

# Introduction to Multiobjective Optimization: Noninteractive Approaches

Kaisa Miettinen

Department of Mathematical Information Technology  
P.O. Box 35 (Agora), FI-40014 University of Jyväskylä, Finland\*  
kaisa.miettinen@jyu.fi

**Abstract.** We give an introduction to nonlinear multiobjective optimization by covering some basic concepts as well as outlines of some methods. Because Pareto optimal solutions cannot be ordered completely, we need extra preference information coming from a decision maker to be able to select the most preferred solution for a problem involving multiple conflicting objectives. Multiobjective optimization methods are often classified according to the role of a decision maker in the solution process. In this chapter, we concentrate on noninteractive methods where the decision maker either is not involved or specifies preference information before or after the actual solution process. In other words, the decision maker is not assumed to devote too much time in the solution process.

## 1.1 Introduction

Many decision and planning problems involve multiple conflicting objectives that should be considered simultaneously (alternatively, we can talk about multiple conflicting criteria). Such problems are generally known as multiple criteria decision making (MCDM) problems. We can classify MCDM problems in many ways depending on the characteristics of the problem in question. For example, we talk about multiattribute decision analysis if we have a discrete, predefined set of alternatives to be considered. Here we study multiobjective optimization (also known as multiobjective mathematical programming) where the set of feasible solutions is not explicitly known in advance but it is restricted by constraint functions. Because of the aims and scope of this book, we concentrate on nonlinear multiobjective optimization (where at least one function in the problem formulation is nonlinear) and ignore approaches designed only for multiobjective linear programming (MOLP) problems (where all the functions are linear).

---

\* In 2007 also Helsinki School of Economics, Helsinki, Finland

Reviewed by: Nirupam Chakraborti, Indian Institute of Technology, India

Hiroataka Nakayama, Konan University, Japan

Roman Słowiński, Poznan University of Technology, Poland

In multiobjective optimization problems, it is characteristic that no unique solution exists but a set of mathematically equally good solutions can be identified. These solutions are known as nondominated, efficient, noninferior or Pareto optimal solutions (defined in Preface). In the MCDM literature, these terms are usually seen as synonyms. Multiobjective optimization problems have been intensively studied for several decades and the research is based on the theoretical background laid, for example, in (Edgeworth, 1881; Koopmans, 1951; Kuhn and Tucker, 1951; Pareto, 1896, 1906). As a matter of fact, many ideas and approaches have their foundation in the theory of mathematical programming. For example, while formulating optimality conditions of nonlinear programming, Kuhn and Tucker (1951) did also formulate them for multiobjective optimization problems.

Typically, in the MCDM literature, the idea of solving a multiobjective optimization problem is understood as helping a human decision maker (DM) in considering the multiple objectives simultaneously and in finding a Pareto optimal solution that pleases him/her the most. Thus, the solution process needs some involvement of the DM in the form of specifying preference information and the final solution is determined by his/her preferences in one way or the other. In other words, a more or less explicit preference model is built from preference information and this model is exploited in order to find solutions that better fit the DM's preferences. Here we assume that a single DM is involved. Group decision making with several DMs is discussed, e.g., in (Hwang and Lin, 1987; Fandel, 1990).

In general, the DM is a person who is assumed to know the problem considered and be able to provide preference information related to the objectives and/or different solutions in some form. Besides a DM, we usually also need an analyst when solving a multiobjective optimization problems. An *analyst* is a person or a computer program responsible for the mathematical modelling and computing sides of the solution process. The analyst is supposed to help the DM at various stages of the solution process, in particular, in eliciting preference information and in interpreting the information coming from the computations (see also Chapter 15).

We can list several desirable properties of multiobjective optimization methods. Among them are, for example, that the method should generate Pareto optimal solutions reliably, it should help the DM to get an overview of the set of Pareto optimal solutions, it should not require too much time from the DM, the information exchanged (given by the method and asked from the DM) should be understandable and not too demanding or complicated (cognitively or otherwise) and the method should support the DM in finding the most preferred solution as the final one so that the DM could be convinced of its relative goodness. The last-mentioned aim could be characterized as psychological convergence (differing from mathematical convergence which is emphasized in mathematical programming).

Surveys of methods developed for multiobjective optimization problems include (Chankong and Haimes, 1983; Hwang and Masud, 1979; Marler and

Arora, 2004; Miettinen, 1999; Sawaragi *et al.*, 1985; Steuer, 1986; Vincke, 1992). For example, in (Hwang and Masud, 1979; Miettinen, 1999), the methods are classified into the four following classes according to the role of the DM in the solution process. Sometimes, there is no DM and her/his preference information available and in those cases we must use so-called *no-preference methods*. Then, the task is to find some neutral compromise solution without any additional preference information. This means that instead of asking the DM for preference information, some assumptions are made about what a “reasonable” compromise could be like. In all the other classes, the DM is assumed to take part in the solution process.

In *a priori methods*, the DM first articulates preference information and one’s aspirations and then the solution process tries to find a Pareto optimal solution satisfying them as well as possible. This is a straightforward approach but the difficulty is that the DM does not necessarily know the possibilities and limitations of the problem beforehand and may have too optimistic or pessimistic expectations. Alternatively, it is possible to use *a posteriori* methods, where a representation of the set of Pareto optimal solutions is first generated and then the DM is supposed to select the most preferred one among them. This approach gives the DM an overview of different solutions available but if there are more than two objectives in the problem, it may be difficult for the DM to analyze the large amount of information (because visualizing the solutions is no longer as straightforward as in a biobjective case) and, on the other hand, generating the set of Pareto optimal solutions may be computationally expensive. Typically, evolutionary multiobjective optimization algorithms (see Chapter 3) belong to this class but, when using them, it may happen that the real Pareto optimal set is not reached. This means that the solutions produced are nondominated in the current population but not necessarily actually Pareto optimal (if, e.g., the search is stopped too early).

In this chapter, we concentrate on the three classes of noninteractive methods where either no DM takes part in the solution process or (s)he expresses preference relations before or after the process. The fourth class devoted to *interactive methods* is the most extensive class of methods and it will be covered in Chapter 2. In interactive approaches, an iterative solution algorithm (which can be called a solution pattern) is formed and repeated (typically several times). After each iteration, some information is given to the DM and (s)he is asked to specify preference information (in the form that the method in question can utilize, e.g., by answering some questions). One can say that the analyst aims at determining the preference structure of the DM in an interactive way. What is noteworthy is that the DM can specify and adjust one’s preferences between each iteration and at the same time learn about the interdependencies in the problem as well as about one’s own preferences.

Methods in different classes have their strengths and weaknesses and for that reason different approaches are needed. Let us point out that the classification we use here is not complete or absolute. Overlapping and combinations of classes are possible and some methods can belong to more than one class

depending on different interpretations. Other classifications are given, for example, by Cohon (1985); Rosenthal (1985).

The rest of this chapter is organized as follows. In Section 1.2, we augment the basic terminology and notation introduced in Preface. In other words, we discuss some more concepts of multiobjective optimization including optimality and elements of a solution process. After that we introduce two widely used basic methods, the weighting method and the  $\varepsilon$ -constraint method in Section 1.3. Sections 1.4–1.6 are devoted to some methods belonging to the three above-described classes, that is, no-preference methods, a posteriori methods and a priori methods, respectively. We also give references to further details. In Section 1.7, we summarize some properties of the methods described and, finally, we conclude with Section 1.8.

## 1.2 Some Concepts

### 1.2.1 Optimality

Continuous multiobjective optimization problems typically have an infinite number of Pareto optimal solutions (whereas combinatorial multiobjective optimization problems have a finite but possibly very large number of Pareto optimal solutions) and the Pareto optimal set (consisting of the Pareto optimal solutions) can be nonconvex and disconnected. Because the basic terminology and concepts of multiobjective optimization were defined in Preface, we do not repeat them here. However, it is important to note that the definitions of Pareto optimality and weak Pareto optimality (given in Preface) introduce *global Pareto optimality* and *global weak Pareto optimality*. Corresponding to nonlinear programming, we can also define local (weak) Pareto optimality in a small environment of the point considered. Let us emphasize that a locally Pareto optimal objective vector has no practical relevance (if it is not global) because it may be located in the interior of the feasible objective region (i.e., it is possible to improve all objective function values) whereas globally Pareto optimal solutions are always located on its boundary. Thus, it is important to use appropriate tools to get globally Pareto optimal solutions. We shall get back to this when we discuss scalarizing functions.

Naturally, any globally Pareto optimal solution is locally Pareto optimal. The converse is valid for convex problems, see, for example, (Miettinen, 1999). A multiobjective optimization problem can be defined to be convex if the feasible objective region is convex or if the feasible region is convex and the objective functions are quasiconvex with at least one strictly quasiconvex function.

Before we continue, it is important to briefly touch the existence of Pareto optimal solutions. It is shown in (Sawaragi *et al.*, 1985) that Pareto optimal solutions exist if we assume that the (nonempty) feasible region is compact and all the objective functions are lower semicontinuous. Alternatively, we can formulate the assumption in the form that the feasible objective region is

nonempty and compact. We do not go into details of theoretical foundations here but assume in what follows that Pareto optimal solutions exist. Another important question besides the existence of Pareto optimal solutions is the stability of the Pareto optimal set with respect to perturbations of the feasible region, objective functions or domination structures of the DM. This topic is extensively discussed in (Sawaragi *et al.*, 1985) and it is also touched in Chapter 9. Let us mention that sometimes, like by Steuer (1986), Pareto optimal decision vectors are referred to as efficient solutions and the term nondominated solution is used for Pareto optimal objective vectors.

If the problem is correctly specified, the final solution of a rational DM is always Pareto optimal. Thus, we can restrict our consideration to Pareto optimal solutions. For that reason, it is important that the multiobjective optimization method used can meet the following two needs: firstly, it must be able to cover, that is, find any Pareto optimal solution and, secondly, generate only Pareto optimal solutions (Sawaragi *et al.*, 1985). However, weakly Pareto optimal solutions are often relevant from a technical point of view because they are sometimes easier to generate than Pareto optimal ones.

One more widely used optimality concepts is proper Pareto optimality. The properly Pareto optimal set is a subset of the Pareto optimal set which is a subset of the weakly Pareto optimal set. For an example of these three concepts of optimality and their relationships, see Figure 1.1. In the figure, the set of weakly Pareto optimal solutions is denoted by a bold line. The endpoints of the Pareto optimal set are denoted by circles and the endpoints of the properly Pareto optimal set by short lines (note that the sets can also be disconnected).

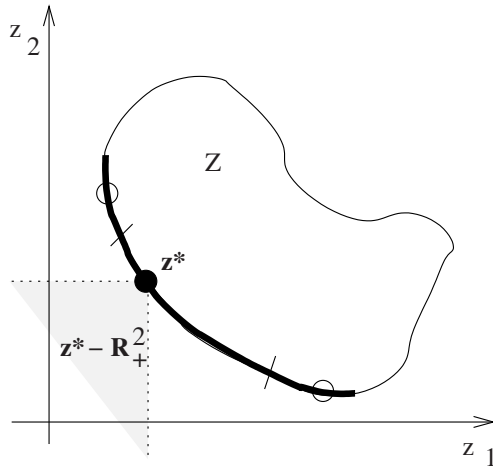


Fig. 1.1. Sets of properly, weakly and Pareto optimal solutions.

As a matter of fact, Pareto optimal solutions can be divided into improperly and properly Pareto optimal ones depending on whether unbounded trade-offs between objectives are allowed or not. Practically, a properly Pareto optimal solution with a very high trade-off does not essentially differ from a weakly Pareto optimal solution for a human DM. There are several definitions for proper Pareto optimality and they are not equivalent. The first definition was given by Kuhn and Tucker (1951) while they formulated optimality conditions for multiobjective optimization. Some of the definitions are collected, for example, in (Miettinen, 1999) and relationships between different definitions are analyzed in (Sawaragi *et al.*, 1985; Makarov and Rachkovski, 1999).

The idea of proper Pareto optimality is easily understandable in the definition of Geoffrion (1968): A decision vector  $\mathbf{x}' \in S$  is *properly Pareto optimal* (in the sense of Geoffrion) if it is Pareto optimal and if there is some real number  $M$  such that for each  $f_i$  and each  $\mathbf{x} \in S$  satisfying  $f_i(\mathbf{x}) < f_i(\mathbf{x}')$  there exists at least one  $f_j$  such that  $f_j(\mathbf{x}') < f_j(\mathbf{x})$  and

$$\frac{f_i(\mathbf{x}') - f_i(\mathbf{x})}{f_j(\mathbf{x}) - f_j(\mathbf{x}')} \leq M.$$

An objective vector is properly Pareto optimal if the corresponding decision vector is properly Pareto optimal. We can see from the definition that a solution is properly Pareto optimal if there is at least one pair of objectives for which a finite decrement in one objective is possible only at the expense of some reasonable increment in the other objective.

Let us point out that optimality can be defined in more general ways (than above) with the help of ordering cones (pointed convex cones)  $D$  defined in  $\mathbf{R}^k$ . The cone  $D$  can be used to induce a partial ordering in  $Z$ . In other words, for two objective vectors  $\mathbf{z}$  and  $\mathbf{z}'$  we can say that  $\mathbf{z}'$  dominates  $\mathbf{z}$  if

$$\mathbf{z} \in \mathbf{z}' + D \setminus \{\mathbf{0}\}.$$

Now we can say that a feasible decision vector is efficient and the corresponding objective vector is nondominated with respect to  $D$  if there exists no other feasible objective vector that dominates it. This definition is equivalent to Pareto optimality if we set

$$D = \mathbf{R}_+^k = \{\mathbf{z} \in \mathbf{R}^k \mid z_i \geq 0 \text{ for } i = 1, \dots, k\},$$

that is,  $D$  is the nonnegative orthant of  $\mathbf{R}^k$ . For further details of ordering cones and different spaces we refer, for example, to (Jahn, 2004; Luc, 1989) and references therein.

As said, we can give an equivalent formulation to the definition of Pareto optimality (given in Preface) as follows: A feasible decision vector  $\mathbf{x}^* \in S$  and the corresponding objective vector  $\mathbf{z}^* = \mathbf{f}(\mathbf{x}^*) \in Z$  are Pareto optimal if

$$\mathbf{z}^* - \mathbf{R}_+^k \setminus \{\mathbf{0}\} \cap Z = \emptyset.$$

For a visualization of this, see Figure 1.1, where a shifted cone at  $\mathbf{z}^*$  is illustrated. This definition clearly shows why Pareto optimal objective vectors must be located on the boundary of the feasible objective region  $Z$ . After having introduced the definition of Pareto optimality in this form, we can give another definition for proper Pareto optimality. This definition (introduced by Wierzbicki (1986)) is both computationally usable and intuitive.

The above-defined vectors  $\mathbf{x}^* \in S$  and  $\mathbf{z}^* \in Z$  are  $\rho$ -properly Pareto optimal if

$$\mathbf{z}^* - \mathbf{R}_\rho^k \setminus \{\mathbf{0}\} \cap Z = \emptyset,$$

where  $\mathbf{R}_\rho^k$  is a slightly broader cone than  $\mathbf{R}_+^k$ . Now, trade-offs are bounded by  $\rho$  and  $1/\rho$  and we have a relationship to  $M$  used in Geoffrion's definition as  $M = 1 + 1/\rho$ . For details, see, for example (Miettinen, 1999; Wierzbicki, 1986).

### 1.2.2 Solution Process and Some Elements in It

Mathematically, we cannot order Pareto optimal objective vectors because the objective space is only partially ordered. However, it is generally desirable to obtain one point as a final solution to be implemented and this solution should satisfy the preferences of the particular DM. Finding a solution to problem (1) defined in Preface is called a *solution process*. As mentioned earlier, it usually involves co-operation of the DM and an analyst. The analyst is supposed to know the specifics of the methods used and help the DM at various stages of the solution process. It is important to emphasize that the DM is not assumed to know MCDM or methods available but (s)he is supposed to be an expert in the problem domain, that is, understand the application considered. Sometimes, finding the set of Pareto optimal solutions is referred to as *vector optimization*. However, here by solving a multiobjective optimization problem we mean finding a feasible and Pareto optimal decision vector that satisfies the DM. Assuming such a solution exists, it is called a *final solution*.

The concepts of ideal and nadir objective vectors were defined in Preface for getting information about the ranges of the objective function values in the Pareto optimal set; provided the objective functions are bounded over the feasible region. As mentioned then, there is no constructive method for calculating the nadir objective vector for nonlinear problems. A *payoff table* (suggested by Benayoun *et al.* (1971)) is often used but it is not a reliable way as demonstrated, for example, by Korhonen *et al.* (1997); Weistroffer (1985). The payoff table has  $k$  objective vectors as its rows where objective function values are calculated at points optimizing each objective function individually. In other words, components of the ideal objective vector are located on the diagonal of the payoff table. An estimate of the nadir objective vector is obtained by finding the worst objective values in each column. This method gives accurate information only in the case of two objectives. Otherwise, it may be an over- or an underestimation (because of alternative optima, see,

e.g., (Miettinen, 1999) for details). Let us mention that the nadir objective vector can also be estimated using evolutionary algorithms (Deb *et al.*, 2006).

Multiobjective optimization problems are usually solved by scalarization. *Scalarization* means that the problem involving multiple objectives is converted into an optimization problem with a single objective function or a family of such problems. Because this new problem has a real-valued objective function (that possibly depends on some parameters coming, e.g., from preference information), it can be solved using appropriate single objective optimizers. The real-valued objective function is often referred to as a *scalarizing function* and, as discussed earlier, it is justified to use such scalarizing functions that can be proven to generate Pareto optimal solutions. (However, sometimes it may be computationally easier to generate weakly Pareto optimal solutions.) Depending on whether a local or a global solver is used, we get either locally or globally Pareto optimal solutions (if the problem is not convex). As discussed earlier, locally Pareto optimal objective vectors are not of interest and, thus, we must pay attention that an appropriate solver is used. We must also keep in mind that when using numerical optimization methods, the solutions obtained are not necessarily optimal in practice (e.g., if the method used does not converge properly or if the global solver fails in finding the global optimum).

It is sometimes assumed that the DM makes decisions on the basis of an underlying function. This function representing the preferences of the DM is called a *value function*  $v : \mathbf{R}^k \rightarrow \mathbf{R}$  (Keeney and Raiffa, 1976). In some methods, the value function is assumed to be known implicitly and it has been important in the development of solution methods and as a theoretical background. A utility function is often used as a synonym for a value function but we reserve that concept for stochastic problems which are not treated here. The value function is assumed to be non-increasing with the increase of objective values because we here assume that all objective functions are to be minimized, while the value function is to be maximized. This means that the preference of the DM will not decrease but will rather increase if the value of an objective function decreases, while all the other objective values remain unchanged (i.e., less is preferred to more). In this case, the solution maximizing  $v$  can be proven to be Pareto optimal. Regardless of the existence of a value function, it is usually assumed that less is preferred to more by the DM.

Instead of as a maximum of a value function, a final solution can be understood as a satisficing one. *Satisficing decision making* means that the DM does not intend to maximize any value function but tries to achieve certain aspirations (Sawaragi *et al.*, 1985). A Pareto optimal solution which satisfies all the aspirations of the DM is called a *satisficing solution*. In some rare cases, DMs may regard solutions satisficing even if they are not Pareto optimal. This may, for example, mean that not all relevant objectives are explicitly expressed. However, here we assume DMs to be rational and concentrate on Pareto optimal solutions.



Not only value functions but, in general, any preference model of a DM may be explicit or implicit in multiobjective optimization methods. Examples of local preference models include aspiration levels and different distance measures. During solution processes, various kinds of information can be solicited from the DM. *Aspiration levels*  $\bar{z}_i$  ( $i = 1, \dots, k$ ) are such desirable or acceptable levels in the objective function values that are of special interest and importance to the DM. The vector  $\bar{\mathbf{z}} \in \mathbf{R}^k$  consisting of aspiration levels is called a *reference point*.

According to the definition of Pareto optimality, moving from one Pareto optimal solution to another necessitates trading off. This is one of the basic concepts in multiobjective optimization. A *trade-off* reflects the ratio of change in the values of the objective functions concerning the increment of one objective function that occurs when the value of some other objective function decreases. For details, see, e.g., (Chankong and Haimes, 1983; Miettinen, 1999) and Chapters 2 and 9.

As mentioned earlier, it is sometimes easier to generate weakly Pareto optimal solutions than Pareto optimal ones (because some scalarizing functions produce weakly Pareto optimal solutions). There are different ways to get solutions that can be proven to be Pareto optimal. Benson (1978) has suggested to check the Pareto optimality of the decision vector  $\mathbf{x}^* \in S$  by solving the problem

$$\begin{aligned} & \text{maximize} && \sum_{i=1}^k \varepsilon_i \\ & \text{subject to} && f_i(\mathbf{x}) + \varepsilon_i = f_i(\mathbf{x}^*) \quad \text{for all } i = 1, \dots, k, \\ & && \varepsilon_i \geq 0 \quad \text{for all } i = 1, \dots, k, \\ & && \mathbf{x} \in S, \end{aligned} \tag{1.1}$$

where both  $\mathbf{x} \in \mathbf{R}^n$  and  $\boldsymbol{\varepsilon} \in \mathbf{R}_+^k$  are variables. If the optimal objective function value of (1.1) is zero, then  $\mathbf{x}^*$  can be proven to be Pareto optimal and if the optimal objective function value is finite and nonzero corresponding to a decision vector  $\mathbf{x}'$ , then  $\mathbf{x}'$  is Pareto optimal. Note that the equality constraints in (1.1) can be replaced by inequalities  $f_i(\mathbf{x}) + \varepsilon_i \leq f_i(\mathbf{x}^*)$ . However, we must point out that problem (1.1) is computationally badly conditioned because it has only one feasible solution ( $\varepsilon_i = 0$  for each  $i$ ) if  $\mathbf{x}^*$  is Pareto optimal and computational difficulties must be handled in practice, for example, using penalty functions. We shall introduce other ways to guarantee Pareto optimality in what follows in connection with some scalarizing functions.

Let us point out that in this chapter we do not concentrate on the theory behind multiobjective optimization, necessary and sufficient optimality conditions, duality results, etc. Instead, we refer, for example, to (Jahn, 2004; Luc, 1989; Miettinen, 1999; Sawaragi *et al.*, 1985) and references therein.

In the following sections, we briefly describe some methods for solving multiobjective optimization problems. We introduce several philosophies and ways of approaching the problem. As mentioned in the introduction, we concentrate on the classes devoted to no-preference methods, a posteriori methods

and a priori methods and remind that overlapping and combinations of classes are possible because no classification can fully cover the plethora of existing methods.

Methods in each class have their strengths and weaknesses and selecting a method to be used should be based on the desires and abilities of the DM as well as properties of the problem in question. Naturally, an analyst plays a crucial role when selecting a method because (s)he is supposed to know the properties of different methods available. Her/his recommendation should fit the needs and the psychological profile of the DM in question. In different methods, different types of information are given to the DM, the DM is assumed to specify preference information in different ways and different scalarizing functions are used. Besides the references given in each section, further details about the methods to be described, including proofs of theorems related to optimality, can be found in (Miettinen, 1999).

### 1.3 Basic Methods

Before we concentrate on the three classes of methods described in the introduction, we first discuss two well-known methods that can be called basic methods because they are so widely used. Actually, in many applications one can see them being used without necessarily recognizing them as multiobjective optimization methods. In other words, the difference between a modelling and an optimization phase are often blurred and these methods are used in order to convert the problem into a form where one objective function can be optimized with single objective solvers available. The reason for this may be that methods of single objective optimization are more widely known as those of multiobjective optimization. One can say that these two basic methods are the ones that first come to one's mind if there is a need to optimize multiple objectives simultaneously. Here we consider their strengths and weaknesses (which the users of these methods are not necessarily aware of) as well as show that many other (more advanced) approaches exist.

#### 1.3.1 Weighting Method

In the weighting method (see, e.g., (Gass and Saaty, 1955; Zadeh, 1963)), we solve the problem

$$\begin{aligned} & \text{minimize} && \sum_{i=1}^k w_i f_i(\mathbf{x}) \\ & \text{subject to} && \mathbf{x} \in S, \end{aligned} \tag{1.2}$$

where  $w_i \geq 0$  for all  $i = 1, \dots, k$  and, typically,  $\sum_{i=1}^k w_i = 1$ . The solution of (1.2) can be proven to be weakly Pareto optimal and, furthermore, Pareto optimal if we have  $w_i > 0$  for all  $i = 1, \dots, k$  or if the solution is unique (see, e.g., (Miettinen, 1999)).

The weighting method can be used as an a posteriori method so that different weights are used to generate different Pareto optimal solutions and then the DM is asked to select the most satisfactory one. Alternatively, the DM can be asked to specify the weights in which case the method is used as an a priori method.

As mentioned earlier, it is important in multiobjective optimization that Pareto optimal solutions are generated and that any Pareto optimal solution can be found. In this respect, the weighting method has a serious shortcoming. It can be proven that any Pareto optimal solution can be found by altering the weights only if the problem is convex. Thus, it may happen that some Pareto optimal solutions of nonconvex problems cannot be found no matter how the weights are selected. (Conditions under which the whole Pareto optimal set can be generated by the weighting method with positive weights are presented in (Censor, 1977).) Even though linear problems are not considered here, we should point out that despite MOLP problems being convex, the weighting method may not behave as expected even when solving them. This is because, when altering the weights, the method may jump from one vertex to another leaving intermediate solutions undetected. This is explained by the fact that linear solvers typically produce vertex solutions.

Unfortunately, people who use the weighting method do not necessarily know that that it does not work correctly for nonconvex problems. This is a serious and important aspect because it is not always easy to check the convexity in real applications if the problem is based, for example, on some simulation model or solving some systems like systems of partial differential equations. If the method is used in nonconvex problems for generating a representation of the Pareto optimal set, the DM gets a completely misleading impression about the feasible solutions available when some parts of the Pareto optimal set remain uncovered.

It is advisable to normalize the objectives with some scaling so that different magnitudes do not confuse the method. Systematic ways of perturbing the weights to obtain different Pareto optimal solutions are suggested, e.g., in (Chankong and Haimes, 1983). However, as illustrated by Das and Dennis (1997), an evenly distributed set of weights does not necessarily produce an evenly distributed representation of the Pareto optimal set, even if the problem is convex.

On the other hand, if the method is used as an a priori method, the DM is expected to be able to represent her/his preferences in the form of weights. This may be possible if we assume that the DM has a linear value function (which then corresponds to the objective function in problem (1.2)). However, in general, the role of the weights may be greatly misleading. They are often said to reflect the relative importance of the objective functions but, for example, Roy and Mousseau (1996) show that it is not at all clear what underlies this notion. Moreover, the relative importance of objective functions is usually understood globally, for the entire decision problem, while many practical applications show that the importance typically varies for

different objective function values, that is, the concept is meaningful only locally. (For more discussion on ordering objective functions by importance, see, e.g., (Podinovski, 1994).)

One more reason why the DM may not get satisfactory solutions with the weighting method is that if some of the objective functions correlate with each other, then changing the weights may not produce expected solutions at all but, instead, seemingly bad weights may result with satisfactory solutions and vice versa (see, e.g., (Steuer, 1986)). This is also shown in (Tanner, 1991) with an example originally formulated by P. Korhonen. With this example of choosing a spouse (where three candidates are evaluated with five criteria) it is clearly demonstrated how weights representing the preferences of the DM (i.e., giving the clearly biggest weight to the most important criterion) result with a spouse who is the worst in the criterion that the DM regarded as the most important one. (In this case, the undesired outcome may be explained by the compensatory character of the weighting method.)

In particular for MOLP problems, weights that produce a certain Pareto optimal solution are not necessarily unique and, thus, dramatically different weights may produce similar solutions. On the other hand, it is also possible that a small change in the weights may cause big differences in objective values. In all, we can say that it is not necessarily easy for the DM (or the analyst) to control the solution process with weights because weights behave in an indirect way. Then, the solution process may become an interactive one where the DM tries to guess such weights that would produce a satisfactory solution and this is not at all desirable because the DM can not be properly supported and (s)he is likely to get frustrated. Instead, in such cases it is advisable to use real interactive methods where the DM can better control the solution process with more intuitive preference information. For further details, see Chapter 2.

### 1.3.2 $\varepsilon$ -Constraint Method

In the  $\varepsilon$ -constraint method, one of the objective functions is selected to be optimized, the others are converted into constraints and the problem gets the form

$$\begin{aligned} & \text{minimize} && f_\ell(\mathbf{x}) \\ & \text{subject to} && f_j(\mathbf{x}) \leq \varepsilon_j \quad \text{for all } j = 1, \dots, k, \quad j \neq \ell, \\ & && \mathbf{x} \in S, \end{aligned} \tag{1.3}$$

where  $\ell \in \{1, \dots, k\}$  and  $\varepsilon_j$  are upper bounds for the objectives ( $j \neq \ell$ ). The method has been introduced in (Haimes *et al.*, 1971) and widely discussed in (Chankong and Haimes, 1983).

As far as optimality is concerned, the solution of problem (1.3) can be proven to always be weakly Pareto optimal. On the other hand,  $\mathbf{x}^* \in S$  can be proven to be Pareto optimal if and only if it solves (1.3) for every  $\ell = 1, \dots, k$ , where  $\varepsilon_j = f_j(\mathbf{x}^*)$  for  $j = 1, \dots, k$ ,  $j \neq \ell$ . In addition, a unique solution of

(1.3) can be proven to be Pareto optimal for any upper bounds. In other words, to ensure Pareto optimality we must either solve  $k$  different problems (and solving many problems for each Pareto optimal solution increases computational cost) or obtain a unique solution (which is not necessarily easy to verify). However, a positive fact is that finding any Pareto optimal solution does not necessitate convexity (as was the case with the weighting method). In other words, this method works for both convex and nonconvex problems.

In practice, it may be difficult to specify the upper bounds so that the resulting problem (1.3) has solutions, that is, the feasible region will not become empty. This difficulty is emphasized when the number of objective functions increases. Systematic ways of perturbing the upper bounds to obtain different Pareto optimal solutions are suggested in (Chankong and Haimes, 1983). In this way, the method can be used as an a posteriori method. Information about the ranges of objective functions in the Pareto optimal set is useful in perturbing the upper bounds. On the other hand, it is possible to use the method in an a priori way and ask the DM to specify the function to be optimized and the upper bounds. Specifying upper bounds can be expected to be easier for the DM than, for example, weights because objective function values are understandable as such for the DM. However, the drawback here is that if there is a promising solution really close to the bound but on the infeasible side, it will never be found. In other words, the bounds are a very stiff way of specifying preference information.

In what follows, we discuss three method classes described in the introduction and outline some methods belonging to each of them. Again, proofs of theorems related to optimality as well as further details about the methods can be found in (Miettinen, 1999).

## 1.4 No-Preference Methods

In no-preference methods, the opinions of the DM are not taken into consideration in the solution process. Thus, the problem is solved using some relatively simple method and the idea is to find some compromise solution typically ‘in the middle’ of the Pareto optimal set because there is no preference information available to direct the solution process otherwise. These methods are suitable for situations where there is no DM available or (s)he has no special expectations of the solution. They can also be used to produce a starting point for interactive methods.

One can question the name of no-preference methods because there may still exist an underlying preference model (e.g., the acceptance of a global criterion by a DM, like the one in the method to be described in the next subsection, can be seen as a preference model). However, we use the term of no-preference method in order to emphasize the fact that no explicit preferences from the DM are available and, thus, they cannot be used. These methods can also be referred to as methods of neutral preferences.

### 1.4.1 Method of Global Criterion

In the method of global criterion or compromise programming (Yu, 1973; Zeleny, 1973), the distance between some desirable reference point in the objective space and the feasible objective region is minimized. The analyst selects the reference point used and a natural choice is to set it as the ideal objective vector. We can use, for example, the  $L_p$ -metric or the Chebyshev metric (also known as the  $L_\infty$ -metric) to measure the distance to the ideal objective vector  $\mathbf{z}^*$  or the utopian objective vector  $\mathbf{z}^{**}$  (see definitions in Preface) and then we need to solve the problem

$$\begin{aligned} & \text{minimize} && \left( \sum_{i=1}^k |f_i(\mathbf{x}) - z_i^*|^p \right)^{1/p} \\ & \text{subject to} && \mathbf{x} \in S, \end{aligned} \tag{1.4}$$

(where the exponent  $1/p$  can be dropped) or

$$\begin{aligned} & \text{minimize} && \max_{i=1, \dots, k} [ |f_i(\mathbf{x}) - z_i^*| ] \\ & \text{subject to} && \mathbf{x} \in S, \end{aligned} \tag{1.5}$$

respectively. Note that if we here know the real ideal objective vector, we can ignore the absolute value signs because the difference is always positive (according to the definition of the ideal objective vector).

It is demonstrated, for example, in (Miettinen, 1999) that the choice of the distance metric affects the solution obtained. We can prove that the solution of (1.4) is Pareto optimal and the solution of (1.5) is weakly Pareto optimal. Furthermore, the latter can be proven to be Pareto optimal if it is unique.

Let us point out that if the objective functions have different magnitudes, the method works properly only if we scale the objective functions to a uniform, dimensionless scale. This means, for example, that we divide each absolute value term involving  $f_i$  by the corresponding range of  $f_i$  in the Pareto optimal set characterized by nadir and utopian objective vectors (defined in Preface), that is, by  $z_i^{\text{nad}} - z_i^{**}$  (for each  $i$ ). As the utopian objective vector dominates all Pareto optimal solutions, we use the utopian and not the ideal objective values in order to avoid dividing by zero in all occasions. (Connections of this method to utility or value functions are discussed in (Ballesteros and Romero, 1991).)

### 1.4.2 Neutral Compromise Solution

Another simple way of generating a solution without the involvement of the DM is suggested in (Wierzbicki, 1999) and referred to as a *neutral compromise solution*. The idea is to project a point located ‘somewhere in the middle’ of the ranges of objective values in the Pareto optimal set to become feasible. Components of such a point can be obtained as the average of the ideal (or utopian) and nadir values of each objective function. We can get a neutral compromise solution by solving the problem

$$\begin{aligned} & \text{minimize} && \max_{i=1,\dots,k} \left[ \frac{f_i(\mathbf{x}) - ((z_i^* + z_i^{\text{nad}})/2)}{z_i^{\text{nad}} - z_i^{**}} \right] \\ & \text{subject to} && \mathbf{x} \in S. \end{aligned} \tag{1.6}$$

As can be seen, this problem uses the utopian and the nadir objective vectors or other reliable approximations about the ranges of the objective functions in the Pareto optimal set for scaling purposes (in the denominator), as mentioned above. The solution is weakly Pareto optimal. We shall later return to scalarizing functions of this type later and discuss how Pareto optimality can be guaranteed. Naturally, the average in the numerator can be taken between components of utopian and nadir objective vectors, instead of the ideal and nadir ones.

## 1.5 A Posteriori Methods

In what follows, we assume that we have a DM available to take part in the solution process. A posteriori methods can be called *methods for generating Pareto optimal solutions*. Because there usually are infinitely many Pareto optimal solutions, the idea is to generate a representation of the Pareto optimal set and present it to the DM who selects the most satisfactory solution of them as the final one. The idea is that once the DM has seen an overview of different Pareto optimal solutions, it is easier to select the most preferred one. The inconveniences here are that the generation process is usually computationally expensive and sometimes in part, at least, difficult. On the other hand, it may be hard for the DM to make a choice from a large set of alternatives. An important question related to this is how to represent and display the alternatives to the DM in an illustrative way (Miettinen, 2003, 1999). Plotting the objective vectors on a plane is a natural way of displaying them only in the case of two objectives. In that case, the Pareto optimal set can be generated parametrically (see, e.g., (Benson, 1979; Gass and Saaty, 1955)). The problem becomes more complicated with more objectives. For visualizing sets of Pareto optimal solutions, see Chapter 8. Furthermore, visualization and approximation of Pareto optimal sets are discussed in Chapter 9. It is also possible to use so-called box-indices to represent Pareto optimal solutions to be compared by using a rough enough scale in order to let the DM easily recognize the main characteristics of the solutions at a glance (Miettinen *et al.*, 2008).

Remember that the weighting method and the  $\varepsilon$ -constraint method can be used as a posteriori methods. Next we outline some other methods in this class.

### 1.5.1 Method of Weighted Metrics

In the method of weighted metrics, we generalize the idea of the method of global criterion where the distance between some reference point and the

feasible objective region is minimized. The difference is that we can produce different solutions by weighting the metrics. The weighted approach is also sometimes called compromise programming (Zeleny, 1973).

Again, the solution obtained depends greatly on the distance measure used. For  $1 \leq p < \infty$ , we have a problem

$$\begin{aligned} & \text{minimize} && \left( \sum_{i=1}^k w_i (f_i(\mathbf{x}) - z_i^*)^p \right)^{1/p} \\ & \text{subject to} && \mathbf{x} \in S. \end{aligned} \quad (1.7)$$

The exponent  $1/p$  can be dropped. Alternatively, we can use a *weighted Chebyshev problem*

$$\begin{aligned} & \text{minimize} && \max_{i=1, \dots, k} [w_i (f_i(\mathbf{x}) - z_i^*)] \\ & \text{subject to} && \mathbf{x} \in S. \end{aligned} \quad (1.8)$$

Note that we have here ignored the absolute values assuming we know the global ideal (or utopian) objective vector. As far as optimality is concerned, we can prove that the solution of (1.7) is Pareto optimal if either the solution is unique or all the weights are positive. Furthermore, the solution of (1.8) is weakly Pareto optimal for positive weights. Finally, (1.8) has at least one Pareto optimal solution. On the other hand, convexity of the problem is needed in order to be able to prove that every Pareto optimal solution can be found by (1.7) by altering the weights. However, any Pareto optimal solution can be found by (1.8) assuming that the utopian objective vector  $\mathbf{z}^{**}$  is used as a reference point.

The objective function in (1.8) is nondifferentiable and, thus single objective optimizers using gradient information cannot be used to solve it. But if all the functions in the problem considered are differentiable, we can use an equivalent differentiable variant of (1.8) by introducing one more variable and new constraints of the form

$$\begin{aligned} & \text{minimize} && \alpha \\ & \text{subject to} && \alpha \geq w_i (f_i(\mathbf{x}) - z_i^*) \text{ for all } i = 1, \dots, k, \\ & && \mathbf{x} \in S, \end{aligned} \quad (1.9)$$

where both  $\mathbf{x} \in \mathbf{R}^n$  and  $\alpha \in \mathbf{R}$  are variables. With this formulation, single objective solvers assuming differentiability can be used.

Because problem (1.8) with  $\mathbf{z}^{**}$  seems a promising approach (as it can find any Pareto optimal solution), it would be nice to be able to avoid weakly Pareto optimal solutions. This can be done by giving a slight slope to the contours of the scalarizing function used (see, e.g., (Steuer, 1986)). In other words, we can formulate a so-called *augmented Chebyshev problem* in the form

$$\begin{aligned} & \text{minimize} && \max_{i=1, \dots, k} [w_i (f_i(\mathbf{x}) - z_i^{**})] + \rho \sum_{i=1}^k (f_i(\mathbf{x}) - z_i^{**}) \\ & \text{subject to} && \mathbf{x} \in S, \end{aligned} \quad (1.10)$$

where  $\rho$  is a sufficiently small positive scalar. Strictly speaking, (1.10) generates properly Pareto optimal solutions and any properly Pareto optimal



solution can be found (Kaliszewski, 1994). In other words, we are not actually able to find any Pareto optimal solution but only such solutions having a finite trade-off. However, when solving real-life problems, it is very likely that the DM is not interested in improperly Pareto optimal solutions after all. Here  $\rho$  corresponds to the bound for desirable or acceptable trade-offs (see definition of  $\rho$ -proper Pareto optimality in Section 1.2.1). Let us mention that an augmented version of the differentiable problem formulation (1.9) is obtained by adding the augmentation term (i.e., the term multiplied by  $\rho$ ) to the objective function  $\alpha$ .

Alternatively, it is possible to generate provably Pareto optimal solutions by solving two problems in a row. In other words, problem (1.8) is first solved and then another optimization problem is solved in the set of optimal solutions to (1.8). To be more specific, let  $\mathbf{x}^*$  be the solution of the first problem (1.8). Then the second problem is the following

$$\begin{aligned} & \text{minimize} && \sum_{i=1}^k (f_i(\mathbf{x}) - z_i^{**}) \\ & \text{subject to} && \max_{i=1, \dots, k} [w_i (f_i(\mathbf{x}) - z_i^{**})] \leq \max_{i=1, \dots, k} [w_i (f_i(\mathbf{x}^*) - z_i^{**})], \\ & && \mathbf{x} \in S. \end{aligned}$$

One should mention that the resulting problem may be computationally badly conditioned if the problem has only one feasible solution. With this so-called lexicographic approach it is possible to reach any Pareto optimal solution. Unfortunately, the computational cost increases because two optimization problems must be solved for each Pareto optimal solution (Miettinen *et al.*, 2006).

### 1.5.2 Achievement Scalarizing Function Approach

Scalarizing functions of a special type are called *achievement (scalarizing) functions*. They have been introduced, for example, in (Wierzbicki, 1982, 1986). These functions are based on an arbitrary reference point  $\bar{\mathbf{z}} \in \mathbf{R}^k$  and the idea is to project the reference point consisting of desirable aspiration levels onto the set of Pareto optimal solutions. Different Pareto optimal solutions can be produced with different reference points. The difference to the previous method (i.e., method of weighted metrics) is that no distance metric is used and the reference point does not have to be fixed as the ideal or utopian objective vector. Because of these characteristics, Pareto optimal solutions are obtained no matter how the reference point is selected in the objective space.

Achievement functions can be formulated in different ways. As an example we can mention the problem

$$\begin{aligned} & \text{minimize} && \max_{i=1, \dots, k} [w_i (f_i(\mathbf{x}) - \bar{z}_i)] + \rho \sum_{i=1}^k (f_i(\mathbf{x}) - \bar{z}_i) \\ & \text{subject to} && \mathbf{x} \in S, \end{aligned} \tag{1.11}$$

where  $\mathbf{w}$  is a fixed normalizing factor, for example,  $w_i = 1/(z_i^{\text{nad}} - z_i^{**})$  for all  $i$  and  $\rho > 0$  is an augmentation multiplier as in (1.10). And corresponding

to (1.10), we can prove that solutions of this problem are properly Pareto optimal and any properly Pareto optimal solution can be found. To be more specific, the solutions obtained are  $\rho$ -properly Pareto optimal (as defined in Section 1.2). If the augmentation term is dropped, the solutions can be proven to be weakly Pareto optimal. Pareto optimality can also be guaranteed and proven if the lexicographic approach described above is used. Let us point out that problem (1.6) uses an achievement scalarizing function where the reference point is fixed. The problem could be augmented as in (1.11).

Note that when compared to the method of weighted metrics, we do not use absolute value signs here in any case. No matter which achievement function formulation is used, the idea is the same: if the reference point is feasible, or actually to be more exact,  $\bar{z} \in Z + \mathbf{R}_+^k$ , then the minimization of the achievement function subject to the feasible region allocates slack between the reference point and Pareto optimal solutions producing a Pareto optimal solution. In other words, in this case the reference point is a Pareto optimal solution for the problem in question or it is dominated by some Pareto optimal solution. On the other hand, if the reference point is infeasible, that is,  $\bar{z} \notin Z + \mathbf{R}_+^k$ , then the minimization produces a solution that minimizes the distance between  $\bar{z}$  and  $Z$ . In both cases, we can say that we project the reference point on the Pareto optimal set. Discussion on how the projection direction can be varied in the achievement function can be found in (Luque *et al.*, 2009).

As mentioned before, achievement functions can be formulated in many ways and they can be based on so-called reservation levels, besides aspiration levels. For more details about them, we refer, for example, to (Wierzbicki, 1982, 1986, 1999, 2000) and Chapter 2.

### 1.5.3 Approximation Methods

During the years, many methods have been developed for approximating the set of Pareto optimal solutions in the MCDM literature. Here we do not go into their details. A survey of such methods is given in (Ruzika and Wiecek, 2005). Other approximation algorithms (not included there) are introduced in (Lotov *et al.*, 2004). For more information about approximation methods we also refer to Chapter 9.

## 1.6 A Priori Methods

In a priori methods, the DM must specify her/his preference information (for example, in the form of aspirations or opinions) before the solution process. If the solution obtained is satisfactory, the DM does not have to invest too much time in the solution process. However, unfortunately, the DM does not necessarily know beforehand what it is possible to attain in the problem and how realistic her/his expectations are. In this case, the DM may be disappointed

at the solution obtained and may be willing to change one's preference information. This easily leads to a desire of using an interactive approach (see Chapter 2). As already mentioned, the basic methods introduced earlier can be used as a priori methods. It is also possible to use the achievement scalarizing function approach as an a priori method where the DM specifies the reference point and the Pareto optimal solution closest to it is generated. Here we briefly describe three other methods.

### 1.6.1 Value Function Method

The value function method (Keeney and Raiffa, 1976) was already mentioned in Section 1.2.2. It is an excellent method if the DM happens to know an explicit mathematical formulation for the value function and if that function can capture and represent all her/his preferences. Then the problem to be solved is

$$\begin{aligned} & \text{maximize} && v(\mathbf{f}(\mathbf{x})) \\ & \text{subject to} && \mathbf{x} \in S. \end{aligned}$$

Because the value function provides a complete ordering in the objective space, the best Pareto optimal solution is found in this way. Unfortunately, it may be difficult, if not impossible, to get that mathematical expression of  $v$ . For example, in (deNeufville and McCord, 1984), the inability to encode the DM's underlying value function reliably is demonstrated by experiments. On the other hand, the value function can be difficult to optimize because of its possible complicated nature. Finally, even if it were possible for the DM to express her/his preferences globally as a value function, the resulting preference structure may be too simple since value functions cannot represent intransitivity or incomparability. In other words, the DM's preferences must satisfy certain conditions (like consistent preferences) so that a value function can be defined on them. For more discussion see, for example, (Miettinen, 1999).

### 1.6.2 Lexicographic Ordering

In lexicographic ordering (Fishburn, 1974), the DM must arrange the objective functions according to their absolute importance. This means that a more important objective is infinitely more important than a less important objective. After the ordering, the most important objective function is minimized subject to the original constraints. If this problem has a unique solution, it is the final one and the solution process stops. Otherwise, the second most important objective function is minimized. Now, a new constraint is introduced to guarantee that the most important objective function preserves its optimal value. If this problem has a unique solution, the solution process stops. Otherwise, the process goes on as above. (Let us add that computationally it is not trivial to check the uniqueness of solutions. Then the next problem must be solved just to be sure. However, if the next problem has a unique solution, the problem is computationally badly conditioned, as discussed earlier.)

The solution of lexicographic ordering can be proven to be Pareto optimal. The method is quite simple and one can claim that people often make decisions successively. However, the DM may have difficulties in specifying an absolute order of importance. Besides, the method is very rough and it is very likely that the process stops before less important objective functions are taken into consideration. This means that all the objectives that were regarded as relevant while formulating the problem are not taken into account at all, which is questionable.

The notion of absolute importance is discussed in (Roy and Mousseau, 1996). Note that lexicographic ordering does not allow a small increment of an important objective function to be traded off with a great decrement of a less important objective. Yet, the DM might find this kind of trading off appealing. If this is the case, lexicographic ordering is not likely to produce a satisficing solution.

### 1.6.3 Goal Programming

Goal programming is one of the first methods expressly created for multiobjective optimization (Charnes *et al.*, 1955; Charnes and Cooper, 1961). It has been originally developed for MOLP problems (Ignizio, 1985).

In goal programming, the DM is asked to specify aspiration levels  $\bar{z}_i$  ( $i = 1, \dots, k$ ) for the objective functions. Then, deviations from these aspiration levels are minimized. An objective function jointly with an aspiration level is referred to as a *goal*. For minimization problems, goals are of the form  $f_i(\mathbf{x}) \leq \bar{z}_i$  and the aspiration levels are assumed to be selected so that they are not achievable simultaneously. After the goals have been formed, the *deviations*  $\delta_i = \max[0, f_i(\mathbf{x}) - \bar{z}_i]$  of the objective function values are minimized.

The method has several variants. In the *weighted goal programming approach* (Charnes and Cooper, 1977), the weighted sum of the deviations is minimized. This means that in addition to the aspiration levels, the DM must specify positive weights. Then we solve a problem

$$\begin{aligned} & \text{minimize} && \sum_{i=1}^k w_i \delta_i \\ & \text{subject to} && f_i(\mathbf{x}) - \delta_i \leq \bar{z}_i \quad \text{for all } i = 1, \dots, k, \\ & && \delta_i \geq 0 \quad \text{for all } i = 1, \dots, k, \\ & && \mathbf{x} \in S, \end{aligned} \tag{1.12}$$

where  $\mathbf{x} \in \mathbf{R}^n$  and  $\delta_i$  ( $i = 1, \dots, k$ ) are the variables.

On the other hand, in the *lexicographic goal programming approach*, the DM must specify a lexicographic order for the goals in addition to the aspiration levels. After the lexicographic ordering, the problem with the deviations as objective functions is solved lexicographically subject to the constraints of (1.12) as explained in Section 1.6.2. It is also possible to use a combination of the weighted and the lexicographic approaches. In this case, several

objective functions may belong to the same class of importance in the lexicographic order. In each priority class, a weighted sum of the deviations is minimized. Let us also mention a so-called *min-max goal programming approach* (Flavell, 1976) where the maximum of deviations is minimized and meta-goal programming (Rodríguez Uría *et al.*, 2002), where different variants of goal programming are incorporated.

Let us next discuss optimality. The solution of a goal programming problem can be proven to be Pareto optimal if either the aspiration levels form a Pareto optimal reference point or all the variables  $\delta_i$  have positive values at the optimum. In other words, if the aspiration levels form a feasible point, the solution is equal to that reference point which is not necessarily Pareto optimal. We can say that the basic formulation of goal programming presented here works only if the aspiration levels are overoptimistic enough. Pareto optimality of the solutions obtained is discussed, for example, in (Jones *et al.*, 1998).

Goal programming is a very widely used and popular solution method. Goal-setting is an understandable and easy way of making decisions. The specification of the weights or the lexicographic ordering may be more difficult (the weights have no direct physical meaning). For further details, see (Romero, 1991). Let us point out that goal programming is related to the achievement scalarizing function approach (see Section 1.5.2) because they both are based on reference points. The advantage of the latter is that it is able to produce Pareto optimal solutions independently of how the reference point is selected.

Let us finally add that goal programming has been used in a variety of further developments and modifications. Among others, goal programming is related to some fuzzy multiobjective optimization methods where fuzzy sets are used to express degrees of satisfaction from the attainment of goals and from satisfaction of soft constraints (Rommelfanger and Slowinski, 1998). Some more applications of goal programming will be discussed in further chapters of this book.

## 1.7 Summary

In this section we summarize some of the properties of the nine methods discussed so far. We provide a collection of different properties in Figure 1.2. We pay attention to the class the method can be regarded to belong to as well as properties of solutions obtained. We also briefly comment the format of preference information used. In some connections, we use the notation (X) to indicate that the statement or property is true under assumptions mentioned when describing the method.

	weighting method	e-constraint method	method of global criterion	neutral compromise solution	method of weighted metrics	achievement scalarizing function	value function method	lexicographic ordering	goal programming
no-preference method			×	×					
a priori method	×	×				×	×	×	×
a posteriori method	×	×			×	×			
can find any Pareto optimal solution		×			(×)	×			×
solution always Pareto optimal	(×)	(×)	(×)	(×)	(×)	(×)	×	×	
type of preference information									
weights	×								(×)
bounds		×							
reference point						×			×
value function							×		
lexicographic order								×	(×)

Fig. 1.2. Summary of some properties of the methods described.

## 1.8 Conclusions

The aim of this chapter has been to briefly describe some basics of MCDM methods. For this, we have concentrated on some noninteractive methods developed for multiobjective optimization. A large variety of methods exists and it is impossible to cover all of them. In this chapter, we have concentrated on methods where the DM either specifies no preferences or specifies them after or before the solution process. The methods can be combined, hybridized and further developed in many ways, for example, with evolutionary algorithms. Other chapters of this book will discuss possibilities of such developments more.

None of the methods can be claimed to be superior to the others in every aspect. When selecting a solution method, the specific features of the problem

to be solved must be taken into consideration. In addition, the opinions and abilities of the DM are important. The theoretical properties of the methods can rather easily be compared but, in addition, practical applicability also plays an important role in the selection of an appropriate method. One can say that selecting a multiobjective optimization method is a problem with multiple objectives itself! Some methods may suit some problems and some DMs better than others. A decision tree is provided in (Miettinen, 1999) for easing the selection. Specific methods for different areas of application that take into account the characteristics of the problems may also be useful.

### Acknowledgements

I would like to give my most sincere thanks to Professors Alexander Lotov, Francisco Ruiz and Andrzej Wierzbicki for their valuable comments that improved this chapter. This work was partly supported by the Foundation of the Helsinki School of Economics.

### References

- Ballestero, E., Romero, C.: A theorem connecting utility function optimization and compromise programming. *Operations Research Letters* 10(7), 421–427 (1991)
- Benayoun, R., de Montgolfier, J., Tergny, J., Laritchev, O.: Programming with multiple objective functions: Step method (STEM). *Mathematical Programming* 1(3), 366–375 (1971)
- Benson, H.P.: Existence of efficient solutions for vector maximization problems. *Journal of Optimization Theory and Application* 26(4), 569–580 (1978)
- Benson, H.P.: Vector maximization with two objective functions. *Journal of Optimization Theory and Applications* 28(3), 253–257 (1979)
- Censor, Y.: Pareto optimality in multiobjective problems. *Applied Mathematics and Optimization* 4(1), 41–59 (1977)
- Chankong, V., Haimes, Y.Y.: *Multiobjective Decision Making: Theory and Methodology*. Elsevier Science Publishing, New York (1983)
- Charnes, A., Cooper, W.W.: *Management Models and Industrial Applications of Linear Programming*, vol. 1. Wiley, New York (1961)
- Charnes, A., Cooper, W.W.: Goal programming and multiple objective optimization; part 1. *European Journal of Operational Research* 1(1), 39–54 (1977)
- Charnes, A., Cooper, W.W., Ferguson, R.O.: Optimal estimation of executive compensation by linear programming. *Management Science* 1(2), 138–151 (1955)
- Cohon, J.L.: Multicriteria programming: Brief review and application. In: Gero, J.S. (ed.) *Design Optimization*, pp. 163–191. Academic Press, London (1985)
- Das, I., Dennis, J.E.: A closer look at drawbacks of minimizing weighted sums of objectives for Pareto set generation in multicriteria optimization problems. *Structural Optimization* 14(1), 63–69 (1997)
- Deb, K., Chaudhuri, S., Miettinen, K.: Towards estimating nadir objective vector using evolutionary approaches. In: Keijzer, M., et al. (eds.) *Proceedings of the 8th Annual Genetic and Evolutionary Computation Conference (GECCO-2006)*, Seattle, vol. 1, pp. 643–650. ACM Press, New York (2006)

- deNeufville, R., McCord, M.: Unreliable measurement of utility: Significant problems for decision analysis. In: Brans, J.P. (ed.) *Operational Research '84*, pp. 464–476. Elsevier, Amsterdam (1984)
- Edgeworth, F.Y.: *Mathematical Psychics: An Essay on the Application of Mathematics to the Moral Sciences*. C. Kegan Paul & Co., London (1881), University Microfilms International (Out-of-Print Books on Demand) (1987)
- Fandel, G.: Group decision making: Methodology and applications. In: Bana e Costa, C. (ed.) *Readings in Multiple Criteria Decision Aid*, pp. 569–605. Berlin (1990)
- Fishburn, P.C.: Lexicographic orders, utilities and decision rules: A survey. *Management Science* 20(11), 1442–1471 (1974)
- Flavell, R.B.: A new goal programming formulation. *Omega* 4(6), 731–732 (1976)
- Gass, S., Saaty, T.: The computational algorithm for the parametric objective function. *Naval Research Logistics Quarterly* 2, 39–45 (1955)
- Geoffrion, A.M.: Proper efficiency and the theory of vector maximization. *Journal of Mathematical Analysis and Applications* 22(3), 618–630 (1968)
- Haimes, Y.Y., Lasdon, L.S., Wismer, D.A.: On a bicriterion formulation of the problems of integrated system identification and system optimization. *IEEE Transactions on Systems, Man, and Cybernetics* 1, 296–297 (1971)
- Hwang, C.-L., Lin, M.-J.: *Group Decision Making under Multiple Criteria: Methods and Applications*. Springer, New York (1987)
- Hwang, C.L., Masud, A.S.M.: *Multiple Objective Decision Making – Methods and Applications: A State-of-the-Art Survey*. Springer, Berlin (1979)
- Ignizio, J.P.: *Introduction to Linear Goal Programming*. Sage Publications, Beverly Hills (1985)
- Jahn, J.: *Vector Optimization*. Springer, Berlin (2004)
- Jones, D.F., Tamiz, M., Mirrazavi, S.K.: Intelligent solution and analysis of goal programmes: the GPSYS system. *Decision Support Systems* 23(4), 329–332 (1998)
- Kaliszewski, I.: *Quantitative Pareto Analysis by Cone Separation Technique*. Kluwer, Dordrecht (1994)
- Keeney, R.L., Raiffa, H.: *Decisions with Multiple Objectives: Preferences and Value Tradeoffs*. Wiley, Chichester (1976)
- Koopmans, T.: Analysis and production as an efficient combination of activities. In: Koopmans, T. (ed.) *Activity Analysis of Production and Allocation: Proceedings of a Conference*, pp. 33–97. Wiley, New York (1951), Yale University Press, London (1971)
- Korhonen, P., Salo, S., Steuer, R.E.: A heuristic for estimating nadir criterion values in multiple objective linear programming. *Operations Research* 45(5), 751–757 (1997)
- Kuhn, H., Tucker, A.: Nonlinear programming. In: Neyman, J. (ed.) *Proceedings of the Second Berkeley Symposium on Mathematical Statistics and Probability*, pp. 481–492. University of California Press, Berkeley (1951)
- Lotov, A.V., Bushenkov, V.A., Kamenev, G.K.: *Interactive Decision Maps. Approximation and Visualization of Pareto Frontier*. Kluwer Academic Publishers, Boston (2004)
- Luc, D.T.: *Theory of Vector Optimization*. Springer, Berlin (1989)
- Luque, M., Miettinen, K., Eskelinen, P., Ruiz, F.: Incorporating preference information in interactive reference point methods for multiobjective optimization. *Omega* 37(2), 450–462 (2009)



- Makarov, E.K., Rachkovski, N.N.: Unified representation of proper efficiency by means of dilating cones. *Journal of Optimization Theory and Applications* 101(1), 141–165 (1999)
- Marler, R., Arora, J.: Survey of multi-objective optimization methods for engineering. *Structural and Multidisciplinary Optimization* 26(6), 369–395 (2004)
- Miettinen, K.: *Nonlinear Multiobjective Optimization*. Kluwer Academic Publishers, Boston (1999)
- Miettinen, K.: Graphical illustration of Pareto optimal solutions. In: Tanino, T., Tanaka, T., Inuiguchi, M. (eds.) *Multi-Objective Programming and Goal Programming: Theory and Applications*, pp. 197–202. Springer, Berlin (2003)
- Miettinen, K., Mäkelä, M.M., Kaario, K.: Experiments with classification-based scalarizing functions in interactive multiobjective optimization. *European Journal of Operational Research* 175(2), 931–947 (2006)
- Miettinen, K., Molina, J., González, M., Hernández-Díaz, A., Caballero, R.: Using box indices in supporting comparison in multiobjective optimization. *European Journal of Operational Research*, to appear (2008), doi:10.1016/j.ejor.2008.05.103
- Pareto, V.: *Cours d'Economie Politique*. Rouge, Lausanne (1896)
- Pareto, V.: *Manuale di Economia Politica*. Piccola Biblioteca Scientifica, Milan (1906), Translated into English by Schwier, A.S., *Manual of Political Economy*, MacMillan, London (1971)
- Podinovski, V.V.: Criteria importance theory. *Mathematical Social Sciences* 27(3), 237–252 (1994)
- Rodríguez Uría, M., Caballero, R., Ruiz, F., Romero, C.: Meta-goal programming. *European Journal of Operational Research* 136(2), 422–429 (2002)
- Romero, C.: *Handbook of Critical Issues in Goal Programming*. Pergamon Press, Oxford (1991)
- Rommelfanger, H., Slowinski, R.: Fuzzy linear programming with single or multiple objective functions. In: Slowinski, R. (ed.) *Fuzzy Sets in Decision Analysis, Operations Research and Statistics*, pp. 179–213. Kluwer Academic Publishers, Boston (1998)
- Rosenthal, R.E.: Principles of Multiobjective Optimization. *Decision Sciences* 16(2), 133–152 (1985)
- Roy, B., Mousseau, V.: A theoretical framework for analysing the notion of relative importance of criteria. *Journal of Multi-Criteria Decision Analysis* 5(2), 145–159 (1996)
- Ruzika, S., Wiecek, M.M.: Approximation methods in multiobjective programming. *Journal of Optimization Theory and Applications* 126(3), 473–501 (2005)
- Sawaragi, Y., Nakayama, H., Tanino, T.: *Theory of Multiobjective Optimization*. Academic Press, Orlando (1985)
- Steuer, R.E.: *Multiple Criteria Optimization: Theory, Computation, and Application*. Wiley, New York (1986)
- Tanner, L.: Selecting a text-processing system as a qualitative multiple criteria problem. *European Journal of Operational Research* 50(2), 179–187 (1991)
- Vincke, P.: *Multicriteria Decision-Aid*. Wiley, Chichester (1992)
- Weistroffer, H.R.: Careful usage of pessimistic values is needed in multiple objectives optimization. *Operations Research Letters* 4(1), 23–25 (1985)
- Wierzbicki, A.P.: A mathematical basis for satisficing decision making. *Mathematical Modelling* 3, 391–405 (1982)

- Wierzbicki, A.P.: On the completeness and constructiveness of parametric characterizations to vector optimization problems. *OR Spectrum* 8(2), 73–87 (1986)
- Wierzbicki, A.P.: Reference point approaches. In: Gal, T., Stewart, T.J., Hanne, T. (eds.) *Multicriteria Decision Making: Advances in MCDM Models, Algorithms, Theory, and Applications*, pp. 9-1–9-39. Kluwer, Boston (1999)
- Wierzbicki, A.P.: Reference point methodology. In: Wierzbicki, A.P., Makowski, M., Wessels, J. (eds.) *Model-Based Decision Support Methodology with Environmental Applications*, pp. 71–89. Kluwer Academic Publishers, Dordrecht (2000)
- Yu, P.L.: A class of solutions for group decision problems. *Management Science* 19(8), 936–946 (1973)
- Zadeh, L.: Optimality and non-scalar-valued performance criteria. *IEEE Transactions on Automatic Control* 8, 59–60 (1963)
- Zeleny, M.: Compromise programming. In: Cochrane, J.L., Zeleny, M. (eds.) *Multiple Criteria Decision Making*, pp. 262–301. University of South Carolina, Columbia, SC (1973)