Chapter 1

A Survey on Multiobjective Optimization

This chapter is devoted to the basic concepts of multiobjective optimization and introduces the main methods and ideas developed in the field of multiobjective optimization on the multiple criteria decision making side (including both noninteractive and interactive approaches) and on the evolutionary multiobjective optimization side. The chapter lays a foundation for the rest of the thesis and should also allow newcomers to the field to get familiar with the topic.

1.1 Introduction

Many decision and planning problems involve multiple conflicting objectives that should be considered simultaneously. Such problems are generally known as Multiple Criteria Decision Making (MCDM) problems. Here we focus on Multiobjective Optimization (MOO). In general, a Multiobjective Optimization Problem (MOP) has a set of solutions. These solutions are known as nondominated, efficient, noninferior or Pareto-optimal solutions. MOPs have been intensively studied for several decades and the research is based on the theoretical background laid, for example, in (Kuhn and Tucker, 1951).

Typically, in the MCDM literature, the idea of solving a MOP 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.

The methods developed for MOPs are classified into four 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*. 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. 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.

The three classes, where either no DM takes part in the solution process or (s)he expresses preference relations before or after the process, belong to *noninteractive methods*. The fourth class devoted to *interactive methods* is the most extensive class of methods. In interactive approaches, an iterative solution algorithm is formed and repeated several times. After each iteration, some information is given to the DM and (s)he is asked to specify preference information.

Evolutionary Multiobjective Optimization (EMO), in its current state, is an established field of research and application with a biannual conference series running successfully since 2001. For solving MOPs, EMO procedures attempt to find a set of well-distributed Pareto-optimal points, so that an idea of the extent and shape of the Pareto-optimal front can be obtained. One of the major current research thrusts is to combine EMO procedures with other MCDM tools so as to develop hybrid and interactive MOO algorithms for finding a set of trade-off optimal solutions and then choose a preferred solution for implementation.

The chapter is organized as follows. The fundamental concepts and definitions of MOO are introduced in Section 1.2. Section 1.3 presents the noninteractive approaches while interactive approaches are discussed in Section 1.4. Finally, the principles of EMO are given in Section 1.5.

1.2 Basic Concepts

1.2.1 Main Terminology and Notations

We handle MOP in the form

$$\text{MOP:} \quad \begin{array}{ll} \text{minimize} & \{ f_1(\mathbf{x}), f_2(\mathbf{x}),..., f_k(\mathbf{x})\} \\ \text{subject to} & \mathbf{x} \in X \end{array} \tag{1.1}$$

Involving k ($\geq$ 2) conflicting *objective functions* $f_i : \mathbb{R}^n \rightarrow \mathbb{R}$ that wanted to be minimized simultaneously. The *decision* (*variable*) *vectors* $\mathbf{x} = (x_1, x_2,..., x_n)^T$ belong to the nonempty *feasible region* $X \subset \mathbb{R}^n$ formed by a set of constraints. In this general problem formulation the types of constraints forming the feasible region are not fixed. The *objective vectors* are images of decision vectors and written as $\mathbf{z} = \mathbf{f}(\mathbf{x})$ or $(z_1, z_2,\ldots, z_k) = (f_1(\mathbf{x}), f_2(\mathbf{x}),..., f_k(\mathbf{x}))$. Furthermore, the image of the feasible region in the objective space is called a *feasible objective region* $Z = \mathbf{f}(X) \subset \mathbb{R}^k$.

Definition 1.1: A decision vector $\bar{\mathbf{x}} \in X$ is called *Pareto-optimal* solution for problem (1.1) if there does not exist another $\mathbf{x} \in X$ such that $f_i(\mathbf{x}) \leq f_i(\bar{\mathbf{x}})$, i = 1,..., k and $f_i(\mathbf{x}) < f_i(\bar{\mathbf{x}})$ for at least one index i.

An objective vector is Pareto-optimal if the corresponding decision vector is Pareto-optimal. All Pareto-optimal points make up a front when viewed together on the objective space. This front is called *Pareto-optimal front*.

The ranges of the Pareto-optimal solutions in the feasible objective region provide valuable information about the problem considered:

- Lower bounds of the Pareto-optimal set are available in the *ideal objective vector* $\mathbf{z}^{\mathbf{ideal}} \in \mathbb{R}^k$. The component z_i^{ideal} is obtained by minimizing the objective function z_i subject to the feasible region X.
- A vector strictly better than $\mathbf{z}^{\mathbf{ideal}}$ is called *utopian objective vector* $\mathbf{z}^{\mathbf{utop}}$. In practice, we set $z_i^{utop} = z_i^{ideal} - \varepsilon$ for $i = 1,..., k$, where ε is some small positive scalar.
- The upper bounds of the Pareto-optimal set, that is, the components of a *nadir objective vector* $\mathbf{z}^{\mathbf{nadir}}$. The component z_i^{nadir} is obtained by maximizing the objective function z_i subject to Pareto-optimal set.

1.2.2 The Solution Process and Its Elements

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 (1.1) defined above is called a *solution process*. 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 MOP 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*.

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: $\mathbb{R}^k \rightarrow \mathbb{R}$ (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.

Not only value functions but, in general, any preference model of a DM may be explicit or implicit in MOO 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* z_i^* ($i = 1, \ldots, k$) are such desirable or acceptable levels in the objective function values that are of special interest and importance to the DM. The vector $\mathbf{z}^* \in \mathbb{R}^k$ 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 MOO. 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).

1.3 Noninteractive Approaches

In this section, we concentrate on the three classes of noninteractive approaches which are devoted to no-preference methods, a posteriori methods and a priori methods and note 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. 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 below, further details about the methods to be described, including proofs of theorems related to optimality, can be found in (Miettinen, 1999).

1.3.1 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 MOO methods. In other words, the difference between a modeling 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 MOO. One can say that these two basic methods are the ones that first come to one's mind