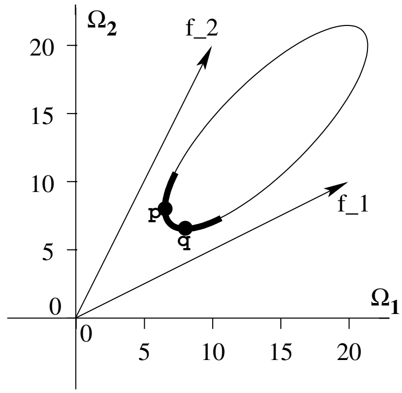

Fig. 6.7. The guided dominance principle is equivalent to the original dominance principle and appropriately transformed objective space (Branke and Deb, 2005).

weights gradually over time during the optimization. For two objectives, it is suggested to set $w_1(t) = |\sin(2\pi t/F)|$ and $w_2(t) = 1 - w_1(t)$, where t is the generation counter and F is a parameter to influence the oscillation period. The range of weights used in this process can be easily restricted to reflect the preferences of the DM by specifying a maximal and minimal weight $w_1^{\max}$ and $w_1^{\min}$, setting $w_1(t) = w_1^{\min} + (w_1^{\max} - w_1^{\min}) \cdot (sin(2\pi t/F) + 1)/2$ and adjusting w_2 accordingly. The effect is a population moving along the Pareto front, covering the part of the front which is optimal with respect to the range of possible weight values. Because the population will not converge but keep oscillating along the front, it is necessary to collect all non-dominated solutions found in an external archive. Note also the slight difference in effect to restricting the maximal and minimal trade-off as do the other approaches in this section. While the other approaches enforce these trade-offs locally, on a one-to-one comparison, the dynamic weighting modifies the global fitness function. Therefore, the approach runs into problems if the Pareto front is concave, because a small weight change would require the population to make a big "jump".

6.6 Approaches Based on Marginal Contribution

Several authors have recently proposed to replace the crowding distance as used in NSGA-II by a solution's contribution to a given performance measure, i.e., the loss in performance if that particular solution would be absent from the population (Branke *et al.*, 2004; Emmerich *et al.*, 2005; Zitzler and Künzli, 2004). In the following, we call this a solution's marginal contribution. The algorithm then looks similar to Algorithm 2.

Algorithm 2 Marginal contribution MOEA

```
Initialize population of size μ
Determine Pareto-ranking
Compute marginal contributions
repeat
  Select parents
  Generate λ offspring by crossover and mutation and add them to the
  population
  Determine Pareto-ranking
  Compute marginal contributions
  while (population size > μ) do {Environmental selection}
    From worst Pareto rank, remove individual with least marginal contri-
    bution
    Recompute marginal contributions
  end while
until termination condition
```

In Zitzler and Künzli (2004) and Emmerich *et al.* (2005), the performance measure used is the hypervolume. The hypervolume is the area (in 2D) or part of the objective space dominated by the solution set and bounded by a reference point **p**, see Chapter 14. Figure 6.8 gives an example for the hypervolume, and the parts used to rank the different solutions. The marginal contribution is then calculated only based on the individuals with the same Pareto rank. In the given example, Solution B has the largest marginal contribution. An obvious difficulty with hypervolume calculations is the determination of a proper reference point **p**, as this strongly influences the marginal contribution of the extreme solutions.

Zitzler *et al.* (2007) extend this idea by defining a weighting function over the objective space, and use the weighted hypervolume as indicator. This allows to incorporate preferences into the MOEA by giving preferred regions of the objective space a higher weight. In Zitzler *et al.* (2007), three different weighting schemes are proposed: a weight distribution which favors extremal solutions, a weight distribution which favors one objective over the other (but still keeping the best solution with respect to the less important objective), and a weight distribution based on a reference point, which generates a ridge-like function through the reference point (a, b) parallel to the diagonal. To calculate the weighted hypervolume marginal contributions, numerical integration is used.

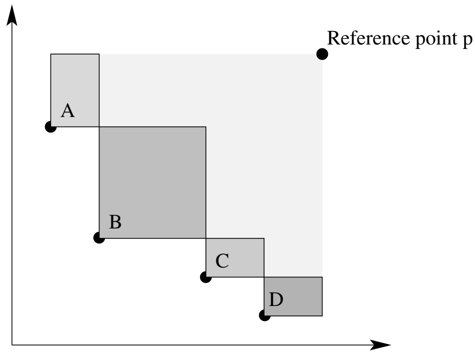

Fig. 6.8. Marginal contributions as calculated according to the hypervolume performance measure. The marginal contributions correspond to the respective shaded areas.

Another measure discussed in Zitzler and Künzli (2004) is the ϵ-Indicator. Basically, it measures the minimal distance by which an individual needs to be improved in each objective to become non-dominated (or can be worsened before it becomes dominated). Recently, Thiele *et al.* (2007) suggested to weight the ϵ-Indicator by an achievement scalarizing function based on a user specified reference point. The paper demonstrates that this allows to focus the search on the area around the specified reference point, and find interesting solutions faster.

Branke *et al.* (2004) proposed to use the "expected utility" as performance measure, i.e., a solution is evaluated by the expected loss in utility if this solution would be absent from the population. To calculate the expected utility, Branke *et al.* (2004) assumed that the DM has a linear utility function of the form $u(\mathbf{x}) = \lambda f_1(\mathbf{x}) + (1-\lambda) f_2(\mathbf{x})$, and λ is unknown but follows a uniform distribution over $[0, 1]$. The expected marginal utility (emu) of a solution $\mathbf{x}$ is then the utility difference between the best and second best solution, integrated over all utility functions where solution $\mathbf{x}$ is best:

$$\mathrm{emu}(\mathbf{x}) = \int_{\lambda=0}^{1} \max\{0, \min_{\mathbf{y}}\{u(\mathbf{y}) - u(\mathbf{x})\}\} d\lambda \tag{6.10}$$

While the expected marginal utility can be calculated exactly in the case of two objectives, numerical integration is required for more objectives. The result of using this performance measure is a natural focus of the search on so-called "knees", i.e., convex regions with strong curvature. In these regions, an improvement in either objective requires a significant worsening of the other objective, and such solutions are often preferred by DMs (Das, 1999). An example of the resulting distribution of individuals along a Pareto front with a single knee is shown in Figure 6.9. Although this approach does not take into account individual user preferences explicitly, it favors the often preferred knee regions of the Pareto front. Additional explicit user preferences can be taken into account by allowing the user to specify the probability distribution for λ. For example, a probable preference for objective f_2 could be expressed by a linearly decreasing probability density of λ in the interval $[0..1]$, $p_\lambda(\alpha) = 2 - 2\alpha$. The effect of integrating such a preference information can be seen in Figure 6.10.

6.7 Other Approaches

The method by Cvetkovic and Parmee (2002) assigns each criterion a weight w_i, and additionally requires a minimum level for dominance τ, which corresponds to the concordance criterion of the ELECTRE method Figueira *et al.* (2005). Accordingly, the following weighted dominance criterion is used as dominance relation in the MOEA.

$$\mathbf{x} \succ_w \mathbf{y} \Leftrightarrow \sum_{i: f_i(\mathbf{x}) \leq f_i(\mathbf{y})} w_i \geq \tau.$$

To facilitate specification of the required weights, they suggest a method to turn fuzzy preferences into specific quantitative weights. However, since for every criterion the dominance scheme only considers whether one solution is better than another solution, and not by how much it is better, this approach allows only a very coarse guidance and is difficult to control. A somewhat similar dominance criterion has been proposed in Schmiedle *et al.* (2002). As

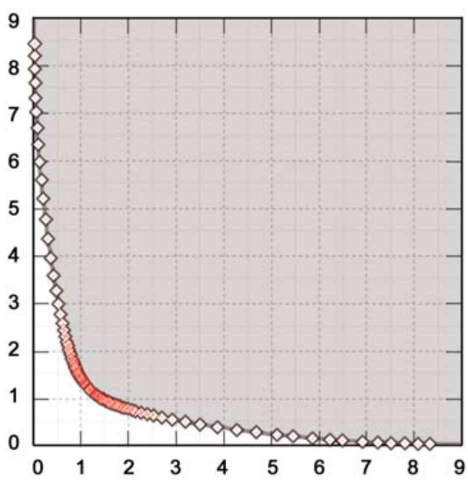

Fig. 6.9. Marginal contribution calculated according to expected utility result in a concentration of the individuals in knee areas.

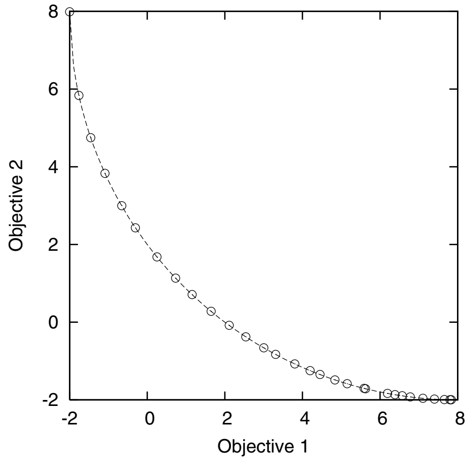

Fig. 6.10. Resulting distribution of individuals with the marginal expected utility approach and a linearly decreasing probability distribution for λ.

an additioanal feature, cycles in the preference relation graph are treated by considering all alternatives in a cycle as equivalent, and merging them into a single meta-node in the preference relation graph.

Hughes (2001) is concerned with MOEAs for noisy objective functions. The main idea to cope with the noise is to rank individuals by the sum of probabilities of being dominated by any other individual. To take preferences into account, the paper proposes a kind of weighting of the domination probabilities.

Some papers (Rekiek *et al.*, 2000; Coelho *et al.*, 2003; Parreiras and Vasconcelos, 2005) use preference flow according to Promethee II (Brans and Mareschal, 2005). Although this generates a preference order of the individuals, it does so depending on the different alternatives present in the population, not in absolute terms as, e.g., a weighted aggregation would do.

6.8 Discussions and Conclusions

If a single solution is to be selected in a multiobjective optimization problem, at some point during the process, the DM has to reveal his/her preferences. Specifying these preferences a priori, i.e., before alternatives are known, often means to ask too much of the DM. On the other hand, searching for all non-dominated solutions as most MOEA do may result in a waste of optimization efforts to find solutions that are clearly unacceptable to the DM.

This chapter overviewed intermediate approaches, that ask for partial preference information from the DM a priori, and then focus the search to those regions of the Pareto optimal front that seem most interesting to the DM.

That way, it is possible to provide a larger number of relevant solutions. It seems intuitive that this should also allow to reduce the computation time, although this aspect has explicitly only been shown in Branke and Deb (2005) and Thiele *et al.* (2007).

Table 6.1 summarizes some aspects of some of the most prominent approaches. It lists the information required from the DM (Information), the part of the MOEA modified (Modification), and whether the result is a bounded region of the Pareto optimal front or a biased distribution (Influence). What method is most appropriate certainly depends on the application (e.g., whether the Pareto front is convex or concave, or whether the DM has a good conception of what is reachable) and on the kind of information the DM feels comfortable to provide. Many of the ideas can be combined, allowing the DM to provide preference information in different ways. For example, it would be straightforward to combine a reference point based approach which leads to sharp boundaries of the area in objective space considered as interesting with a marginal contribution approach which alters the distribution

Table 6.1. Comparison of some selected approaches to incorporate partial user preferences.

Name	Information	Modification	Influence
Constraints Coello (2002)	constraint	miscellaneous	region
Preference relation Fonseca and Fleming (1993)	reference point	dominance	region
Reference point based EMO, Deb *et al.* (2006)	reference point	crowding dist.	region
Light beam search based EMO, Deb and Kumar (2007b)	reference direction thresholds	crowding dist.	region
Imprecise value function Greenwood *et al.* (1997)	solution ranking	dominance	region
Guided MOEA Branke *et al.* (2001)	maximal/minimal trade-off	objectives	region
Weighted integration Zitzler *et al.* (2007)	weighting of objective space	crowding dist.	distribution
Marginal expected utility Branke *et al.* (2004)	trade-off prob- ability distribution	crowding dist.	distribution
Biased crowding Branke and Deb (2005)	desired trade-off	crowding dist.	distribution

within this area. Furthermore, many of the ideas can be used in an interactive manner, which will be the focus of the following chapter (Chapter 7).

References

Branke, J., Deb, K.: Integrating user preference into evolutionary multi-objective optimization. In: Jin, Y. (ed.) Knowledge Incorporation in Evolutionary Computation, pp. 461–478. Springer, Heidelberg (2005)

Branke, J., Kaußler, T., Schmeck, H.: Guidance in evolutionary multi-objective optimization. Advances in Engineering Software 32, 499–507 (2001)

Branke, J., Deb, K., Dierolf, H., Osswald, M.: Finding knees in multi-objective optimization. In: Yao, X., et al. (eds.) PPSN 2004. LNCS, vol. 3242, pp. 722–731. Springer, Heidelberg (2004)

Brans, J.-P., Mareschal, B.: PROMETHEE methods. In: Figueira, J., et al. (eds.) Multiple criteria decision analysis, pp. 163–196. Springer, Heidelberg (2005)

Coelho, R.F., Bersini, H., Bouillard, P.: Parametrical mechanical design with constraints and preferences: Application to a purge valve. Computer Methods in Applied Mechanics and Engineering 192, 4355–4378 (2003)

Coello Coello, C.A.: Handling preferences in evolutionary multiobjective optimization: A survey. In: Congress on Evolutionary Computation, vol. 1, pp. 30–37. IEEE Computer Society Press, Los Alamitos (2000)

Coello Coello, C.A.: Theoretical and numerical constraint-handling techniques used with evolutionary algorithms: A survey of the state of the art. Computer Methods in Applied Mechanics and Engineering 191(11-12), 1245–1287 (2002)

Coello Coello, C.A., van Veldhuizen, D.A., Lamont, G.B.: Evolutionary Algorithms for Solving Multi-Objective Problems. Kluwer Academic Publishers, Dordrecht (2002)

Cvetkovic, D., Parmee, I.C.: Preferences and their application in evolutionary multiobjective optimisation. IEEE Transactions on Evolutionary Computation 6(1), 42–57 (2002)

Das, I.: On characterizing the 'knee' of the pareto curve based on normal-boundary intersection. Structural Optimization 18(2/3), 107–115 (1999)

Deb, K.: Solving goal programming problems using multi-objective genetic algorithms. In: Proceedings of Congress on Evolutionary Computation, pp. 77–84 (1999)

Deb, K.: An efficient constraint handling method for genetic algorithms. Computer Methods in Applied Mechanics and Engineering 186(2-4), 311–338 (2000)

Deb, K.: Multi-Objective Optimization using Evolutionary Algorithms. Wiley, Chichester (2001)

Deb, K.: Multi-objective evolutionary algorithms: Introducing bias among Pareto-optimal solutions. In: Ghosh, A., Tsutsui, S. (eds.) Advances in Evolutionary Computing: Theory and Applications, pp. 263–292. Springer, Heidelberg (2003)

Deb, K., Kumar, A.: Interactive evolutionary multi-objective optimization and decision-making using reference direction method. In: Genetic and Evolutionary Computation Conference, pp. 781–788. ACM Press, New York (2007a)

Deb, K., Kumar, A.: Light beam search based multi-objective optimization using evolutionary algorithms. In: Congress on Evolutionary Computation, pp. 2125–2132. IEEE Computer Society Press, Los Alamitos (2007b)

Deb, K., Sundar, J.: Reference point based multi-objective optimization using evolutionary algorithms. In: Genetic and Evolutionary Computation Conference, pp. 635–642. ACM Press, New York (2006)

Deb, K., Agrawal, S., Pratap, A., Meyarivan, T.: A fast and elitist multi-objective genetic algorithm: NSGA-II. IEEE Transactions on Evolutionary Computation 6(2), 182–197 (2002a)

Deb, K., Thiele, L., Laumanns, M., Zitzler, E.: Scalable multi-objective optimization test problems. In: Congress on Evolutionary Computation (CEC), pp. 825–830 (2002b)

Deb, K., Sundar, J., Reddy, U.B., Chaudhuri, S.: Reference point based multi-objective optimization using evolutionary algorithms. International Journal of Computational Intelligence Research 2(3), 273–286 (2006)

Emmerich, M.T.M., Beume, N., Naujoks, B.: An EMO algorithm using the hypervolume measure as selection criterion. In: Coello Coello, C.A., Hernández Aguirre, A., Zitzler, E. (eds.) EMO 2005. LNCS, vol. 3410, pp. 62–76. Springer, Heidelberg (2005)

Figueira, J., Mousseau, V., Roy, B.: ELECTRE methods. In: Figueia, J., Greco, S., Ehrgott, M. (eds.) Multiple Criteria Decision Analysis: State of the Art Surveys, pp. 134–162. Springer, Heidelberg (2005)

Fonseca, C.M., Fleming, P.J.: Genetic algorithms for multiobjective optimization: Formulation, discussion, and generalization. In: International Conference on Genetic Algorithms, pp. 416–423 (1993)

Fonseca, C.M., Fleming, P.J.: Multiobjective optimization and multiple constraint handling with evolutionary algorithms - part I: A unified fomulation. IEEE Transactions on Systems, Man, and Cybernetics - Part A 28(1), 26–37 (1998)

Greenwood, G.W., Hu, X.S., D'Ambrosio, J.G.: Fitness functions for multiple objective optimization problems: combining preferences with Pareto rankings. In: Belew, R.K., Vose, M.D. (eds.) Foundations of Genetic Algorithms, pp. 437–455. Morgan Kaufmann, San Francisco (1997)

Horn, J.: Multicriterion Decision making. In: Bäck, T., Fogel, D., Michalewicz, Z. (eds.) Handbook of Evolutionary Computation, vol. 1, pp. F1.9:1–F1.9:15. Oxford University Press, Oxford (1997)

Hughes, E.J.: Constraint handling with uncertain and noisy multi-objective evolution. In: Congress on Evolutionary Computation, pp. 963–970. IEEE Computer Society Press, Los Alamitos (2001)

Jaszkiewicz, A., Slowinski, R.: The light beam search over a non-dominated surface of a multiple-objective programming problem. European Journal of Operational Research 113(2), 300–314 (1999)

Jiménez, F., Verdegay, J.L.: Evolutionary techniques for constrained optimization problems. In: Zimmermann, H.-J. (ed.) European Congress on Intelligent Techniques and Soft Computing, Verlag Mainz, Aachen (1999)

Jin, Y., Sendhoff, B.: Incorporation of fuzzy preferences into evolutionary multi-objective optimization. In: Asia-Pacific Conference on Simulated Evolution and Learning, Nanyang Technical University, Singapore, pp. 26–30 (2002)

Korhonen, P., Laakso, J.: A visual interactive method for solving the multiple criteria problem. European Journal of Operational Research 24, 277–287 (1986)

Molina, J., Santana, L.V., Hernandez-Diaz, A.G., Coello Coello, C.A., Caballero, R.: g-dominance: Reference point based dominance. European Journal of Operational Research (2009)

Parreiras, R.O., Vasconcelos, J.A.: Decision making in multiobjective optimization problems. In: Nedjah, N., de Macedo Mourelle, L. (eds.) Real-World Multi-Objective System Engineering, pp. 29–52. Nova Science Publishers, New York (2005)

Rachmawati, L., Srinivasan, D.: Preference incorporation in multi-objective evolutionary algorithms: A survey. In: Congress on Evolutionary Computation, pp. 3385–3391. IEEE Computer Society Press, Los Alamitos (2006)

Rekiek, B., Lit, P.D., Fabrice, P., L'Eglise, T., Emanuel, F., Delchambre, A.: Dealing with users's preferences in hybrid assembly lines design. In: Binder, Z., et al. (eds.) Management and Control of Production and Logistics Conference, pp. 989–994. Pergamon Press, Oxford (2000)

Sait, S.M., Youssef, H., Ali, H.: Fuzzy simulated evolution algorithm for multi-objective optimization of VLSI placement. In: Congress on Evolutionary Computation, pp. 91–97. IEEE Computer Society Press, Los Alamitos (1999)

Sakawa, M., Yauchi, K.: An interactive fuzzy satisficing method for multiobjective nonconvex programming problems through floating point genetic algorithms. European Journal of Operational Research 117, 113–124 (1999)

Schmiedle, F., Drechsler, N., Große, D., Drechsler, R.: Priorities in multi-objective optimization for genetic programming. In: Spector, L., et al. (eds.) Genetic and Evolutionary Computation Conference, pp. 129–136. Morgan Kaufmann, San Francisco (2002)

Tan, K.C., Lee, T.H., Khor, E.F.: Evolutionary algorithms with goal and priority information for multi-objective optimization. In: Congress on Evolutionary Computation, pp. 106–113. IEEE Computer Society Press, Los Alamitos (1999)

Thiele, L., Miettinen, K., Korhonen, P.J., Molina, J.: A preference-based interactive evolutionary algorithm for multiobjective optimization. Technical Report W-412, Helsinki School of Economics, Helsinki, Finland (2007)

Trautmann, H., Mehnen, J.: A method for including a-priori-preference in multicriteria optimization. Technical Report 49/2005, SFG 475, University of Dortmund, Germany (2005)

Van Veldhuizen, D., Lamont, G.B.: Multiobjective evolutionary algorithms: Analyzing the state-of-the-art. Evolutionary Computation Journal 8(2), 125–148 (2000)

White, C., Sage, A., Dozono, S.: A model of multiattribute decision-making and tradeoff weight determination under uncertainty. IEEE Transactions on Systems, Man, and Cybernetics 14, 223–229 (1984)

Wierzbicki, A.P.: Basic properties of scalarizing functions for multiobjective optimization. Optimization 8(1), 55–60 (1977)

Wierzbicki, A.P.: On the completeness and constructiveness of parametric characterizations to vector optimization problems. OR Spektrum 8(2), 73–87 (1986)

Zitzler, E., Künzli, S.: Indicator-based selection in multiobjective search. In: Yao, X., et al. (eds.) PPSN 2004. LNCS, vol. 3242, pp. 832–842. Springer, Heidelberg (2004)

Zitzler, E., Laumanns, M., Thiele, L.: SPEA2: Improving the Strength Pareto Evolutionary Algorithm for multiobjective optimization. In: Giannakoglou, K.C., et al. (eds.) Evolutionary Methods for Design, Optimisation and Control with Application to Industrial Problems (EUROGEN 2001), pp. 95–100. International Center for Numerical Methods in Engineering, CIMNE (2002)

Zitzler, E., Brockhoff, D., Thiele, L.: The hypervolume indicator revisited: On the design of pareto-compliant indicators via weighted integration. In: Obayashi, S., Deb, K., Poloni, C., Hiroyasu, T., Murata, T. (eds.) EMO 2007. LNCS, vol. 4403, pp. 862–876. Springer, Heidelberg (2007)