UNIVERSITÄT JYVÄSKYLÄ
MATHEMATISCHES INSTITUT

BERICHT 60

UNIVERSITY OF JYVÄSKYLÄ
DEPARTMENT OF MATHEMATICS

REPORT 60

ON THE METHODOLOGY OF MULTIOBJECTIVE OPTIMIZATION WITH APPLICATIONS

KAISA MIETTINEN

JYVÄSKYLÄ
1994

UNIVERSITÄT JYVÄSKYLÄ
MATHEMATISCHES INSTITUT
BERICHT 60

UNIVERSITY OF JYVÄSKYLÄ
DEPARTMENT OF MATHEMATICS
REPORT 60

ON THE METHODOLODY OF MULTIOBJECTIVE OPTIMIZATION WITH APPLICATIONS

KAISA MIETTINEN

To be presented, with the permission of the Faculty of Mathematics and Natural Sciences of the University of Jyväskylä, for public criticism in Auditorium S 212 of the University, on July 29th, 1994, at 12 o'clock noon.

JYVÄSKYLÄ
1994

Editor: Pertti Mattila
University of Jyväskylä
Department of Mathematics
P.O. Box 35
FIN-40351 Jyväskylä
Finland

URN:ISBN:978-951-39-9052-7
ISBN 978-951-39-9052-7 (PDF)
ISSN 0075-4641

Jyväskylän yliopisto, 2022

ISBN 951-34-0316-5
ISSN 0075-4641

Copyright © 1994, by Kaisa
Miettinen and University of Jyväskylä

Jyväskylän yliopistopaino
Jyväskylä 1994

Acknowledgements

I wish to express my gratitude to several individuals. First, it is in order to thank Professor Pekka Neittaanmäki, who originally proposed multiobjective optimization for my research subject. I want to take this opportunity to give special thanks to Professors Pekka Korhonen and Aimo Törn for the careful reading of the manuscript and for their valuable comments.

I am extremely grateful to Doctor Marko Mäkelä for his cooperation and comments and for the innumerable number of profitable discussions we have had. I also appreciate his efforts in reading parts of the manuscript. His expertise and computer programs in nondifferentiable optimization have been highly valuable. Furthermore, I would like to thank Doctor Timo Männikkö for providing his expert knowledge and the software on the continuous casting process.

I am indebted to Mrs. Tuula Blåfield for her trouble in the linguistic proofreading and to Doctor Ari Lehtonen for making me familiar with several peculiarities of $\mathcal{AMS}$-TeX.

My thanks go to the University of Jyväskylä and the Academy of Finland for financial support.

With sincere gratitude I want to acknowledge the support of my parents, Anna-Liisa and Kauko. Finally, to my husband, Kari-Pekka, goes my deepest appreciation for his continuous encouragement and patience. He has also earned my genuine gratitude for his help in discovering the name for NIMBUS and for developing the user interface to it.

Kaisa Miettinen

Jyväskylä, June 1994

Contents

Preface

The origin of this presentation is optimization – searching for the optimal solution, selection and decision. Optimization problems occur, for example, in everyday life when buying things (like a house or a television) or selecting a means of transport to work, in engineering when designing focussing systems, spacecraft structures, bridges, robots, or camera lenses, in economics when planning production systems or pricing products, and in environmental control when managing pollution problems. What is common in most of those optimization problems is that they have several at least partly conflicting criteria to be taken into consideration at the same time. A number of different goals are desired to attain simultaneously. In this case, methods of traditional (single objective) optimization are not enough, but we need new ways of thinking, new concepts, and new methods.

A general term in this presentation for problems with multiple criteria is *multiple criteria optimization problems*. They can be divided into two distinct parts according to [MacCrimmon, 1973]. The classes are called multiattribute decision analysis and multiobjective optimization, according to the properties of the feasible region. In *multiattribute decision analysis* the set of feasible alternatives is discrete, predetermined and finite. Specific and current examples of multiattribute problems are the selection of the locations of power plants and dumping places. In *multiobjective optimization problems* the feasible alternatives are not explicitly known in advance. There are an infinite number of them and they are represented by decision variables restricted by constraint functions. These problems can be called continuous. In this case, one has to generate the alternatives before they can be valuated. A short collection of history, basic ideas and references handling both of the classes has been gathered in [Dyer, Fishburn, Steuer, Wallenius, Zionts, 1992].

We have devoted this presentation solely to multiobjective optimization. The field of multiple criteria optimization is so extensive that there is a reason to restrict the handling. As far as multiattribute decision analysis is concerned we refer to the monographs [Keeney, Raiffa, 1976] and [Hwang, Yoon, 1981]. More references with seventeen major methods in the area with simple examples can be found in the latter monograph. A review of research in multiobjective optimization and multiattribute decision analysis, problems and future directions has been collected in the paper [Korhonen, Moskowitz, Wallenius, 1992]. It contains short descriptions of many concepts and areas of multiple criteria optimization and decision making, which are not included in this presentation.

A large number of application areas of multiobjective optimization have been presented in the literature. A good conception of the possibilities and the importance of multiobjective optimization can be comprehended from the fact that over 500 papers describing different applications have been listed in [White, 1990] (from the period 1955–1986). They cover, for example, problems of agriculture, banking, health service, energy, industry, water and wildlife.

Even though we have restricted ourselves to handling only multiobjective optimization problems, it is still a wide research area and we must further cut off several topics. Such special types of multiobjective optimization problems are those where the feasible decision variables must have integer values (multiple criteria integer programming) or 0-1 values, trajectory optimization problems (where the multiple criteria have multiple observation points), multiple criteria networks (e.g., best path problems, where several parameters, such as, cost and distance, are attached to each arc), multiple criteria transportation networks (handled in [Current, Min, 1986] and [Current, Marsh, 1993]) and multiple criteria dynamic programming (treated in [Li, Haimes, 1989]).

One more topic not handled here are problems where there are uncertainties involved. They can be divided into stochastic and fuzzy problems. In stochastic programming it is usually assumed that uncertainty is due to a lack of information about prevailing states and that this uncertainty only concerns the occurrence of states and not the definition of states, results, or criteria themselves. A problem containing random variables on some probability space as coefficients is called a stochastic programming problem (handled, e.g., in the monographs [Stancu-Minasian, 1984] and [Guddat, Guerra Vasquez, Tammer, Wendler, 1985]). When decision making takes place in an environment in which the goals, constraints, and consequences of possible actions are not precisely known, it is called "decision in fuzzy environments" (handled, e.g., in [Kacprzyk, Orlovski, 1987]). We assume here that the problems are deterministic, that is, the outcome of any feasible decision vector is known for certain.

The aim of this presentation has been twofold. The first of the basic objectives has been to provide an extensive, up-to-date, self-contained and consistent survey and review of the literature and the state-of-the-art around multiobjective optimization. The second aim has been to create new methodology for nondifferentiable multiobjective optimization. Moreover, we propose a new way to solve certain state-constrained problems of optimal control. By applying the new algorithms, improved solutions are obtained for them.

The amount of the literature on multiobjective optimization is immense. In addition to several monographs, a lot of journal papers and conference proceedings have been published. The most important source when searching for them has been the MathSci Disc database on CD-ROM. For practical reasons the searches have been limited to include English material and the main interest has been in publications after the year 1980. Almost 1000 papers and monographs have been examined while preparing this presentation. About half of them are cited and listed in the bibliography. The monographs [Cohon, 1978], [Hwang, Masud, 1979], [Chankong, Haimes, 1983(b)], [Osyczka, 1984], [Sawaragi, Nakayama, Tanino, 1985], [Yu, 1985] and [Steuer, 1986] have given a general basis for this presentation and they provide an extensive view on the area of multiobjective optimization. Further, noteworthy monographs on the topic are [Rietveld, 1980], [Zeleny, 1982] and [Vincke, 1992]. A significant part of the latter reference deals with multiattribute decision analysis, though. The monograph [Ringuest, 1992] mostly treats behavioural aspects of multiobjective optimization. The monographs [Jahn, 1986(a)] and [Luc, 1989] handle theoretical aspects extensively.

Theory and methods for multiobjective optimization have mainly been developed during the last three decades. Here we do not go deeply into the history as the origin and the achievements of this research field from 1776 to 1960 have been widely handled in [Stadler, 1979].

At the beginning of this presentation, important concepts and definitions of multiobjective optimization are put forward. In addition, several theoretical aspects are handled. For example, analogous optimality conditions for differentiable and nondifferentiable problems are considered. The whole presentation through we keep to problems involving only finite-dimensional Euclidean spaces. In [Dauer, Stadler, 1986], there is a survey on multiobjective optimization in infinite-dimensional spaces.

The state-of-the-art of the method development is portrayed by describing a number of different methods and introducing their good and weak properties with references to extensions and applications. The methods are classified into four classes according to the role of a (single) decision maker in the solution process. The class of interactive methods contains most methods and it is handled most extensively. The basic emphasis when selecting methods to be included has been in nonlinear problems. Only such linear methods have made the exception that contain some specially interesting ideas or have played an important role in the method development in general. In connection with every method described, some comments of the author have been collected under the title "concluding remarks".

Despite the fact that only multiobjective optimization problems are handled, it does not mean that some of the method types presented were not applicable to multiattribute decision analysis. Nevertheless, most of the methods have been designed only for either of the problem types exploiting certain special characters.

In addition to describing solution methods, we introduce some existing software packages. Compared with the great amount of methods, there are only relatively few implementations widely known and available. Only such programs that have been available to the author for testing are presented. Some practical experiences of each software package are collected at the end of its description. Some of the programs included are capable of solving only linear multiobjective optimization problems.

As computers and monitors have developed, the graphical illustration has increased in importance and has also become easier to realize. Here we gather some ways of graphical illustration and some matters to be taken into consideration.

There are a number of complex problems in the area of optimal control that have been widely solved and treated in different connections at the University of Jyväskylä. They contain nondifferentiable functions and are of multiobjective nature. Originally, they were solved (e.g., in [Haslinger, Neittaanmäki, 1988] and [Laitinen, 1989]) by first scalarizing the multiple objective functions into one by some simple method (like summing up all the functions) and then regularizing the nondifferentiabilities into a differentiable form. After discretization, the problems could be solved by traditional, differentiable single objective optimization methods. However, both scalarization and regularization simplify the problem and cause errors in the models.

The first step in trying to make the treatment more accurate was to leave the regularization and employ nondifferentiable analysis and nondifferentiable methods. Such treatment has been presented, for example, in [Mäkelä, 1990], [Mäkelä, Neittaanmäki, 1992] and [Neittaanmäki, Tiba, 1994]. Anyway, the scalarization still remained. In the scalarization, the relative importances of the criteria are not usually known in advance and the method of summing up the criteria is artificial. As some of the criteria originate from technological constraints, the summing may bring about inaccuracies and the solution may be irrelevant in a technological sense. For this reason, it is important to use interactive methods, where the user can direct the solution process into a desirable direction.

The reason for not treating the problems as multiobjective optimization problems earlier was the small number of suitable methods capable of handling nondifferentiable functions. This impression was confirmed while examining the literature. It turned out that nondifferentiable multiobjective optimization problems have thus far been treated relatively little and there is still room for new methods.

After this reasoning it was logical that the efforts of creating new multiobjective methods and thus fulfilling the second aim of this presentation directed towards nondifferentiable problems. We introduce here two new interactive multiobjective optimization methods, called subgradient GDF method and NIMBUS method, applicable also to nondifferentiable problems. These two methods are very different. Even the starting points in their development have been different. The subgradient GDF method is based on an existing method for differentiable problems, whereas the NIMBUS method has been founded on an approach of nondifferentiable calculus with special interest in the easiness of use. We illustrate the methods by some numerical examples. Finally, we consider and solve two optimal control problems involving multiple nondifferentiable objective functions, namely a model of an elastic string and a process of continuous casting of steel.

After presenting a set of different solution methods, some comparison is in order. Naturally, no absolute order of superiority can be given but some points can be brought up. We present brief summaries of some comparisons available in the literature. Moreover, we handle the important question of selecting a method. In addition to considering some significant factors, we present a decision tree for aiding in the selection. The tree contains some basic assumptions of the methods with different ways of exchanging information between the method and its user. Also a table on a subjective basis comparing some features of the interactive methods described is presented.

The aim has been to collect a consistent and self-contained presentation of multiobjective optimization starting from some basic results and moving ahead towards the challenges of the future. Even many simple theorems are proved for the convenience of the reader and to lay firm cornerstones for the continuation. However, to keep the text at a reasonable length, some of the proofs have been omitted, but appropriate references in the literature are indicated.

The contents of this thesis have been arranged as follows. The basic concepts and notations of multiobjective optimization are presented in Chapter 1. Some related theorems are stated and optimality conditions are considered. A solid, conceptual basis for the continuation is created. Chapter 2 introduces some theoretical background and several solution methods. The methods are divided into four classes according to the role of the decision maker. Some of the methods are depicted in more detail, some in general outline and some just mentioned. Appropriate references to the literature are always pointed out. Some comments on the methods described are collected as concluding remarks. At the end of the chapter two new methods, the subgradient GDF method and the NIMBUS method, are introduced.

Some computer implementations are reported in Chapter 3. Practical experiences with each software package are also given. Chapter 4 is devoted to graphical illustration. Potentialities and restrictions of graphics are handled and some clarifying figures are enclosed. Comparison of the methods is the topic of Chapter 5. Summaries of some published comparisons are stated and some outlines for selecting an appropriate method are suggested. Also a tree diagram containing all the methods that have been described in some detail in this presentation is announced. The chapter ends with a comparative table of the interactive methods handled.

Results on numerical test examples of the subgradient GDF and the NIMBUS method are depicted in Chapter 6. Two optimal control problems are the topic of Chapter 7. First, they are briefly introduced with references to a more thorough treatment. Then the solution processes by the subgradient GDF method and the NIMBUS method are presented. Future directions are charted in Chapter 8, and finally, some conclusions are drawn in Chapter 9.

Notation and Symbols

$\mathbf{R}^n$	n-dimensional Euclidean space	
S	feasible region	page 1
$\mathbf{x}$	decision vector	page 1
f_i	objective function	page 1
k	number of objective functions	page 1
$\mathbf{f}$	vector of objective functions	page 1
Z	feasible criterion region	page 1
$\mathbf{z}$	criterion vector	page 1
$\|\mathbf{x}\|$	Euclidean norm	page 2
$\mathrm{dist}(\mathbf{x}, E)$	Euclidean distance function	page 2
$B(\mathbf{x}, \delta)$	open ball	page 2
$\mathrm{conv}\ E$	convex hull of set E	page 2
$\mathrm{int}\ E$	interior of set E	
$\nabla f_j(\mathbf{x})$	gradient of f_j at $\mathbf{x}$	page 3
$\frac{d\, f_j(\mathbf{x})}{d\, x_i}$	partial derivative of f_j subject to x_i	page 3
D	ordering cone	page 5
$\bar{\mathbf{z}}$	reference point	page 7
U	value function	Definition 1.4.1
$\mathbf{z}^\star$	ideal criterion vector	Definition 1.5.1
$\mathbf{z}^{\mathrm{nad}}$	nadir point	page 9
λ_{ij}	trade-off rate	Definition 1.7.3
m_{ij}	marginal rate of substitution	Definition 1.7.4
$\partial f_j(\mathbf{x})$	subdifferential of f_j at $\mathbf{x}$	Definition 1.11.3
ξ	subgradient	Definition 1.11.3
$\nabla_{\mathbf{x}} U(\mathbf{f}(\mathbf{x}^*))$	gradient of U with respect to $\mathbf{x}$ at $\mathbf{f}(\mathbf{x}^*)$	
$\partial_{\mathbf{x}} U(\mathbf{f}(\mathbf{x}^*))$	subdifferential of U with respect to $\mathbf{x}$ at $\mathbf{f}(\mathbf{x}^*)$	
P	number of alternative criterion vectors	
$\mathbf{z}^{\star\star}$	utopian vector	page 87
$s_{\bar{\mathbf{z}}}$	achievement (scalarizing) function	page 94

1. Concepts and Theoretical Considerations

We begin by laying a conceptual and theoretical basis for the presentation. First, we present the deterministic, continuous problem formulation and some general notations. Then we introduce several concepts and definitions of multiobjective optimization as well as their interconnections. The concepts and terms used in the literature around multiobjective optimization have not completely been fixed. The terminology in this presentation is just a part and partly slightly different from the prevailing terminology. In some cases, only one of the existing consistent terms is employed. Somewhat different definitions for concepts are presented, for example, in [Zionts, 1989].

To deepen the theoretical basis, we handle optimality conditions for differentiable and nondifferentiable multiobjective optimization problems. We also briefly touch the topics of sensitivity analysis and stability.

Throughout the presentation, even some simple results are proved for the convenience of the reader. If the idea of the proof is based on some reference, it is mentioned in the context. If some proof can be directly found as such in the literature, it is not repeated here. Instead, appropriate references are indicated. This is also in order to keep the text at a reasonable length.

Even though multiobjective optimization methods are presented in the Chapter 2, we emphasize as early as now that the methods and the theory of single objective optimization are presumed to be known. Multiobjective optimization problems are usually solved by scalarization. *Scalarization* means that the problem is converted into a single (scalar) or a family of single objective optimization problems, that is, the new problem has a real-valued objective function possibly depending on some parameters. After the problem has been scalarized, the widely developed theory and methods for single objective optimization can be used.

1.1. Problem Setting and General Notations

In this presentation we study a multiobjective optimization problem of the form

$$\begin{aligned} &\text{minimize} \quad \{f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_k(\mathbf{x})\} \\ &\text{subject to} \quad \mathbf{x} \in S, \end{aligned} \tag{1.1.1}$$

where we have k (≥ 2) *objective functions* $f_i \colon \mathbf{R}^n \rightarrow \mathbf{R}$. We denote the vector of objective functions by $\mathbf{f}(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_k(\mathbf{x}))^T$. The *decision variable vectors* $\mathbf{x} = (x_1, x_2, \ldots, x_n)^T$ belong to the (nonempty) *feasible region* (set) S, which is a subset of the *decision variable space* $\mathbf{R}^n$. We do not yet fix the form of the *constraint functions* forming S, but refer to S in general. The word "minimize" means that we want to minimize all the objective functions simultaneously. If there is no conflict between the objective functions, then a solution can be found in which every objective function attains its optimum. In this case, no special methods are needed. To avoid such trivial cases we suppose that there does not exist a single solution which is optimal with respect to every objective function. This means that the objective functions are at least partly conflicting. They may also be noncommensurable.

In the following, we denote the image of the feasible region by Z $(= \mathbf{f}(S))$. It is a subset of the *criterion space* $\mathbf{R}^k$. The elements of Z are called *criterion vectors* and denoted by $\mathbf{z} = (z_1, z_2, \ldots, z_k)^T$, where $z_i = f_i(\mathbf{x})$ for all $i = 1, \ldots, k$ are *criterion values*.

For clarity and simplicity of the treatment we suppose that all the objective functions are to be minimized. If an objective function f_i is to be maximized, it is equivalent to minimize the function $-f_i$.

First, we present some general concepts and notations. We use bold face and superscripts for vectors, for example, $\mathbf{x}^1$ and subscripts for components of vectors, for example, x_1. All the vectors here are supposed to be column vectors. For two vectors, $\mathbf{x}$ and $\mathbf{y} \in \mathbf{R}^n$, the notation $\mathbf{x}^T\mathbf{y}$ denotes their *scalar product* and the inequality $\mathbf{x} \leq \mathbf{y}$ means that $x_i \leq y_i$ for all $i = 1, \ldots, n$.

The *Euclidean norm* of a vector $\mathbf{x} \in \mathbf{R}^n$ is denoted by $\|\mathbf{x}\| = \left(\sum_{i=1}^n x_i^2\right)^{1/2}$. The *Euclidean distance function* between a point $\mathbf{x}$ and a set E is denoted by $\operatorname{dist}(\mathbf{x}, E) = \inf_{\mathbf{y} \in E} \|\mathbf{x} - \mathbf{y}\|$. The symbol $B(\mathbf{x}, \delta)$ denotes an *open ball* with centre $\mathbf{x}$ and radius $\delta > 0$, $B(\mathbf{x}, \delta) = \{\mathbf{y} \in \mathbf{R}^n \mid \|\mathbf{x} - \mathbf{y}\| < \delta\}$.

The sum $\sum_{i=1}^n \beta_i \mathbf{x}^i$ is called a *convex combination* of the vectors $\mathbf{x}^1, \ldots, \mathbf{x}^n \in E$, if $\beta_i \geq 0$ for all i and $\sum_{i=1}^n \beta_i = 1$. A *convex hull* of a set $E \subset \mathbf{R}^n$, denoted by $\operatorname{conv}(E)$, is a set of all the convex combinations of vectors in E. A set $E \subset \mathbf{R}^n$ is a *cone* if $\beta\mathbf{x} \in E$ whenever $\mathbf{x} \in E$ and $\beta \geq 0$. A negative of a cone is $-E = \{-\mathbf{x} \in \mathbf{R}^n \mid \mathbf{x} \in E\}$. A cone E is said to be *pointed* if it satisfies $E \cap -E = \{\mathbf{0}\}$.

It is said that $\mathbf{d} \in \mathbf{R}^n$ is a *feasible direction* emanating from $\mathbf{x} \in E$ if there exists $\alpha^* > 0$ such that $\mathbf{x} + \alpha\mathbf{d} \in E$ for $0 \leq \alpha \leq \alpha^*$. In some connections, we assume that the feasible region S is formed of inequality constraints. An inequality constraint is said to be *active* at some point if it is fulfilled as an equality at that point.

In the following, we present some types of multiobjective optimization problems.

Definition 1.1.2. *When all the objective and the constraint functions are linear, then the multiobjective optimization problem is called linear. In brief, it is an MOLP (multiobjective linear programming) problem.*

A large variety of solution techniques have been created so that they take into account the special characteristics of the MOLP problems. This presentation concentrates on cases where nonlinear functions are included and, thus, methods for nonlinear problems are needed. Methods and details of MOLP problems are mentioned only incidentally. (Some pitfalls and misunderstandings in linear multiobjective optimization are presented in [Korhonen, Wallenius, 1989(a)]. Suggestions to avoid them are also given.)

Before we define convex multiobjective optimization problems, we briefly state that a function $f_i \colon \mathbf{R}^n \to \mathbf{R}$ is *convex* if for all $\mathbf{x}^1, \mathbf{x}^2 \in \mathbf{R}^n$ is valid that $f_i(\beta\mathbf{x}^1 + (1 - \beta)\mathbf{x}^2) \leq \beta f_i(\mathbf{x}^1) + (1 - \beta) f_i(\mathbf{x}^2)$ for all $0 \leq \beta \leq 1$, and a set $S \in \mathbf{R}^n$ is *convex* if $\mathbf{x}^1, \mathbf{x}^2 \in S$ implies that $\beta\mathbf{x}^1 + (1 - \beta)\mathbf{x}^2 \in S$ for all $0 \leq \beta \leq 1$.

Definition 1.1.3. *The multiobjective optimization problem is convex if all the objective functions f_i $(i = 1, \ldots, k)$ and the feasible region S are convex.*

A convex multiobjective optimization problem is an important concept in the continuation. We shall also need related concepts, quasiconvex and pseudoconvex functions. The pseudoconcavity of a function calls for differentiability. For completeness, we write down the definitions of differentiable and continuously differentiable functions. A function $f_i\colon \mathbf{R}^n \to \mathbf{R}$ is *differentiable* at $\mathbf{x}^*$ if $f_i(\mathbf{x}^* + \mathbf{d}) - f_i(\mathbf{x}^*) = \nabla f_i(\mathbf{x}^*)^T\mathbf{d} + \|\mathbf{d}\|\, \varepsilon(\mathbf{x}^*, \mathbf{d})$, where $\nabla f_i(\mathbf{x}^*)$ is the *gradient* of f_i at $\mathbf{x}^*$ and $\varepsilon(\mathbf{x}^*, \mathbf{d}) \to 0$ as $\|\mathbf{d}\| \to 0$. In addition, f_i is *continuously differentiable* at $\mathbf{x}^*$ if all of its partial derivatives $\frac{d\, f_i(\mathbf{x}^*)}{d\, x_j}$ $(j = 1, \dots, n)$, that is, all the components of the gradient are continuous at $\mathbf{x}^*$.

Now we can define quasiconvex and pseudoconvex functions. A function $f_i\colon \mathbf{R}^n \to \mathbf{R}$ is *quasiconvex* if $f_i(\beta\mathbf{x}^1 + (1-\beta)\mathbf{x}^2) \leq \max\,[f_i(\mathbf{x}^1), f_i(\mathbf{x}^2)]$ for all $0 \leq \beta \leq 1$ and for all $\mathbf{x}^1, \mathbf{x}^2 \in \mathbf{R}^n$. Let f_i be differentiable. Then it is *pseudoconvex* if for all $\mathbf{x}^1, \mathbf{x}^2 \in \mathbf{R}^n$ such that $\nabla f_i(\mathbf{x}^1)^T(\mathbf{x}^2 - \mathbf{x}^1) \geq 0$, we have $f_i(\mathbf{x}^2) \geq f_i(\mathbf{x}^1)$.

The definition of convex functions can be modified for concave functions by replacing "$\leq$" by "$\geq$". Correspondingly, the definition of quasiconvex functions becomes appropriate for quasiconcave functions by the exchange of "$\leq$" to "$\geq$" and "max" to "min". In the definition of pseudoconvex functions we replace "$\geq$" by "$\leq$" to get the definition for pseudoconcave functions. Notice that if a function f_i is quasiconcave, all of its level sets $\{\mathbf{x} \in \mathbf{R}^n \mid f_i(\mathbf{x}) \geq \alpha\}$ are convex.

An important class of problems in this presentation are also nondifferentiable multiobjective optimization problems.

Definition 1.1.4. *The multiobjective optimization problem is nondifferentiable if some of the objective functions f_i $(i = 1, \dots, k)$ or the constraint functions forming the feasible region S are nondifferentiable.*

Special concepts and properties of nondifferentiable functions are introduced in Section 1.11, in the context where nondifferentiability is handled.

1.2. Pareto Optimality and Efficiency

In this section, we handle a crucial concept in optimization, namely optimality. Because of the conflictness and possible noncommensurability of the objective functions, it is not possible to find a single solution that would be optimal to all the objectives simultaneously. Multiobjective optimization problems are in a sense ill-defined. There is no natural ordering in the criterion space because it is only partially ordered (meaning that, for example, $(1,1)^T$ can be said to be less than $(3,3)^T$, but how to compare $(1,3)^T$ and $(3,1)^T$). This is always true when vectors are compared in real spaces (see also [Chankong, Haimes, 1983(b)], pp. 65–67).

Anyway, a part of the criterion vectors can be extracted for examination. Such vectors are those in which none of the components can be improved without deteriorating at least one of the other components. In 1881, F. Edgeworth presented this definition in [Edgeworth, 1987]. However, the definition is called Pareto optimality after a French-Italian (welfare) economist Vilfredo Pareto ([Pareto, 1964, 1971]), who in 1896 developed it further. (In [Stadler, 1988(a)], the term Edgeworth-Pareto optimality is used for the above-mentioned reason.) T. C. Koopmans was one of the first to employ the concept of Pareto optimality in [Koopmans, 1971] in 1951. A more formal definition of Pareto optimality is the following.

Definition 1.2.1. *A decision vector $\mathbf{x}^* \in S$ is Pareto optimal if there does not exist another decision vector $\mathbf{x} \in S$ such that $f_i(\mathbf{x}) \leq f_i(\mathbf{x}^*)$ for all $i = 1, \dots, k$ and $f_j(\mathbf{x}) < f_j(\mathbf{x}^*)$ for at least one objective function f_j.*

A criterion vector $\mathbf{z}^* \in Z$ is Pareto optimal if there does not exist another criterion vector $\mathbf{z} \in Z$ such that $z_i \leq z_i^*$ for all $i = 1, \ldots, k$ and $z_j < z_j^*$ for at least one component z_j; or equivalently, $\mathbf{z}^*$ is Pareto optimal if the decision vector corresponding to it is Pareto optimal.

In the example of Figure 1, a feasible region $S \subset \mathbf{R}^3$ and its image, a feasible criterion region $Z \subset \mathbf{R}^2$, are illustrated. The fat line contains all the Pareto optimal criterion vectors. The vector $\mathbf{z}^*$ is an example of them.

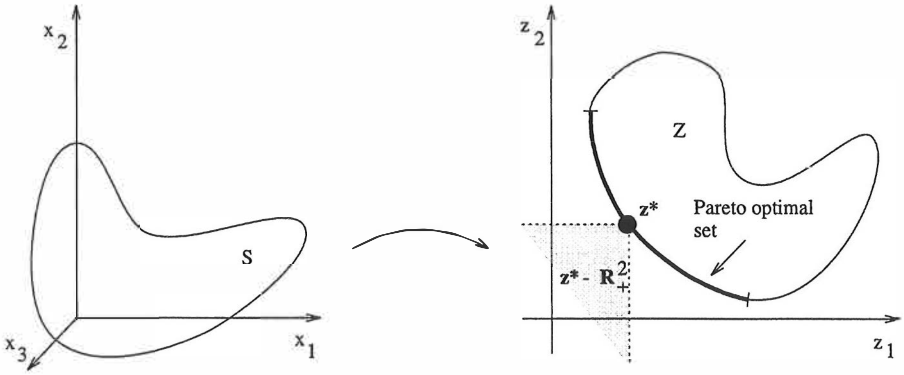


Figure 1. The sets S and Z and the Pareto optimal criterion vectors.

In addition to Pareto optimality, several other terms are sometimes used for the optimality concept described above. These terms are, for example, noninferiority, efficiency and nondominance. Differing from this practice, a more general meaning is given to efficiency later. In this presentation, Pareto optimality is used in general as a concept of optimality, unless stated otherwise.

Definition 1.2.1 introduces a so-called *global Pareto optimality*. Another important concept is a so-called local Pareto optimality.

Definition 1.2.2. *A decision vector $\mathbf{x}^* \in S$ is a locally Pareto optimal solution if there exists $\delta > 0$ such that $\mathbf{x}^*$ is Pareto optimal in $S \cap B(\mathbf{x}^*, \delta)$.*

Naturally, any globally Pareto optimal solution is locally Pareto optimal. In the following, we show that in convex multiobjective optimization problems any locally Pareto optimal solution is also globally Pareto optimal. (This result has been handled also, e.g., in [Censor, 1977].)

Theorem 1.2.3. *Let the multiobjective optimization problem be convex. Then every locally Pareto optimal solution is also globally Pareto optimal.*

Proof. Let $\mathbf{x}^* \in S$ be locally Pareto optimal. Thus there exist some $\delta > 0$ and a neighbourhood $B(\mathbf{x}^*, \delta)$ of $\mathbf{x}^*$ such that there is no $\mathbf{x} \in S \cap B(\mathbf{x}^*, \delta)$ for which $f_i(\mathbf{x}) \leq f_i(\mathbf{x}^*)$ for all $i = 1, \ldots, k$ and for at least one index j is $f_j(\mathbf{x}) < f_j(\mathbf{x}^*)$.

Let us assume that $\mathbf{x}^*$ is not globally Pareto optimal. In this case, there exists some other point $\mathbf{x}^\circ \in S$ such that

$$(1.2.4) \qquad f_i(\mathbf{x}^\circ) \leq f_i(\mathbf{x}^*) \text{ for all } i = 1, \ldots, k \text{ and } f_j(\mathbf{x}^\circ) < f_j(\mathbf{x}^*) \text{ for some } j.$$

Let us define $\hat{\mathbf{x}} = \beta\mathbf{x}^\circ + (1-\beta)\mathbf{x}^*$, where $0 < \beta < 1$ is selected such that $\hat{\mathbf{x}} \in B(\mathbf{x}^*, \delta)$. The convexity of S implies that $\hat{\mathbf{x}} \in S$.

By the convexity of the objective functions and employing (1.2.4), we obtain $f_i(\hat{\mathbf{x}}) \leq \beta f_i(\mathbf{x}^\circ) + (1-\beta)f_i(\mathbf{x}^*) \leq \beta f_i(\mathbf{x}^*) + (1-\beta)f_i(\mathbf{x}^*) = f_i(\mathbf{x}^*)$ for every i. Because $\mathbf{x}^*$ is locally Pareto optimal and $\hat{\mathbf{x}} \in B(\mathbf{x}^*, \delta)$, it has to be $f_i(\hat{\mathbf{x}}) = f_i(\mathbf{x}^*)$ for all i.

On the other hand, $f_i(\mathbf{x}^*) \leq \beta f_i(\mathbf{x}^\circ) + (1-\beta)f_i(\mathbf{x}^*)$ for every i. Because $\beta > 0$, we can divide by it and obtain $f_i(\mathbf{x}^*) \leq f_i(\mathbf{x}^\circ)$ for all i. According to the assumption (1.2.4), we have $f_j(\mathbf{x}^*) > f_j(\mathbf{x}^\circ)$ for some j. Here we have a contradiction. Thus, $\mathbf{x}^*$ is globally Pareto optimal. ▮

For briefness, we shall usually speak only about Pareto optimality in the sequel. In practice we, however, have computationally available only locally Pareto optimal solutions unless some additional requirement, such as convexity, is fulfilled.

It is also possible to define optimality in a multiobjective context in more general ways. Let us have a pointed convex cone D defined in $\mathbf{R}^k$. This cone D is called an *ordering cone* and it is used to induce a partial ordering on Z. Let us have two criterion vectors, $\mathbf{z}^1$ and $\mathbf{z}^2 \in Z$. A criterion vector $\mathbf{z}^1$ dominates $\mathbf{z}^2$, that is, $\mathbf{z}^1 \leq_D \mathbf{z}^2$ if $\mathbf{z}^2 - \mathbf{z}^1 \in D$ and $\mathbf{z}^1 \neq \mathbf{z}^2$, that is, $\mathbf{z}^2 - \mathbf{z}^1 \in D \setminus \{\mathbf{0}\}$. The same can also be put as $\mathbf{z}^2 \in \mathbf{z}^1 + D$ and $\mathbf{z}^1 \neq \mathbf{z}^2$, that is, $\mathbf{z}^2 \in \mathbf{z}^1 + D \setminus \{\mathbf{0}\}$ as illustrated in Figure 2.

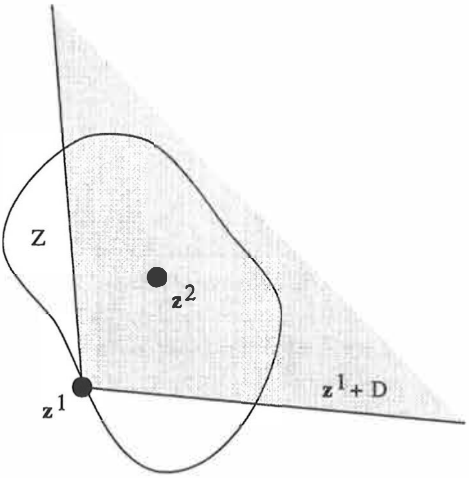


Figure 2. Domination property induced by a cone D.

Now we can present the following definition of optimality which is alternative to the previous ones. When some ordering cone is used in defining the optimality, then the term efficiency is used in this presentation.

Definition 1.2.5. *Let D be a pointed convex cone. A decision vector $\mathbf{x}^* \in S$ is efficient (with respect to D) if there does not exist another decision vector $\mathbf{x} \in S$ such that $\mathbf{f}(\mathbf{x}) \leq_D \mathbf{f}(\mathbf{x}^*)$.*

A criterion vector $\mathbf{z}^* \in Z$ is efficient if there does not exist another criterion vector $\mathbf{z} \in Z$ such that $\mathbf{z} \leq_D \mathbf{z}^*$.

This definition means that a vector is efficient (nondominated) if it is not dominated by any other feasible vector. The definition above can be formulated in many ways.

For short, we present alternative forms of Definition 1.2.5 only for its latter part. If we substitute $\leq_D$ for its definition, we have: *A criterion vector $\mathbf{z}^* \in Z$ is efficient if there does not exist another criterion vector $\mathbf{z} \in Z$ such that $\mathbf{0} \neq \mathbf{z}^* - \mathbf{z} \in D$ or $\mathbf{z}^* - \mathbf{z} \in D \setminus \{\mathbf{0}\}$* (see [Corley, 1980] or [Yu, 1985]).

Other equivalent formulations are, for instance, $\mathbf{z}^* \in Z$ is efficient if $(Z - \mathbf{z}^*) \cap (-D) = \{\mathbf{0}\}$ (see [Weidner, 1988] and [Pascoletti, Serafini, 1984]), if $(\mathbf{z}^* - \tilde{D}) \cap Z = \emptyset$, where $\tilde{D} = D \setminus \{\mathbf{0}\}$ (see [Wierzbicki, 1986(b)] and [Tapia, Murtagh, 1989]), or if $(\mathbf{z}^* - D) \cap Z = \mathbf{z}^*$ (see [Chen, 1984] and [Jahn, 1987]).

Different notions of efficiency have been gathered in [Ester, Tröltzsch, 1986]. Several auxiliary problems are there provided to obtain efficient solutions.

Remark. *The above definitions are equivalent to Pareto optimality if $D = \mathbf{R}^k_+$ (see Figure 1).*

When Pareto optimality (or efficiency) is defined with the help of ordering cones, it is trivial to verify that Pareto optimal (and efficient) criterion vectors always lie on the boundary of the feasible criterion region Z.

Instead of a cone D, which is constant for all criterion vectors, we can use a point-to-set map D from Z into $\mathbf{R}^k$ to represent the domination structure. This convex cone $D(\mathbf{z})$ is dependent on the current criterion vector. For details of ordering cones, see [Yu, 1974] and [Sawaragi, Nakayama, Tanino, 1985].

In the following, we mostly settle for handling Pareto optimality. Some extensions related to efficiency are only mentioned.

1.3. Decision Maker

There are usually a lot (infinite number) of Pareto optimal (or efficient) solutions. We can speak about a set of Pareto optimal solutions or a *Pareto optimal set*. This set can be nonconvex and nonconnected. Mathematically, every Pareto optimal solution is equally good to be a solution of the multiobjective optimization problem. However, it is generally desirable to obtain just one vector as a solution. Selecting one vector out of the set of Pareto optimal solutions needs information that is not contained in the objective functions. This is why – compared with single objective optimization – a new element is added in multiobjective optimization.

We need a decision maker to make the selection. The *decision maker* is a person (or a group of persons) who is supposed to have better insight into the problem and who can express preference relations between different solutions. Usually, the decision maker is responsible for the final solution. Solving a multiobjective optimization problem calls for the cooperation of the decision maker and an analyst. By an *analyst* we here mean a person or a computer program responsible for the mathematical side of the solution process. The analyst generates information for the decision maker to consider and the solution is selected according to the preferences of the decision maker.

In this presentation, it is assumed that we have a single decision maker or a unanimous group of decision makers. In Chapter 2, solution methods are classified according to the role of the decision maker in the solution process. In some methods, various assumptions are made concerning the preference structure and behaviour of the decision maker. Generally, group decision making calls for negotiations and specific methods (see, for example, [Yu, 1973] and [Keeney, Raiffa, 1976]).

Various kinds of information is asked from the decision maker. Among such information may be, for example, desirable or acceptable levels in the values of the objective functions, which are called *aspiration levels*. The point in the criterion space consisting of aspiration levels is called a *reference point* and denoted by $\bar{\mathbf{z}}$. These criterion values (allowed to be feasible or not) are of special interest and importance to the decision maker.

By solving a multiobjective optimization problem we here mean finding a feasible decision vector such that it is Pareto optimal and satisfies the needs and requirements of the decision maker. Such a solution is called a *final solution*.

This presentation does not concentrate on the problems of decision making, which is a research area of its own. Interesting topics in this area are, for instance, decision making with incomplete information and habitual domains. The first-mentioned matter is handled in [Weber, 1987]. Reasons for incomplete information are, for example, lack of knowledge, time pressure and fear of commitment. A *habitual domain* is defined in [Yu, 1991] as a set of ways of thinking, judging, and responding, as well as knowledge and experience on which they are based. Yu stresses that in order to make effective decisions it is important to expand and enrich the habitual domains of the decision makers. Several ways of carrying this out are presented in the reference.

1.4. Value Function

It is usually assumed that the decision maker makes decisions based on some kind of an underlying function. This function is called a value function.

Definition 1.4.1. *A function $U\colon \mathbf{R}^k \to \mathbf{R}$ representing the preferences of the decision maker among the criterion vectors is called a value function.*

Let $\mathbf{z}^1$ and $\mathbf{z}^2 \in Z$ be two different criterion vectors. If $U(\mathbf{z}^1) > U(\mathbf{z}^2)$, then the decision maker prefers $\mathbf{z}^1$ to $\mathbf{z}^2$. If $U(\mathbf{z}^1) = U(\mathbf{z}^2)$, then the decision maker finds the criterion vectors equally desirable, that is, they are *indifferent*.

It must be pointed out that the value function is totally a decision maker-dependent concept. Different decision makers may have different value functions for the same problem. Sometimes the term "utility function" is used instead of the value function. Here we follow the common way of speaking about value functions in deterministic problems. The term "utility function" is reserved for stochastic problems (which are not included in this presentation). See [Keeney, Raiffa, 1976] for more discussion about both of the terms.

If we had at our disposal the mathematical expression of the decision maker's value function, the multiobjective optimization problem would be simple to solve. The value function would just be maximized by some method of single objective optimization. The value function would offer a total (complete) ordering of the criterion vectors. However, there are several reasons why this seemingly easy way is not generally used in practice. The most important reason is that it is extremely difficult, if not impossible, for a decision maker to specify mathematically the function behind her or his preferences. Secondly, even if the function were known, it could be difficult to optimize because of its possible complicated nature. An example of such situations is the nonconcavity of the value function. In this case, only a local maximum may be found instead of a global one. In addition, it is reminded in [Steuer, Gardiner, 1991] that it is not necessarily only positive that optimizing the value function results in a single solution. After specifying the value function, the decision maker may have doubts about its validity. This is why (s)he may want to explore different alternatives before selecting the final solution.

Even though value functions are seldom explicitly used in solving multiobjective optimization problems, they are very important in the development of solution methods and as a theoretical background. In many multiobjective optimization methods, the value function is supposed to be known implicitly and the decision maker is supposed to make selections on its basis. In several methods, convergence results are obtained by setting some assumptions, for example, quasiconcavity on the implicit value function.

Usually, the value function is assumed to be monotonically (componentwise) decreasing. It means that the preference of the decision maker does not decrease if the value of some objective function decreases while all the other objective values remain unchanged (i.e., less is preferred to more). This assumption is justified in [Rosenthal, 1985] by stressing that "Clearly, under the monotonicity assumption a rational decision maker would never deliberately select a dominated point. This is probably the only important statement in multiobjective optimization that can be made without the possibility of generating some disagreement."

However, there are exceptions to this situation. Rosenthal mentions as an (maximization) example the deer population, where more deer are usually preferred to fewer for aesthetic and recreational reasons, but not in the case when the deer population is large enough to remove all the forest undergrowth.

A fact to keep in mind is that a monotone (value) function may be nonconcave. It is illustrated, for instance, in [Steuer, 1986], pp. 154–155.

The following theorem presents an important result about the solutions of componentwise decreasing value functions.

Theorem 1.4.2. *Let the value function $U : \mathbf{R}^k \to \mathbf{R}$ be componentwise decreasing. Let U attain its maximum at $\mathbf{z}^* \in Z$. Then $\mathbf{z}^*$ is Pareto optimal.*

Proof. Let $\mathbf{z}^* \in Z$ be a maximal solution of a componentwise decreasing value function U. Let us assume that $\mathbf{z}^*$ is not Pareto optimal. Then there exists a criterion vector $\mathbf{z} \in Z$ such that $z_i \leq z_i^*$ for all $i = 1, \ldots, k$ and $z_j < z_j^*$ for at least one index j. Because U is componentwise decreasing, we have $U(\mathbf{z}) > U(\mathbf{z}^*)$. Thus U does not attain its maximum at $\mathbf{z}^*$. This contradiction implies that $\mathbf{z}^*$ is Pareto optimal. ∎

Theorem 1.4.2 gives a relationship between Pareto optimal solutions and value functions. To have an impression about the relationship between efficient solutions and value functions let us consider a pseudoconcave value function U. Pseudoconcavity means that whenever $\nabla U(\mathbf{z}^1)^T(\mathbf{z}^2 - \mathbf{z}^1) \leq 0$, we have $U(\mathbf{z}^2) \leq U(\mathbf{z}^1)$. Now we can define an ordering cone as a map $D(\mathbf{z}) = \{\mathbf{d} \in \mathbf{R}^k \mid \nabla U(\mathbf{z})^T \mathbf{d} \leq 0\}$. This ordering cone can be used to determine efficient solutions. Notice that if we have a value function, we can derive its domination structure, but not generally in reverse. See [Yu, 1974] for an example.

Some references handling the existence of value functions are listed in [Stadler, 1979]. Different value functions are also presented. Different properties and forms of value functions are widely treated in [Hemming, 1978].

The way a final solution was earlier defined means that a solution is final if it maximizes the decision maker's value function. Sometimes another concept, a satisficing solution, is distinguished. Satisficing solutions are connected with so-called satisficing decision making. *Satisficing decision making* means that the decision maker does not intend to maximize any general value function but tries to achieve certain aspirations. A solution which satisfies all the aspirations of the decision maker is called a *satisficing solution*. In a most extreme case, one can define a solution to be satisficing independent of whether it is Pareto optimal or not. Here we, however, always assume that a satisficing solution is Pareto optimal (or at least weakly Pareto optimal, see Definition 1.6.1).

1.5. Ranges of the Pareto Optimal Set

Let us for a while investigate the ranges of the Pareto optimal solutions. An optimistic estimate is called an ideal criterion vector.

Definition 1.5.1. *The components $z_i^\star$ of the ideal criterion vector $\mathbf{z}^\star \in \mathbf{R}^k$ are obtained by minimizing each of the objective functions individually subject to the constraints, that is,*

$$\begin{array}{ll} \textit{minimize} & f_i(\mathbf{x}) \\ \textit{subject to} & \mathbf{x} \in S, \end{array}$$

for $i = 1, \ldots, k$.

It is obvious that if the ideal criterion vector were feasible ($\mathbf{z}^\star \in Z$), it would be the solution of the multiobjective optimization problem. This is not possible in general since there is a conflict among the objectives. Even though the ideal criterion vector is not attainable, it can be considered a reference point, something to go for. From the ideal criterion vector we obtain the lower bounds of the Pareto optimal set for each objective function.

The upper bounds of the Pareto optimal set, the components of a so-called *nadir point*, $\mathbf{z}^{\mathrm{nad}}$ are much more difficult to obtain. They can be estimated from a payoff table. A *payoff table* is formed by using the decision vectors obtained when calculating the ideal criterion vector. On the ith row of the payoff table there are the values of all the objective functions calculated at the point where f_i obtained its minimal value. So, $z_i^\star$ is at the main diagonal of the table. The maximal value of the ith column in the payoff table is selected as an estimate of the upper bound of the ith objective over the Pareto optimal set. Notice that the criterion vectors in the rows of the payoff table are Pareto optimal if they are unique.

The black points in Figure 3 represent ideal criterion vectors, and the grey ones are nadir points. The nadir point may be feasible or not, as illustrated in Figure 3. The Pareto optimal set is represented by the fat lines.

Weistroffer has presented examples in [Weistroffer, 1985] to illustrate the fact that the estimates from the payoff table are not necessarily equal to the real components of the nadir point. The difference between the complete Pareto optimal set and the subset of the Pareto optimal set bounded by the ideal criterion vector and the upper bounds obtained from the payoff table in linear cases is explored also in [Reeves, Reid, 1988]. It is proposed that relaxing (increasing) the approximated upper bounds by a relatively small tolerance should improve the approximation, although it is ad hoc

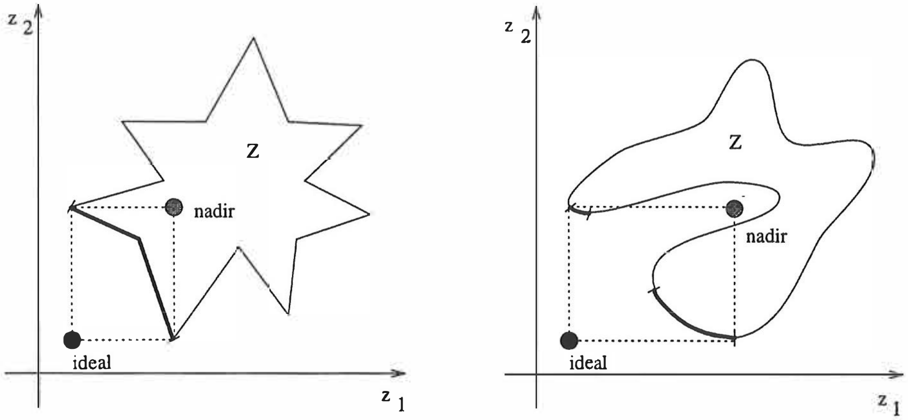


Figure 3. Ideal criterion vectors and nadir points.

in nature. Some linear multiobjective optimization problems are also studied in [Isermann, Steuer, 1988]. It is examined how many of the Pareto optimal extreme solutions are above the upper bounds obtained from the payoff table. Three methods for determining the exact nadir point in a linear case are suggested. None of them is especially economical computationally. In [Dessouky, Ghiassi, Davis, 1986], three heuristics are presented to calculate the nadir point when the problem is linear. For nonlinear problems, there is no constructive method for calculating the nadir point. Anyway, the payoff table may be used as a rough estimate as long as its robustness is kept in mind. Because of the above-described difficulty of calculating the actual nadir point, we usually refer to the approximate nadir point as $\mathbf{z}^{\mathrm{nad}}$ in what follows.

It is possible that (some) objective functions are unbounded, for instance, from below. In this case some caution is in order. In multiobjective optimization problems this does not necessarily mean that the problem is formulated incorrectly. There may still exist Pareto optimal solutions. However, if, for instance, some component of the ideal criterion vector is unbounded and it is replaced by a small but finite number, methods utilizing the ideal criterion vector may not be able to overcome the replacement.

The ranges of the Pareto optimal set are of interest also in [Benson, Sayin, 1994]. The authors deal with the maximization of a linear function over the Pareto optimal set of an MOLP problem.

1.6. Weak Pareto Optimality

In addition to Pareto optimality, other related concepts are widely used. They are weak and proper Pareto optimality. The relationship between these concepts is that the properly Pareto optimal set is a subset of the Pareto optimal set which is a subset of the weakly Pareto optimal set.

A vector is weakly Pareto optimal if there does not exist any other vector for which all the components are better. In a more formal way it means the following.

Definition 1.6.1. *A decision vector $\mathbf{x}^* \in S$ is weakly Pareto optimal if there does not exist another decision vector $\mathbf{x} \in S$ such that $f_i(\mathbf{x}) < f_i(\mathbf{x}^*)$ for all $i = 1, \ldots, k$.*

A criterion vector $\mathbf{z}^* \in Z$ is weakly Pareto optimal if there does not exist another criterion vector $\mathbf{z} \in Z$ such that $z_i < z_i^*$ for all $i = 1, \ldots, k$; or equivalently, if the decision vector corresponding to it is weakly Pareto optimal.

The fat line in Figure 4 represents the set of weakly Pareto optimal criterion vectors. The fact that the Pareto optimal set is a subset of the weakly Pareto optimal set can also be seen in the figure. The Pareto optimal criterion vectors are situated along the line between the black points.

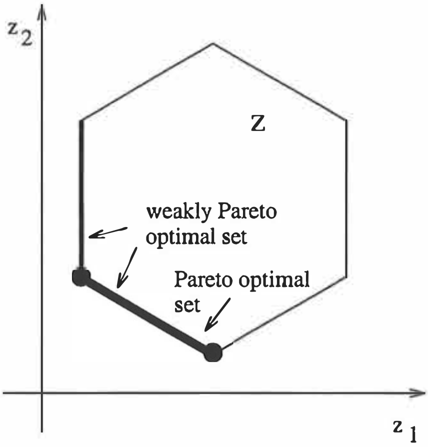


Figure 4. Weakly Pareto optimal points.

If the set Z of criterion vectors is ordered by an ordering cone D, weakly efficient vectors may be characterized in the following way. A criterion vector $\mathbf{z}^* \in Z$ is weakly efficient if $(Z - \mathbf{z}^*) \cap (-\text{int}(D)) = \emptyset$ (see [Sawaragi, Nakayama, Tanino, 1985]), or if $(\mathbf{z}^* - \text{int}(D)) \cap Z = \emptyset$ (see [Jahn, 1987] and [Wierzbicki, 1986(b)]), where $\text{int}(D)$ denotes the interior of the cone D.

The connectedness of the sets of Pareto optimal and weakly Pareto optimal solutions has not been widely treated. The Pareto optimal set of an MOLP problem is proved to be connected in [Steuer, 1986]. It is stated in [Warburton, 1983] as a generally known fact that the Pareto optimal set is connected in convex multiobjective optimization problems. Warburton shows that if the feasible region is convex and compact and the objective functions are quasiconvex, then the set of weakly Pareto optimal solutions is connected. The connectedness of the Pareto optimal set is guaranteed for a certain subclass of quasiconvex functions. Also a noncompact case is studied in [Warburton, 1983]. Connectedness of the sets of weakly efficient and efficient points is studied in [Helbig, 1990(a)].

Although weakly Pareto optimal solutions are important for theoretical considerations, they are not always useful in practice, because of the big size of the weakly Pareto optimal set. A more restricting concept than Pareto optimality is proper Pareto optimality. To clarify its practical meaning and for other further purposes we first define trade-offs and marginal rates of substitution.

1.7. Trade-Off and Marginal Rate of Substitution

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.

Definition 1.7.1. (From [Chankong, Haimes, 1983(b)]) *Let $\mathbf{x}^1$ and $\mathbf{x}^2 \in S$ be two decision vectors and let $(f_1(\mathbf{x}^1), \ldots, f_k(\mathbf{x}^1))^T$ and $(f_1(\mathbf{x}^2), \ldots, f_k(\mathbf{x}^2))^T$ be the corresponding criterion vectors, respectively. We denote the ratio of change between the functions f_i and f_j by*

$$\Lambda_{ij} = \Lambda_{ij}(\mathbf{x}^1, \mathbf{x}^2) = \frac{f_i(\mathbf{x}^1) - f_i(\mathbf{x}^2)}{f_j(\mathbf{x}^1) - f_j(\mathbf{x}^2)},$$

where $f_j(\mathbf{x}^1) - f_j(\mathbf{x}^2) \neq 0$.

Now, Λ_{ij} is called a partial trade-off involving f_i and f_j between $\mathbf{x}^1$ and $\mathbf{x}^2$ if $f_l(\mathbf{x}^1) = f_l(\mathbf{x}^2)$ for all $l = 1, \ldots, k$, $l \neq i, j$. If $f_l(\mathbf{x}^1) \neq f_l(\mathbf{x}^2)$ for at least one $l = 1, \ldots, k$, and $l \neq i, j$, then Λ_{ij} is called a total trade-off involving f_i and f_j between $\mathbf{x}^1$ and $\mathbf{x}^2$.

Notice that in the case of two objective functions there is no difference between partial and total trade-offs. If partial trade-offs are presented to the decision maker, (s)he can compare changes in two objective functions at a time. This is usually more comfortable than comparing several objectives. If the points $\mathbf{x}^1$ and $\mathbf{x}^2$ are Pareto optimal, then there always exist some objective functions f_i and f_j for which the trade-off is negative. A concept related to the trade-off is the trade-off rate.

Definition 1.7.2. *Let $\mathbf{x}^* \in S$ be a decision vector and let $\mathbf{d}^*$ be a feasible direction emanating from $\mathbf{x}^*$. The total trade-off rate at $\mathbf{x}^*$ involving f_i and f_j along the direction $\mathbf{d}^*$ is given by*

$$\lambda_{ij} = \lambda_{ij}(\mathbf{x}^*, \mathbf{d}^*) = \lim_{\alpha \to 0} \Lambda_{ij}(\mathbf{x}^* + \alpha \mathbf{d}^*, \mathbf{x}^*).$$

If $\mathbf{d}^*$ is a feasible direction so that there exists $\bar{\alpha} > 0$ satisfying $f_l(\mathbf{x}^* + \alpha \mathbf{d}^*) = f_l(\mathbf{x}^*)$ for all $l \neq i, j$ and for all $0 \leq \alpha \leq \bar{\alpha}$, then the corresponding λ_{ij} is called a partial trade-off rate.

Remark. *If the objective functions are continuously differentiable, then*

$$\lambda_{ij} = \frac{\nabla f_i(\mathbf{x}^*)^T \mathbf{d}^*}{\nabla f_j(\mathbf{x}^*)^T \mathbf{d}^*},$$

where the denominator differs from zero.

For continuously differentiable objective functions we can alternatively give the following definition.

Definition 1.7.3. *Let the objective functions be continuously differentiable and let $\mathbf{x}^* \in S$ be a decision vector. Then a partial trade-off rate at $\mathbf{x}^*$ involving f_i and f_j is given by*

$$\lambda_{ij} = \lambda_{ij}(\mathbf{x}^*) = \frac{d f_i(\mathbf{x}^*)}{d f_j(\mathbf{x}^*)}.$$

Notice that the trade-off is defined mathematically and the decision maker cannot affect it. If we take into consideration the opinions of the decision maker, we can define indifference curves and marginal rates of substitution.

It is said that two feasible solutions are situated on the same *indifference curve* (or isopreference curve) if the decision maker finds them equally desirable, that is, neither of them is preferred to the other one. This means that indifference curves are contours of the underlying value function. There may also be a "wider" indifference band. Then we do not have any well-defined boundary between preferences, but a band where indifference occurs. This concept is studied in [Passy, Levanon, 1984].

For any two solutions on the same indifference curve there is a trade-off involving a certain increment in the value of one objective function (f_j) that the decision maker is willing to tolerate in exchange for a certain amount of decrement for some other objective function (f_i) while the preferences of the two solutions remain the same. This is called a marginal rate of substitution. This kind of trading between different solutions is characteristic of multiobjective optimization problems when moving from one Pareto optimal solution to another. The marginal rate of substitution (sometimes also called indifference trade-off) is the negative of the slope of the tangent to the indifference curve at a certain point.

Definition 1.7.4. *A marginal rate of substitution $m_{ij} = m_{ij}(\mathbf{x}^*)$ ($i, j = 1, \ldots, k$, $i \neq j$) represents the preferences of the decision maker at a decision vector $\mathbf{x}^* \in S$. It is the amount of decrement in the value of the objective function f_i that compensates to the decision maker the increment in the value of the objective function f_j by one unit, while the values of all the other objectives remain unaltered.*

Notice that in the definition the starting and the resulting criterion vectors lie on the same indifference curve.

It can be stated that a final solution of a multiobjective optimization problem is a Pareto optimal solution where the indifference curve is tangent to the Pareto optimal set. This tangency condition means finding an indifference curve intersecting the feasible criterion region that is farthest to the southwest. Figure 5 illustrates this property.

Remark. *If the partial derivatives exist, then*

$$m_{ij}(\mathbf{x}^*) = \frac{d U(\mathbf{f}(\mathbf{x}^*))}{d f_j} \Big/ \frac{d U(\mathbf{f}(\mathbf{x}^*))}{d f_i}. \tag{1.7.5}$$

If the Pareto optimal set is smooth (that is, at every Pareto optimal point there exists a unique tangent), we can have the following result. When one examines the definition of a trade-off rate at some point, one sees that it is the slope of the tangent of the Pareto optimal set at that point. We can also define that when a Pareto optimal

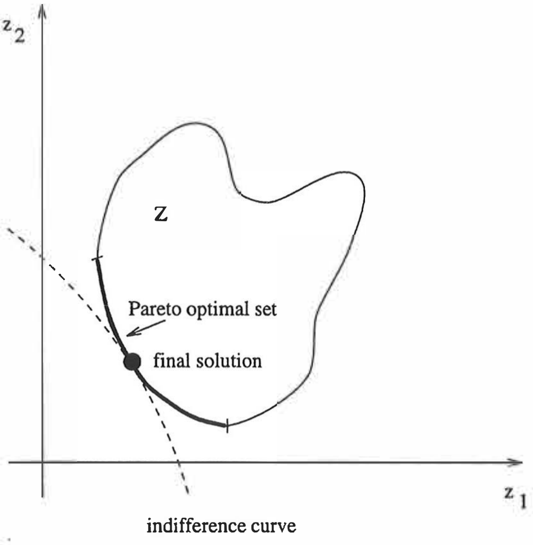


Figure 5. The final solution.

solution is a final solution, then the tangents of the indifference curve and the Pareto optimal set coincide at it, that is, $-m_{ij} = \lambda_{ij}$ for all $i, j = 1, \ldots, k$, $i \neq j$. Thus with the help of the negative of the marginal rate of substitution and trade-off rate one can get a local linear approximation of the indifference curve and the Pareto optimal set, respectively.

Usually, one of the objective functions is selected to be a *reference function* when trade-offs and marginal rates of substitution are treated. The trade-offs and the marginal rates of substitution are generated with respect to it. In the notations above, f_i is the reference function. When cooperating with decision makers, it is important to select the reference function in a meaningful way. Important criteria in the selection are, for example, that the reference function is in familiar units or that it is dominant.

1.8. Proper Pareto Optimality

Kuhn and Tucker were the first to notice that some of the Pareto optimal solutions had undesirable properties (see [Kuhn, Tucker, 1951]). They introduced so-called properly Pareto optimal solutions. The idea of properly Pareto optimal solutions is that unbounded trade-offs between objectives are not allowed. Pareto optimal solutions may be divided into properly and improperly Pareto optimal solutions. Practically, a properly Pareto optimal solution with very high or very low trade-offs does not essentially differ from a weakly Pareto optimal solution.

There exist several definitions for proper Pareto optimality. The idea is easiest to understand from the definition of Geoffrion.

Definition 1.8.1. (From [Geoffrion, 1968]) *A decision vector $\mathbf{x}^* \in S$ is properly Pareto optimal if it is Pareto optimal and if there is some real number $M > 0$ such that for each 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*

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

In other words, a solution is properly Pareto optimal if there is at least one pair of objectives for which a finite decrement of the other objective is possible only at the expense of some reasonable increment of the other objective.

A method for obtaining all the properly Pareto optimal solutions satisfying prescribed trade-offs in a convex case is proposed in [Geromel, Ferreira, 1991]. Also upper estimates for properly Pareto optimal solutions are given.

Durier has studied in [Durier, 1988] the relationships between Pareto optimal and properly Pareto optimal sets in a convex case. He, for example, concludes that if the set of properly Pareto optimal solutions is closed, then the two sets are equal. A property called a locally flat surface, which guarantees the very same equality in convex and differentiable problems, is presented in [Zhou, Mokhtarian, Zlobec, 1993].

Results about Pareto optimal and properly Pareto optimal solutions have been gathered in [Gal, 1986]. In [Chew, Choo, 1984], it is proved in that every Pareto optimal solution is also properly Pareto optimal for a nonlinear problem involving only pseudolinear functions (i.e., differentiable functions which are also pseudoconvex and pseudoconcave). The results of Chew and Choo can be considered special cases of more general results in [Weir, 1990]. In [Gulati, Islam, 1990], it is shown that the preceding result can be generalized by assuming quasiconvexity of the active constraints (of the form $\mathbf{g}(\mathbf{x}) \leq \mathbf{0}$) with some regularity properties. Pseudolinearity is extended by defining semilocally pseudolinear functions in [Kaul, Lyall, Kaur, 1988]. We shall present some results about the relationships between Pareto optimal, weakly and properly Pareto optimal solutions in the context of solution methods in Chapter 2.

Next, we introduce ε-proper Pareto optimality, which is easy to illustrate graphically.

Definition 1.8.2. (From [Wierzbicki, 1980(b)]) *A decision vector* $\mathbf{x}^* \in S$ *and the corresponding criterion vector* $\mathbf{z}^* \in Z$ *are* ε*-properly Pareto optimal if*

$$Z \cap (\mathbf{z}^* - \mathbf{R}^k_\varepsilon \setminus \{\mathbf{0}\}) = \emptyset,$$

where $\mathbf{R}^k_\varepsilon = \{\mathbf{z} \in \mathbf{R}^k \mid \mathit{dist}(\mathbf{z}, \mathbf{R}^k_+) \leq \varepsilon \|\mathbf{z}\|\}$ *and* $\varepsilon > 0$ *is a predetermined scalar.*

The set of ε-properly Pareto optimal solutions is depicted in Figure 6 and denoted by a fat line. The solutions are obtained by intersecting the feasible criterion region with a blunt cone. The end points of the Pareto optimal curve, $\mathbf{z}^1$ and $\mathbf{z}^2$, have also been marked to ease the comparison.

An interesting thing with ε-properly Pareto optimal solutions is that the trade-offs are bounded between ε and $1/\varepsilon$. We return to this definition in Section 2.19. Now we continue by the original definition of Kuhn and Tucker.

Kuhn and Tucker derived also necessary and sufficient conditions for proper Pareto optimality in [Kuhn, Tucker, 1951]. Those conditions will be presented in the next section.

The feasible region is now supposed to consist of inequality constraints. So we assume that $S = \{\mathbf{x} \in \mathbf{R}^n \mid \mathbf{g}(\mathbf{x}) = (g_1(\mathbf{x}), g_2(\mathbf{x}), \ldots, g_m(\mathbf{x}))^T \leq \mathbf{0}\}$. In addition, all the

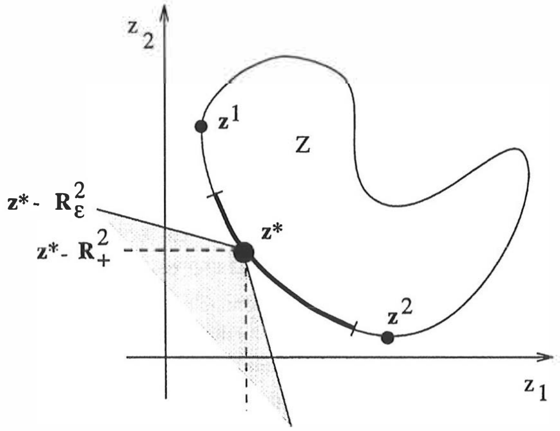


Figure 6. ε-properly Pareto optimal solutions.

objective and the constraint functions are supposed to be continuously differentiable. Thus this definition is not applicable to nondifferentiable multiobjective optimization problems.

Definition 1.8.3. (From [Kuhn, Tucker, 1951]) *A decision vector* $\mathbf{x}^* \in S$ *is properly Pareto optimal if it is Pareto optimal* and *if there does not exist* any *vector* $\mathbf{d} \in \mathbf{R}^n$ *such that*

$$\nabla f_i(\mathbf{x}^*)^T \mathbf{d} \leq 0$$

for all $i = 1, \ldots, k$, *for some* j

$$\nabla f_j(\mathbf{x}^*)^T \mathbf{d} < 0,$$

and

$$\nabla g_l(\mathbf{x}^*)^T \mathbf{d} \leq 0$$

for all l *satisfying* $g_l(\mathbf{x}^*) = 0$.

A comparison of the definitions of Kuhn and Tucker and Geoffrion is presented in [Geoffrion, 1968]. Several practical examples are given in [Tamura, Arai, 1982] to illustrate the fact that properly Pareto optimal solutions according to the definitions of Geoffrion and Kuhn and Tucker (and one more definition by Klinger, see [Klinger, 1967]) are not necessarily consistent. Also conditions under which proper Pareto optimality in the sense of Kuhn and Tucker implies proper Pareto optimality in the sense of Geoffrion are proved. Conditions for the reverse result are given and proved, for example, in [Sawaragi, Nakayama, Tanino, 1985]. More mathematical results concerning the properties and relationships of the definitions of Geoffrion, Kuhn and Tucker and Klinger are given in [White, 1983(a)].

Borwein, Benson and Henig have defined proper efficiency when a closed convex cone is used as an ordering cone. Even those definitions are not equivalent with each other. However, for instance, the definitions of Geoffrion and Benson are equal when $D = \mathbf{R}^k_+$ (see [Benson, 1983]). All the five definitions (excluding that of Klinger's) are presented and compared in [Sawaragi, Nakayama, Tanino, 1985]. In [Henig, 1982(b)], necessary and sufficient conditions for the existence of properly efficient solutions are given. A new kind of proper efficiency, called super efficiency, is presented in [Borwein, Zhuang, 1989, 1991].

In the following, proper Pareto optimality is understood in the sense of Geoffrion's definition unless stated otherwise.

1.9. Existence of Pareto Optimal Solutions

Let us have a look at the existence of Pareto optimal and properly Pareto optimal solutions. The procedures presented can also be used to test a feasible decision vector for Pareto optimality and to find an initial Pareto optimal solution.

Specific results for MOLP problems are presented in [Ecker, Kouada, 1975]. General problems are examined in [Wendell, Lee, 1977] by relying on duality theory and generalizing the results of Ecker and Kouada into nonlinear problems. The treatment is based on an auxiliary problem

$$
\begin{aligned}
&\text{infimum} && \sum_{i=1}^{k} f_i(\mathbf{x}) \\
&\text{subject to} && \mathbf{f}(\mathbf{x}) \leq \mathbf{f}(\hat{\mathbf{x}}) \\
&&& \mathbf{x} \in S,
\end{aligned}
\tag{1.9.1}
$$

where $\hat{\mathbf{x}}$ is any vector in S and the optimal objective function value is denoted by $\phi(\hat{\mathbf{x}})$.

Theorem 1.9.2. *Let a decision vector $\mathbf{x}^* \in S$ be given. The vector $\mathbf{x}^*$ is Pareto optimal if and only if $\phi(\mathbf{x}^*) = \sum_{i=1}^{k} f_i(\mathbf{x}^*)$.*

Proof. See [Wendell, Lee, 1977].

When studying the (primal) problem (1.9.1) and its dual, a ***duality gap*** is said to occur if the optimal value of the primal problem is not equivalent to the optimal value of the dual problem.

Theorem 1.9.3. *Let a decision vector $\hat{\mathbf{x}} \in S$ be given and suppose that $\phi(\hat{\mathbf{x}}) = -\infty$. Then some $\mathbf{x}^* \in S$ is Pareto optimal only if there is a duality gap between the primal (1.9.1) and its dual problem at $\mathbf{x}^*$.*

Proof. See [Wendell, Lee, 1977].

The significance of Theorem 1.9.3 is that precluding duality gaps the nonexistence of Pareto optimal points is characterized by the condition that $\phi(\hat{\mathbf{x}}) = -\infty$ for some $\hat{\mathbf{x}} \in S$. It can also be proved that if a multiobjective optimization problem is convex and if $\phi(\hat{\mathbf{x}}) = -\infty$ for some $\hat{\mathbf{x}} \in S$, then no properly Pareto optimal solutions exist. See the details in [Wendell, Lee, 1977].

Also in [Benson, 1978], the existence of Pareto optimal and properly Pareto optimal solutions is investigated. The results can be combined into the following theorem.

Theorem 1.9.4. *Let a decision vector $\mathbf{x}^* \in S$ be given. Solve the problem*

$$
\begin{aligned}
&\textit{maximize} && \sum_{i=1}^{k} \varepsilon_i \\
&\textit{subject to} && f_i(\mathbf{x}) + \varepsilon_i = f_i(\mathbf{x}^*) \textit{ for all } i = 1, \dots, k, \\
&&& \varepsilon_i \geq 0 \textit{ for all } i = 1, \dots, k, \\
&&& \mathbf{x} \in S.
\end{aligned}
\tag{1.9.5}
$$

(1) *The vector* $\mathbf{x}^*$ *is Pareto optimal if and only if the problem (1.9.5) has an optimal objective function value of zero.*
(2) *If the problem (1.9.5) has a finite nonzero optimal objective function value obtained at a point* $\hat{\mathbf{x}}$*, then* $\hat{\mathbf{x}}$ *is Pareto optimal.*
(3) *If the multiobjective optimization problem is convex and if the problem (1.9.5) does not have a finite optimal objective function value, then the set of properly Pareto optimal solutions is empty.*
(4) *If in addition to the conditions in (3), the set* $\{\hat{\mathbf{z}} \in \mathbf{R}^k \mid \hat{\mathbf{z}} \leq \mathbf{f}(\mathbf{x})$ *for some* $\mathbf{x} \in S\}$ *is closed, then the Pareto optimal set is empty.*

Proof. See [Benson, 1978] or [Chankong, Haimes, 1983(b)], pp. 151–152.

The auxiliary problems (1.9.1) and (1.9.5) can also be used to produce Pareto optimal solutions from weakly Pareto optimal solutions. However, in some practical problems it is very expensive to carry out these additional optimizations.

Several ways of determining the Pareto optimality of a particular point in an MOLP problem are presented in [Eiselt, Pederzoli, Sandblom, 1987]. They all apply to special situations. The existence of Pareto optimal solutions when there is an infinite number of objective functions is considered in [Alekseichik, Naumov, 1981].

The existence and characterization of efficient solutions with respect to ordering cones are studied in [Henig, 1982(a)]. The existence of efficient solutions in linear spaces is treated in [Borwein, 1983]. In addition, the existence of weakly and properly efficient (in the sense of Borwein) and efficient solutions in the presence of ordering cones is studied in [Jahn, 1986(b)]. The existence of efficient solutions is studied in [Cambini, Martein, 1994] by introducing so-called quasi-D-bounded sets.

A phenomenon called complete efficiency is studied in [Benson, 1991]. *Complete efficiency* occurs when every feasible decision vector of a multiobjective optimization problem is Pareto optimal. Tests are presented to check for complete efficiency in linear and nonlinear cases. A significant saving of computational efforts can be attained if the problem is tested for complete efficiency before it is solved. If the problem is completely efficient, no time, effort and special machinery for generating a part or all of the Pareto optimal solutions is needed. To Benson's knowledge, no solution algorithm first checks for complete efficiency. Benson reminds that nobody has yet studied the frequency of completely efficient problems among multiobjective optimization problems. He points out that it may be more common than one can think, especially when the feasible region S has no interior. An example of such problems are transportation problems. Complete efficiency is also treated in [Weidner, 1990].

In [Benson, 1983], Benson examines conditions for the situation when there always exists an efficient solution that is superior to any nonefficient solution, that is, for each $\mathbf{x} \in S$ and corresponding $\mathbf{z} \in Z$ there exists an efficient point $\mathbf{x}^*$ and corresponding $\mathbf{z}^*$ such that $\mathbf{z} - \mathbf{z}^* \in D$, where D is the ordering cone. This is called a *domination property*. The results of Benson are corrected and necessary and sufficient conditions for the domination property to hold are supplied in [Luc, 1984(a)]. The domination property and its sufficient conditions are treated in [Henig, 1986]. It is demonstrated that efficiency, proper efficiency, and the domination property are equivalent under convexity. The domination property in infinite dimensional spaces and for the sum of two sets is handled in [Luc, 1990].

In the sequel, we present optimality conditions for multiobjective optimization problems. Because the conditions are different for differentiable and nondifferentiable problem, they are handled separately.

1.10. Optimality Conditions

Optimality conditions are an important sector in optimization. As elsewhere in this presentation, we restrict the treatment also here to finite dimensional Euclidean spaces. We consider problems of the form

$$(1.10.1)\quad \begin{array}{ll} \text{minimize} & \{f_1(\mathbf{x}), f_2(\mathbf{x}), \dots, f_k(\mathbf{x})\} \\ \text{subject to} & \mathbf{x} \in S = \{\mathbf{x} \in \mathbf{R}^n \mid \mathbf{g}(\mathbf{x}) = (g_1(\mathbf{x}), g_2(\mathbf{x}), \dots, g_m(\mathbf{x}))^T \leq \mathbf{0}\}. \end{array}$$

We denote the set of active constraints at a point $\mathbf{x}^*$ by $J(\mathbf{x}^*) = \{j \in \{1, \dots, m\} \mid g_j(\mathbf{x}^*) = 0\}$. We assume in this section that the objective and the constraint functions are continuously differentiable. In Section 1.11, we handle nondifferentiable functions.

The presentation is mainly based on the results stated in [Kuhn, Tucker, 1951], [Da Cunha, Polak, 1967], [Marusciac, 1982] and [Simon, 1986]. The theorems are here presented in a simplified form when compared to the general practice in the literature. For this reason, the proofs have been modified. At first, we present a necessary condition of the Fritz John-type.

Theorem 1.10.2. *Let the objective and the constraint functions of the problem (1.10.1) be continuously differentiable. A necessary condition for a decision vector* $\mathbf{x}^* \in S$ *to be Pareto optimal is that there exist vectors* $\mathbf{0} \leq \boldsymbol{\lambda} \in \mathbf{R}^k$ *and* $\mathbf{0} \leq \boldsymbol{\mu} \in \mathbf{R}^m$ *for which* $(\boldsymbol{\lambda}, \boldsymbol{\mu}) \neq (\mathbf{0}, \mathbf{0})$ *such that*

$$(1)\quad \sum_{i=1}^{k} \lambda_i \nabla f_i(\mathbf{x}^*) + \sum_{j=1}^{m} \mu_j \nabla g_j(\mathbf{x}^*) = 0$$

$$(2)\quad \mu_j g_j(\mathbf{x}^*) = 0 \quad \textit{for all } j = 1, \dots, m.$$

Proof. See, for instance, [Da Cunha, Polak, 1967].

We do not present the proof here because it is quite extensive. The theorem can be considered a special case of the corresponding theorem for nondifferentiable problems, which is proved in Section 1.11. For convex problems, necessary optimality conditions can be derived by using separating hyperplanes. This has been realized, for example, in [Zadeh, 1963]. A separation theorem is also employed in the proof of the general case in [Da Cunha, Polak, 1967].

The difference between the Fritz John and Karush-Kuhn-Tucker-type optimality conditions in single objective optimization is that the multiplier (λ) of the objective function is assumed to be positive in the latter case. This eliminates degeneracy since it implies that the objective function plays its important role in the optimality conditions. To guarantee the positivity of λ, some regularity has to be assumed from the problem. Different regularity conditions exist and they are called *constraint qualifications*.

In the multiobjective case it is equally important that all the multipliers of the objective functions are not equal to zero. From several different constraint qualifications we here present a so-called Kuhn-Tucker constraint qualification.

Definition 1.10.3. *Let the constraint functions g_j of the problem (1.10.1) be continuously differentiable. The problem satisfies the Kuhn-Tucker constraint qualification at $\mathbf{x}^* \in S$ if for any $\mathbf{d} \in \mathbf{R}^n$ such that $\nabla g_j(\mathbf{x}^*)^T\mathbf{d} \leq 0$ for all $j \in J(\mathbf{x}^*)$, there exists a function $\mathbf{a}\colon [0,1] \to \mathbf{R}^n$ which is continuously differentiable at 0, and some real scalar $\alpha > 0$, such that*

$$\mathbf{a}(0) = \mathbf{x}^*, \quad \mathbf{g}(\mathbf{a}(t)) \leq \mathbf{0} \quad \textit{for all} \quad 0 \leq t \leq 1 \quad \textit{and} \quad \mathbf{a}'(0) = \alpha\mathbf{d}.$$

Before we can continue, we write down a so-called Motzkin's theorem of the alternative. It will be needed in the proof of the following necessary condition.

Theorem 1.10.4. **(Motzkin's theorem)** *Let A and C be given matrices, where A contains at least one element. Then either the system of inequalities*

$$A\mathbf{x} < \mathbf{0}, \quad C\mathbf{x} \leq \mathbf{0}$$

has a solution $\mathbf{x}$, or the system

$$A^T\boldsymbol{\lambda} + C^T\boldsymbol{\mu} = 0, \quad \boldsymbol{\lambda} \geq 0, \quad \boldsymbol{\lambda} \neq \mathbf{0}, \quad \boldsymbol{\mu} \geq \mathbf{0}$$

has a solution $(\boldsymbol{\lambda}, \boldsymbol{\mu})$, but never both.

Proof. See, for example, [Mangasarian, 1969], pp. 28–29.

Theorem 1.10.5. *Let the assumptions of Theorem 1.10.2 be satisfied with the Kuhn-Tucker constraint qualification. Then Theorem 1.10.2 is valid with the addition that $\boldsymbol{\lambda} \neq \mathbf{0}$.*

Proof. Let $\mathbf{x}^*$ be Pareto optimal. The idea of this proof is to apply Theorem 1.10.4. For this reason we prove that there does not exist any $\mathbf{d} \in \mathbf{R}^n$ such that

$$\begin{aligned} &\nabla f_i(\mathbf{x}^*)^T\mathbf{d} < 0 \text{ for all } i = 1, \ldots, k, \text{ and} \\ &\nabla g_j(\mathbf{x}^*)^T\mathbf{d} \leq 0 \text{ for all } j \in J(\mathbf{x}^*). \end{aligned} \tag{1.10.6}$$

Let us on the contrary assume that there exists some $\mathbf{d}^* \in \mathbf{R}^n$ satisfying (1.10.6). Then from the Kuhn-Tucker constraint qualification we know that there exists a function $\mathbf{a}\colon [0,1] \to \mathbf{R}^n$ which is continuously differentiable at 0 and some real scalar $\alpha > 0$ such that $\mathbf{a}(0) = \mathbf{x}^*$, $\mathbf{g}(\mathbf{a}(t)) \leq \mathbf{0}$ for all $0 \leq t \leq 1$ and $\mathbf{a}'(0) = \alpha\mathbf{d}^*$.

Because the functions f_i are continuously differentiable, we can approximate $f_i(\mathbf{a}(t))$ linearly as

$$\begin{aligned} f_i(\mathbf{a}(t)) &= f_i(\mathbf{x}^*) + \nabla f_i(\mathbf{x}^*)^T(\mathbf{a}(t) - \mathbf{x}^*) + \|\mathbf{a}(t) - \mathbf{x}^*\|\varphi(\mathbf{a}(t), \mathbf{x}^*) \\ &= f_i(\mathbf{x}^*) + \nabla f_i(\mathbf{x}^*)^T(\mathbf{a}(t) - \mathbf{a}(0)) + \|\mathbf{a}(t) - \mathbf{a}(0)\|\varphi(\mathbf{a}(t), \mathbf{a}(0)) \\ &= f_i(\mathbf{x}^*) + t\nabla f_i(\mathbf{x}^*)^T\left(\frac{\mathbf{a}(0+t) - \mathbf{a}(0)}{t}\right) + \|\mathbf{a}(t) - \mathbf{a}(0)\|\varphi(\mathbf{a}(t), \mathbf{a}(0)), \end{aligned}$$

where $\varphi(\mathbf{a}(t), \mathbf{a}(0)) \to 0$ as $\|\mathbf{a}(t) - \mathbf{a}(0)\| \to 0$. As $t \to 0$ tends $\|\mathbf{a}(t) - \mathbf{a}(0)\|$ to zero and we have $\frac{\mathbf{a}(0+t) - \mathbf{a}(0)}{t} = \mathbf{a}'(0) = \alpha\mathbf{d}^*$.

After utilizing the assumption $\nabla f_i(\mathbf{x}^*)^T \mathbf{d}^* < 0$ for all $i = 1, \dots, k$ (and $t \geq 0$), we have $f_i(\mathbf{a}(t)) < f_i(\mathbf{x}^*)$ for all i for a sufficiently small t. This contradicts the Pareto optimality of $\mathbf{x}^*$.

Thus we have proved the statement (1.10.6). Now we conclude from Theorem 1.10.4 that there exists multipliers $\lambda_i \geq 0$ for $i \in \{1, \dots, k\}, \boldsymbol{\lambda} \neq \mathbf{0}$, and $\mu_j \geq 0$ for $j \in J(\mathbf{x}^*)$ such that $\sum_{i=1}^k \lambda_i \nabla f_i(\mathbf{x}^*) + \sum_{j \in J(\mathbf{x}^*)} \mu_j \nabla g_j(\mathbf{x}^*) = 0$. We obtain the statement (1) of Theorem 1.10.2 by setting $\mu_j = 0$ for all $j \in \{1, \dots, m\} \setminus J(\mathbf{x}^*)$.

If $g_j(\mathbf{x}^*) < 0$ for some j, then according to the above setting $\mu_j = 0$ and the equalities (2) of Theorem 1.10.2 follow. ▮

A sort of similar proof basically but different in realization is presented in [Marusciac, 1982].

Constraint qualifications based on the linear independence of gradient vectors have been stated in [Da Cunha, Polak, 1967]. A collection of other constraint qualifications has been gathered in [Simon, 1986]. A new constraint qualification for convex problems has been introduced in [Zhou, Mokhtarian, Zlobec, 1993].

Corollary 1.10.7. *The conditions of Theorems 1.10.2 and 1.10.5 are also necessary for a decision vector $\mathbf{x}^* \in S$ to be weakly Pareto optimal.*

If the multiobjective optimization problem is convex, then we can state the following sufficient condition for Pareto optimality. Let us first recall the sufficient condition of optimality in the single objective case.

Theorem 1.10.8. *A sufficient condition for a point $\mathbf{x}^* \in \mathbf{R}^n$ to be a (global) minimum of the problem*

$$(1.10.9) \qquad \begin{array}{ll} \textit{minimize} & f_i(\mathbf{x}) \\ \textit{subject to} & \mathbf{g}(\mathbf{x}) = (g_1(\mathbf{x}), g_2(\mathbf{x}), \dots, g_m(\mathbf{x}))^T \leq \mathbf{0}, \end{array}$$

where the functions $f_i \colon \mathbf{R}^n \to \mathbf{R}$ and $g_j \colon \mathbf{R}^n \to \mathbf{R}$, $j = 1, \dots, m$, are continuously differentiable and convex, is that there exist multipliers $\mathbf{0} \leq \boldsymbol{\mu} \in \mathbf{R}^m$ such that

$$(1) \quad \nabla f_i(\mathbf{x}^*) + \sum_{j=1}^m \mu_j \nabla g_j(\mathbf{x}^*) = 0$$

$$(2) \quad \mu_j g_j(\mathbf{x}^*) = 0 \qquad \textit{for all } j = 1, \dots, m.$$

Proof. See, for example, [Simon, 1986].

Theorem 1.10.10. *Let the problem (1.10.1) be convex. A sufficient condition for a decision vector $\mathbf{x}^* \in S$ to be a Pareto optimal solution of the problem (1.10.1) is that there exist multipliers $\mathbf{0} < \boldsymbol{\lambda} \in \mathbf{R}^k$ and $\mathbf{0} \leq \boldsymbol{\mu} \in \mathbf{R}^m$ such that*

$$(1) \quad \sum_{i=1}^k \lambda_i \nabla f_i(\mathbf{x}^*) + \sum_{j=1}^m \mu_j \nabla g_j(\mathbf{x}^*) = 0$$

$$(2) \quad \mu_j g_j(\mathbf{x}^*) = 0 \qquad \textit{for all } j = 1, \dots, m.$$

Proof. Let the vectors $\boldsymbol{\lambda}$ and $\boldsymbol{\mu}$ be such that the conditions stated are satisfied. We define a function $F \colon \mathbf{R}^n \to \mathbf{R}$ as $F(\mathbf{x}) = \sum_{i=1}^k \lambda_i f_i(\mathbf{x})$, where $\mathbf{x} \in S$. Trivially F is convex because all the functions f_i are and $\boldsymbol{\lambda} > \mathbf{0}$. Now we have, from the statements (1) and (2), that $\nabla F(\mathbf{x}^*) + \sum_{j=1}^{m} \mu_j \nabla g_j(\mathbf{x}^*) = 0$ and $\mu_j g_j(\mathbf{x}^*) = 0$ for all $j = 1, \ldots, m$. Thus, according to Theorem 1.10.8, a sufficient condition for F to attain its minimum at $\mathbf{x}^*$ is satisfied. So $F(\mathbf{x}^*) \leq F(\mathbf{x})$ for all $\mathbf{x} \in S$. In other words,

$$\sum_{i=1}^{k} \lambda_i f_i(\mathbf{x}^*) \leq \sum_{i=1}^{k} \lambda_i f_i(\mathbf{x}) \tag{1.10.11}$$

for all $\mathbf{x} \in S$.

Let us assume that $\mathbf{x}^*$ is not Pareto-optimal. Then there exists some point $\hat{\mathbf{x}} \in S$ such that $f_i(\hat{\mathbf{x}}) \leq f_i(\mathbf{x}^*)$ for all i and for at least one index j is $f_j(\hat{\mathbf{x}}) < f_j(\mathbf{x}^*)$. Because every λ_i was assumed to be positive, we have $\sum_{i=1}^{k} \lambda_i f_i(\hat{\mathbf{x}}) < \sum_{i=1}^{k} \lambda_i f_i(\mathbf{x}^*)$. This is a contradiction with the inequality (1.10.11) and $\mathbf{x}^*$ is thus Pareto optimal. ▮

Notice that because the multiobjective optimization problem is assumed to be convex, Theorem 1.10.10 provides a sufficient condition for global Pareto optimality. This was proved in Theorem 1.2.3.

Theorem 1.10.12. *The condition stated in Theorem 1.10.10 is sufficient for a decision vector* $\mathbf{x}^* \in S$ *to be a weakly Pareto optimal solution of the problem (1.10.1) for* $\mathbf{0} \leq \boldsymbol{\lambda} \in \mathbf{R}^k$ *with* $\boldsymbol{\lambda} \neq \mathbf{0}$.

Proof. The proof is a straightforward modification of the proof of Theorem 1.10.10.

The convexity assumption in Theorem 1.10.10 can be relaxed. The stated sufficient condition is also valid if the objective functions are pseudoconvex and the constraint functions are quasiconvex at $\mathbf{x}^*$. This extension is handled, for example, in [Marusciac, 1982] and [Simon, 1986].

Second-order optimality conditions (presuming twice continuously differentiable objective and constraint functions) have been examined substantially less than the first-order optimality conditions. Second-order necessary conditions for Pareto optimality are handled in [Simon, 1986], and necessary and sufficient conditions for Pareto and weakly Pareto optimal solutions are presented in [Wang, 1991].

If an ordering cone D is used in defining efficiency, then the optimality conditions are similar to those presented above except for the multipliers λ_i. Now they are not only nonnegative real scalars but belong to a dual cone D^*, where $D^* = \{\boldsymbol{\lambda} \in \mathbf{R}^k \mid \boldsymbol{\lambda}^T\mathbf{y} \geq 0 \text{ for all } \mathbf{y} \in D\}$. Because of the close resemblance, we do not handle optimality conditions separately for efficiency here. For details see, for example, [Chen, 1984] and [Luc, 1989].

For completeness we also present the original necessary optimality condition formulated for proper Pareto optimality (in the sense of Kuhn and Tucker) as stated by Kuhn and Tucker. To begin with, we write down Tucker's theorem of the alternative, which will be utilized in the proof.

Theorem 1.10.13. (**Tucker's theorem**) *Let A and C be given matrices, where A contains at least one element. Then either the system of inequalities*

$$A\mathbf{x} \leq \mathbf{0}, \quad A\mathbf{x} \neq \mathbf{0}, \quad C\mathbf{x} \leq \mathbf{0}$$

has a solution $\mathbf{x}$, *or the system*

$$A^T\boldsymbol{\lambda} + C^T\boldsymbol{\mu} = 0, \quad \boldsymbol{\lambda} > \mathbf{0}, \quad \boldsymbol{\mu} \geq \mathbf{0}$$

has a solution $(\boldsymbol{\lambda}, \boldsymbol{\mu})$, *but never both.*

Proof. The proof is similar to the proof of Theorem 1.10.4.

Theorem 1.10.14. *A necessary condition for a decision vector* $\mathbf{x}^* \in S$ *to be a properly Pareto optimal solution (in the sense of Kuhn and Tucker) of the problem (1.10.1) is that there exist vectors* $\mathbf{0} < \boldsymbol{\lambda} \in \mathbf{R}^k$ *and* $\mathbf{0} \leq \boldsymbol{\mu} \in \mathbf{R}^m$ *such that*

$$\begin{aligned} &(1) \quad \sum_{i=1}^{k} \lambda_i \nabla f_i(\mathbf{x}^*) + \sum_{j=1}^{m} \mu_j \nabla g_j(\mathbf{x}^*) = 0 \\ &(2) \quad \mu_j g_j(\mathbf{x}^*) = 0 \quad \text{for all } j = 1, \ldots, m. \end{aligned}$$

Proof. Let $\mathbf{x}^*$ be properly Pareto optimal (in the sense of Kuhn and Tucker). Then, from the definition we know that there does not exist any vector $\mathbf{d} \in \mathbf{R}^n$ such that $\nabla f_i(\mathbf{x}^*)^T\mathbf{d} \leq 0$ for all $i = 1, \ldots, k$, $\nabla f_j(\mathbf{x}^*)^T\mathbf{d} < 0$ for some index j, and $\nabla g_l(\mathbf{x}^*)^T\mathbf{d} \leq 0$ for all $l \in J(\mathbf{x}^*)$. Then, from Theorem 1.10.13 we know that there exist multipliers $\lambda_i > 0$ for $i \in \{1, \ldots, k\}$ and $\mu_j \geq 0$ for $j \in J(\mathbf{x}^*)$ such that $\sum_{i=1}^{k} \lambda_i \nabla f_i(\mathbf{x}^*) + \sum_{j \in J(\mathbf{x}^*)} \mu_j \nabla g_j(\mathbf{x}^*) = 0$. We obtain the statement (1) by setting $\mu_j = 0$ for all $j \in \{1, \ldots, m\} \setminus J(\mathbf{x}^*)$.

If $g_j(\mathbf{x}^*) < 0$ for some j, then according to the above setting $\mu_j = 0$ and the equalities (2) follow. ▮

It is proved in [Geoffrion, 1968] and [Sawaragi, Nakayama, Tanino, 1985] that if the Kuhn-Tucker constraint qualification (Definition 1.10.3) is satisfied at a decision vector $\mathbf{x}^* \in S$, then the condition in Theorem 1.10.14 is also necessary for $\mathbf{x}^*$ to be properly Pareto optimal in the sense of Geoffrion.

Theorem 1.10.15. *If the problem (1.10.1) is convex, then the condition in Theorem 1.10.14 is also sufficient for a decision vector* $\mathbf{x}^* \in S$ *to be properly Pareto optimal (in the sense of Kuhn and Tucker).*

Proof. See [Sawaragi, Nakayama, Tanino, 1985], p. 90.

Necessary and sufficient conditions for proper Pareto optimality in the sense of Geoffrion are presented in [Gulati, Islam, 1990] for pseudolinear objective and quasiconvex constraint functions.

1.11. Nondifferentiable Optimality Conditions

In this section, we no longer necessitate differentiability but put forward nondifferentiable counterparts for the optimality conditions presented in Section 1.10. Usually, when the assumption of continuous differentiability is given up, functions are supposed to be locally Lipschitzian.

Definition 1.11.1. *A function $f_i\colon \mathbf{R}^n \to \mathbf{R}$ is locally Lipschitzian at a point $\mathbf{x}^* \in \mathbf{R}^n$ if there exist scalars $K > 0$ and $\delta > 0$ such that*

$$|f_i(\mathbf{x}^1) - f_i(\mathbf{x}^2)| \leq K\|\mathbf{x}^1 - \mathbf{x}^2\| \quad \textit{for all } \mathbf{x}^1, \mathbf{x}^2 \in B(\mathbf{x}^*, \delta).$$

A function is here called nondifferentiable if it is locally Lipschitzian (and not necessarily continuously differentiable).

In every other way the multiobjective optimization problem to be solved is still of the form

$$\begin{array}{ll} \text{minimize} & \{f_1(\mathbf{x}), f_2(\mathbf{x}), \dots, f_k(\mathbf{x})\} \\ \text{subject to} & \mathbf{x} \in S = \{\mathbf{x} \in \mathbf{R}^n \mid \mathbf{g}(\mathbf{x}) = (g_1(\mathbf{x}), g_2(\mathbf{x}), \dots, g_m(\mathbf{x}))^T \leq \mathbf{0}\}. \end{array} \tag{1.11.2}$$

According to Rademacher's Theorem (see, e.g., [Federer, 1969]), we know that a locally Lipschitzian function, defined in an open set, is differentiable almost everywhere in that set. The set, where a function f_i is not differentiable, is denoted here by Ω_{f_i}. In the sequel, we employ the concept subdifferential as defined by Clarke in [Clarke, 1983]. It corresponds to the gradient in the differentiable case.

Definition 1.11.3. *Let the function $f_i\colon \mathbf{R}^n \to \mathbf{R}$ be locally Lipschitzian at a point $\mathbf{x}^* \in \mathbf{R}^n$. The set*

$$\partial f_i(\mathbf{x}^*) = \operatorname{conv}\{\xi \in \mathbf{R}^n \mid \xi = \lim_{l \to \infty} \nabla f_i(\mathbf{x}^l);\ \mathbf{x}^l \to \mathbf{x}^*,\ \mathbf{x}^l \in \mathbf{R}^n \setminus \Omega_{f_i}\}$$

is called a subdifferential of the function f_i evaluated at the point $\mathbf{x}^*$. In addition, the vectors $\xi \in \partial f_i(\mathbf{x}^*)$ are called subgradients.

In the following we briefly present several properties of subdifferentials without any proofs.

Theorem 1.11.4. *If the functions $f_i\colon \mathbf{R}^n \to \mathbf{R}$, $i = 1, \dots, k$, are locally Lipschitzian at the point $\mathbf{x}^* \in \mathbf{R}^n$, then we have for real scalars w_i*

$$\partial\Big(\sum_{i=1}^{k} w_i f_i\Big)(\mathbf{x}^*) \subset \sum_{i=1}^{k} w_i \partial f_i(\mathbf{x}^*).$$

If the functions f_i are convex, then the inclusion is valid as equality.

Proof. See, for example, [Mäkelä, Neittaanmäki, 1992], p. 39.

Theorem 1.11.5. *Let the functions $f_i\colon \mathbf{R}^n \to \mathbf{R}$, $i = 1, \dots, k$, be locally Lipschitzian at the point $\mathbf{x}^* \in \mathbf{R}^n$. Then the function $f\colon \mathbf{R}^n \to \mathbf{R}$*

$$f(\mathbf{x}^*) = \max_{1 \leq i \leq k} f_i(\mathbf{x}^*)$$

is also locally Lipschitzian at $\mathbf{x}^*$. In addition,

$$\partial f(\mathbf{x}^*) \subset \operatorname{conv}\{\partial f_i(\mathbf{x}^*) \mid i \in I(\mathbf{x}^*)\},$$

where $I(\mathbf{x}^*) \subset \{1, \dots, k\}$ denotes the set of indices i for which $f(\mathbf{x}^*) = f_i(\mathbf{x}^*)$.

Proof. See, for example, [Mäkelä, Neittaanmäki, 1992], pp. 47–49.

Theorem 1.11.6. *If the function $f_i \colon \mathbf{R}^n \to \mathbf{R}$ is locally Lipschitzian at the point $\mathbf{x}^* \in \mathbf{R}^n$ and attains its local minimum at $\mathbf{x}^*$, then*

$$0 \in \partial f_i(\mathbf{x}^*).$$

If the function f_i is convex, then the condition is also sufficient.

Proof. See, for example, [Mäkelä, Neittaanmäki, 1992], pp. 70–71.

Theorem 1.11.7. *A necessary condition for a point $\mathbf{x}^* \in \mathbf{R}^n$ to be a local minimum of the problem*

$$\text{(1.11.8)} \qquad \begin{array}{ll} \textit{minimize} & f_i(\mathbf{x}) \\ \textit{subject to} & \mathbf{g}(\mathbf{x}) = (g_1(\mathbf{x}), g_2(\mathbf{x}), \dots, g_m(\mathbf{x}))^T \leq \mathbf{0}, \end{array}$$

where the functions $f_i \colon \mathbf{R}^n \to \mathbf{R}$ and $g_j \colon \mathbf{R}^n \to \mathbf{R}$, $j = 1, \dots, m$, are locally Lipschitzian at $\mathbf{x}^*$, is that there exist multipliers $0 \leq \lambda \in \mathbf{R}$ and $\mathbf{0} \leq \boldsymbol{\mu} \in \mathbf{R}^m$ for which $(\lambda, \boldsymbol{\mu}) \neq (0, \mathbf{0})$ such that

$$\begin{array}{ll} (1) & 0 \in \lambda \partial f_i(\mathbf{x}^*) + \sum_{j=1}^{m} \mu_j \partial g_j(\mathbf{x}^*) \\ (2) & \mu_j g_j(\mathbf{x}^*) = 0 \quad \textit{for all } j = 1, \dots, m. \end{array}$$

Proof. See, for example, [Clarke, 1983], pp. 228–230, or [Kiwiel, 1985(c)], p. 16.

Now we have gathered a collection of tools for continuation and we can handle the actual optimality conditions. The presentation is mainly based on the references [Minami, 1980–81, 1981, 1983], [Kanniappan, 1983], [Wang, 1984], [Doležal, 1985] and [Craven, 1989]. The theorems are here presented in a simplified form when compared to the general practice in the literature. For this reason, the proofs have been modified.

Theorem 1.11.9. *Let the objective and the constraint functions of the problem (1.11.2) be locally Lipschitzian at a point $\mathbf{x}^* \in S$. A necessary condition for the point $\mathbf{x}^*$ to be a Pareto optimal solution of the problem (1.11.2) is that there exist multipliers $\mathbf{0} \leq \boldsymbol{\lambda} \in \mathbf{R}^k$ and $\mathbf{0} \leq \boldsymbol{\mu} \in \mathbf{R}^m$ for which $(\boldsymbol{\lambda}, \boldsymbol{\mu}) \neq (\mathbf{0}, \mathbf{0})$ such that*

$$\begin{array}{ll} (1) & 0 \in \sum_{i=1}^{k} \lambda_i \partial f_i(\mathbf{x}^*) + \sum_{j=1}^{m} \mu_j \partial g_j(\mathbf{x}^*) \\ (2) & \mu_j g_j(\mathbf{x}^*) = 0 \quad \textit{for all } j = 1, \dots, m. \end{array}$$

Proof. Because it is assumed that $(\boldsymbol{\lambda}, \boldsymbol{\mu}) \neq (\mathbf{0}, \mathbf{0})$, we can normalize the multipliers to sum up to one. We shall here prove a stronger condition, where $\sum_{i=1}^{k} \lambda_i + \sum_{j=1}^{m} \mu_j = 1$.

Let $\mathbf{x}^* \in S$ be Pareto optimal. At first we define an additional function $F\colon \mathbf{R}^n \to \mathbf{R}$ by

$$F(\mathbf{x}) = \max_{i,j}[f_i(\mathbf{x}) - f_i(\mathbf{x}^*), g_j(\mathbf{x})]$$

and show that for all $\mathbf{x} \in \mathbf{R}^n$ is

$$F(\mathbf{x}) \geq 0. \tag{1.11.10}$$

Let us on the contrary assume that for some $\mathbf{x}^\circ \in \mathbf{R}^n$ is $F(\mathbf{x}^\circ) < 0$. Then $g_j(\mathbf{x}^\circ) < 0$ for all $j = 1, \dots, m$ and the point is thus feasible for the problem (1.11.2). In addition, $f_i(\mathbf{x}^\circ) < f_i(\mathbf{x}^*)$ for all $i = 1, \dots, k$, which contradicts the Pareto optimality of $\mathbf{x}^*$. Thus (1.11.10) must be true.

Noting that the point $\mathbf{x}^*$ is feasible to the problem (1.11.2), we have $\mathbf{g}(\mathbf{x}^*) \leq 0$. This implies $F(\mathbf{x}^*) = 0$. Combining this fact with the property (1.11.10), we know that F attains its (global) minimum at $\mathbf{x}^*$. As all the functions f_i and g_j are locally Lipschitzian at $\mathbf{x}^*$, likewise is F (according to Theorems 1.11.4 and 1.11.5). We deduce from Theorem 1.11.6 that $0 \in \partial F(\mathbf{x}^*)$.

Let us denote, for short, $h_i(\mathbf{x}) = f_i(\mathbf{x}) - f_i(\mathbf{x}^*)$ for all $i = 1, \dots, k$. Note that

$$\partial h_i(\mathbf{x}^*) \subset \partial f_i(\mathbf{x}^*), \tag{1.11.11}$$

applying Theorem 1.11.4.

We designate the set of indices i for which $F(\mathbf{x}^*) = h_i(\mathbf{x}^*)$ by $I(\mathbf{x}^*) \subset \{1, \dots, k\}$ and the set of indices j for which $F(\mathbf{x}^*) = g_j(\mathbf{x}^*)$ by $J(\mathbf{x}^*) \subset \{1, \dots, m\}$. Now we can employ Theorem 1.11.5 and obtain

$$0 \in \operatorname{conv}\{\partial h_i(\mathbf{x}^*), \partial g_j(\mathbf{x}^*) \mid i \in I(\mathbf{x}^*),\ j \in J(\mathbf{x}^*)\}.$$

Employing (1.11.11) and the definition of a convex hull, we know that there exist vectors $\boldsymbol{\lambda}$ and $\boldsymbol{\mu}$ of real multipliers for which $\lambda_i \geq 0$ for all $i \in I(\mathbf{x}^*)$, $\mu_j \geq 0$ for all $j \in J(\mathbf{x}^*)$ and $\sum_{i \in I(\mathbf{x}^*)} \lambda_i + \sum_{j \in J(\mathbf{x}^*)} \mu_j = 1$, such that

$$0 \in \sum_{i \in I(\mathbf{x}^*)} \lambda_i \partial f_i(\mathbf{x}^*) + \sum_{j \in J(\mathbf{x}^*)} \mu_j \partial g_j(\mathbf{x}^*).$$

Now we can set $\lambda_i = 0$ for all $i \in \{1, \dots, k\} \setminus I(\mathbf{x}^*)$ and $\mu_j = 0$ for all $j \in \{1, \dots, m\} \setminus J(\mathbf{x}^*)$. The statement (1) follows from this setting.

The part (2) is trivial. If $g_j(\mathbf{x}^*) < 0$ for some j, then $j \in \{1, \dots, m\} \setminus J(\mathbf{x}^*)$ and we have $\mu_j = 0$. This completes the proof. ▮

Theorem 1.11.9 can also be proved employing first some scalarization method and then Theorem 1.11.7 for the resulting single objective optimization problem. A reasoning based on results concerning the ε-constraint method (to be presented in Section 2.3 in Theorem 2.3.3) is sketched in [Kanniappan, 1983]. There the multiobjective optimization problem is assumed to be convex but the proof is valid for nonconvex problems, too.

Next, we examine some constraint qualifications more closely. It is obvious that they differ from the differentiable case.

Notice that when the necessary optimality conditions are derived, for example, with the help of the ε-constraint problem, it is easy to generalize constraint qualifications from single objective optimization into the multiobjective case. One just assumes that both the original constraints and the possible additional constraints satisfy some constraint qualification. This has been expressed in [Doležal, 1985]. The constraint qualifications used there are those of calmness and Mangasarian-Fromovitz.

In the following theorem a so-called Cottle constraint qualification is used. Different constraint qualifications are presented, for example, in [Ishizuka, Shimizu, 1984].

Definition 1.11.12. *Let the objective and the constraint functions of the problem (1.11.2) be locally Lipschitzian at a point* $\mathbf{x}^* \in S$*. The problem (1.11.2) satisfies a Cottle constraint qualification at* $\mathbf{x}^*$ *if either* $g_j(\mathbf{x}^*) < 0$ *for all* $j = 1, \ldots, m$*, or* $0 \notin \operatorname{conv}\{\partial g_j(\mathbf{x}^*) \mid g_j(\mathbf{x}^*) = 0\}$.

Assuming the Cottle constraint qualification, we obtain the following necessary condition for Pareto optimality.

Theorem 1.11.13. *Let the assumptions of Theorem 1.11.9 be satisfied with the Cottle constraint qualification. Then Theorem 1.11.9 is valid with the addition that* $\boldsymbol{\lambda} \neq \mathbf{0}$.

Proof. The proof of Theorem 1.11.9 is here valid up till the observation $0 \in \partial F(\mathbf{x}^*)$ and the definition of the surrogate function h_i. We prove also this theorem in a stronger form, where the multipliers sum up to one.

From the definition of F we know that

$$F(\mathbf{x}^*) = h_i(\mathbf{x}^*) \ (= 0) \tag{1.11.14}$$

for all $i = 1, \ldots, k$. We continue by first assuming that $g_j(\mathbf{x}^*) < 0$ for all $j = 1, \ldots, m$. In this case, $F(\mathbf{x}^*) > g_j(\mathbf{x}^*)$ for all j. Now we can apply Theorem 1.11.5 and obtain

$$0 \in \operatorname{conv}\{\partial h_i(\mathbf{x}^*) \mid i = 1, \ldots, k\}.$$

Next we can apply (1.11.11) to eliminate the surrogate functions h_i. From the definition of a convex hull we know that there exists a vector $\mathbf{0} \leq \boldsymbol{\lambda} \in \mathbf{R}^k$ of multipliers for which $\sum_{i=1}^k \lambda_i = 1$ (thus $\boldsymbol{\lambda} \neq \mathbf{0}$) such that

$$0 \in \sum_{i=1}^{k} \lambda_i \partial f_i(\mathbf{x}^*).$$

We obtain the statement to be proved (denoted by (1) in Theorem 1.11.9) by setting $\mu_j = 0$ for all $j = 1, \ldots, m$.

On the other hand, if there exists some index j such that $g_j(\mathbf{x}^*) = 0$, we denote the set of such indices by $J(\mathbf{x}^*)$. By the Cottle constraint qualification we know that

$$0 \notin \operatorname{conv}\{\partial g_j(\mathbf{x}^*) \mid j \in J(\mathbf{x}^*)\}. \tag{1.11.15}$$

In this case, we have from (1.11.14) and Theorem 1.11.5 the result

$$0 \in \operatorname{conv}\{\partial h_i(\mathbf{x}^*), \partial g_j(\mathbf{x}^*) \mid i = 1, \ldots, k, \ j \in J(\mathbf{x}^*)\}.$$

Applying (1.11.11) and the definition of a convex hull, we know that there exist multipliers $\lambda_i \geq 0$, $i \in \{1, \dots, k\}$, and $\mu_j \geq 0$, $j \in J(\mathbf{x}^*)$ for which $\sum_{i=1}^k \lambda_i + \sum_{j \in J(\mathbf{x}^*)} \mu_j = 1$, and by the assumption (1.11.15) especially $\boldsymbol{\lambda} \neq \mathbf{0}$, such that

$$0 \in \sum_{i=1}^k \lambda_i \partial f_i(\mathbf{x}^*) + \sum_{j \in J(\mathbf{x}^*)} \mu_j \partial g_j(\mathbf{x}^*).$$

Again, we obtain the statement to be proved by setting $\mu_j = 0$ for all $j \in \{1, \dots, m\} \setminus J(\mathbf{x}^*)$.

The proof of part (2) is the same as in Theorem 1.11.9. ▮

Corollary 1.11.16. *The conditions of Theorems 1.11.9 and 1.11.13 are also necessary for a decision vector $\mathbf{x}^* \in S$ to be a weakly Pareto optimal solution of the problem (1.11.2).*

If we assume the multiobjective optimization problem to be convex and $\boldsymbol{\lambda} > \mathbf{0}$, we get a sufficient condition.

Theorem 1.11.17. *Let the problem (1.11.2) be convex with locally Lipschitzian functions. A sufficient condition for a decision vector $\mathbf{x}^* \in S$ to be a Pareto optimal solution of the problem (1.11.2) is that there exist multipliers $\mathbf{0} < \boldsymbol{\lambda} \in \mathbf{R}^k$ and $\mathbf{0} \leq \boldsymbol{\mu} \in \mathbf{R}^m$ such that*

$$(1) \quad 0 \in \sum_{i=1}^k \lambda_i \partial f_i(\mathbf{x}^*) + \sum_{j=1}^m \mu_j \partial g_j(\mathbf{x}^*)$$

$$(2) \quad \mu_j g_j(\mathbf{x}^*) = 0 \quad \textit{for all } j = 1, \dots, m.$$

Proof. To start with, we define an additional function $F\colon \mathbf{R}^n \to \mathbf{R}$ by $F(\mathbf{x}) = \sum_{i=1}^k \lambda_i f_i(\mathbf{x}) + \sum_{j=1}^m \mu_j g_j(\mathbf{x})$, where the multipliers λ_i and μ_j satisfy the assumptions above. Because the functions f_i and g_j are convex, $\boldsymbol{\lambda} > \mathbf{0}$ and $\boldsymbol{\mu} \geq \mathbf{0}$, then also F is convex, and $\partial F(\mathbf{x}) = \sum_{i=1}^k \lambda_i \partial f_i(\mathbf{x}) + \sum_{j=1}^m \mu_j \partial g_j(\mathbf{x})$ (as stated in Theorem 1.11.4).

From the assumption (1) we know that $0 \in \partial F(\mathbf{x}^*)$, and, according to Theorem 1.11.6, the point $\mathbf{x}^*$ is a (global) optimum of F. This implies that for any $\mathbf{x}^\circ \in \mathbf{R}^n$, especially any $\mathbf{x}^\circ$ satisfying $\mathbf{g}(\mathbf{x}^\circ) \leq \mathbf{0}$, the following is valid:

$$\begin{aligned} 0 &\leq F(\mathbf{x}^\circ) - F(\mathbf{x}^*) \\ &= \sum_{i=1}^k \lambda_i f_i(\mathbf{x}^\circ) + \sum_{j=1}^m \mu_j g_j(\mathbf{x}^\circ) - \sum_{i=1}^k \lambda_i f_i(\mathbf{x}^*) - \sum_{j=1}^m \mu_j g_j(\mathbf{x}^*). \end{aligned}$$

Employing the assumption (2), the fact that $\mathbf{g}(\mathbf{x}^\circ) \leq 0$ and $\boldsymbol{\mu} \geq \mathbf{0}$, we obtain

$$\sum_{i=1}^k \lambda_i f_i(\mathbf{x}^*) \leq \sum_{i=1}^k \lambda_i f_i(\mathbf{x}^\circ) \tag{1.11.18}$$

for any $\mathbf{x}^\circ \in S$.

Let us assume that $\mathbf{x}^*$ is not Pareto-optimal. Then there exists some feasible $\hat{\mathbf{x}}$ such that $f_i(\hat{\mathbf{x}}) \leq f_i(\mathbf{x}^*)$ for all i and for at least one index j is $f_j(\hat{\mathbf{x}}) < f_j(\mathbf{x}^*)$. Because every λ_i was assumed to be positive, we have $\sum_{i=1}^k \lambda_i f_i(\hat{\mathbf{x}}) < \sum_{i=1}^k \lambda_i f_i(\mathbf{x}^*)$. This is a contradiction with the inequality (1.11.18) and $\mathbf{x}^*$ is thus Pareto optimal. ▮

Theorem 1.11.19. *The condition stated in Theorem 1.11.17 is sufficient for a decision vector $\mathbf{x}^* \in S$ to be a weakly Pareto optimal solution of the problem (1.11.2) for $\mathbf{0} \leq \boldsymbol{\lambda} \in \mathbf{R}^k$ with $\boldsymbol{\lambda} \neq \mathbf{0}$.*

Proof. The proof is a trivial modification of the proof of Theorem 1.11.17.

Finally, we introduce one more constraint qualification. It can only be applied to convex problems and it will be needed in connection with one of the multiobjective optimization methods, in Section 2.23. It is called a Slater constraint qualification. It is independent of the differentiability of the functions involved.

Definition 1.11.20. *Let the problem (1.11.2) be convex. The problem (1.11.2) satisfies the Slater constraint qualification if there exists some $\mathbf{x}$ with $g_j(\mathbf{x}) < 0$ for all $j = 1, \ldots, m$.*

Necessary optimality conditions for Pareto optimality in those nondifferentiable problems where the objective functions are fractions of convex and concave functions are formulated in [Bhatia, Datta, 1985].

If an ordering cone D is used in defining efficiency, then the optimality conditions are similar to those presented above except the multipliers λ_i (just as in the differentiable case). The multipliers belong to the dual cone $D^* = \{\boldsymbol{\lambda} \in \mathbf{R}^k \mid \boldsymbol{\lambda}^T \mathbf{y} \geq 0 \text{ for all } \mathbf{y} \in D\}$. Because of the similarity, we do not present here separate optimality conditions for efficiency. Necessary and sufficient conditions for efficiency and weak efficiency are handled in [Wang, 1984]. In [Craven, 1989] and [El Abdouni, Thibault, 1992], necessary conditions for weak efficiency in normed spaces and Banach spaces, respectively, are presented. The objective and the constraint functions are still assumed to be locally Lipschitzian.

Direct counterparts of optimality conditions for proper Pareto optimality in the sense of Kuhn and Tucker, presented in Section 1.10, cannot be stated in the nondifferentiable case. The reason is that the definition of Kuhn and Tucker assumes continuous differentiability. However, a sufficient condition for proper Pareto optimality (in the sense of Geoffrion) when the objective and the constraint functions are compositions of convex, locally Lipschitzian functions are formulated in [Jeyakumar, Yang, 1993]. This treatment naturally includes ordinary convex, locally Lipschitzian functions. The authors also present necessary conditions for weak Pareto optimality and sufficient conditions of their own for Pareto optimality in problems with convex composite functions.

1.12. More Optimality Conditions

Many necessary and sufficient conditions for (weak, proper, or) Pareto optimality (or efficiency) have been suggested. They are based on different kind of assumptions on the properties and form of the problem. Many of them are based on some kind of scalarization of the original problem and conditions are set to both the objective functions and the scalarization parameters (some of such conditions are presented in the following chapter in connection with the scalarization methods). In this presentation, we settle for a closer handling of the Fritz John and the Karush-Kuhn-Tucker-type conditions, presented in the two earlier sections. For an interested reader we list some other references.

Necessary conditions for proper and improper Pareto optimality in the sense of Kuhn and Tucker are derived with the help of cones in [Tamura, Arai, 1982]. Geoffrion was in [Geoffrion, 1968] the first to give basic characterization of properly Pareto optimal solutions in terms of a scalar problem, called a weighting problem (see Section 2.2). He extended the results by a comprehensive theorem into necessary and sufficient conditions for local and global proper Pareto optimality. The treatment of Geoffrion's is closely followed in [Chou, Hsia, Lee, 1985], where properly Pareto optimal solutions are characterized for multiobjective optimization problems with set-valued functions.

In [Chankong, Haimes, 1982], the authors have modified the optimality conditions of Kuhn and Tucker for Pareto optimality to be suitable with certain solution methods (the ε-constraint method and the jth Lagrangian problem, see Section 2.3). Chankong and Haimes also propose optimality conditions for proper Pareto optimality (in the sense of Geoffrion) with the ε-constraint method. In [Benson, Morin, 1977], necessary and sufficient conditions are given for a Pareto optimal solution to be properly Pareto optimal. This is done with the help of the jth Lagrangian problem. Necessary and sufficient conditions for Pareto optimality with convex and differentiable functions partly based on the ε-constraint problem are proved in [Zlobec, 1984].

Necessary and sufficient conditions for Pareto optimality and proper Pareto optimality are proved with the help of duality theory and an auxiliary problem (1.9.1) (presented in Section 1.9) in [Wendell, Lee, 1977]. However, it is stated that nonlinear problems do not generally satisfy the conditions developed. In such cases Pareto optimal solutions have to be tested for proper Pareto optimality on a point-by-point basis.

In [Gulati, Islam, 1988], linear fractional objective functions and generalized convex constraints are handled. Necessary conditions of the Karush-Kuhn-Tucker-type are presented for Pareto optimal solutions and conditions under which Pareto optimal solutions are properly Pareto optimal are stated. Necessary and sufficient conditions for Pareto optimality for nonlinear fractional objective functions with nonlinear constraints are proved in [Lee, 1992]. In [Benson, 1979], Benson gives a necessary and sufficient condition for a point to be Pareto optimal when there are two concave objective functions (problem of maximization) and a convex feasible set.

The following references deal with conditions for efficiency, where the criterion space is ordered by an ordering cone.

In [Zubiri, 1988], necessary and sufficient conditions are proved for weak efficiency in Banach spaces with the help of a weighted L_∞-norm (see Section 2.7). Several necessary and sufficient conditions for efficient, weakly efficient and properly efficient solutions (in the sense of Borwein) in real topological linear spaces have been gathered in [Jahn, 1985]. Necessary and sufficient optimality conditions of the Karush-Kuhn-Tucker-type are derived in [Hazen, 1988] for the cases when preferences are and are not representable by cones.

Let us finally briefly mention some further references handling nondifferentiable cases. Necessary and sufficient conditions for Pareto optimality and proper Pareto optimality are derived in [Bhatia, Aggarwal, 1992] by the weighting method and

Dini derivatives (instead of subdifferentials introduced by Clarke). The functions in the problem are supposed to be nondifferentiable such that the objective functions are pseudoconcave and the constraint functions are quasiconvex. Also some duality results are provided.

Optimality conditions based on optimization theory of Dubovitskii and Milyutin presuming certain convexity assumptions are presented in [Censor, 1977] for Pareto optimality in $\mathbf{R}^n$ and in [Minami, 1981] for weak Pareto optimality in a linear topological space. No differentiability assumptions are needed. Necessary and sufficient conditions for weak, proper and Pareto optimality in finite dimensional normed spaces are presented in [Staib, 1991] under different assumptions.

1.13. Sensitivity Analysis and Stability

The last subject to be mentioned in this chapter is sensitivity analysis. Sensitivity analysis studies situations when the input parameters defining the multiobjective optimization problem change or have errors. In sensitivity analysis, an answer is sought to the question how much the parameters can be altered and varied without affecting the solution. More justification for sensitivity analysis is provided in [Rarig, Haimes, 1983].

Given a family of parametrized multiobjective optimization problems, a set-valued perturbation function is defined in [Tanino, 1990] such that it associates to each parameter value the set of Pareto optimal points of the perturbed feasible region. The behaviour of the perturbation function is analyzed both qualitatively and quantitavely. By stability we mean the study of various continuity properties of the perturbation function of a family of parametrized optimization problems, that is, qualitative analysis. By sensitivity we mean the study of the derivatives of the perturbation function, that is, quantitative analysis.

In general multiobjective optimization problems, plenty of attention has been paid to the stability of the preference structure of the decision maker. In these cases, it is usually assumed that the partial ordering of the criterion space is induced by an ordering cone.

Stability and sensitivity analysis are wide areas of research, and we do not intend to go into details here. Instead, we refer to, for example, [Tanino, Sawaragi, 1980], [Ester, 1984], [Papageorgiou, 1985], [Lucchetti, 1985], [Craven, 1988] and [Tanino, 1988(a), (b), 1990] for further analysis.

In MOLP problems, sensitivity analysis is usually applied to changes in the matrix of objective function coefficients, the vector of the right-hand sides of the constraints or the matrix of constraint coefficients. These matters are considered, for example, in [Deshpande, Zionts, 1980], [Gal, Leberling, 1981], [Fiala, 1983], [Bolintineanu, Craven, 1992] and [Antunes, Clímaco, 1992]. The last-mentioned paper handles the problem through scalarization (i.e., weighting method).

Changes that occur in the solution of an MOLP problem, if the number of objective functions, the number of variables or the number of constraint functions is altered are examined in [Eiselt, Pederzoli, Sandblom, 1987]. This is an interesting topic also for nonlinear problems, because an objective function may be forgotten from the model and it is good to know how this can affect the solution obtained. As an example we can mention that if a convex objective function is added to a convex multiobjective optimization problem, all weakly Pareto optimal solutions remain weakly Pareto optimal (see [Lowe, Thisse, Ward, Wendell, 1984]). The corresponding result is not always valid for Pareto optimal solutions. A counterexample can be found in [Steuer, 1986] on page 179. A corollary regarding the generation of the weakly Pareto optimal set of a convex problem as a union of such Pareto optimal sets where subsets of the objective functions are used, is proved in [Lowe, Thisse, Ward, Wendell, 1984].

Duality theory for nonlinear multiobjective optimization problems has been presented, for example, in [Bitran, 1981], [Luc, 1984(b), 1987], [Nakayama, 1984, 1985(b)], [Göpfert, 1986] and [Weir, 1987], for convex problems in [Martínez-Legaz, Singer, 1987], and for more general convex-like problems in [Wang, Li, 1992] and [Preda, 1992]. An overview of duality theory for linear and nonlinear cases has been gathered in [Nakayama, 1985(c)]. Some regularity results for multiobjective optimization problems are presented in [Martein, 1989].

Finally, we state that an excellent presentation about stability and duality in multiobjective optimization can be found in [Sawaragi, Nakayama, Tanino, 1985]. More than a third of the contents of the monograph handles these topics.

2. Methods for Multiobjective Optimization

Generating Pareto optimal solutions is an important phase in multiobjective optimization, and mathematically the problem is considered to be solved when the Pareto optimal set is found. The term *vector optimization* is sometimes used to denote the problem of identifying the Pareto optimal set. However, this is not always enough. We want to obtain just one solution. This means that we must find a way to order the Pareto optimal solutions in a complete order. This is why we need a decision maker and her or his preference structure. In this chapter, we present several methods for solving multiobjective optimization problems. Usually, this means finding a Pareto optimal solution that satisfies the decision maker best.

We are not here going to interfere with the formulation of a real-life phenomenon into a mathematically well-defined problem. We just emphasize that a proper formulation is important. For example, forgetting or misspelling an objective function may affect the Pareto optimal set. Several critical things to keep in mind in the formulation are listed in [Nijkamp, Rietveld, Spronk, 1988]. From among them we find sufficient representativity of the objective functions and precision of information.

In most methods we are interested in the criterion space instead of the decision variable space. One reason for this is that the dimension of the criterion space is usually considerably smaller than the dimension of the decision variable space. Another reason is that decision makers are more interested in the criterion values.

In general, multiobjective optimization problems are solved by scalarization. The most important exception for this are the MOLP problems. Some simplex-based solution methods have been created so that they can handle multiple objectives as they are without any scalarization to find Pareto optimal extreme points. Several authors have proposed different modifications of simplex tableaus into multiobjective simplex tableaus, see, for example, [Philip, 1972], [Evans, Steuer, 1973], [Zeleny, 1974], [Yu, Zeleny, 1975] and [Ignizio, 1985(b)]. Yu and Zeleny also propose a method for determining the whole Pareto optimal set. This is done after the Pareto optimal extreme solutions have been found by the multiobjective simplex method. Here we shall not examine the simplex variations more but we continue with scalarization methods.

As mentioned in Section 1.1, scalarization means converting the problem into a single or a family of single objective optimization problems with a real-valued objective function depending possibly on some parameters. This enables the use of the theory and methods of scalar optimization. The fundamental thing is that the optimal solutions of multiobjective optimization problems can be characterized as solutions of certain single objective optimization problems. For some scalarization methods the Pareto optimality of the solutions is guaranteed (see, for example, [Wierzbicki, 1986(b)]). Scalarizing functions usually depend on some auxiliary parameters. Some numerical difficulties may appear if the single objective optimization problem has feasible solutions only with very few parameter values or it is not solvable with all the parameter values. Thus the seemingly promising idea of simplifying the problem into single objective optimizations has also its weaknesses.

In [Sawaragi, Nakayama, Tanino, 1985], three requirements are set on a scalarizing function:

(1) It can cover any Pareto optimal solution.
(2) Every solution is Pareto optimal.

If the scalarizing function is based on aspiration levels, then, in addition

(3) Its solution is satisficing if the aspiration levels used are feasible.

Unfortunately, there is no scalarizing function that can satisfy all the three requirements.

An important fact to keep in mind is that standard routines for single objective optimization problems can only find local optima. This is why only locally Pareto optimal solutions are usually obtained and handled when dealing with scalarizing functions. Global Pareto optimality can be guaranteed, for example, if the objective and the constraint functions are convex (as stated in Theorem 1.2.3). A clustering-based method for obtaining globally Pareto optimal solutions is presented in [Törn, 1983]. In the following, the solutions are understood to be local, unless stated otherwise.

Results from other fields of research, for example, game theory, can be used in the solution processes. Rao has studied the relationship between Pareto optimal solutions and game theory in [Rao, 1987]. He has also applied his results to structural optimization.

There is a large variety of methods for multiobjective optimization. None of them can be said to be superior to all the others in general. When selecting a solution method, the specific features of the problem to be solved must be taken into consideration. In addition, the opinions of the decision maker are important. It is not enough if the analyst prefers some method but the decision maker cannot or does not want to use it. The decision maker may be busy and mathematically ignorant. One can say that selecting an appropriate multiobjective optimization method is a multiobjective optimization problem in itself! We shall treat the method selection problem in Section 5.2 after some methods have been presented.

Methods for multiobjective optimization can be classified in many ways according to different criteria. In [Cohon, 1985], they are categorized into two relatively distinct subsets: generating methods and preference-based methods. In generating methods, the set of Pareto optimal (or efficient) solutions is generated for the decision maker, who then chooses one of the alternatives. In preference-based methods, the preferences of the decision maker are taken into consideration as the solution process goes on and the solution that best satisfies the decision maker's preferences is selected.

Rosenthal suggests three classes of solution methods; partial generation of the Pareto optimal set, explicit value function maximization and interactive implicit value function maximization in [Rosenthal, 1985]. In [Carmichael, 1981], the methods are classified according to whether a composite single criterion, a single criterion with constraints, or many single criteria are the basis for the approach. A rough division could be done into interactive and noninteractive methods. Those classes can be further divided in many ways.

Here we apply the classification presented in [Hwang, Masud, 1979]. This classification has been followed, for instance, in [Hwang, Paidy, Yoon, Masud, 1980], [Buchanan, 1986] and [Lieberman, 1991(a), (b)]. Hwang and Masud have classified the methods according to the participation of the decision maker in the solution process. The classes are: 1) methods where no articulation of preference information is used (no-preference methods), 2) methods where a priori articulation of preference information is used (a priori methods), 3) methods where progressive articulation of preference information is used (interactive methods) and 4) methods where a posteriori articulation of preference information is used (a posteriori methods). The names in the parentheses are used in the following for short.

It is, however, stated in [Despontin, Moscarola, Spronk, 1983] that after looking at every possible classification, one has to conclude that every classification considered led to more than 50 % of exception. Thus one must keep in mind that the classifications are not absolute. Overlapping and combinations of the classes are possible and some methods can be considered to belong into more than one class. The presented grouping is only guiding.

We also consider another way of classification into *ad hoc* and *non ad hoc* methods. This way of division, suggested in [Steuer, Gardiner, 1991], has been mainly meant for interactive methods but can be applied to some other methods, too. It is based on the existence of an underlying value function. The common feature of ad hoc methods is that even if one knew the decision maker's value function, one could not exactly know how to respond to the questions posed by the algorithm. On the other hand, in non ad hoc methods the responses can be determined or at least confidently simulated if the decision maker's value function is known.

Before we examine the methods, we mention several references for further information.

A wide collection of methods available (up to the year 1983) has been gathered in [Despontin, Moscarola, Spronk, 1983]. Almost 100 methods for both multiobjective and multiattribute optimization have been included. A set of methods developed up till the year 1973 for both multiattribute decision analysis and multiobjective optimization has been collected in [MacCrimmon, 1973].

Extensive surveys of concepts and methods for multiobjective optimization are provided in the monographs [Chankong, Haimes, 1983(b)] and [Steuer, 1986]. Similar matters have been gathered and studied briefly in [Dyer, Sarin, 1981], [Rosenthal, 1985], [Buchanan, 1986], [Steuer, 1989(b)], [Steuer, Gardiner, 1990] and [Stewart, 1992]. An overview is given in [Evans, 1984] and several methods are also presented in [Cohon, 1985] and [Osyczka, 1984]. In [Hwang, Masud, 1979], a large number of methods is presented and illustrated by solving numerical examples in detail. A similar but shortened presentation is given in [Hwang, Paidy, Yoon, Masud, 1980]. The detailed solution process descriptions have been intended to help in selecting solution methods.

Overviews of multiobjective optimization methods in the former Soviet Union are presented in [Lieberman, 1991(a), (b)] and of theory and applications in China in [Hu, 1990]. Nine multiobjective optimization methods developed in Germany are briefly introduced in [Ester, Holzmüller, 1986].

A great number of interactive multiobjective optimization methods have been collected in [Shin, Ravindran, 1991]. Interactive methods are also presented in [White, 1983(b)] and [Narula, Weistroffer, 1989(a)]. Information about applications of the methods is reported, too. Some literature on interactive multiobjective optimization between the years 1965 and 1988 has been gathered in [Aksoy, 1990].

An overview of methods for MOLP problems can be found in [Zionts, 1980, 1989]. Methods for hierarchical multiobjective optimization problems have been reviewed in [Haimes, Li, 1988]. Such methods are needed in large-scale problems. A wide survey on the literature of hierarchical multiobjective analysis is also provided.

Several groups of methods applicable to computer-aided design systems are presented briefly in [Eiduks, 1983]. Methods for applications in structural optimization are reported in [Jendo, 1986], [Eschenauer, 1987], [Koski, Silvennoinen, 1987] and [Osyczka, Koski, 1989]. Methods with applications to industry and large-scale systems are presented in the monographs [Tabucanon, 1988] and [Haimes, Tarvainen, Shima, Thadathil, 1990], respectively. The collections of papers [Stadler, 1988(b)] and [Eschenauer, Koski, Osyczka, 1990] contain mainly applications in engineering.

In the following, we present several methods (in four classes) for multiobjective optimization. Some of them will be described in detail, some in outline, and some are just briefly mentioned. This presentation has not been intended to cover every existing method.

2.1. Methods Where A Posteriori Articulation of Preference Information Is Used

A posteriori methods can also be called methods for generating Pareto optimal solutions. After the Pareto optimal set (or a part of it) has been generated, it is presented to the decision maker, who selects the most preferred among the alternatives. The inconveniences here are that the generation is usually computationally expensive and sometimes at least partly difficult. On the other hand, it is hard for the decision maker to select from among a large set of alternatives. One more important question is how to present or display the alternatives to the decision maker in an effective way. The working order in these methods is: 1. analyst and 2. decision maker.

If there are only two objective functions the Pareto optimal set can be generated parametrically (see, for example, [Gass, Saaty, 1955] and [Benson, 1979]). When there are more objectives, the problem becomes more complicated.

In the MOLP problems the methods of this class can be divided into two subclasses. The first is for the methods which can find all the Pareto optimal solutions and the second is for the methods which can find only all the Pareto optimal extreme solutions. In the latter case, edges connecting Pareto optimal extreme points may be Pareto optimal or not.

The methods closely presented are called basic methods, since they are used frequently in practical problems, and they are also used as parts of some more developed methods. These basic methods are a weighting method and an ε-constraint method. After them, we give a limited overview of a method combining features of both the weighting and the ε-constraint method. Finally, some other methods of this class are briefly mentioned.

2.2. Weighting Method

In the weighting method, presented, for example, in [Gass, Saaty, 1955] and [Zadeh, 1963], the idea is to associate each objective function with a weighting factor and minimize the weighted sum of the objectives. In this way, the multiple objective functions are transformed into a single objective function. We suppose that the weighting coefficients w_i are real numbers such that $w_i \geq 0$ for all $i = 1, \ldots, k$. It is also usually supposed that the weights are normalized, that is, $\sum_{i=1}^{k} w_i = 1$. Then the multiobjective optimization problem (1.1.1) is modified into the following problem

$$\begin{array}{lll} & \text{minimize} & \sum_{i=1}^{k} w_i f_i(\mathbf{x}) \\ (2.2.1) & \text{subject to} & \mathbf{x} \in S, \\ & \text{where} & w_i \geq 0 \text{ for all } i = 1, \ldots, k, \text{ and} \\ & & \sum_{i=1}^{k} w_i = 1. \end{array}$$

Theoretical Results

In the following, several theoretical results are presented concerning the weighting method.

Theorem 2.2.2. *Let $\mathbf{x}^* \in S$ be a solution of the weighting problem (2.2.1). Then $\mathbf{x}^*$ is weakly Pareto optimal.*

Proof. Let $\mathbf{x}^* \in S$ be a solution of the weighting problem. Let us suppose that it is not weakly Pareto optimal. In this case, there exists a solution $\mathbf{x} \in S$ such that $f_i(\mathbf{x}) < f_i(\mathbf{x}^*)$ for all $i = 1, \ldots, k$. According to the assumptions set to the weighting coefficients, $w_j > 0$ for at least one j. Thus we have $\sum_{i=1}^{k} w_i f_i(\mathbf{x}) < \sum_{i=1}^{k} w_i f_i(\mathbf{x}^*)$. This is a contradiction with the assumption that $\mathbf{x}^*$ is a solution of the weighting problem. Thus $\mathbf{x}^*$ is weakly Pareto optimal. ▮

Theorem 2.2.3. *Let $\mathbf{x}^* \in S$ be a solution of the weighting problem (2.2.1), where the weighting coefficients are positive ($w_i > 0$ for all $i = 1, \ldots, k$). Then $\mathbf{x}^*$ is Pareto optimal.*

Proof. Let $\mathbf{x}^* \in S$ be a solution of the weighting problem. Let us suppose that it is not Pareto optimal. This means that there exists a solution $\mathbf{x} \in S$ such that $f_i(\mathbf{x}) \leq f_i(\mathbf{x}^*)$ for all $i = 1, \ldots, k$ and $f_j(\mathbf{x}) < f_j(\mathbf{x}^*)$ for at least one j. Since $w_i > 0$ for all $i = 1, \ldots, k$, we have $\sum_{i=1}^{k} w_i f_i(\mathbf{x}) < \sum_{i=1}^{k} w_i f_i(\mathbf{x}^*)$. This contradicts the assumption that $\mathbf{x}^*$ is a solution of the weighting problem. ▮

Theorem 2.2.4. *Let $\mathbf{x}^* \in S$ be a unique solution of the weighting problem (2.2.1). Then $\mathbf{x}^*$ is Pareto optimal.*

Proof. Let $\mathbf{x}^* \in S$ be a unique solution of the weighting problem. Let us suppose that it is not Pareto optimal. In this case, there exists a solution $\mathbf{x} \in S$ such that $f_i(\mathbf{x}) \leq f_i(\mathbf{x}^*)$ for all $i = 1, \ldots, k$ and $f_j(\mathbf{x}) < f_j(\mathbf{x}^*)$ for at least one j. Because all the weighting coefficients w_i are nonnegative, we have $\sum_{i=1}^{k} w_i f_i(\mathbf{x}) \leq \sum_{i=1}^{k} w_i f_i(\mathbf{x}^*)$. On the other hand, the uniqueness of $\mathbf{x}^*$ means that $\sum_{i=1}^{k} w_i f_i(\mathbf{x}^*) < \sum_{i=1}^{k} w_i f_i(\hat{\mathbf{x}})$ for all $\hat{\mathbf{x}} \in S$. The two inequalities above are contradictory and thus $\mathbf{x}^*$ must be Pareto optimal. ▮

As Theorems 2.2.3 and 2.2.4 state, the solution of the weighting method is always Pareto optimal if the weighting coefficients are all positive or if the solution is unique, without any further assumptions. The weakness of the weighting method is that not all of the Pareto optimal solutions can be found unless the problem is convex.

Theorem 2.2.5. *Let the multiobjective optimization problem be convex. If* $\mathbf{x}^* \in S$ *is Pareto optimal, then there exists a weighting vector* $\mathbf{w}$ *(*$w_i \geq 0$, $i = 1, \ldots, k$, $\sum_{i=1}^{k} w_i = 1$*) such that* $\mathbf{x}^*$ *is a solution of the weighting problem (2.2.1).*

Proof. The proof is presented after Theorem 2.3.9.

Figure 7 illustrates the contents of Theorem 2.2.5. On the left, every Pareto optimal solution along the fat line can be obtained. On the right, it is not possible to obtain the Pareto optimal solutions in the "hollow".

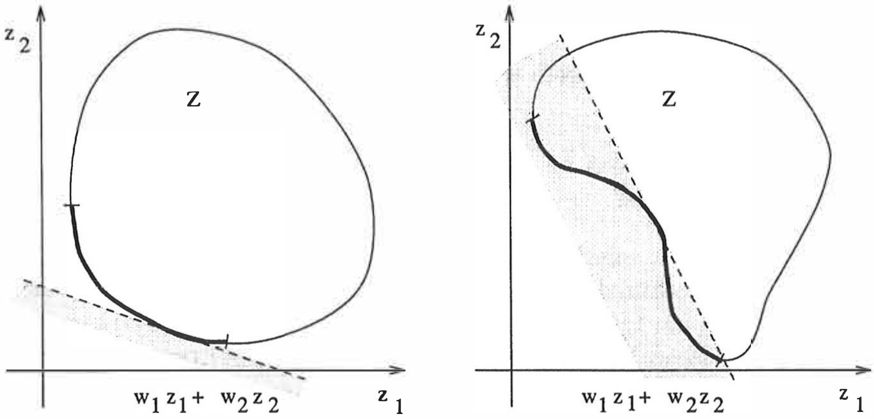


Figure 7. Weighting method with convex and nonconvex problems.

The similar result holds also for weakly Pareto optimal solutions of convex problems.

Remark. *According to Theorem 2.2.5, all the Pareto optimal solutions of the MOLP problems can be found by the weighting method.*

In practice, the result of the above remark is not quite true. The single objective optimization routines for linear problems usually find only extreme point solutions. Thus, if some facet of the feasible region is Pareto optimal, then the infinity of Pareto optimal non-extreme points must be described in terms of linear combinations of the Pareto optimal extreme solutions. On the other hand, notice that if two adjacent Pareto optimal extreme points for an MOLP problem are found, the edge connecting them is not necessarily Pareto optimal.

The conditions under which the whole Pareto optimal set can be generated by the weighting method with positive weighting coefficients are presented in [Censor, 1977]. The solutions that are possible to reach by the weighting method with positive weighting coefficients are characterized in [Belkeziz, Pirlot, 1991]. The authors also give some generalized results. More relations between nonnegative and positive weighting coefficients, convexity of S and Z and Pareto optimality are studied in [Lin, 1976(b)].

If the weighting coefficients in the weighting method are all positive, we can say more about the solutions than the Pareto optimality. The following results concerning proper Pareto optimality were originally presented in [Geoffrion, 1968].

Theorem 2.2.6. *Let $\mathbf{x}^* \in S$ be a solution of the weighting problem (2.2.1), when all the weighting coefficients w_i, $i = 1, \ldots, k$, are positive. Then $\mathbf{x}^*$ is properly Pareto optimal (sufficient condition).*

Proof. Let $\mathbf{x}^* \in S$ be a solution of the weighting problem with positive weighting coefficients. In Theorem 2.2.3 we showed that the solution is Pareto optimal. Now we shall show that $\mathbf{x}^*$ is properly Pareto optimal with $M = (k-1)\max_{i,j}(w_j/w_i)$.

Let us on the contrary suppose that $\mathbf{x}^*$ is not properly Pareto optimal. Then for some i (which we fix) and $\mathbf{x} \in S$ such that $f_i(\mathbf{x}^*) > f_i(\mathbf{x})$ we have

$$f_i(\mathbf{x}^*) - f_i(\mathbf{x}) > M(f_j(\mathbf{x}) - f_j(\mathbf{x}^*))$$

for all j such that $f_j(\mathbf{x}^*) < f_j(\mathbf{x})$. Now we can write

$$f_i(\mathbf{x}^*) - f_i(\mathbf{x}) > (k-1)\frac{w_j}{w_i}(f_j(\mathbf{x}) - f_j(\mathbf{x}^*)).$$

After multiplying both sides by $w_i/(k-1) > 0$, we get

$$\frac{w_i}{k-1}(f_i(\mathbf{x}^*) - f_i(\mathbf{x})) > w_j(f_j(\mathbf{x}) - f_j(\mathbf{x}^*)) \left(> 0 \geq w_l(f_l(\mathbf{x}) - f_l(\mathbf{x}^*))\right),$$

where $l \neq$ the fixed index i, and $l \neq$ the indices j, which were specified earlier. After this reasoning we can sum over all $j \neq i$ and obtain

$$w_i(f_i(\mathbf{x}^*) - f_i(\mathbf{x})) > \sum_{j \neq i}(w_j(f_j(\mathbf{x}) - f_j(\mathbf{x}^*))),$$

which means

$$\sum_{j=1}^{k} w_j f_j(\mathbf{x}^*) > \sum_{j=1}^{k} w_j f_j(\mathbf{x}).$$

Now we have a contradiction with the assumption that $\mathbf{x}^*$ is a solution of the weighting problem. ▮

Theorem 2.2.7. *If the multiobjective optimization problem is convex, then the condition in Theorem 2.2.6 is also necessary.*

Proof. See [Geoffrion, 1968] or [Chou, Hsia, Lee, 1985].

Corollary 2.2.8. *A necessary and sufficient condition for a point to be a properly Pareto optimal solution of an MOLP problem is that it is a solution of a weighting problem with all the weighting coefficients being positive.*

The weighting method is used in [Isermann, 1974] in proving that for linear multiobjective optimization problems all the Pareto optimal solutions are also properly Pareto optimal. Some results about weak, proper and Pareto optimality of the solutions obtained by the weighting method have been combined in [Wierzbicki, 1986(b)]. Proper Pareto optimality and the weighting method are also discussed in [Belkeziz, Pirlot, 1991].

Applications and Extensions

The weighting method is used to generate Pareto optimal solutions in [Sadek, Bruch, Sloss, Adali, 1988–89] when solving a problem of optimal control of a damped beam. The weighting method is also applied in [Weck, Förtsch, 1988] to structural systems in the optimization of a spindle bearing system and in the optimization of a table, and in [ReVelle, 1988] when the reductions of strategic nuclear weapons for the two superpowers are examined. In addition, the weighting method is an essential part in the determination of the optimal size of a batch system in [Friedman, Mehrez, 1992].

The weighting vector that produces a certain Pareto optimal solution is not necessarily unique. This is particularly true for linear problems. A method is presented in [Steuer, 1986] for determining ranges for weighting vectors that produce the same solution. Notice that some weighting vectors may produce unbounded single objective optimization problems. This does not mean that the problem could not have feasible solutions with some other weighting vectors.

A method for reducing the Pareto optimal set (of an MOLP problem) before it is presented to the decision maker is suggested in [Soloveychik, 1983]. Pareto optimal solutions are generated by the weighting method. Statistical analysis (factor analysis) is used to group and partition the Pareto optimal set into groups of relatively homogeneous elements. Finally, typical solutions from the groups are chosen and presented to the decision maker.

In [Hansen, Labbé, Wendell, 1989], sensitivity analysis is considered in solving linear multiobjective optimization problems by the weighting method. The authors determine the maximum percentage by which all the weighting coefficients can deviate simultaneously and independently from their values while the same optimal basis retains. They also consider the situation when ranges are given to the weighting coefficients. This case can be exploited to enlarge the maximum percentage.

The weighting method can be used so that the decision maker specifies a weighting vector representing her or his preference information. In this case, the weighted-sums problem can be considered (a negative of) a value function (remember that value functions are maximized). Notice that, according to Remark (1.7.5), the weighting coefficients provided by the decision maker are now nothing but marginal rates of substitution ($m_{ij} = w_j/w_i$). A method for assisting in the determination of the weighting coefficients is presented in [Batishchev, Anuchin, Shaposhnikov, 1991]. When the weighting method is used in this fashion, it can be considered to belong to the class of a priori methods. This method can also be extended into an interactive form by letting the decision maker modify the weighting vectors after each iteration.

Let us for a while consider the weighting method as an a priori method. A remarkable point is that the objective functions should be normalized or scaled so that their criterion values were approximately of the same magnitude. Only in this way one can control and manoeuvre the method to produce solutions of a desirable nature in proportion to the ranges of the objective functions. Otherwise the role of the weighting coefficients may be greatly misleading. It is suggested in [Steuer, 1986] that every objective function is multiplied by a *range equalization factor* K_i. The range R_i of f_i is estimated by the difference between the (possibly approximated) nadir point and the ideal criterion vector. Now,

$$K_i = \frac{1}{R_i} \frac{1}{\sum_{j=1}^{k} \frac{1}{R_j}}$$

for every $i = 1, \ldots, k$. A simpler way, suggested in [Osyczka, 1984, 1992], is to divide each objective function by its ideal criterion value. It is remarked in [Hobbs, 1986] that instead of relative importance, the weighting factors should represent the rate at which the decision maker is willing to trade off values of objective functions.

It must be noticed that if some of the objective functions correlate with each other, then seemingly "good" weighting vectors may produce poor results and seemingly "bad" weighting vectors may produce good results (see [Steuer, 1986], pp. 198–199, for an illustrative example).

Wierzbicki points out in [Wierzbicki, 1986(b)] that "experience in application of decision support systems shows that weighting coefficients are not easy to be interpreted and understood well by an average user."

A modification of the weighting method has been presented in [Kanniappan, 1988]. The decision maker can give either upper bounds or lower bounds or both of them for the weighting vector and the problem is solved with the aid of this information. An extended version of this method with an implementation is given in [Miettinen, 1990].

It is suggested in [Koski, Silvennoinen, 1987] that the weighting method can be used to reduce the number of the objective functions before the actual solution process. The original objective functions are divided into such groups that a linear combination of the objective functions in each group forms a new objective function, and these new objective functions form a new multiobjective optimization problem. The authors state that every Pareto optimal solution of the new problem is also a Pareto optimal solution of the original problems, but the reverse result is not generally true.

Concluding Remarks

Sometimes, the results concerning the weighting method are presented in a simpler form, assuming that zeros are not accepted as weighting coefficients. It may seem that the weighting factor zero makes no sense. It means that we have included in the problem some objective function that has no significance at all. Nevertheless, zero values have here been included to make the presentation more general. On the other hand, by allowing also zeros as weighting coefficients, it is easy to explore how the solutions change when some objective function is dropped.

Applying Theorem 2.2.3, we know that different Pareto optimal solutions can be obtained by the weighting method by altering the positive weighting coefficients. However, in practical calculations the condition $w_i \geq \varepsilon$, where $\varepsilon > 0$, must be used instead of the condition $w_i > 0$ for all $i = 1, \ldots, k$. All the Pareto optimal solutions in some convex problems may be found if ε is small enough, but the concept of "small enough" is problem-dependent and for this reason difficult to specify in advance, as pointed out in [Korhonen, Wallenius, 1989(a)].

As observed before, the weakness of the weighting method is that all of the Pareto optimal points cannot be found if the problem is nonconvex. If this is the case, a duality gap is said to occur (according to duality theory).

Employing the weighting method as an a priori method presumes that the decision maker's underlying value function is or can be approximated by a linear function (see Section 1.4). It is in many cases a rather simplifying assumption. It must be noticed that altering the weighting vectors linearly does not have to mean that the values of the objective functions should change linearly.

2.3. ε-Constraint Method

In the ε-constraint method, introduced in [Haimes, Lasdon, Wismer, 1971], one of the objective functions is selected to be optimized and all the other objective functions are converted into constraints by giving an upper bound for each of them. Now the problem to be solved is of the form

$$\begin{aligned} &\text{minimize} && f_\ell(\mathbf{x}) \\ &\text{subject to} && f_j(\mathbf{x}) \le \varepsilon_j \quad \text{for all } j = 1, \ldots, k, \; j \neq \ell, \\ & && \mathbf{x} \in S. \end{aligned} \tag{2.3.1}$$

In [Lin, 1976(a), (b)], Lin has proposed a method where proper equality constraints are used instead of the above-mentioned inequalities. The solutions obtained by Lin's method are Pareto optimal under certain assumptions. Here we, however, concentrate on the formulation (2.3.1).

Theoretical Results about Pareto Optimality

First, we prove a result concerning weak Pareto optimality.

Theorem 2.3.2. *Let $\mathbf{x}^* \in S$ be a solution of the ε-constraint problem (2.3.1). Then it is weakly Pareto optimal.*

Proof. Let $\mathbf{x}^* \in S$ be a solution of the ε-constraint problem. Let us assume that $\mathbf{x}^*$ is not weakly Pareto optimal. In this case, there exists some other $\mathbf{x} \in S$ such that $f_j(\mathbf{x}) < f_j(\mathbf{x}^*)$ for all $j = 1, \ldots, k$.

This means that $f_j(\mathbf{x}) < f_j(\mathbf{x}^*) \le \varepsilon_j$ for all $j = 1, \ldots, k$, $j \neq \ell$. Thus $\mathbf{x}$ is feasible for the ε-constraint problem. While in addition $f_\ell(\mathbf{x}) < f_\ell(\mathbf{x}^*)$, we have a contradiction with the assumption that $\mathbf{x}^*$ is a solution of the ε-constraint problem. ▮

Next, we handle Pareto optimality and the ε-constraint method.

Theorem 2.3.3. *A decision vector $\mathbf{x}^* \in S$ is Pareto optimal if and only if it is a solution of the ε-constraint problem (2.3.1) for every $\ell = 1, \ldots, k$, where $\varepsilon_j = f_j(\mathbf{x}^*)$ $(j = 1, \ldots, k, \; j \neq \ell)$.*

Proof. Necessity: Let $\mathbf{x}^*$ be Pareto optimal. Let us assume that it does not solve the ε-constraint problem for some ℓ where $\varepsilon_j = f_j(\mathbf{x}^*)$ $(j = 1, \ldots, k, \; j \neq \ell)$. Then there exists a solution $\mathbf{x} \in S$ such that $f_\ell(\mathbf{x}) < f_\ell(\mathbf{x}^*)$ and $f_j(\mathbf{x}) \le f_j(\mathbf{x}^*)$ when $j \neq \ell$. This contradicts the Pareto optimality of $\mathbf{x}^*$.

Sufficiency: Since $\mathbf{x}^*$ is by assumption a solution of the ε-constraint problem for every $\ell = 1, \ldots, k$, there is no $\mathbf{x} \in S$ such that $f_\ell(\mathbf{x}) < f_\ell(\mathbf{x}^*)$ and $f_j(\mathbf{x}) \le f_j(\mathbf{x}^*)$ when $j \neq \ell$. This is the definition of Pareto optimality for $\mathbf{x}^*$. ▮

Notice that according to the "necessity"-part of Theorem 2.3.3, it is possible to find every Pareto optimal solution of any multiobjective optimization problem by the ε-constraint method regardless of the convexity of the problem.

Theorem 2.3.4. *If $\mathbf{x}^* \in S$ is a unique solution of the ε-constraint problem (2.3.1) for some ℓ with $\varepsilon_j = f_j(\mathbf{x}^*)$ $(j = 1, \ldots, k,\ j \neq \ell)$, then $\mathbf{x}^*$ is Pareto optimal.*

Proof. Let $\mathbf{x}^* \in S$ be a unique solution of the ε-constraint problem for some ℓ. Let us assume that it is not Pareto optimal. In other words, there exists some point $\mathbf{x}^\circ \in S$ such that $f_i(\mathbf{x}^\circ) \leq f_i(\mathbf{x}^*)$ for all $i = 1, \ldots, k$ and for at least one index j is $f_j(\mathbf{x}^\circ) < f_j(\mathbf{x}^*)$. The uniqueness of $\mathbf{x}^*$ means that for all $\mathbf{x} \in S$ such that $f_i(\mathbf{x}) \leq f_i(\mathbf{x}^*)$, $i \neq \ell$, is $f_\ell(\mathbf{x}^*) < f_\ell(\mathbf{x})$. Here we have a contradiction with the preceding inequalities and $\mathbf{x}^*$ must be Pareto optimal. ∎

The following theorem is a straightforward extension of Theorem 2.3.4.

Theorem 2.3.5. *For any given upper bound vector $\boldsymbol{\varepsilon} = (\varepsilon_1, \ldots, \varepsilon_{\ell-1}, \varepsilon_{\ell+1}, \ldots, \varepsilon_k)^T$, an optimal solution of the ε-constraint problem (2.3.1) is Pareto optimal if it is unique.*

Proof. Let $\mathbf{x}^* \in S$ be a unique solution of the problem (2.3.1). This means that $f_\ell(\mathbf{x}^*) < f_\ell(\mathbf{x})$ for all $\mathbf{x} \in S$ when $f_j(\mathbf{x}^*) \leq \varepsilon_j$ for all $j \neq \ell$. Let us assume that $\mathbf{x}^*$ is not Pareto optimal. In this case, there exists a vector $\mathbf{x}^\circ \in S$ such that $f_i(\mathbf{x}^\circ) \leq f_i(\mathbf{x}^*)$ for all $i = 1, \ldots, k$ and the inequality is strict for at least one index j.

If $j = \ell$, this means that $f_\ell(\mathbf{x}^\circ) < f_\ell(\mathbf{x}^*)$ and $f_i(\mathbf{x}^\circ) \leq f_i(\mathbf{x}^*) \leq \varepsilon_j$ for all $i \neq \ell$. Here we have a contradiction with the fact that $\mathbf{x}^*$ is a solution of the ε-constraint problem.

On the other hand, if $j \neq \ell$, then $f_j(\mathbf{x}^\circ) < f_j(\mathbf{x}^*) \leq \varepsilon_j$, $f_i(\mathbf{x}^\circ) \leq f_i(\mathbf{x}^*) \leq \varepsilon_i$ for all $i \neq j$ and ℓ, and $f_\ell(\mathbf{x}^\circ) \leq f_\ell(\mathbf{x}^*)$. This is a contradiction with $\mathbf{x}^*$ being a unique solution of the ε-constraint problem. ∎

In Figure 8, different upper bounds for the objective function f_2 are given, while the function f_1 is to be minimized. The Pareto optimal set has been illustrated by a fat line. The upper bound level ε^1 is too tight and the feasible region becomes empty. On the other hand, the level ε^4 does not restrict the region at all. If it is used as the upper bound, the point $\mathbf{z}^4$ is obtained as a solution. It is Pareto optimal according to Theorem 2.3.5. Correspondingly, for the upper bound ε^3 the point $\mathbf{z}^3$ is obtained as a Pareto optimal solution. The point $\mathbf{z}^2$ is the optimal solution for the upper bound ε^2. Its Pareto optimality can be proved according to Theorem 2.3.4. Also Theorem 2.3.3 can be applied.

The relationships between the weighting method and the ε-constraint method are presented in the following theorems.

Theorem 2.3.6. *Let $\mathbf{w}$ be a weighting vector such that $\mathbf{x}^* \in S$ is a solution of the corresponding weighting problem (2.2.1). Then*

(1) *if $w_\ell > 0$, $\mathbf{x}^*$ is a solution of the ε-constraint problem for f_ℓ as the objective function and $\varepsilon_j = f_j(\mathbf{x}^*)$ $(j = 1, \ldots, k,\ j \neq \ell)$;* or

(2) *if $\mathbf{x}^*$ is a unique solution of the weighting problem (2.2.1), then $\mathbf{x}^*$ is a solution of the ε-constraint problem when $\varepsilon_j = f_j(\mathbf{x}^*)$ for every f_ℓ, $\ell = 1, \ldots, k$, as the objective function.*

Proof. Let $\mathbf{x}^* \in S$ be a solution of the weighting problem (2.2.1) for some weighting vector $\mathbf{w}$.

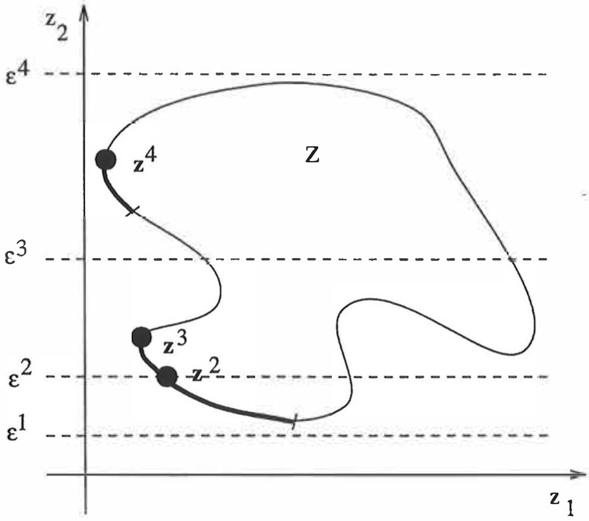


Figure 8. Different upper bounds for the ε-constraint method.

(1) In this case,

$$\sum_{i=1}^{k} w_i f_i(\mathbf{x}) \geq \sum_{i=1}^{k} w_i f_i(\mathbf{x}^*) \tag{2.3.7}$$

for all $\mathbf{x} \in S$.

Let us assume that $\mathbf{x}^*$ is not a solution of the ε-constraint problem. Then there exists a point $\hat{\mathbf{x}} \in S$ such that $f_\ell(\hat{\mathbf{x}}) < f_\ell(\mathbf{x}^*)$ and $f_j(\hat{\mathbf{x}}) \leq f_j(\mathbf{x}^*)$ when $j \neq \ell$. We assumed that $w_\ell > 0$ and $w_i \geq 0$ when $i \neq \ell$. Now we have

$$0 > w_\ell(f_\ell(\hat{\mathbf{x}}) - f_\ell(\mathbf{x}^*)) + \sum_{i \neq \ell} w_i(f_i(\hat{\mathbf{x}}) - f_i(\mathbf{x}^*)),$$

which is a contradiction with the inequality (2.3.7). Thus $\mathbf{x}^*$ is a solution of the ε-constraint problem.

(2) If $\mathbf{x}^*$ is a unique solution of the weighting problem, then for all $\mathbf{x} \subset S$ is

$$\sum_{i=1}^{k} w_i f_i(\mathbf{x}^*) < \sum_{i=1}^{k} w_i f_i(\mathbf{x}). \tag{2.3.8}$$

If there is some objective function f_ℓ such that $\mathbf{x}^*$ does not solve the ε-constraint problem when f_ℓ is to be minimized, then we can find a solution $\hat{\mathbf{x}} \in S$ such that $f_\ell(\hat{\mathbf{x}}) < f_\ell(\mathbf{x}^*)$ and $f_j(\hat{\mathbf{x}}) \leq f_j(\mathbf{x}^*)$ when $j \neq \ell$. This means that for any $\mathbf{w} \geq \mathbf{0}$ is $\sum_{i=1}^{k} w_i(f_i(\hat{\mathbf{x}}) - f_i(\mathbf{x}^*)) \leq 0$. This contradicts the inequality (2.3.8). Thus $\mathbf{x}^*$ is a solution of the ε-constraint problem for all f_ℓ to be minimized. ∎

Theorem 2.3.9. *Let the multiobjective optimization problem be convex. If $\mathbf{x}^* \in S$ is a solution of the ε-constraint problem (2.3.1) for any given f_ℓ to be minimized and $\varepsilon_j = f_j(\mathbf{x}^*)$ when $j \neq \ell$, then there exists a weighting vector $\mathbf{w}$ ($w_i \geq 0$, $\sum_{i=1}^{k} w_i = 1$) such that $\mathbf{x}^*$ is also a solution of the weighting problem (2.2.1).*

Proof. The proof needs a so-called generalized Gordon theorem. See [Chankong, Haimes, 1983(b)], p. 121.

Now we have appropriate tools for proving Theorem 2.2.5 from the previous section.

Proof of Theorem 2.2.5. Since $\mathbf{x}^*$ is Pareto optimal, it is by Theorem 2.3.3 a solution of the ε-constraint problem for every objective function f_ℓ to be minimized. The proof is completed with the aid of the convexity assumption and Theorem 2.3.9. ▌

A diagram representing several results about the characterization of Pareto optimal solutions and optimality conditions of the weighting method, the ε-constraint method and a so-called jth Lagrangian method, their relations and connections is presented in [Chankong, Haimes, 1982, 1983(b)]. The jth Lagrangian method (presented in [Benson, Morin, 1977]) means solving the problem

$$\begin{aligned} &\text{minimize} \quad f_j(\mathbf{x}) + \sum_{i \neq j} u_i f_i(\mathbf{x}) \\ &\text{subject to} \quad \mathbf{x} \in S, \end{aligned} \tag{2.3.10}$$

where $\mathbf{u} = (u_1, \ldots, u_{j-1}, u_{j+1}, \ldots, u_k)^T$ and $u_i \geq 0$ for all $i \neq j$. The jth Lagrangian method is in a computational viewpoint almost equal to the weighting method. This is why it is not studied more closely in this presentation. Chankong and Haimes have treated the problems separately to emphasize two ways of arriving at the same point.

Theoretical Results about Proper Pareto Optimality

Let us now return to the ε-constraint problem and the proper Pareto optimality of its solutions. In [Benson, Morin, 1977] an auxiliary function, called perturbation function, $v\colon \mathbf{R}^{k-1} \to \mathbf{R}$ associated with the ε-constraint problem has been defined in the form (modified here for the minimization problem)

$$v(\mathbf{y}) = \inf_{\mathbf{x} \in S} \{ f_\ell(\mathbf{x}) \mid f_j(\mathbf{x}) - \varepsilon_j \leq y_j \text{ for all } j \neq \ell \}.$$

(The optimal value of the objective function of the ε-constraint problem is $v(\mathbf{0})$.) Now we can define the stability of ε-constraint problems.

Definition 2.3.11. *The ε-constraint problem (2.3.1) is said to be **stable** when $v(\mathbf{0})$ is finite and there exists a scalar $R > 0$ such that, for all $\mathbf{y} \neq \mathbf{0}$*

$$\frac{v(\mathbf{0}) - v(\mathbf{y})}{\|\mathbf{y}\|} \leq R.$$

After this, a theorem concerning the proper Pareto optimality of the solutions of the ε-constraint problem can be presented.

Theorem 2.3.12. *Let the multiobjective optimization problem be convex and let $\mathbf{x}^* \in S$ be a Pareto optimal solution. Then $\mathbf{x}^*$ is properly Pareto optimal if and only if the ε-constraint problem (2.3.1) is stable for each $\ell = 1, \ldots, k$, where $\varepsilon_j = f_j(\mathbf{x}^*)$ for all $j \neq \ell$.*

Proof. See [Benson, Morin, 1977] or [Sawaragi, Nakayama, Tanino, 1985], p. 88.

Let us now suppose that the feasible region is of the form

$$S = \{\mathbf{x} \in \mathbf{R}^n \mid \mathbf{g}(\mathbf{x}) = (g_1(\mathbf{x}), g_2(\mathbf{x}), \dots, g_m(\mathbf{x}))^T \leq \mathbf{0}\}.$$

The ε-constraint problem is a constrained single objective optimization problem and it can be converted into an unconstrained problem by formulating a Lagrange function of the form

$$f_\ell(\mathbf{x}) + \sum_{j \neq \ell} \lambda_j (f_j(\mathbf{x}) - \varepsilon_j) + \sum_{i=1}^{m} \mu_i g_i(\mathbf{x}).$$

Setting some assumptions on the Lagrange multipliers $\boldsymbol{\lambda} \in \mathbf{R}^{k-1}$ and $\boldsymbol{\mu} \in \mathbf{R}^m$, we can get more conditions for proper Pareto optimality.

Before we can go on, we must define a so-called regular point.

Definition 2.3.13. *A point $\mathbf{x}^* \in S$ is said to be a regular point of the constraints of the ε-constraint problem if the gradients of the active constraints at $\mathbf{x}^*$ are linearly independent.*

(The vectors $\mathbf{v}^i$, $i = 1, \dots, m$, are linearly independent if the only weighting coefficients w_i for which $\sum_{i=1}^{m} w_i \mathbf{v}^i = 0$ are $w_i = 0$, $i = 1, \dots, m$.)

For clarity, we shall now formulate the classical necessary Karush-Kuhn-Tucker conditions for optimality (see [Kuhn, Tucker, 1951]) applied to the ε-constraint problem. The proof for general nonlinear problems is presented, for example, in [Kuhn, Tucker, 1951] and [Luenberger, 1984], p. 315. The conditions can also be derived from the optimality conditions for multiobjective optimization problems which were presented in Section 1.10. Notice that we assume here the above-defined constraint qualification, regularity.

Remark 2.3.14. *Necessary Karush-Kuhn-Tucker optimality conditions applied to the ε-constraint problem: Let the objective and the constraint functions be continuously differentiable. Let $\mathbf{x}^* \in S$ be a regular point of the constraints of the ε-constraint problem. If $\mathbf{x}^*$ is a (local) minimum of the ε-constraint problem, then there exists a set of (Kuhn-Tucker) multipliers $\mathbf{0} \leq \boldsymbol{\lambda} \in \mathbf{R}^{k-1}$ and $\mathbf{0} \leq \boldsymbol{\mu} \in \mathbf{R}^m$ such that*

$$\nabla f_\ell(\mathbf{x}^*) + \sum_{j \neq \ell} \lambda_j \nabla (f_j(\mathbf{x}^*) - \varepsilon_j) + \sum_{i=1}^{m} \mu_i \nabla g_i(\mathbf{x}^*) = 0$$

and

$$\lambda_j (f_j(\mathbf{x}^*) - \varepsilon_j) = 0 \text{ for all } j \neq \ell, \quad \mu_i g_i(\mathbf{x}^*) = 0 \text{ for all } i = 1, \dots, m.$$

Notice that the Lagrange multipliers are here called *Kuhn-Tucker multipliers*, when they are associated with the Karush-Kuhn-Tucker optimality conditions. Now we can present the following theorem.

Theorem 2.3.15. *Let $\mathbf{x}^* \in S$ be a regular point of the constraints of the ε-constraint problem and let all the objective and the constraint functions be continuously differentiable.*

(1) If $\mathbf{x}^*$ is a properly Pareto optimal solution, then $\mathbf{x}^*$ solves the ε-constraint problem for some f_ℓ being minimized and $\varepsilon_j = f_j(\mathbf{x}^*)$ $(j \neq \ell)$ with all the Kuhn-Tucker multipliers associated with the constraints $f_j(\mathbf{x}) \leq \varepsilon_j$ $(j \neq \ell)$ being strictly positive.

(2) If the multiobjective optimization problem is convex, then $\mathbf{x}^*$ is a properly Pareto optimal solution if it is a solution of the ε-constraint problem with the Kuhn-Tucker multipliers associated with the constraints being positive for all $j \neq \ell$.

Proof. See [Chankong, Haimes, 1983(b)], p. 143.

It can also be proved that if some solution is improperly Pareto optimal and the problem is convex, then some of the associated Kuhn-Tucker multipliers equal zero. On the other hand, if some of the Kuhn-Tucker multipliers equals zero, then the solution of the ε-constraint problem is improperly Pareto optimal (see, e.g., [Chankong, Haimes, 1983(a)]).

If the multiobjective optimization problem is solved by the ε-constraint method, the proper Pareto optimality can be checked by employing the Lagrange function. In the previous section in connection with the weighting method, we also presented some conditions for proper Pareto optimality. There are many methods where proper Pareto optimality is difficult to guarantee algorithmically. For the following two scalarization types, proper Pareto optimality has been studied.

In [Benson, Morin, 1977], proper Pareto optimality is characterized with the help of the jth Lagrangian problem (2.3.10). Wendell and Lee use in [Wendell, Lee, 1977] the function (1.9.1) that has characteristics from both the weighting method and the ε-constraint method. Optimality conditions for Pareto optimality and proper Pareto optimality are presented by assuming certain properties on the function and on the weighting coefficients.

Connections with Trade-Off Rates

The relationships between the Kuhn-Tucker multipliers and the trade-off rates have been studied in [Haimes, Chankong, 1979] and [Chankong, Haimes, 1983(b)]. Indeed, under certain conditions which are presented in the following, the Kuhn-Tucker multipliers of the Lagrange problem are equivalent to the trade-off rates.

For the convenience of notations we state the second-order sufficiency conditions for the ε-constraint problem. See [Chankong, Haimes, 1983(b)] for an exact mathematical formulation.

The *second-order sufficiency conditions* for the ε-constraint problem require that the optimality conditions of Remark 2.3.14 are satisfied and the Hessian matrix of the corresponding Lagrange function is positive definite on the subspace corresponding to the supporting hyperplane to the active constraints surface where the Kuhn-Tucker multipliers of the active constraints are strictly positive.

The connection between Kuhn-Tucker multipliers and trade-off rates is presented in the following theorem. The upper bound vector is denoted by $\boldsymbol{\varepsilon}^\circ \in \mathbf{R}^{k-1}$.

Theorem 2.3.16. *Let $\mathbf{x}^* \in S$ be a solution of the ε-constraint problem for some f_ℓ ($\ell = 1, \dots, k$) to be minimized (where the vector of upper bounds $\boldsymbol{\varepsilon}^\circ \in \mathbf{R}^{k-1}$ is chosen such that feasible solutions exist). Let $\lambda_j = \lambda_{\ell j}$ ($j \neq \ell$) be the corresponding Kuhn-Tucker multipliers associated with the constraints $f_j(\mathbf{x}) \leq \varepsilon_j^\circ$ ($j \neq \ell$). If*

(1) *$\mathbf{x}^*$ is a regular point of the constraints of the ε-constraint problem,*
(2) *the second-order sufficiency conditions are satisfied at $\mathbf{x}^*$, and*
(3) *there are no degenerate constraints at $\mathbf{x}^*$ (i.e., the Kuhn-Tucker multipliers of all the constraints are strictly positive),*

then $\lambda_{\ell j} = -\frac{d\, f_\ell(\mathbf{x}^*)}{d\, \varepsilon_j}$ for all $j \neq \ell$.

Proof. The proof is based on the implicit function theorem, see [Luenberger, 1984], p. 313.

From the assumption $\lambda_j(f_j(\mathbf{x}^*) - \varepsilon_j) = 0$ for all $j \neq \ell$ of the Karush-Kuhn-Tucker optimality conditions and the nondegeneracy of the constraints we know that $f_j(\mathbf{x}^*) = \varepsilon_j$ for all $j \neq \ell$. Thus from Theorem 2.3.16 we have trade-off rates

$$\lambda_{\ell j} = -\frac{d\, f_\ell(\mathbf{x}^*)}{d\, f_j(\mathbf{x}^*)} \text{ for all } j \neq \ell.$$

An important result about the relationships between the Kuhn-Tucker multipliers and trade-off rates in a more general situation, where also zero-valued multipliers are accepted, is presented in the following. For notational reasons we now suppose that the function to be minimized in the ε-constraint problem is f_k (i.e., we set $f_\ell = f_k$). This does not lose any generality. For details and a more extensive form of the theorem we refer to [Chankong, Haimes, 1983(b)].

Theorem 2.3.17. *Let $\mathbf{x}^* \in S$ be a solution of the ε-constraint problem (when f_k is minimized and $\boldsymbol{\varepsilon}^\circ \in \mathbf{R}^{k-1}$ is chosen so that feasible solutions exist) such that*

(1) *$\mathbf{x}^*$ is a regular point of the constraints of the ε-constraint problem,*
(2) *the second-order sufficiency conditions are satisfied at $\mathbf{x}^*$, and*
(3) *all the active constraints at $\mathbf{x}^*$ are nondegenerate.*

Let λ_{kj} be the optimal Kuhn-Tucker multipliers associated with the constraints $f_j(\mathbf{x}) \leq \varepsilon_j^\circ$, $j \neq k$. Without loss of generality we can assume that the first p ($1 \leq p \leq k-1$) of the multipliers are strictly positive (i.e., $\lambda_{kj} > 0$ for $j = 1, \dots, p$) and the rest $k-1-p$ multipliers equal zero (i.e., $\lambda_{kj} = 0$ for $j = p+1, \dots, k-1$). We denote the criterion vector corresponding to $\mathbf{x}^*$ by $\mathbf{z}^* \in Z$.

(1) *If $p = k-1$, that is, all the Kuhn-Tucker multipliers are strictly positive, then the Pareto optimal surface in the feasible criterion region in the neighbourhood of $\mathbf{z}^*$ can be represented by a continuously differentiable function $\bar{f}_k$ such that for each $(z_1, \dots, z_k)$ in the neighbourhood of $\mathbf{z}^*$ is $z_k = \bar{f}_k(z_1, \dots, z_{k-1})$. Moreover, for all $1 \leq j \leq p$*

$$\lambda_{kj} = -\frac{d\, \bar{f}_k}{d\, f_j}(z_1^*, \dots, z_{k-1}^*).$$

Thus λ_{kj} represents the partial trade-off rate between f_k and f_j at $\mathbf{x}^*$.

(2) *If $1 \leq p < k-1$, that is, some of the Kuhn-Tucker multipliers equal zero, then the Pareto optimal surface in the feasible criterion region in the*

neighbourhood of $\mathbf{z}^*$ *can be represented by continuously differentiable functions* $z_j = \bar{f}_j(z_1, \ldots, z_p, \varepsilon^\circ_{p+1}, \ldots, \varepsilon^\circ_{k-1})$ *for* $j = p+1, \ldots, k$. *Moreover, for all* $1 \le i \le p$

$$\lambda_{ki} = -\frac{d\bar{f}_k}{d f_i}(z_1^*, \ldots, z_p^*, \varepsilon^\circ_{p+1}, \ldots, \varepsilon^\circ_{k-1}) = \frac{\nabla f_k(\mathbf{x}^*)^T \mathbf{d}_i^*}{\nabla f_i(\mathbf{x}^*)^T \mathbf{d}_i^*},$$

where $\mathbf{d}^*$ *is the direction of* $d\mathbf{x}(\boldsymbol{\varepsilon}^\circ)/d\varepsilon_i$. *In addition, for all* $p+1 \le j \le k-1$,

$$\frac{d\bar{f}_j}{d f_i}(z_1^*, \ldots, z_p^*, \varepsilon^\circ_{p+1}, \ldots, \varepsilon^\circ_{k-1}) = \nabla f_j(\mathbf{x}^*)^T \frac{d\mathbf{x}(\boldsymbol{\varepsilon}^\circ)}{d\varepsilon_i}.$$

Thus λ_{ki} *represents the total trade-off rate between* f_k *and* f_i *at* $\mathbf{x}^*$ *in the direction of* $d\mathbf{x}(\boldsymbol{\varepsilon}^\circ)/d\varepsilon_i$.

Proof. See [Chankong, Haimes, 1983(b)], pp. 163–165.

Let us now take a look at the contents of Theorem 2.3.17. Part (1) says that under the specified conditions there are exactly $k-1$ degrees of freedom in specifying a point on the (locally) Pareto optimal surface in the criterion space in the neighbourhood of $\mathbf{z}^*$. In other words, when the values for $z_1, \ldots, z_{k-1}$ have been chosen from the neighbourhood of $\mathbf{z}^*$, then the value for z_k can be calculated from the given function and the resulting point $\mathbf{z}$ lies on the (locally) Pareto optimal surface in the criterion space.

Part (2) of Theorem 2.3.17 extends the result of part (1) by relaxing the assumption that all the constraints $f_j(\mathbf{x}) \le \varepsilon^\circ_j$, $j = 1, \ldots, k-1$, should be active and nondegenerate ($\lambda_{kj} > 0$ for all $j = 1, \ldots, k-1$). When the number of nondegenerate constraints is p ($< k-1$), then the degree of freedom in specifying a point on the (locally) Pareto optimal surface in the criterion space in the neighbourhood of $\mathbf{z}^*$ is the number of nondegenerate active constraints (p). The results of Theorem 2.3.17 will be needed in Section 2.13 when the ε-constraint method is used as a part of an interactive method.

Applications and Extensions

The ε-constraint method is used for generating Pareto optimal solutions in [Osman, Ragab, 1986(b)]. Then the solutions are clustered and a global Pareto optimum is located.

Sensitivity analysis with the ε-constraint method is dealt with in [Rarig, Haimes, 1983]. The authors define an index approximating the standard deviation of the optimal solution. The objective and the constraint functions are not supposed to be known for a certainty. The parameters describing the problem are treated as independent random variables with known finite mean values and variances. This index conveys to the decision maker information about the possibility of the actual solution deviating from the nominal (calculated) solution.

At this point it is worthwhile to mention a method for nonlinear problems presented in [Osman, Ragab, 1986(a)] which combines features from both the weighting method and the ε-constraint method. The nonconvex feasible criterion region is divided into convex and nonconvex parts. The positive feature of the weighting method that the feasible region is not disturbed in the solution process is utilized in convex parts, and the capability of the ε-constraint method to find all the Pareto optimal solutions is utilized in nonconvex parts. Therefore, merits of both these basic methods are exploited.

Concluding Remarks

Theoretically, every Pareto optimal solution of multiobjective optimization problems can be found by the ε-constraint method by altering the upper bounds and the function to be minimized. It must be underlined that even duality gaps in nonconvex problems (see, e.g., Section 1.9 and [Chankong, Haimes, 1983(b)]) do not disturb the functioning of the ε-constraint method. However, computationally, the conditions set by Theorems 2.3.3, 2.3.4 and 2.3.5 are not always very practical. For example, according to Theorem 2.3.3 the ε-constraint problem needs to be solved for all f_ℓ as objective functions in order to generate one Pareto optimal solution. On the other hand, the uniqueness of the solution demanded in the other theorems is not always so easy to check, either.

It may be difficult to specify the upper bounds for the objective functions. The components of the ideal criterion vector can be used to help in the specification. Then we can set $\varepsilon_j = z_j^\star + e_j$, where e_j is some relatively small positive real number which can be altered.

The ε-constraint method can also be used as an a priori method, where the decision maker specifies f_ℓ and the upper bounds. Then it can characterized as an ad hoc method. It means that one can never be completely sure how to select the objective function and the upper bounds to obtain a desired solution. This is a common weakness with the a priori weighting method. Computationally, the ε-constraint method is more laborious than the weighting method because the number of constraints increases.

2.4. Method of Corley

In [Corley, 1980], a method based on the minimization of a real-valued function subject to parametric constraints is introduced. Optimality is supposed to be defined by an ordering cone D as presented after Definition 1.2.5. Thus the method deals with efficient solutions. Features from both the weighting method and the ε-constraint method are incorporated. The original method has been here modified for minimization problems.

The problem to be solved is of the form

$$\begin{aligned} &\text{minimize} && \mathbf{d}^T\mathbf{f}(\mathbf{x}) \\ &\text{subject to} && \mathbf{f}(\mathbf{x}) - \mathbf{y} \in -D, \\ &&& \mathbf{x} \in S, \end{aligned} \tag{2.4.1}$$

where $\mathbf{y} \in \mathbf{R}^k$ is a parameter and $\mathbf{d} \in D^+ = \{\mathbf{d} \in \mathbf{R}^k \mid \mathbf{d}^T\mathbf{y} > \mathbf{0} \text{ for all } \mathbf{0} \neq \mathbf{y} \in D\}$ can be regarded as a weighting vector. Now we have the following result.

Theorem 2.4.2. *If $\mathbf{x}^* \in S$ solves the problem (2.4.1) for any $\mathbf{y} \in \mathbf{R}^k$, then $\mathbf{x}^*$ is efficient. On the other hand, if $\mathbf{x}^*$ is efficient, then $\mathbf{x}^*$ solves the problem (2.4.1) for $\mathbf{y} = \mathbf{f}(\mathbf{x}^*)$.*

Proof. See [Corley, 1980].

The set of efficient solutions can be found by solving the problem (2.4.1) with methods for parametric constraints (where the parameter is $\mathbf{y}$), see, for example, [Rao, 1984]. The Pareto optimal set is found by setting $D = \mathbf{R}^k_+$ and $D^+ = \{\boldsymbol{\beta} \in \mathbf{R}^k \mid \beta_i > 0,\ i = 1, \ldots, k\}$, see details in [Corley, 1980]. This approach for finding Pareto optimal solutions is closely related to the problem (1.9.1) in [Wendell, Lee, 1977].

2.5. Other A Posteriori Methods

Finally, we briefly mention some other methods of a posteriori type.

A so-called hyperplane method is introduced in [Yano, Sakawa, 1989] for generating Pareto optimal or properly Pareto optimal solutions. It is shown that the weighting method, the ε-constraint method and the weighted L_p-metric methods (in Sections 2.2, 2.3 and 2.7, respectively) can be viewed as special cases of the hyperplane methods. A theory concerning trade-off rates in the hyperplane method is provided in [Sakawa, Yano, 1990]. A generalized hyperplane method for generating all the efficient solutions (with respect to some cone) is presented in [Sakawa, Yano, 1992].

Another method for a general characterization of the Pareto optimal set is suggested in [Soland, 1979]. For example, the weighting method, the method of global criterion (see Section 2.7) and goal programming (see Section 2.11) can be seen as special cases of the general scalar problem of Soland. The weighting method and the ε-constraint method are utilized in a so-called envelope approach for determining Pareto optimal solutions in [Li, Haimes, 1987, 1988]. An application to dynamic multiobjective programming is also treated.

The noninferior (meaning here Pareto optimality) set estimation (NISE) method for MOLP problems can be considered to belong to this class of a posteriori methods, too. It is a technique for generating the Pareto optimal set of two objective functions (see [Cohon, 1978]). In [Balachandran, Gero, 1985], the method is extended for three objective functions. The weighting method is the basis of the NISE method. In [Armann, 1989], a method is presented for generating a dispersed subset of the Pareto optimal set which is then presented to the decision maker.

Multiobjective optimization problems with polynomial objective and constraint functions are treated in [Kostreva, Ordoyne, Wiecek, 1992]. The method for determining Pareto optimal solutions is based on the problem (1.9.5) and a so-called homotopy continuation. Notice that problems with polynomial functions are highly nonlinear, nonconvex and nonconcave. Such problems have not been handled before.

A scalarization method for multiobjective optimization problems, where optimality is defined through ordering cones, is suggested in [Pascoletti, Serafini, 1984]. By varying the parameters of the scalar problem it is possible to find all the efficient solutions. Further investigation is made in [Sterna-Karwat, 1987].

2.6. Methods Where No Articulation of Preference Information Is Used

In no-preference methods, where the opinions of the decision maker are not taken into consideration, the multiobjective optimization problem is solved with some relatively simple method and the solution obtained is presented to the decision maker. The decision maker may either accept or reject the solution. It seems quite unlikely that the solution best satisfying the decision maker could be found with these methods. That is why no-preference methods are suitable for such situations where the decision maker does not have any special expectations of the solution and (s)he is satisfied with just some optimal solution. The working order here is 1. analyst, 2. none.

As an example of this class we present the method of global criterion. This approach is also sometimes called compromise programming.

2.7. Method of Global Criterion

In the method of global criterion, the distance between some reference point and the feasible criterion region is minimized. The analyst has to select the reference point and the metric for measuring the distances. All the objective functions are thought to be equally important. Here we examine the situation where the ideal criterion vector is used as a reference point and L_p-metrics are used for measuring. In this case, the problem to be solved is

$$\text{(2.7.1)} \qquad \begin{aligned} &\text{minimize} \quad \left(\sum_{i=1}^{k} (f_i(\mathbf{x}) - z_i^\star)^p \right)^{1/p} \\ &\text{subject to} \quad \mathbf{x} \in S. \end{aligned}$$

From the definition of the ideal criterion vector $\mathbf{z}^\star$ we know that $f_i(\mathbf{x}) \geq z_i^\star$ for all $i = 1, \ldots, k$ and all $\mathbf{x} \in S$. This is why no absolute values are needed. The exponent $1/p$ may be dropped. The problems with or without the exponent $1/p$ are equivalent for $1 \leq p < \infty$ since the problem (2.7.1) is an increasing function of the corresponding problem without the exponent.

If $p = \infty$, the metric is also called a Tchebycheff metric and the problem is of the form

$$\text{(2.7.2)} \qquad \begin{aligned} &\text{minimize} \quad \max_{1 \leq i \leq k} [f_i(\mathbf{x}) - z_i^\star] \\ &\text{subject to} \quad \mathbf{x} \in S. \end{aligned}$$

Notice that the problem (2.7.2) is nondifferentiable. Anyway, it can be transformed into a differentiable form as in (2.7.11).

The solution obtained depends greatly on the value chosen for p. Widely used choices are $p = 1, 2$ or ∞. In Figure 9, the contours of these three different metrics are shown. The black point is the ideal criterion vector and the fat line represents the Pareto optimal set. It is worth noticing that if the original problem is linear, the choice $p = 1$ results in a linear problem. As the value of p increases, the nonlinear minimization problem becomes more difficult and badly conditioned to solve.

For linear problems, the solutions obtained by the L_p-metrics where $1 < p < \infty$ are situated between the solutions obtained by the L_1- and L_∞-metrics. It is illustrated in [Zeleny, 1973] that this set of solutions is a part of the Pareto optimal set, but only a substantially small part.

Instead of the terms $f_i(\mathbf{x}) - z_i^\star$, denominators may be added to the problems (2.7.1) and (2.7.2) to normalize the components, that is, to use the terms $\frac{f_i(\mathbf{x}) - z_i^\star}{|z_i^\star|}$ instead. Some other denominators can also be used. The reason for employing denominators is that sometimes it is worthwhile to use relative distances in the calculations. For example, using the components of $\mathbf{z}^\star$ forms the contour of the metric to reflect better the location of the ideal criterion vector. Of course, the denominators $z_i^\star$ cannot be used if some of them equals zero.

A variation of the L_∞-metric is suggested in [Osyczka, 1989(a), 1992]. There the problem to be solved is

$$\text{(2.7.3)} \qquad \begin{aligned} &\text{minimize} \quad \max_{1 \leq i \leq k} \left[\max \left\{ \left| \frac{f_i(\mathbf{x}) - z_i^\star}{z_i^\star} \right|, \left| \frac{f_i(\mathbf{x}) - z_i^\star}{f_i(\mathbf{x})} \right| \right\} \right] \\ &\text{subject to} \quad \mathbf{x} \in S. \end{aligned}$$

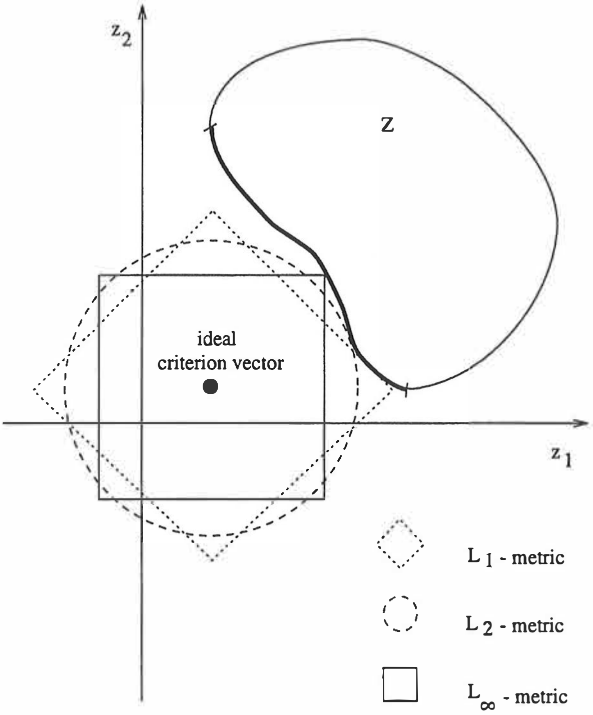


Figure 9. Different metrics.

The objective functions may be normalized by

$$\bar{f}_i(\mathbf{x}) = \frac{f_i(\mathbf{x}) - z_i^\star}{\max_{\mathbf{x}\in S} f_i(\mathbf{x}) - z_i^\star} \tag{2.7.4}$$

before the distance is minimized. In this case, the range of the new objective functions is $[0, 1]$. This normalizing is possible only if the objectives are bounded. However, it is usually better to employ the ranges of the Pareto optimal set and replace the max term by the component of the approximated nadir point z_i^{nad} in (2.7.4).

Theoretical Results

Next, we present some theoretical results concerning the method of global criterion.

Theorem 2.7.5. *Every solution of the problem (2.7.1) (where $1 \le p < \infty$) is Pareto optimal.*

Proof. Let $\mathbf{x}^* \in S$ be a solution of the problem (2.7.1) with $1 \le p < \infty$. Let us suppose that $\mathbf{x}^*$ is not Pareto optimal. Then there exists a point $\mathbf{x} \in S$ such that $f_i(\mathbf{x}) \le f_i(\mathbf{x}^*)$ for all $i = 1, \dots, k$ and $f_j(\mathbf{x}) < f_j(\mathbf{x}^*)$ for at least one j. Now $(f_i(\mathbf{x}) - z_i^\star)^p \le (f_i(\mathbf{x}^*) - z_i^\star)^p$ for all i and $(f_j(\mathbf{x}) - z_j^\star)^p < (f_j(\mathbf{x}^*) - z_j^\star)^p$. From this we obtain

$$\sum_{i=1}^{k} (f_i(\mathbf{x}) - z_i^\star)^p < \sum_{i=1}^{k} (f_i(\mathbf{x}^*) - z_i^\star)^p.$$

When both sides of the inequality are raised into the $1/p$th power we get a contradiction with the assumption that $\mathbf{x}^*$ is a solution of the problem (2.7.1). This completes the proof. ∎

Yu has pointed out in [Yu, 1973] that if Z is a convex set, then for $1 < p < \infty$ the solution of the problem (2.7.1) is unique.

Theorem 2.7.6. *Every solution of the problem (2.7.2) is weakly Pareto optimal.*

Proof. Let $\mathbf{x}^* \in S$ be a solution of the problem (2.7.2). Let us suppose that $\mathbf{x}^*$ is not weakly Pareto optimal. In this case, there exists a point $\mathbf{x} \in S$ such that $f_i(\mathbf{x}) < f_i(\mathbf{x}^*)$ for all $i = 1, \ldots, k$. It means that, $f_i(\mathbf{x}) - z_i^\star < f_i(\mathbf{x}^*) - z_i^\star$ for all i. Thus, $\mathbf{x}^*$ cannot be a solution of the problem (2.7.2). Here we have a contradiction which completes the proof. ∎

Theorem 2.7.7. *The problem (2.7.2) has at least one Pareto optimal solution.*

Proof. Let us suppose that none of the optimal solutions of the problem (2.7.2) is Pareto optimal. Let $\mathbf{x}^* \in S$ be an optimal solution of the problem (2.7.2). Since we assume that it is not Pareto optimal, there must exist a solution $\mathbf{x} \in S$ which is not optimal for the problem (2.7.2) but for which $f_i(\mathbf{x}) \leq f_i(\mathbf{x}^*)$ for all i (and $f_j(\mathbf{x}) < f_j(\mathbf{x}^*)$ for at least one j).

Now we have $f_i(\mathbf{x}) - z_i^\star \leq f_i(\mathbf{x}^*) - z_i^\star$ for all i with the strict inequality holding for at least one index j, and further $\max_i[f_i(\mathbf{x}) - z_i^\star] \leq \max_i[f_i(\mathbf{x}^*) - z_i^\star]$. Because $\mathbf{x}^*$ is an optimal solution of the problem (2.7.2), $\mathbf{x}$ has to be an optimal solution, too. This contradiction completes the proof. ∎

Corollary 2.7.8. *If the problem (2.7.2) has a unique solution, it is Pareto optimal.*

The L_∞-metric is utilized in [Ecker, Shoemaker, 1980, 1981] to obtain certain subsets of the Pareto optimal set in a linear case. Sufficient conditions for the solution of an L_p-metric problem to be stable with respect to changes of the feasible region S are presented in [Jurkiewicz, 1983].

Concluding Remarks

The method of global criterion is a simple method to be used if the aim is just to obtain a solution where no special hopes are set. The properties of the metrics imply that if the objective functions are not normalized anyhow, then such an objective function gets more importance whose ideal criterion value is situated nearer the feasible criterion region.

The solutions obtained with the L_p-metric ($1 \leq p < \infty$) are guaranteed to be Pareto optimal. If the L_∞-metric is used, the solution may be weakly Pareto optimal. In the latter case, for instance, the problem (1.9.5) can be used to produce Pareto optimal solutions. It is up to the analyst to select an appropriate metric.

Weighted L_p-Metrics

The method of global criterion can also be used to generate Pareto optimal solutions. In this case, weighting coefficients w_i are included in the metrics. At this point, we also briefly present such a method. It can be considered to belong to the classes of a posteriori or a priori methods.

We suppose that $w_i \geq 0$ for all i and $\sum_{i=1}^{k} w_i = 1$. Now the problems for minimizing distances are of the form

$$\begin{aligned} \text{minimize} \quad & \left(\sum_{i=1}^{k} w_i \big(f_i(\mathbf{x}) - z_i^\star\big)^p \right)^{1/p} \\ \text{subject to} \quad & \mathbf{x} \in S \end{aligned} \tag{2.7.9}$$

and

$$\begin{aligned} \text{minimize} \quad & \max_{1 \leq i \leq k} [w_i(f_i(\mathbf{x}) - z_i^\star)] \\ \text{subject to} \quad & \mathbf{x} \in S. \end{aligned} \tag{2.7.10}$$

Again, denominators may be included. Weighting vectors can also be used in connection with the problem of the form (2.7.3).

If $p = 1$, the sum of weighted deviations is minimized (and the problem to be solved is equal to the weighting method except a constant). If $p = 2$, we have a method of least squares. When p gets larger, the minimization of the largest deviation becomes more important. Finally, when $p = \infty$, the only thing that matters is the weighted relative deviation of one objective function.

The problem (2.7.10) is nondifferentiable as its unweighted counterpart. It can, however, be solved in a differentiable form as long as the objective and the constraint functions are differentiable. In this case, instead of the problem (2.7.10), the problem

$$\begin{aligned} \text{minimize} \quad & \alpha \\ \text{subject to} \quad & \alpha \geq w_i\big(f_i(\mathbf{x}) - z_i^\star\big), \quad \text{for all } i = 1, \ldots, k, \\ & \mathbf{x} \in S, \end{aligned} \tag{2.7.11}$$

is solved, where both $\mathbf{x}$ and $\alpha \in \mathbf{R}$ are variables. This formulation will be utilized later in this presentation.

In the following, we present some theoretical results concerning the weighted metrics. Most of the proofs so closely remind those presented earlier that there is no reason to repeat them.

Theorem 2.7.12. *The solution $\mathbf{x}^* \in S$ of the problem (2.7.9) (when $1 \leq p < \infty$) is Pareto optimal if (i) the solution is unique; or (ii) the weighting coefficients are positive ($w_i > 0$ for all $i = 1, \ldots, k$).*

Proof. The proof is not presented here since it follows directly from the proofs of Theorems 2.2.3, 2.2.4 and 2.7.5. See [Yu, 1973], or [Chankong, Haimes, 1983(b)], p. 144.

Theorem 2.7.13. *Every solution of the problem (2.7.10) is weakly Pareto optimal if $w_i > 0$ for all $i = 1, \ldots, k$.*

Proof. The proof is a straightforward modification of the proof of Theorem 2.7.6.

Theorem 2.7.14. *The problem (2.7.10) has at least one Pareto optimal solution.*

Proof. The proof follows directly from the proof of Theorem 2.7.7.

Convexity of the multiobjective optimization problem is needed in order to guarantee that every Pareto optimal solution can be found by the weighted L_p-metric (2.7.9). The following theorem shows that, on the other hand, every Pareto optimal solution can be found by the weighted L_∞-metric (2.7.10).

Theorem 2.7.15. *Let $\mathbf{x}^* \in S$ be a Pareto optimal solution. Then there exists a weighting vector $\mathbf{w}$, where $w_i > 0$ for all $i = 1, \ldots, k$, such that $\mathbf{x}^*$ is an optimal solution of the problem (2.7.10), where the reference point is $\mathbf{z}^{**} = \mathbf{z}^\star - \boldsymbol{\varepsilon}$ (where $\boldsymbol{\varepsilon} \in \mathbf{R}^k$ is any vector with $\varepsilon_i > 0$ for all $i = 1, \ldots, k$).*

Proof. Let $\mathbf{x}^* \in S$ be a Pareto optimal solution. Let us assume that there does not exist a weighting vector $\mathbf{w} > \mathbf{0}$ such that $\mathbf{x}^*$ is a solution of the problem (2.7.10). We know that $f_i(\mathbf{x}) > z_i^{**}$ for all $i = 1, \ldots, k$ and for all $\mathbf{x} \in S$. Now we choose $w_i = \frac{\beta}{f_i(\mathbf{x}^*) - z_i^{**}}$ for all $i = 1, \ldots, k$, where $\beta > 0$ is some normalizing factor.

According to the antithesis, there exists another point $\mathbf{x}^\circ \in S$ that is not Pareto optimal and that is a solution of the problem (2.7.10), meaning that

$$\max_i \left[w_i(f_i(\mathbf{x}^\circ) - z_i^{**})\right] \leq \max_i \left[w_i(f_i(\mathbf{x}^*) - z_i^{**})\right]$$
$$= \max_i \left[\frac{\beta}{f_i(\mathbf{x}^*) - z_i^{**}}(f_i(\mathbf{x}^*) - z_i^{**})\right] = \beta.$$

Thus $w_i(f_i(\mathbf{x}^\circ) - z_i^{**}) \leq \beta$ for all $i = 1, \ldots, k$. This means that

$$\frac{\beta}{f_i(\mathbf{x}^*) - z_i^{**}}(f_i(\mathbf{x}^\circ) - z_i^{**}) \leq \beta,$$

and after simplifying the expression we have

$$f_i(\mathbf{x}^\circ) \leq f_i(\mathbf{x}^*)$$

for all $i = 1, \ldots, k$. Because $\mathbf{x}^*$ is Pareto optimal, we must have $f_i(\mathbf{x}^\circ) = f_i(\mathbf{x}^*)$ for all i. Because $\mathbf{x}^\circ$ is a solution of the problem (2.7.10), also $\mathbf{x}^*$ must be. Here we have a contradiction which completes the proof. ▮

The theorem above sounds quite promising for the weighted L_∞-metric. Unfortunately, the metric is not so overwhelming as one could deduce. In addition to the fact that every Pareto optimal solution can be found, weakly Pareto optimal solutions may also be included. Auxiliary calculation is needed in order to identify the weak ones.

An interesting result concerning trade-off rates and the weighted L_∞-metric has been proved in [Yano, Sakawa, 1987]. The problem (2.7.11) is formulated as a Lagrange function and the trade-off rates are obtained from the Kuhn-Tucker multipliers under certain assumptions. This procedure is not treated here in more detail because of its close similarities with the approach presented in Section 2.3 in connection with the ε-constraint method. A differentiating factor is that the weighting coefficients have an essential role in these trade-off rates.

Applications and Extensions of Weighted L_p-Metrics

A shape optimization problem of a spillway profile is solved by the weighted L_2-metric (with denominators) in [Wang, Zhou, 1990]. An extension of the weighted method of global criterion called composite programming is presented in [Bárdossy, Bogárdi, Duckstein, 1985]. The method is applied to a problem of multiobjective watershed management.

More results about the properties of the L_p-metrics ($1 \leq p < \infty$) with and without the weighting coefficients can be found in [Yu, 1973], [Chankong, Haimes, 1983(b)], [Nakayama, 1985(a)], [Koski, Silvennoinen, 1987] and [Bowman, 1976], the last one treating especially the L_∞-metric. Some results about the proper efficiency (in the sense of Henig) of the solutions of the weighted L_p-metric are presented briefly in [Wierzbicki, 1986(b)].

One way to apply the weighted L_∞-metric successfully will be introduced in Section 2.17. The weighted L_∞-metric may also be augmented (see Section 2.17). In this case, all the solutions are Pareto optimal but not all of them are necessarily found. More information about the augmented weighted L_∞-metric is presented, for example, in [Steuer, Choo, 1983] and [Steuer, 1986]. Necessary and sufficient conditions for proper efficiency (in the sense of Henig) are provided in [Kaliszewski, 1985, 1986] with the help of the augmented weighted L_∞-metric.

Results for a generalized L_∞-metric approach are presented in [Dauer, Osman, 1985]. An extended generalized L_∞-metric is presented to characterize properly Pareto optimal solutions of nonconvex multiobjective optimization problems in [Choo, Atkins, 1983].

2.8. Methods Where A Priori Articulation of Preference Information Is Used

In the a priori methods, the decision maker must specify her or his preferences, hopes and opinions before the solution process. The difficulty is that the decision maker does not necessarily know beforehand what is possible to attain in the problem and how realistic the expectations are. The working order in these methods is: 1. decision maker and 2. analyst.

In the following, we handle three a priori methods. The approach of value function optimization was already mentioned earlier. Here we present it again briefly. Then we introduce the lexicographic ordering and goal programming.

2.9. Value Function Method

Introduction

In the value function method, the decision maker must be able to give an accurate and explicit mathematical form of the value function $U\colon \mathbf{R}^k \to \mathbf{R}$ that represents her or his preferences globally. The function provides a complete ordering in the criterion space. Then the problem

$$(2.9.1) \qquad \begin{aligned} &\text{maximize} && U(\mathbf{f}(\mathbf{x})) \\ &\text{subject to} && \mathbf{x} \in S \end{aligned}$$

is to be solved by some method for single objective optimization as illustrated in Figure 10. The fat line represents the Pareto optimal set. Remember Theorem 1.4.2, which says that the solution of the problem (2.9.1) is Pareto optimal if the value function is componentwise decreasing.

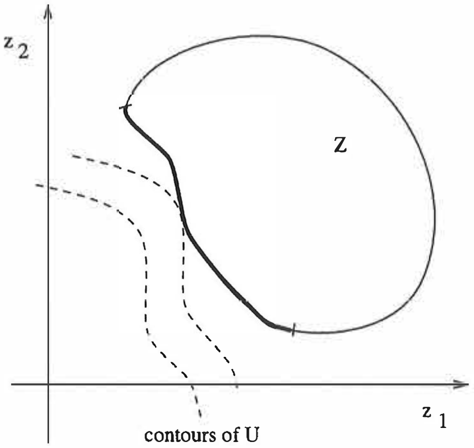


Figure 10. Contours of the value function.

The value function method seems to be a very simple method, but the difficulty lies in specifying the mathematical expression of the value function. This unability to encode the decision maker's underlying value function reliably is demonstrated in [deNeufville, McCord, 1984] by experiments. It is shown that encoding methods which should theoretically produce identical value functions fail; the functions may differ from each other by more than 50 %. It is also pointed out that there is no actual analysis of the accuracy of the value function assessment. The consistency checks, that is, whether decision makers provide consistent answers to similar questions, are not adequate: a biased instrument can provide consistent data.

On the other hand, if it were possible for the decision maker to express her or his preferences globally, the resulting preference structure might be too simple, since value functions cannot represent intransitivity or incomparability (see [Rosinger, 1985]). More features and weaknesses were presented in connection with the definition of the value function (Definition 1.4.1) in Section 1.4.

The value function method could be called an "optimal" way of solving multiobjective optimization problems, if the decision maker could reliably present the value function. The use of the value function method is restricted in practice to multiattribute decision analysis problems with a discrete set of feasible alternatives. The theory of value and utility functions for multiattribute problems has been examined broadly in [Keeney, Raiffa, 1976]. It is believed, in [Rosenthal, 1985], that the experiences can be utilized also in continuous cases.

Important results about value functions and conditions for their existence have been gathered in [Dyer, Sarin, 1981]. Two general classes of value functions, additive and multiplicative forms, are presented widely in [Keeney, Raiffa, 1976] and briefly in [Rosenthal, 1985]. General properties and some desirable features of certain types of value functions (e.g., max-min, min-sum and exponential forms) are stated in [Soland, 1979], [Stam, Lee, Yu, 1985], [Choo, Chew, 1985], [Bell, 1986], [Harrison, Rosenthal, 1988] and [Sounderpandian, 1991]. More examples of value functions are given in [Tell, Wallenius, 1979]. Relations between value functions, ordering cones and (proper) efficiency are studied in [Henig, 1990].

In some interactive methods, it is supposed that the value function is of some particular (e.g., additive or exponent) form and then its parameters are fitted according to the decision maker's preferences. Such methods are presented, for example, in [Sakawa, Seo, 1980, 1982(a), (b)] (see Section 2.15) and [Rothermel, Schilling, 1986]. Three kinds of conditions for value functions under which it is not possible to exclude any Pareto optimal or properly Pareto optimal solution from consideration a priori are identified in [Soland, 1979].

Relationships between the method of global criterion (see Section 2.7) and the value function method are investigated in [Ballestero, Romero, 1991]. One could think that there is nothing in common between those methods, since a value function represents the opinions of the decision maker and the method of global criterion does not take the decision maker into consideration. However, conditions can be set to the value function to guarantee that its optimum belongs to the solution set obtainable by the method of global criterion.

Concluding Remarks

The value function method is an excellent method if the decision maker happens to know an explicit mathematical formulation of the value function and if that function represents totally the preferences of the decision maker. These two serious preconditions are the difficulties of the approach.

There are certain conditions which the decision maker's preferences must satisfy so that a value function can be defined on them. The decision maker must, for instance, be able to specify consistent (implying transitive) preferences. Thus, there does not necessarily exist a value function which imposes a total order in the set of feasible criterion vectors. It is reminded in [Polak, Payne, 1976] that the assumption of a total order is often contrary to our intuitive aims and hence is quite likely to lead to less than ideal selections. This fact must be kept in mind in the following, when several methods are introduced which assume the existence of a value function (at least implicitly).

One important thing to keep in mind in practice is that the aspirations of the decision maker may change during the solution process. A notable question is aroused in [Steuer, Gardiner, 1990]: "Does this mean that the decision maker's value function can change considerably during a short time and is thus unstable or is it so difficult for the decision maker to really know the value function without interaction with the solution process?" More open questions concerning value functions have been listed in [Nijkamp, Rietveld, Spronk, 1988].

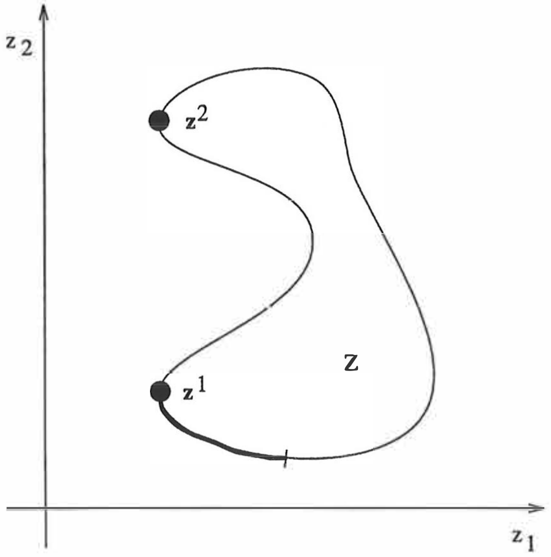
The weighting method presented in Section 2.2 may be regarded as a special case of a value function, where the utilities are linear and additive. If the underlying value function is assumed to be linear, this means that the marginal rates of substitution of the decision maker are constant at every solution. See comments on this feature in Section 2.11.

2.10. Lexicographic Ordering

Introduction

In the lexicographic ordering the decision maker must arrange the objective functions according to their absolute importance. This ordering means that the more important objective is infinitely more important than the 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 solution of the whole multiobjective optimization problem. Otherwise, the second most important objective function is minimized. Now, in addition to the original constraints, a new constraint is added. This new constraint is there to guarantee that the most important objective function preserves its optimal value. If this problem has a unique solution, it is the solution of the original problem. Otherwise, the process goes on accordingly. Lexicographic orders and utilities are widely examined in [Fishburn, 1974].

An example of lexicographic ordering is presented in Figure 11. There are two objective functions where the first one is the most important. After minimizing the first objective, there are two points left and after minimizing the second objective, the point $\mathbf{z}^1$ is obtained as the final solution. The fat line represents the Pareto optimal set in the figure. This example is somewhat too positive since all the objective functions have their effect on the solution process.



Figurc 11. Lcxicographic ordering.

Now we can present the following result about the Pareto optimality of the solutions.

Theorem 2.10.1. *A solution obtained by the lexicographic ordering is Pareto optimal.*

Proof. Let $\mathbf{x}^* \in S$ be a solution obtained by the lexicographic ordering. Let us assume that it is not Pareto optimal. In this case, there exists a point $\mathbf{x} \in S$ ($\mathbf{x} \neq \mathbf{x}^*$)

such that $f_i(\mathbf{x}) \leq f_i(\mathbf{x}^*)$ for all $i = 1, \ldots, k$ and for at least one j the inequality is strict, that is, $f_j(\mathbf{x}) < f_j(\mathbf{x}^*)$.

Let be $i = 1$. From the definition of the lexicographic ordering we know that f_1 attains its minimum at $\mathbf{x}^*$. Since also $f_1(\mathbf{x}) \leq f_1(\mathbf{x}^*)$, it is only possible that $f_1(\mathbf{x}) = f_1(\mathbf{x}^*)$.

There are two possibilities in determining the lexicographic optimum. Either a unique solution is found during the optimization process, or optimizations are performed for every $i = 1, \ldots, k$. In the latter case, when $i = 2$, we also have $f_2(\mathbf{x}) = f_2(\mathbf{x}^*)$ and with similar reasoning we have that $f_i(\mathbf{x}) = f_i(\mathbf{x}^*)$ for every $i = 1, \ldots, k$. This contradicts the assumption of strict inequality. Thus, $\mathbf{x}^*$ is Pareto optimal.

On the other hand, if the lexicographic ordering stops before every objective function has been examined, this means that a unique solution $\mathbf{x}^*$ has been obtained for f_i. The assumption $f_i(\mathbf{x}) \leq f_i(\mathbf{x}^*)$ implies that $f_i(\mathbf{x}) = f_i(\mathbf{x}^*)$, which is a contradiction. Thus, $\mathbf{x}^*$ is Pareto optimal. ∎

In [Ben-Tal, 1980], Pareto and lexicographic optima are characterized in convex problems. Duality theory for convex problems with the help of the lexicographic ordering is developed in [Martínez-Legaz, 1988].

The lexicographic ordering corresponds to the weighting method when the weighting coefficients are of very different magnitude. The question whether there exist weighting vectors such that the optimal solution of the weighting method is identical to the solution obtained by the lexicographic ordering is considered in [Sherali, 1982] and [Sherali, Soyster, 1983]. The answer is positive for linear problems and several discrete problems. In practice, this means that the problem of lexicographic ordering can be solved as a weighting problem with standard programs.

Concluding Remarks

The justification of using the lexicographic ordering is its simplicity and the fact that people usually make decisions successively. However, this method has several drawbacks. The decision maker may have difficulties in putting the objective functions into an absolute order of importance. On the other hand, the method is usually robust. It is very likely that the less important objective functions are not taken into the consideration at all. If the most important objective function has a unique solution, the other objectives do not have any influence on the solution. And even if the most important objective had alternative optima and the second most important objective could be used, it is very unlikely that this problem would have alternative optima, and the third or other less important objectives could be used.

Notice that the 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 function. Yet, many times this kind of trading might be appealing to the decision maker.

The lexicographic ordering may be used as a part of the following solution method, called goal programming.

2.11. Goal Programming

The ideas of goal programming were originally introduced in [Charnes, Cooper, Ferguson, 1955] but the term goal programming was fixed in [Charnes, Cooper, 1961]. It is one of the first methods really created for multiobjective optimization. Among more recent papers, an easy-to-understand presentation of goal programming is given in [Ignizio, 1983(a), 1985(a)].

Introduction

The basic idea in goal programming is that the decision maker specifies (optimistic) aspiration levels to the objective functions and the deviations from these aspiration levels are minimized. An objective function jointly with an aspiration level forms a *goal*. We can say that minimizing the prize of a product is an objective function, but if we want the prize to be less than 500 FIM, it is a goal (and if the prize must be less than 500 FIM, it is a constraint). We denote the aspiration level of the ith objective function by $\bar{z}_i$ $(i = 1, \dots, k)$.

For minimization problems, the goals are of the form $f_i(\mathbf{x}) \leq \bar{z}_i$. The goals may also be represented as equalities or ranges (for the latter situation see [Charnes, Cooper, 1977]). The aspiration levels are supposed to be selected such that they are not achievable simultaneously.

It is worth noticing that the goals are of the same form as the constraints of the problem. This is why the constraints may be regarded as a subset of the goals. This way of formulating the problem is called *generalized goal programming*. In this case, the goals can be thought of being divided into flexible and inflexible goals, where the constraints are the inflexible (or rigid) ones. More detailed presentations and practical applications of generalized goal programming are given, for example, in [Ignizio, 1983(a)] and [Korhonen, 1991(a)]. See also Section 2.21.

After the aspiration levels have been specified, the following task is to minimize the under- and overachievements of the objective function values with respect to the aspiration levels. It is sufficient to study the deviational variables $\delta_i = \bar{z}_i - f_i(\mathbf{x})$. The deviational variable δ_i may have positive or negative values, depending on the problem. We can present it as a difference of two positive variables, that is, $\delta_i = \delta_i^- - \delta_i^+$. Now we can investigate how well each of the aspiration levels is attained by studying the deviational variables. We can write $f_i(\mathbf{x}) + \delta_i^- - \delta_i^+ = \bar{z}_i$ for all i, where δ_i^- is a negative deviation (underachievement) and δ_i^+ is a positive deviation (overachievement) from the aspiration level. It is valid that $\delta_i^- \cdot \delta_i^+ = 0$ for all i.

Now we have the multiobjective optimization problem in a form where we minimize the deviational variables. For minimization problems it is sufficient to minimize δ_i^+:s. If the ith goal is in the form of an equality, we minimize $\delta_i^- + \delta_i^+$.

Two Approaches

So far, we have only written the multiobjective optimization problem in an equivalent form, where we have deviational variables as the objective functions. There are several possibilities to proceed from this point. Here we present an Archimedian and a preemptive approach. More methods are handled in [Ignizio, 1983(a)] and some formulations are explored in [De Kluyver, 1979].

In the *Archimedian* approach (originally presented by Charnes and Cooper), the weighted sum of the deviational variables is minimized. This means that in addition to the aspiration levels, the decision maker must specify information about the importance of attaining the aspiration levels (in the form of weighting coefficients). The weighting coefficients are supposed to be positive and sum up to one. The bigger the weighting factor is, the more important is the attainment of that aspiration level. (Sometimes negative weighting coefficients are used to represent a premium instead of a penalty.)

To put the above-presented introduction into a mathematical form, we can say that the problem

$$\text{(2.11.1)} \qquad \begin{array}{ll} \text{minimize} & \sum_{i=1}^{k} w_i |\bar{z}_i - f_i(\mathbf{x})| \\ \text{subject to} & \mathbf{x} \in S \end{array}$$

is converted into a new form by adding the overachievement variables

$$\delta_i^+ = \max\,[0,\ f_i(\mathbf{x}) - \bar{z}_i] \text{ or } \delta_i^+ = \frac{1}{2}\Big[|\bar{z}_i - f_i(\mathbf{x})| + f_i(\mathbf{x}) - \bar{z}_i\Big]$$

and underachievement variables

$$\delta_i^- = \max\,[0,\ \bar{z}_i - f_i(\mathbf{x})] \text{ or } \delta_i^- = \frac{1}{2}\Big[|\bar{z}_i - f_i(\mathbf{x})| + \bar{z}_i - f_i(\mathbf{x})\Big].$$

The resulting (Archimedian goal programming) problem is

$$\text{(2.11.2)} \qquad \begin{array}{ll} \text{minimize} & \sum_{i=1}^{k} (w_i^- \delta_i^- + w_i^+ \delta_i^+) \\ \text{subject to} & f_i(\mathbf{x}) + \delta_i^- - \delta_i^+ = \bar{z}_i, \quad i = 1, \dots, k, \\ & \delta_i^-, \delta_i^+ \geq 0, \quad i = 1, \dots, k, \\ & \mathbf{x} \in S, \end{array}$$

where we give separate weighting coefficients for underachievements and overachievements, and $\mathbf{x}$, δ_i^- and δ_i^+ $(i = 1, \dots, k)$ are the variables. If all the goals are in the form $f_i(\mathbf{x}) \leq \bar{z}_i$, we can leave the underachievement variables and write the problem in the form

$$\text{(2.11.3)} \qquad \begin{array}{ll} \text{minimize} & \sum_{i=1}^{k} w_i^+ \delta_i^+ \\ \text{subject to} & f_i(\mathbf{x}) - \delta_i^+ \leq \bar{z}_i, \quad i = 1, \dots, k, \\ & \delta_i^+ \geq 0, \quad i = 1, \dots, k, \\ & \mathbf{x} \in S, \end{array}$$

where $\mathbf{x}$ and δ_i^+ $(i = 1, \dots, k)$ are the variables.

Figure 12 portrays how the problem (2.11.3) is solved. The black point is the reference point of the aspiration levels. Every weighting vector produces different contours by which the feasible criterion region is to be intersected. Thus, different solutions can be obtained by altering the weights. The contours with two weighting vectors have been depicted in the figure. The fat line illustrates the Pareto optimal set.

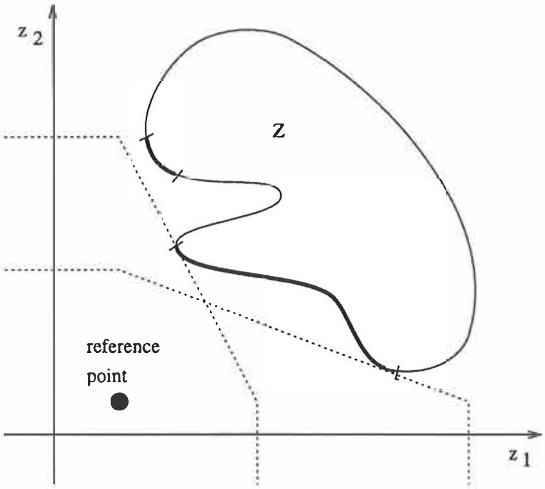


Figure 12. Contours with different weighting vectors.

Even though the constraints $\delta_i^- \cdot \delta_i^+ = 0$ for all i are not usually included in the problem formulations, some attention must be paid to guarantee that they are valid (see details in [Rosenthal, 1983]). An example of the required conditions is given in [Sawaragi, Nakayama, Tanino, 1985], p. 253. The Archimedian problem may be solved by standard single objective optimization methods. If the original problem is linear, then the corresponding Archimedian problem is also linear.

Notice that goal programming is closely related to the method of global criterion or compromise programming, presented in Section 2.7. It can be seen particularly well in the formulation (2.11.1). Instead of the ideal criterion vector, the reference point of the decision maker is used in goal programming. The distances can be measured by many metrics but the L_1-metric is widely used in connection with goal programming.

In the *preemptive* approach, the decision maker must specify a lexicographic order of the goals in addition to the aspiration levels. The goal of the highest priority level is supposed to be infinitely more important than the goal of the second priority level, etc. This means that no matter how large a multiplier is selected, the lower priority goal multiplied by it cannot be made equally important than the higher priority goal. After the lexicographic ordering, the problem with the deviational variables as objective functions and the constraints as in (2.11.2) is solved as explained in Section 2.10. In order to be able to use the preemptive approach, the decision maker's preference order for the objectives must be definite and rigid.

It is computationally advisable not to add new constraints in the preemptive approach, if the problem is linear. Instead, the variables with nonzero reduced cost values should be deleted (see, e.g., [Eiselt, Pederzoli, Sandblom, 1987]).

Also a combination of the Archimedian and the preemptive approaches can be used. 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 deviational variables is minimized. The same weaknesses presented in connection with the lexicographic ordering are valid for this and the preemptive approach.

Next, we prove a result concerning the Pareto optimality of the solutions of goal programming.

Theorem 2.11.4. *If either the aspiration levels form a Pareto optimal reference point or all the deviational variables δ_i^+ for functions to be minimized and δ_i^- for functions to be maximized have positive values at the optimum, then the solution of an Archimedian or preemptive goal programming problem is Pareto optimal.*

Proof. For the preemptive approach, the proof corresponds to that of Theorem 2.10.1. Here, we only present a proof for the Archimedian approach. For simplicity of notations, we assume that the problem is of the form (2.11.3). A more general case is straightforward.

Let $\mathbf{x}^* \in S$ be a solution of the problem (2.11.3), where the deviational variables (denoted here for clarity by δ_i^*) are positive. Let us assume that $\mathbf{x}^*$ is not Pareto optimal. In this case, there exists a vector $\mathbf{x}^\circ \in S$ such that $f_i(\mathbf{x}^\circ) \leq f_i(\mathbf{x}^*)$ for all $i = 1, \ldots, k$ and $f_j(\mathbf{x}^\circ) < f_j(\mathbf{x}^*)$ for at least one index j.

We denote $f_j(\mathbf{x}^*) - f_j(\mathbf{x}^\circ) = \beta > 0$. Then we set $\delta_i^\circ = \delta_i^* > 0$ for $i \neq j$ and $\delta_j^\circ = \max\,[0, \delta_j^* - \beta] \geq 0$, where δ_i° is the deviational variable corresponding to $\mathbf{x}^\circ$ for $i = 1, \ldots, k$.

Now we have $f_i(\mathbf{x}^\circ) - \delta_i^\circ \leq f_i(\mathbf{x}^*) - \delta_i^* \leq \bar{z}_i$ for all $i \neq j$. If $\delta_j^* - \beta > 0$, then $f_j(\mathbf{x}^\circ) - \delta_j^\circ = f_j(\mathbf{x}^\circ) - \delta_j^* + f_j(\mathbf{x}^*) - f_j(\mathbf{x}^\circ) \leq \bar{z}_j$, and if $\delta_j^* - \beta \leq 0$, then $f_j(\mathbf{x}^\circ) - \delta_j^\circ = f_j(\mathbf{x}^\circ) + f_j(\mathbf{x}^*) - f_j(\mathbf{x}^*) = f_j(\mathbf{x}^*) - \beta \leq f_j(\mathbf{x}^*) - \delta_j^* \leq \bar{z}_j$.

This means that $\mathbf{x}^\circ$ satisfies the constraints of the problem (2.11.3). We have $\delta_j^\circ < \delta_j^*$ (this is also valid if $\delta_j^\circ = 0$ since $\delta_i^* > 0$ for all i), and $\delta_i^\circ \leq \delta_i^*$ for all $i \neq j$. As the weighting coefficients are positive, we have $\sum w_i^+ \delta_i^\circ < \sum w_i^+ \delta_i^*$, which contradicts the fact that $\mathbf{x}^*$ is a solution of the problem (2.11.3).

For aspiration levels forming a Pareto optimal point the proof is self-evident. ∎

If the optimal objective function value of the goal programming problem equals zero, some caution is in order, since the solution obtained may not be Pareto optimal. The reason is that if the aspiration levels are all feasible, then the value zero for all the deviational variables gives the minimum value (zero) for the goal programming objective function. Thus the solution is equal to the reference point, and there exist normally many feasible points which are not Pareto optimal. If the solutions are intended to be Pareto optimal despite the selection of the aspiration levels, then we must maximize the distance if the aspiration levels are feasible and minimize the distance if the aspiration levels are infeasible. This is the case with so-called achievement scalarizing functions as explained in Section 2.19. Tests for Pareto optimality in goal programming are provided in [Romero, 1991].

In [Dyer, Sarin, 1981], it is pointed out that although it is not readily apparent, goal programming implicitly assumes that there is a measurable, additive and rigid piecewise linear underlying value function. Rosenthal stresses, in [Rosenthal, 1983], that the Archimedian problem (2.11.2) is equivalent to the value function maximization problem where

$$\frac{dU(\mathbf{f}(\mathbf{x}))}{d f_i} = \begin{cases} w_i^- \text{ if } f_i(\mathbf{x}) < \bar{z}_i \\ -w_i^+ \text{ if } f_i(\mathbf{x}) > \bar{z}_i, \end{cases}$$

which means that the marginal utility is constant on either side of the aspiration level. This is contrary to the economic idea that a decision maker considers the next unit of decrease of f_i more important when f_i is plentiful than when f_i is scarce. This idea is even more evident when we look at the marginal rates of substitution in goal programming problems. In connection with Definition 1.7.4, it was mentioned that marginal rates of substitution may be defined as $m_{ij}(\mathbf{x}) = \frac{d\,U(\mathbf{f}(\mathbf{x}))}{d\,f_j} / \frac{d\,U(\mathbf{f}(\mathbf{x}))}{d\,f_i}$. Thus, goal programming does not take into consideration the possibility that it is easier for the decision maker to let something increase a little if (s)he has got it little than if (s)he has got it much. The reason for this is that goal programming implicitly assumes that the marginal rates of substitution are piecewise constant. This critique also applies to the preemptive approach (see details in [Rosenthal, 1983, 1985]). More critical observations about goal programming are presented in [Rosenthal, 1983] and [Romero, 1991].

Applications and Extensions

An extensive presentation on goal programming and its extensions is given in [Ignizio, 1976] and a survey of different variations of goal programming is provided in [Charnes, Cooper, 1977]. In addition, a wide survey of the literature around goal programming up till the year 1983 is presented in [Soyibo, 1985]. Several modifications and improvements as well as applications are reviewed. A survey of goal programming is also given in [Kornbluth, 1973] and the Archimedian and the preemptive approaches are applied to problems with fractional objective functions.

Preemptive goal programming is applied in [Benito-Alonso, Devaux, 1981] to a problem concerning the location and size of day nurseries, in [Sinha, Sastry, Misra, 1988] to storage problems of agriculture, and in [Mitra, Patankar, 1990] to aid manufacturers in selecting the price and warranty time of their products. Preemptive goal programming is also applied in [Kumar, Singh, Tewari, 1991] to nonlinear multistage decision problems of manufacturing systems, in [Ng, 1992] to aircraft loading and in [Brauer, Naadimuthu, 1992] to solve a mixed integer MOLP problem involving inventory and distribution planning. Archimedian goal programming with equal weighting factors is employed in the planning of public works in [Yoshikawa, Haruna, Kobayashi, 1982] with two illustrative examples.

A combination of the Archimedian and the preemptive goal programming approaches is applied in [Levary, 1986] to problems of optimal control, in [Giokas, Vassiloglou, 1991] to the (linear) management of the bank assets and liabilities of a Greek bank, and in [Ghosh, Pal, Basu, 1992] to the resource planning of university management. In [Sankaran, 1990], the combined approach is used to solve an integer MOLP problem in cell formation, and, in [Schniederjans, Hoffman, 1992], combined zero-one goal programming is applied to a problem concerning international business expansion analysis. The ideas of combined goal programming are adapted in [Miyaji, Ohno, Mine, 1988], when solving a transportation problem-type problem of partitioning students into groups.

The applications mentioned here are just a minor part of what exists. The popularity of goal programming is well affirmed by the fact that in a bibliography collected by White on multiobjective optimization applications (covering the years from 1955 to 1986) more than a half involved goal programming (see [White, 1990]).

Four different goal interpretations in multiobjective optimization are presented in [Dinkelbach, 1980]. Ignizio applies goal programming to multiobjective generalized networks for integer problems in [Ignizio, 1983(b)]. In [Inuiguchi, Kume, 1991], goal programming is extended to linear problems where the coefficients and the aspiration levels are given by intervals. The aspiration level intervals do not here represent regions within which the decision maker is satisfied, but regions where the aspiration levels may vary. A generalization of goal programming through the theory of variational inequalities is presented in [Thore, Nagurney, Pan, 1992]. An extension of goal programming to MOLP problems is given in [Martel, Aouni, 1990]. Instead of the deviational variables, some functions describing the wishes of the decision maker about attaining the goals are used in the Archimedian approach. An illustrating example is also provided.

A multiphase simplex method, called MULTIPLEX, is proposed in [Ignizio, 1985(b)] for solving linear (Archimedian and preemptive) goal programming problems. A graphic solution technique for linear goal programming problems with two decision variables is presented in [Eiselt, Pederzoli, Sandblom, 1987]. The same authors present also simplex-based algorithms for linear goal programming problems. Among them is a sequential method consisting of successive solutions of linear problems with changing polytopes. A modified simplex method for linear goal programming is presented in [Lee, 1981]. Lee also mentions some ideas of further research in the areas of goal programming with uncertainty, integer-valued problems and interactive goal programming. In [Hartley, 1985], it is shown how the simplex algorithm can efficiently be used in solving linear preemptive goal programming problems.

Concluding Remarks

Goal programming is a widely used and popular solution method for practical multiobjective optimization problems. One of the reasons is that goal-setting is an understandable and easy way of making decisions. The specification of the weighting coefficients or the lexicographic ordering may be more difficult. However, the weights do not have so direct an effect on the solution obtained as in the a priori weighting method.

One must be careful with the selection of the aspiration levels so that the Pareto optimality of the solutions can be guaranteed. The correct selection may be difficult for a decision maker who does not know what the feasible region looks like. Presenting the ranges of the Pareto optimal set, or at least the ideal criterion vector, to the decision maker may help in the selection.

Goal programming is not an appropriate method to be used if trade-offs are desired to obtain. Another restricting property is the underlying assumption of a piecewise linear value function and thus piecewise constant marginal rates of substitution.

2.12. Methods Where Progressive Articulation of Preference Information Is Used (Interactive Methods)

The class of interactive methods is the most developed one of the four method classes presented here. The interest devoted to this class can be explained by the fact that assuming the decision maker has enough time and capabilities for cooperation, interactive methods can be presumed to produce most satisfactory results. Many weak points of the methods in the other three classes are overcome. Now only a part of the Pareto optimal points has to be generated and evaluated and the decision maker can specify and correct the preferences and selections as the solution process goes on and (s)he gets to know the problem and its potentialities better. This also means that the decision maker does not have to know any global preference structure. In addition, the decision maker can be assumed to have more confidence in the final solution since (s)he is involved throughout the solution process.

In interactive methods the decision maker works together with an analyst or an interactive computer program. One can say that the analyst tries to find out the preference structure of the decision maker in an interactive way. A solution pattern is formed and repeated several times. After every iteration some information is given to the decision maker and (s)he is asked to answer some questions or provide some other type of information. The working order in these methods is 1. analyst, 2. decision maker, 3. analyst, 4. decision maker, ... After a reasonable (finite) number of iterations every interactive method should yield a solution so that the decision maker can be satisfied with it and (s)he can be convinced that there does not exist any considerably better solution. The basic steps of interactive algorithms can be expressed as

(a) find an initial feasible solution,
(b) interact with the decision maker, and
(c) obtain a new solution (or a set of new solutions). If it (or some of them) or some of the previous solutions is acceptable for the decision maker, stop. Otherwise, go to step (b).

Interactive methods differ from each other by the form information is given to the decision maker, by the form information is provided by the decision maker and how the problem is transformed into a single objective optimization problem. One problem to be solved when designing an interactive method is what kind of data one should use to interact with the decision maker. It should be meaningful and easy to comprehend for the decision maker. The decision maker should understand the meaning of the parameters to which (s)he is asked to supply values. On the other hand, data provided to the decision maker should be easily obtainable by the analyst and contain information of the system. Too much information should not be used. To ensure that the greatest possible benefit can be obtained from the interactive method the decision maker must find the method worthwhile and acceptable and (s)he must be able to use the method properly. Usually, this means that the method must be sufficiently easy to use and understandable.

Interactive methods have been classified in many ways, mainly according to their solution approaches. Here we do not follow any of those classifications.

Two different conceptions of interactive approaches are handled in [Vanderpooten, 1989(a), (b)]. The approaches are searching and learning. In searching-oriented procedures a converging sequence of solution proposals is presented to the decision maker. It is supposed that the decision maker provides consistent preference information. In learning-oriented procedures a free exploration of alternatives is possible allowing trial and error. The latter does not guide the decision maker and the convergence is not guaranteed. A combination of these two approaches containing positive features of them both can be recommended. This approach would support the learning of preferences, while it would also include guiding properties.

From experimental tests with interactive methods it is concluded, in [Korhonen, Moskowitz, Wallenius, 1990], that interactive procedures should converge well right in the few initial iterations. The decision makers are not willing to wait for progress for a long time. Another conclusion is that interactive procedures should have built-in mechanisms to deal with inconsistencies.

Consistency of the responses of the decision maker is one of the most important factors in order to guarantee the success of most interactive solution methods. Because of the subjectivity of the decision makers, different starting points, different types of questions or interaction styles may lead to different final solutions. Some methods are more sensitive with respect to consistency than others. The handling of inconsistency for several interactive methods has been compared in [Shin, Ravindran, 1991]. In general, inconsistency can be reduced by consistency tests during the solution process or by minimizing the decision maker's cognitive burden. The latter way is one of the motivations in developing new methods for multiobjective optimization.

A critical factor in interactive methods is the stopping criterion. There are mainly three stopping criteria. Either the decision maker gets tired of the solution process, some algorithmic stopping (convergence) rule is fulfilled, or the decision maker finds a desired solution and wants to stop. It is difficult to define precisely when a solution is desirable enough to become a final solution. The convergence of the method has been considered to be an important factor when selecting a method. However, it is stated, in [Vanderpooten, Vincke, 1989], that "the procedure should not be stopped because of any convergence test but only if the decision maker is satisfied with a solution or when he has the feeling that he has enough information about his problem."

Before we present the methods, some critical comments are in order. Repeatedly in this presentation, it has been and will be assumed that the decision maker makes consistent decisions or that (s)he has an underlying (implicitly known) value function upon which decisions are made. The purpose is not to go deeply into the theories of decision making. However, it is worth mentioning that those assumptions can be called into question because they are difficult to verify. Decision making is, for example, in [Zeleny, 1989] described as "searching for harmony in a chaos". One can criticize the way how decision makers are forced into a priori formulas, patterns or contexts (like wandering around the Pareto optimal set). Instead, the decision maker should be guided through her or his own creative search process since "decision making is a process of continuous redefinition of the problem."

On the other hand, after the existence of the underlying, implicit value function is supposed, several assumptions are set on it. How can one guarantee and verify, for example, the pseudoconcavity of a function which is not explicitly known? Of course, something can be concluded if we find out enough about the decision maker's preference structure. Steps in that direction are, however, very laborious and yet the results are likely to be controversial.

One more interesting concept is the convergence of an interactive method. One can understand several different features as convergence. It may be said that the method converges into Pareto optimal points if the final solution can be proved to be Pareto optimal. One can also say that the method converges into a satisficing solution, if the final solution is satisficing. On the other hand, convergence may mean that the final solution is optimal to the underlying value function. This kind of convergence result necessitates certain assumptions on the underlying value function. In this case, the observations of the previous paragraph are valid. If the method is not based on the assumption on any underlying value function, this conception of convergence cannot always be applied. To sum up, is not unequivocal, what is meant by convergence and how it should be proved. For this reason, it is difficult to provide convergence results for the different methods considered. Thus, the convergence properties have been left into a secondary position in what follows.

Even though interactive methods can be regarded as most promising solution methods for multiobjective optimization problems, there are still cases where these methods are not practical to be applied regardless of the availability of the decision maker. Such problems are, for instance, many engineering problems that require extensive and expensive calculations (like large-scale finite element approximations). One must, however, remember that computational facilities have developed greatly during the last few years. Thus, the number of problems which cannot be solved by interactive methods has decreased. See [Osyczka, Zajac, 1990] for a suggestion of handling computationally expensive functions.

In the following, we present several interactive methods. Some of them are quite old and much tested and developed, whereas some others are new and deserve further refinements. The methods to be described are the ISWT and the GDF method, SPOT, the ZW and the IWT method, STEM, the reference point and the satisficing trade-off method, the visual interactive approach, the subgradient GDF and the NIMBUS method. Some of them are only briefly outlined and the rest are described in more detail. In developing the last two of the methods, one has tried to overcome some of the drawbacks observed in the other methods. Even though specialities of MOLP problems have been avoided in this presentation, we yet describe some linear interactive methods. The reason for this is that the ideologies of these methods are interesting and important in the general history of multiobjective optimization method development.

In connection with the methods, some applications reported in the literature are mentioned. We also indicate whether the methods belong to the class of ad hoc or non ad hoc methods. (The classes were introduced at the beginning of this chapter.) The description of each method ends by concluding remarks, where some important aspects and opinions of the author have been collected.

Throughout the presentation the iteration counter is denoted by h and the decision variable vector at the current iteration by $\mathbf{x}^h$. In addition, the number of alternative criterion vectors presented to the decision maker is denoted by P.

2.13. Interactive Surrogate Worth Trade-Off Method

An interactive surrogate worth trade-off (ISWT) method, put forward in [Chankong, Haimes, 1978, 1983(b)], is an extension of a surrogate worth trade-off (SWT) method presented in [Haimes, Hall, 1974] and [Haimes, Hall, Freedman, 1975]. We do not go into details of the SWT method here, but present directly the interactive version.

Introduction

It is assumed in the ISWT method that the decision maker's underlying value function is implicitly known. The aim of the method is to find a local maximum for the value function by local approximations.

One of the basic ideas inspiring the ISWT method has been the thought that it is easier for the decision maker to compare alternatives than to give numerical information. Further, it has been assumed that it is easiest to compare two alternatives at a time. The motivation is that if the information presented and requested is simple and easy for the decision maker to understand, it can be expected that the solution process becomes more reliable and fluent. Another important feature in the ISWT method is that the ε-constraint method is used as a means of generating new alternatives. This is why it is guaranteed that the alternatives are Pareto optimal and any Pareto optimal solution can be found.

The main features of the ISWT method are the following. First, an ε-constraint problem is solved and the opinions of the decision maker are asked about the trade-off rates at the solution point. If the decision maker does not want to change the current solution, the solving process can be stopped. Otherwise, the direction of the steepest ascent of the value function is approximated according to the opinions of the decision maker about the trade-off rates. New upper bounds are obtained for the ε-constraint problem. The step-size, how far to proceed into the new direction, is obtained by solving the ε-constraint problem several times and letting the decision maker select the best Pareto optimal solution. The procedure continues as described until the decision maker does not want to change any criterion value or some other stopping criterion is satisfied. Hereby, a sequence of Pareto optimal solutions is generated such that the next solution is preferable to the previous one.

It is assumed that

1. The underlying value function U exists and is implicitly known to the decision maker. In addition, U is continuously differentiable and a monotone non-increasing function on Z.
2. All the objective and the constraint functions are twice continuously differentiable.
3. The feasible region S (with the feasible region of the ε-constraint problem) is compact.

Important additional assumptions are that the Pareto optimal set has a smooth surface (implying that at every point the trade-off rates are unique) and every solution of the ε-constraint problem satisfies the requirements and hypotheses of Theorem 2.3.17. In this way, we make sure that the trade-off rates are at our disposal. Since it is assumed that the regularity and second-order sufficiency conditions are satisfied at every solution $\mathbf{x}^h$ of the ε-constraint problem (implying uniqueness in some neighbourhood), we know that $\mathbf{x}^h$ is Pareto optimal (see [Chankong, Haimes, 1983(b)]). Remember that only local Pareto optimality can be guaranteed, as has been stated earlier.

An important factor in the ISWT method is the reference function (see Section 1.7). It is the function to be minimized in the ε-constraint problem (see Section 2.3). Careful consideration must be given to the selection of the reference function because trade-off rates are calculated with respect to it. It is suggested in [Tarvainen, 1984] that the decision maker specifies such an objective function to be the reference function with respect to whose values (s)he is flexible. This means that there is no sudden limit under which the criterion values are satisfactory and above which the criterion values are unsatisfactory. In addition, the trade-off rates must be sensible with respect to the reference function.

ISWT Algorithm

The main features of the ISWT method can be presented cursorily as follows.

(1) Select the reference function to be minimized and give upper bounds to the other objective functions.

(2) Solve the current ε-constraint problem to get a Pareto optimal solution. Trade-off information is obtained from the optimal Kuhn-Tucker multipliers.

(3) Ask the opinions of the decision maker with respect to the trade-off rates at the current solution point.

(4) If some stopping criterion is satisfied, stop. Otherwise update the upper bounds of the objective functions with the help of the answers obtained earlier and solve several ε-constraint problems to determine an appropriate step-size. Let the decision maker choose the most preferred alternative. Go to step (3).

First, we examine how the trade-off information is obtained from the optimal Kuhn-Tucker multipliers. As noted in Theorem 2.3.17 of Section 2.3, the optimal Kuhn-Tucker multipliers represent trade-off rates under the specified assumptions.

Let $\mathbf{x}^h \in S$ be a solution of the ε-constraint problem at the iteration h, where f_ℓ is the function to be minimized and the upper bounds are ε_i^h for $i \neq \ell$. We suppose that $\mathbf{x}^h$ satisfies the assumptions specified in Theorem 2.3.17. If the Kuhn-Tucker multipliers $\lambda_{\ell i}^h$ associated with the constraints $f_i(\mathbf{x}) \leq \varepsilon_i^h$ are strictly positive for all $i = 1, \ldots, k$, $i \neq \ell$, then $\lambda_{\ell i}^h$ represents the partial trade-off rate at $\mathbf{x}^h$ between f_ℓ and f_i. Now we know that to move from $\mathbf{x}^h$ to some other (locally) Pareto optimal point in the neighbourhood of $\mathbf{x}^h$, the value of the function f_ℓ decreases by $\lambda_{\ell i}^h$ units for every unit of increment in the value of the function f_i (or vice versa), while the values of all the other objective functions remain unaltered. The opinion of the decision maker with regard to this kind of trade-off for all $i \neq \ell$ is found out by posing the following question.

> Let a criterion vector $(f_1(\mathbf{x}^h), \ldots, f_k(\mathbf{x}^h))^T = \mathbf{z}^h$ be given. If the value of f_ℓ is decreased by $\lambda_{\ell i}^h$ units, then the value of f_i is increased by one unit (or vice versa) and the other criterion values remain unaltered. How desirable do you find this trade-off?

If the situation is not so convenient as above, that is, some of the optimal Kuhn-Tucker multipliers $\lambda_{\ell i}^h$ equal zero, then another type of question is needed. Let us suppose that $\lambda_{\ell i}^h > 0$ for $i \in N^>$ and $\lambda_{\ell j}^h = 0$ for $j \in N^=$, where $N^> \cup N^= = \{i \mid i = 1, \ldots, k, \ i \neq \ell\}$. As noted in Theorem 2.3.17, increasing the value of f_i, where $i \in N^>$ decreases the value of f_ℓ and in addition, the values of all f_j also change, where $j \in N^=$. Now the question to the decision maker for all $i \in N^>$ is of the form

> Let a criterion vector $(f_1(\mathbf{x}^h), \ldots, f_k(\mathbf{x}^h))^T = \mathbf{z}^h$ be given. If the value of f_ℓ is decreased by $\lambda_{\ell i}^h$ units, then the value of f_i is increased by one unit and the values of f_j for $j \in N^=$ change by $\nabla f_j(\mathbf{x}^h)^T \frac{d\,\mathbf{x}(\boldsymbol{\varepsilon}^h)}{d\,\varepsilon_i}$ units (or vice versa). How desirable do you find these trade-offs?

A problem with the question above is that the values of $\frac{d\,\mathbf{x}(\boldsymbol{\varepsilon}^h)}{d\,\varepsilon_i}$ for $i \in N^>$ are not known. One of the ways suggested in [Chankong, Haimes, 1983(b)] for coping with this is that the values can be approximated by solving the ε-constraint problem with a slightly modified upper bound vector as $\boldsymbol{\varepsilon}^h(i) = (\varepsilon_1^h, \ldots, \varepsilon_{\ell-1}^h, \varepsilon_{\ell+1}^h, \ldots, \varepsilon_i^h + \epsilon, \ldots, \varepsilon_k^h)$, where $\epsilon \neq 0$ is a scalar with a small absolute value. Let the solution of this ε-constraint problem be $\mathbf{x}(\boldsymbol{\varepsilon}^h(i))$. Now we get the approximation by

$$\frac{d\mathbf{x}(\boldsymbol{\varepsilon}^h)}{d\varepsilon_i} \approx \frac{\mathbf{x}(\boldsymbol{\varepsilon}^h(i)) - \mathbf{x}^h}{\epsilon}.$$

Notice that the decision maker's opinions are asked of certain amounts of change in the values of the objective functions, and not of changes in general. The following problem to be handled is the form of answers expected from the decision maker. It is suggested in [Chankong, Haimes, 1978, 1983(b)] that the decision maker must specify an integer between 10 and -10 to indicate her or his degree of preference. If the decision maker is completely satisfied with the trade-off suggested, the answer is 10. Positive numbers less that 10 indicate the degree of satisfaction (less than complete). Correspondingly, negative answers reflect the decision maker's satisfaction with the trade-off which is converse to that in the question. The answer 0 means that the decision maker is indifferent to the given trade-off.

In [Tarvainen, 1984], it is suggested that much less choices are given to the decision maker. The possible answers are $2, 1, 0, -1, -2$ and their meaning corresponds to the previously presented. The justification is that it is easier for the decision maker to give the answer when there are less alternatives. These five alternatives are enough to represent the direction and rough degree of the decision maker's preferences and satisfaction.

Regardless of the scale selected, the response of the decision maker is called a *surrogate worth* of the trade-off between f_ℓ and f_i at $\mathbf{x}^h$ and denoted by $W_{\ell i}^h$. At each point, $k-1$ (or less, if $N^= \neq \emptyset$) questions of the previously described form are presented to the decision maker and the values for $W_{\ell i}^h$ $(i \neq \ell)$ are obtained.

According to Theorem 2.3.17, there exists a Pareto optimal solution in the neighbourhood of $\mathbf{x}^h$ when the values of the objective functions are changed according to the information given in the trade-off rates. The problem is how much the values of the objective functions can be changed in order to remain on the Pareto optimal surface and obtain the best possible solution. We must find a way to update the upper bounds of the objective functions in an appropriate way.

The way how to proceed from this point depends on the scale chosen for the surrogate worth values. The idea is to obtain an estimate for the gradient of the value function with the help of the surrogate worth values. Then a steepest ascent-type formulation is used. The upper bounds of the ε-constraint problem are revised and a new solution is obtained. It is supposed to satisfy the preferences of the decision maker indicated by the surrogate worth values as well as possible.

In the original version of Chankong and Haimes, it is suggested that the upper bounds are updated from iteration h to $h+1$ by

$$\varepsilon_i^{h+1} = \varepsilon_i^h + t(W_{\ell i}^h |f_i(\mathbf{x}^h)|)$$

for $i \in N^>$ and

$$\varepsilon_j^{h+1} = \varepsilon_j^h + t\left(\nabla f_j(\mathbf{x}^h)^T \left(\frac{d\,\mathbf{x}(\boldsymbol{\varepsilon}^h)}{d\,\varepsilon_i}\right) W_{\ell i}^h |f_i(\mathbf{x}^h)|\right)$$

for $j \in N^=$, where $i \in N^>$ and t is a step-size to be determined. For details, see [Chankong, Haimes, 1978, 1983(b)] and references therein.

For simplicity, it is assumed in [Tarvainen, 1984] that the optimal Kuhn-Tucker multipliers are all strictly positive. The decision maker is asked to specify small and meaningful amounts Δf_i for all $i \neq \ell$. The scalar Δf_i represents the amount of change in the value of f_i that is relevant to the decision maker. Now the upper bounds are updated by

$$\varepsilon_i^{h+1} = \varepsilon_i^h + t(W_{\ell i}^h \Delta f_i)$$

for $i \in N^>$, where t denotes the step-size.

Several discrete values may be given to the step-size t in each updating formula. Then the ε-constraint problem is solved for every value. The resulting criterion vectors are presented to the decision maker, who is asked to choose the most preferred one. A graphical representation of the alternatives may be helpful. This topic is handled in Chapter 4. After choosing the step-size and thus the new solution, trade-off information at that solution is obtained from the optimal Kuhn-Tucker multipliers (as earlier). The procedure continues by asking the decision maker for the surrogate worth values.

In practice, when the decision maker is asked to express her or his preferences about the trade-offs, (s)he is implicitly asked to compare the trade-off rates with her or his marginal rates of substitution. (Naturally, the decision maker does not have to be able to specify the marginal rates of substitution explicitly.) If $m_{\ell i} < \lambda_{\ell i}$, then the surrogate worth value is positive (and the contrary respectively). If $m_{\ell i} = \lambda_{\ell i}$ for all $i \neq \ell$, meaning $W_{\ell i} = 0$ then the stopping criterion introduced in Section 1.7 is valid. Thus, the condition $W^h_{\ell i} = 0$ for all $i \neq \ell$ is a common stopping criterion of the algorithm. Another possible stopping situation is that the decision maker wants to proceed only into an infeasible direction. The latter condition is more difficult to check.

The ISWT method can be classified to be non ad hoc in nature. If the value function is known, then the trade-off rates are easy to compare with the marginal rates of substitution. The convergence rate of the ISWT method greatly depends on the accuracy and consistency of the answers of the decision maker. The comments about the importance of selecting the reference function well were given partly because of the convergence properties. If there is a sharp limit in the values of the reference function where the satisfaction changes from "very satisfactory" to "very unsatisfactory", the solution procedure may stop too early. Further references are cited in [Chankong, Haimes, 1978] for convergence results.

A method related to the ISWT method is presented in [Chen, Wang, 1984]. The method is an interactive version of the SWT method, where new solution alternatives are generated by Lin's proper equality method (see Section 2.3), and the decision maker has to specify only the sign of the surrogate worth values.

There are many other modifications of the SWT method in the literature. Among others, it has been generalized for multiple decision makers in [Hall, Haimes, 1976] and [Haimes, 1980].

Concluding Remarks

The role of the decision maker is quite easy to understand in the ISWT method. The complicatedness of giving the answers depends on which variation of the method is employed. The selection of 21 different answers in the original version is quite a lot to select from. It may be difficult for the decision maker to provide consistent surrogate worth values during the whole decision process. In addition, if there is a large number of objective functions, the decision maker has to specify a lot of surrogate worth values at each iteration. At least for some decision makers it may be easier to maintain consistency when there are less alternative values for the surrogate worth available (as suggested by Tarvainen).

Trade-off rates play an important role in the ISWT method, and this is why the decision maker has to understand the concept of trade-off properly. Attention must also be paid to the understandable and careful formulation of the questions about the trade-off rates. Careless formulation may, for example, cause the sign of the surrogate worth value to be changed.

It is good that all the alternatives during the solution process are Pareto optimal. Thus the decision maker is not bothered with unsatisfactory solutions.

A negative feature is that there are a lot of different assumptions to be satisfied so that the algorithm is guaranteed to work. It may be difficult (and at least laborious) in many practical problems to make sure that the assumptions are fulfilled.

2.14. Geoffrion-Dyer-Feinberg Method

The Geoffrion-Dyer-Feinberg (GDF) method, proposed in [Geoffrion, Dyer, Feinberg, 1972], is an interactive method based in principle on the same idea as the ISWT method above; maximization of the underlying (implicitly known) value function. The realization is quite different, though.

Introduction

The basic thought of the GDF and the ISWT method is the same. At each iteration, a local approximation of the value function is generated and maximized. In the GDF method, the idea is somewhat clearer to be seen. Marginal rates of substitution given by the decision maker are used to approximate the direction of steepest ascent of the value function. Then the value function is maximized by a gradient-based method. A gradient method of Frank and Wolfe (FW) (see [Frank, Wolfe, 1956]) has been selected for optimization because of its simplicity and robust convergence (rapid initial convergence) properties. The GDF method is also sometimes called an interactive Frank-Wolfe method, because it has been constructed on the basis of the FW method.

The problem to be solved here is

$$\begin{aligned} &\text{maximize} && u(\mathbf{x}) = U(\mathbf{f}(\mathbf{x})) \\ &\text{subject to} && \mathbf{x} \in S. \end{aligned} \tag{2.14.1}$$

It is assumed that

1. The underlying value function $U\colon \mathbf{R}^k \to \mathbf{R}$ exists and is implicitly known to the decision maker. In addition, $u\colon \mathbf{R}^n \to \mathbf{R}$ is a continuously differentiable and concave function on S (sufficient conditions for the concavity are, for example, that U is a concave decreasing function and the objective functions are convex; or U is concave and the objective functions are linear), and U is strictly decreasing with respect to the reference function (denoted here by f_ℓ).
2. All the objective functions are continuously differentiable.
3. The feasible region S is compact and convex.

Let us begin by presenting the main principles of the FW method. Let a point $\mathbf{x}^h \in S$ be given. The idea of the FW method is that when maximizing some objective function $u\colon \mathbf{R}^n \to \mathbf{R}$ subject to constraints $\mathbf{x} \in S$, instead of u, a linear approximation of it at some point $\mathbf{x}^h \in S$ is optimized. If the solution is $\mathbf{y}^h$, then the direction $\mathbf{d}^h = \mathbf{y}^h - \mathbf{x}^h$ is a good direction where to seek an increased value for the objective function u.

At any feasible point $\mathbf{x}^h$ a linear approximation to $\mathbf{y} \to u(\mathbf{y})$ is

$$u(\mathbf{x}^h) + \nabla_x u(\mathbf{x}^h)^T(\mathbf{y} - \mathbf{x}^h).$$

The maximization of the linear approximation after excluding constant terms is equivalent to the problem

$$\begin{aligned} &\text{maximize} && \nabla_x u(\mathbf{x}^h)^T \mathbf{y} \\ &\text{subject to} && \mathbf{y} \in S, \end{aligned} \tag{2.14.2}$$

where $\mathbf{x}^h$ is fixed and $\mathbf{y}$ is the variable. Let $\mathbf{y}^h \in S$ be the solution.

A well-known condition for $\mathbf{x}^h$ to be an optimal solution of the problem (2.14.1) is that $\nabla_x u(\mathbf{x}^h)^T\mathbf{d} \leq 0$ for all $\mathbf{d} \in S$. Therefore, if after solving the problem (2.14.2) is $\mathbf{y}^h = \mathbf{x}^h$, then we know that $0 = \nabla_x u(\mathbf{x}^h)^T(\mathbf{y}^h - \mathbf{x}^h) \geq \nabla_x u(\mathbf{x}^h)^T(\mathbf{y} - \mathbf{x}^h)$ for all $\mathbf{y} \in S$, and, thus, the optimality condition is fulfilled at $\mathbf{x}^h$.

If $\mathbf{y}^h \neq \mathbf{x}^h$, then we set $\mathbf{d}^h = \mathbf{y}^h - \mathbf{x}^h$. The points $\mathbf{y}^h$ and $\mathbf{x}^h$ are feasible, and, because of the convexity assumption of S, any new point $\mathbf{x}^{h+1} = \mathbf{x}^h + t\mathbf{d}^h$ where $0 \leq t \leq 1$ is feasible. Finally, we must determine an appropriate step-size into the direction $\mathbf{d}^h$ by maximizing $u(\mathbf{x}^h + t\mathbf{d}^h)$ subject to $0 \leq t \leq 1$.

GDF Algorithm

In the following, we shall show that even though we do not know the value function explicitly, we can obtain a local linear approximation for it (i.e., its gradient) with the help of marginal rates of substitution. This is enough so that the FW method can be applied. Before going into details we present the basic phases of the GDF algorithm.

(1) Ask the decision maker to specify a reference function f_ℓ. Choose a feasible starting point $\mathbf{x}^1$. Set $h = 1$.

(2) Ask the decision maker to specify marginal rates of substitution between f_ℓ and the other objectives at the current solution point $\mathbf{x}^h$.

(3) Solve the problem (2.14.3), where the approximation of the value function is maximized. Denote the solution by $\mathbf{y}^h \in S$. Set the direction $\mathbf{d}^h = \mathbf{y}^h - \mathbf{x}^h$. If $\mathbf{y}^h = \mathbf{x}^h$, go to step (6).

(4) Determine with the help of the decision maker the appropriate step-size t^h to be taken into the direction $\mathbf{d}^h$. Denote the corresponding solution by $\mathbf{x}^{h+1} = \mathbf{x}^h + t^h\mathbf{d}^h$.

(5) Set $h = h + 1$. If the decision maker wants to continue, go to step (2).

(6) Stop. The final solution is $\mathbf{x}^h$.

In the algorithm above we need a local linear approximation of the value function at the point $\mathbf{x}^h$. As explained earlier, we only need to know the gradient of the value function at $\mathbf{x}^h$. According to the chain rule, we know that the gradient of the objective function of the problem (2.14.1) at the point $\mathbf{x}^h \in S$ can be written in the form

$$\nabla_x U(\mathbf{f}(\mathbf{x}^h)) = \sum_{i=1}^{k} \left(\frac{d\,U(\mathbf{f}(\mathbf{x}^h))}{d\,f_i} \right) \nabla_x f_i(\mathbf{x}^h).$$

In assumption 1 we supposed that $\frac{d\,U(\mathbf{f}(\mathbf{x}^h))}{d\,f_\ell} < 0$, where f_ℓ is the reference function (we shall return to it later). Positive scaling does not affect the direction of the gradient, so we can divide the gradient of the value function by a positive scalar $-\frac{d\,U(\mathbf{f}(\mathbf{x}^h))}{d\,f_\ell}$. Now we have the direction of the gradient of the value function at the point $\mathbf{x}^h$ in the form

$$\sum_{i=1}^{k} -m_i \nabla_x f_i(\mathbf{x}^h),$$

where $m_i = \frac{d\,U(\mathbf{f}(\mathbf{x}^h))}{d\,f_i} / \frac{d\,U(\mathbf{f}(\mathbf{x}^h))}{d\,f_\ell}$ for all $i \neq \ell$. The numbers m_i $(= m_{\ell i})$ represent the marginal rates of substitution at $\mathbf{x}^h$ between f_ℓ and f_i (see (1.7.5)). The role of the reference function is significant, because marginal rates of substitution are generated with respect to it. The decision maker must be asked to specify the reference function so that the marginal rates of substitution are sensible. Notice that if the underlying value function is linear, then only one iteration is needed to achieve the final solution (and the marginal rates of substitution are constant).

It may be difficult for the decision maker to specify the marginal rates of substitution. If (s)he cannot specify them straight away, some auxiliary procedures may be used to assist. One such procedure is presented in [Dyer, 1973(a)]. The idea there is to determine (at the point $\mathbf{f}(\mathbf{x}^h)$) small amounts of f_ℓ and f_i, denoted by Δ_{f_ℓ} and Δ_{f_i}, respectively, such that an increase in the value of f_i by Δ_{f_i} is compensated to the decision maker by a decrease by Δ_{f_ℓ} in the value of f_ℓ, while the values of all the other objective functions remain unaltered. In other words, the vectors $(f_1(\mathbf{x}^h), \ldots, f_k(\mathbf{x}^h))^T$ and $(f_1(\mathbf{x}^h), \ldots, f_\ell(\mathbf{x}^h) - \Delta_{f_\ell}, \ldots, f_i(\mathbf{x}^h) + \Delta_{f_i}, \ldots, f_k(\mathbf{x}^h))^T$ are indifferent to the decision maker. Now we get

$$m_i \approx \frac{\Delta_{f_\ell}}{\Delta_{f_i}},$$

where the approximation becomes arbitrarily exact when the Δ-amounts of change approach 0. Notice that $m_\ell = 1$.

The approximation of marginal rates of substitution is illustrated in Figure 13. The fat curve is a contour of the value function and the solid line its tangent at $\mathbf{z}^h$. The marginal rate of substitution at $\mathbf{z}^h$ is the negative of the slope of that tangent. The slope of the approximating dash line is quite different.

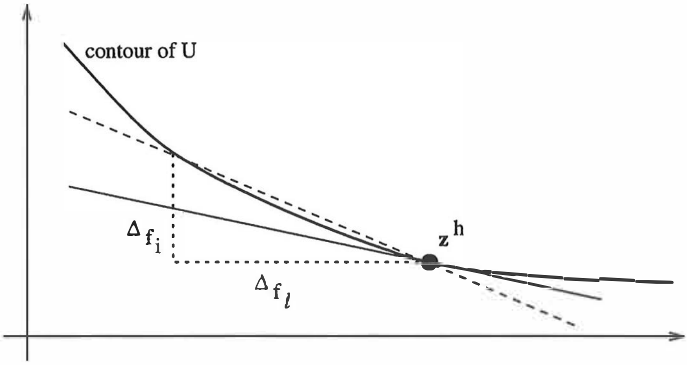


Figure 13. An approximation of the marginal rate of substitution.

It is remarked in [Sawaragi, Nakayama, Tanino, 1985] that, in practice, the Δ-amounts of change cannot be made arbitrarily small near 0, because human beings cannot recognize small changes within some extent. This threshold of human recognition is called a *just noticeable difference*. This is why the marginal rates of substitution are always approximations of the correct values. An example of the effects of the just noticeable difference is given in [Nakayama, 1985(a)] by illustrating how the solution process may terminate at a wrong solution. This is why one may have doubts that marginal rates of substitution are not adequate as a means of providing preference information. They seem to be difficult for the decision maker to answer and their accuracy is questionable.

However, we must now assume that the marginal rates of substitution are provided accurately enough. According to the FW method the maximization of the linear approximation of U is equivalent to the problem

$$\text{(2.14.3)} \qquad \begin{array}{ll} \text{maximize} & \left(\sum_{i=1}^{k} -m_i^h \nabla_x f_i(\mathbf{x}^h)\right)^T \mathbf{y} \\ \text{subject to} & \mathbf{y} \in S \end{array}$$

with $\mathbf{y}$ being the variable. The solution is denoted by $\mathbf{y}^h$. The existence of the optimal solution is ensured by the compactness of S and the continuity of all the functions.

The search direction is now $\mathbf{d}^h = \mathbf{y}^h - \mathbf{x}^h$. Provided that the marginal rates of substitution are reasonably accurate, the search direction should be usable. A scaling idea presented in [Clinton, Troutt, 1988] can be included in the method. Heterogeneous objective functions can be scaled to have equal effect in the problem (2.14.3) by adjusting the norms of the gradients of the objective functions with scalar coefficients.

The following problem is to find the step-size for how far to go into the search direction. Now the only variable is the step-size. The decision maker can be offered criterion vectors, where $z_i = f_i(\mathbf{x}^h + t\mathbf{d}^h)$ for $i = 1, \ldots, k$, and t varies stepwise between 0 and 1 (e.g., $t = \frac{j-1}{P-1}$ where $j = 1, \ldots, P$, and P is the number of the alternative criterion vectors to be presented). Another possibility is to draw the criterion values as a function of t, provided no serious scaling problems exist. An example of the graphical presentation is given in [Hwang, Masud, 1979]. The graphical illustration of the alternative criterion vectors will be handled in Chapter 4. Notice the fact that the alternatives are not necessarily Pareto optimal. From the information given to the decision maker (s)he selects the most preferred criterion vector and the corresponding value of t is selected as t^h. It is obvious that the task of selection becomes more difficult for the decision maker as the number of the objective functions increases.

The opinions of the decision maker and the situation $\mathbf{y}^h = \mathbf{x}^h$ are used here as stopping criteria. Other possible criteria have been presented in [Hwang, Masud, 1979] and [Yu, 1985].

The GDF method can be characterized to be a non ad hoc method. If one knows the value function, it is easy to specify the marginal rates of substitution and select the best alternative. The convergence properties of the GDF method are closely related to the convergence properties of the FW method. The convergence of the FW algorithm under the assumptions provided at the beginning of this section, is proved in [Zangwill, 1969]. However, it must be kept in mind that the correctness of the values for marginal rates of substitution and step-sizes affects the convergence remarkably. If it is supposed that the answers of the decision maker become ever more exact as the solution process continues, it is asserted in [Geoffrion, Dyer, Feinberg, 1972] that infinite convergence holds.

More important than infinite convergence in an interactive procedure like this is the initial rate of convergence, since a satisfactory solution should be found in a reasonable number of iterations. It is claimed in [Geoffrion, Dyer, Feinberg, 1972] that the error in the objective function values is at least halved at each of the first H iterations (H is unknown). The convergence becomes slower near the optimum because of the zig-zag-phenomenon. The effects of errors in estimating the gradient of the value function have been investigated in [Dyer, 1974]. The result is that even if the answers of the decision maker are not strictly consistent and the just noticeable difference affects the marginal rates of substitution, the method is stable and converges (only slower) under certain assumptions.

Applications and Extensions

The GDF method has been applied in [Geoffrion, Dyer, Feinberg, 1972] to the operation of an academic department. Numerical examples are also given, for example, in [Hwang, Masud, 1979] and [Steuer, 1986]. The GDF method is adapted for continuous equilibrium network design problems in [Friesz, 1981]. A time-sharing computer program implementing the GDF algorithm is suggested in [Dyer, 1973(a)].

In [Dyer, 1972], Dyer presents a method called interactive goal programming, which is a combination of the GDF method and goal programming. The vector $\mathbf{y}^h$ is obtained by the means of Archimedian goal programming with the marginal rates of substitution as weights. Also some convergence results are given. The GDF method and the interactive goal programming method are applied in [Jedrzejowicz, Rosicka, 1983] to multiobjective reliability optimization problems appearing in multiple classes of system failures.

The GDF method has been a subject of many modifications in the literature. New versions have been mainly developed to overcome some of the weaknesses of the GDF method. In [Hemming, 1981], a simplex-based direction finding problem is proposed for MOLP problems to avoid the specification of the marginal rates of substitution. It is stressed that the convergence properties may impair but the mental burden placed on the decision maker is diminished. Also a revised step-size problem is presented to produce Pareto optimal solutions.

The GDF method is altered in [Rosinger, 1981, 1982] by constructing a wide family of possible inquiry patterns to lead into the determination of the marginal rates of substitution. The decision maker can choose the form of the inquiry at each iteration. The convergence of the method is also proved.

A so-called proxy approach is introduced in [Oppenheimer, 1978]. The value function is no longer approximated linearly. The idea is to give a local proxy to the value function at each iteration. A sum-of-powers or a sum-of-exponentials proxy is fitted locally by specifying parameters connected to the problem. Now direction finding and step-size determination problems are replaced by the maximization of the proxy function. The proxy is not a valid approximation globally, but when used locally, it gives a higher convergence rate than the original GDF method. Even this method does not guarantee that the solutions are Pareto optimal. Oppenheimer has not established any systematic procedure for maximizing the proxy function. A method where Oppenheimer's ideas are improved and utilized is presented in Section 2.15.

Several modifications of the GDF method are presented in [Sadagopan, Ravindran, 1986]. First, the authors replace the FW method by a generalized reduced gradient method. Then, the role of the decision maker is facilitated by asking intervals for the marginal rates of substitution instead of exact values. The step-size is computed with the help of upper and lower bounds for the objective functions without the decision maker. In [Musselman, Talavage, 1980], the idea of the adaptation is to reduce the feasible region according to the marginal rates of substitution given by the decision maker. Such solutions are dropped that have lower values of the value function than the current solution. The method permits sensitivity analysis of the decision maker's inputs. The GDF method is modified for MOLP problems in [Winkels, Meika, 1984] so that when determining the step-size at each iteration, the criterion vectors are projected with a so-called efficiency projection onto the Pareto optimal set. This is done by solving a parametric linear programming problem.

Finally, we mention a modification of the GDF method for nondifferentiable multiobjective optimization problems, presented in [Miettinen, Mäkelä, 1991, 1993, 1994] and handled in Section 2.22. The modified method is called a subgradient GDF method, because nondifferentiable objective functions are allowed.

Concluding Remarks

In spite of the plausible theoretical foundation of the GDF method, it is not so convincing and overwhelming in practice. A drawback of the GDF method is that the final solution obtained is not necessarily Pareto optimal. In fact, when several alternatives are given to the decision maker for selecting the step-size, it is likely that many of them are not Pareto optimal.

Theoretically, the Pareto optimality of the final solution is guaranteed if the value function is componentwise decreasing (by Theorem 1.4.2). Anyway, marginal rates of substitution are crucial in approximating the value function and for many decision makers difficult and troublesome to specify. Even more difficult is to give consistent and correct marginal rates of substitution at every iteration. The difficulties of the decision maker in determining the marginal rates of substitution are demonstrated, for example, in [Wallenius, 1975] by comparative tests. The same point is illustrated by an example in [Hemming, 1981] when a politician is asked "What increase in unemployment would exactly compensate a decrease of 1 % in the inflation rate?"

The non-Pareto optimality can be overcome by projecting the alternatives onto the Pareto optimal set before presenting them to the decision maker. The projection may be done, for instance, by lexicographic ordering or by the means presented in Section 1.9. The use of scalarizing functions is demonstrated more closely in Section 2.22. The weakness in the projection is that the computational burden increases. It is for the analyst and the decision maker to decide which of the two shortcomings is less inconvenient.

The Frank-Wolfe gradient method has been selected as the maximization algorithm for its fast initial convergence. In some cases, other gradient-based methods may be more appropriate. Employing the (Kiev) subgradient method is described in Section 2.22.

There are a lot of assumptions that the problem to be solved must satisfy in order the method to work and converge. Several sufficient conditions on the decision maker's preferences are presented in [Sawaragi, Nakayama, Tanino, 1985] to guarantee the differentiability and the concavity of the value function. Even the new conditions are not very easy to check. For more critical discussion about the GDF method we refer to [Sawaragi, Nakayama, Tanino, 1985].

2.15. Sequential Proxy Optimization Technique

The sequential proxy optimization technique (SPOT), presented in [Sakawa, 1982], is based on the idea of maximizing the decision maker's value function, which is once again supposed to be known (implicitly). Some properties of the previously presented ISWT and GDF methods have been included.

Introduction

As in the two methods presented so far, the search direction in SPOT is obtained by approximating locally the gradient of the value function, and the step-size is determined according to the preferences of the decision maker. Here, both marginal rates of substitution and trade-off rates are used in approximating the value function.

It is assumed that

1. The underlying value function $U: \mathbf{R}^k \to \mathbf{R}$ exists and is implicitly known to the decision maker. In addition, U is a continuously differentiable, strictly decreasing and concave function on the subset of Z where the points are Pareto optimal.
2. All the objective and the constraint functions are convex and twice continuously differentiable.
3. The feasible region S is compact and convex (and there exist some upper bounds for the ε-constraint problem so that the solution is finite).

The ε-constraint problem is used for generating Pareto optimal solutions. Throughout this section it is assumed that all the upper bound constraints are active at the optimum. (If this is not the case, then the upper bounds must be slightly modified.) The solution of the ε-constraint problem (2.3.1) is supposed to be unique (so that Pareto optimality is guaranteed) and it is denoted by $\mathbf{x}^h$. Then, $f_j(\mathbf{x}^h) = \varepsilon_j^h$ for all $j \neq \ell$. The optimal value of f_ℓ, that is, $f_\ell(\mathbf{x}^h)$, is denoted by z_ℓ^h. It is also assumed that all the Kuhn-Tucker multipliers associated with the active constraints are strictly positive. The conditions of Theorem 2.3.17 are supposed to be satisfied so that trade-off information can be obtained from the Kuhn-Tucker multipliers.

The value function is not maximized here in the form (2.9.1) as before. Instead, the set of feasible alternatives is restricted to the Pareto optimal set. According to the above assumption $f_j(\mathbf{x}^h) = \varepsilon_j^h$ for all $j \neq \ell$, we get a new formulation

$$\text{(2.15.1)} \qquad \text{maximize} \quad U(\varepsilon_1^h, \ldots, \varepsilon_{\ell-1}^h, z_\ell^h, \varepsilon_{\ell+1}^h, \ldots, \varepsilon_k^h).$$

No constraints are needed here since the formulation includes the original constraints. The optimization is now carried out in the criterion space $\mathbf{R}^{k-1}$, where the ε_j^h:s are the variables.

It is proved in [Sakawa, 1982] that the new function is concave with respect to those $\boldsymbol{\varepsilon} \in \mathbf{R}^{k-1}$ for which the upper bound constraints are all active. Sakawa also claims that the partial derivative of (2.15.1) with respect to ε_j^h is equivalent to $\frac{d\,U(\cdot)}{d\,f_\ell}(m_{\ell j}^h - \lambda_{\ell j}^h)$ for $j \neq \ell$, where $m_{\ell j}^h$ is the marginal rate of substitution between f_ℓ and f_j at $\mathbf{x}^h$ (obtained from the decision maker, see Section 2.14) and $\lambda_{\ell j}^h$ is the partial trade-off rate between f_ℓ and f_j at $\mathbf{x}^h$ (obtained from the optimal Kuhn-Tucker multipliers, see Sections 2.3 and 2.13).

Because it was assumed that the value function is strictly decreasing, we know that $\frac{d\,U(\cdot)}{d\,f_\ell} < 0$ and we can divide by it. Now we denote

$$-(m_{\ell j}^h - \lambda_{\ell j}^h) = \Delta\varepsilon_j^h$$

for $j \neq \ell$, and it represents the direction of steepest ascent of the value function (2.15.1) at the current point $\mathbf{x}^h$ for $j \neq \ell$. According to Sakawa, the ℓth component of the direction is

$$\sum_{j \neq \ell} \lambda_{\ell j}^h (m_{\ell j}^h - \lambda_{\ell j}^h)$$

denoted by Δz_ℓ^h.

After obtaining the search direction, we have to find the step-size t which in theory maximizes the function

$$\text{(2.15.2)} \quad U(\varepsilon_1^h + t\Delta\varepsilon_1^h, \ldots, \varepsilon_{\ell-1}^h + t\Delta\varepsilon_{\ell-1}^h, z_\ell^h + t\Delta z_\ell^h, \varepsilon_{\ell+1}^h + t\Delta\varepsilon_{\ell+1}^h, \ldots, \varepsilon_k^h + t\Delta\varepsilon_k^h).$$

The step-size could be determined as earlier by presenting different criterion vectors for the decision maker with different values of t and by letting the decision maker choose the most preferred one. The problem with the alternative criterion vectors is that they are not necessarily Pareto optimal. This weakness can be overcome by solving the ε-constraint problem with the upper bounds $\varepsilon_j^h + t\Delta\varepsilon_j^h$, $j \neq \ell$. We still assume that the constraints are active. The amount of calculations increases now since the ε-constraint problem must be solved for several values of t but now the alternatives are guaranteed to be Pareto optimal.

However, the step-size is not determined by the above described way. To aid the decision maker in maximizing (2.15.2), local proxy preference functions p (in the same spirit as presented in Section 2.14 in connection with the GDF method) are now applied. According to the preference structure of the decision maker a sum-of-exponentials, a sum-of-powers or a sum-of-logarithms proxy function of the form $-\sum_{i=1}^{k} a_i e^{-\omega f_i(\mathbf{x})}$, $-\sum_{i=1}^{k} a_i(n_i + f_i(\mathbf{x}))^{\alpha_i}$ or $\sum_{i=1}^{k} a_i \ln(n_i - f_i(\mathbf{x}))$, respectively, is selected. The constants $\boldsymbol{a}_i, \omega_i, n_i$ and α_i are used for tuning the proxy functions to represent the preferences of the decision maker better, see, for example, [Sakawa, 1982] or [Sakawa, Seo, 1982(b)] for further details. This kind of a proxy function is very restrictive globally but reasonable when assumed locally. The parameters needed to fit the selected proxy function into the current problem are obtained from the marginal rates of substitution.

SPOT Algorithm

Now we can present the basic ideas of the SPOT algorithm.

(1) Choose a reference function and upper bounds $\boldsymbol{\varepsilon}^1 \in \mathbf{R}^{k-1}$ for which all the constraints of the ε-constraint problem are active. Set $h = 1$.

(2) Solve the current (active) ε-constraint problem for $\boldsymbol{\varepsilon}^h$ to obtain a Pareto optimal criterion vector $\mathbf{z}^h$ and the optimal Kuhn-Tucker multipliers $\lambda_{\ell j}^h$, $j \neq \ell$.

(3) Obtain the marginal rates of substitution at $\mathbf{z}^h$ from the decision maker. Test the consistency of the marginal rates of substitution and ask the decision maker to respecify them if necessary.

(4) If $|m_{\ell j}^h - \lambda_{\ell j}^h| < \theta$, where θ is a prespecified positive tolerance, then $\mathbf{z}^h$ is the final solution. Otherwise, determine the components $\Delta\varepsilon_j^h$, $j \neq \ell$, and Δz_ℓ^h of the direction vector.

(5) Select the appropriate form of the proxy function and calculate its parameters. If the obtained proxy function is not strictly decreasing and concave, then ask the decision maker to specify new marginal rates of substitution.

(6) Determine the step-size by solving the ε-constraint problem with the upper bounds $\varepsilon_j^h + t\Delta\varepsilon_j^h$, $j \neq \ell$. Denote the optimal value of the objective function by $z_\ell^h(t)$. A step-size t^h is selected which maximizes the proxy function. If the new criterion vector $(\varepsilon_1^h + t^h\Delta\varepsilon_1^h, \ldots, z_\ell^h(t^h), \ldots, \varepsilon_k^h + t^h\Delta\varepsilon_k^h)^T$ is preferred to $\mathbf{z}^h$, then denote it by $\mathbf{z}^{h+1}$, set $h = h + 1$ and go to step (2). If the decision maker prefers $\mathbf{z}^h$ to the new solution, reduce t^h to be $\frac{1}{2}t^h, \frac{1}{4}t^h, \ldots$ until improvement is achieved.

The maximum of the proxy function is determined by altering the step-size t, obtaining the corresponding Pareto optimal solution and searching for three t values, t_1, t^h and t_2 so that $t_1 < t^h < t_2$ and $p(t_1) < p(t^h) > p(t_2)$, where p is the proxy function. When the above condition is satisfied, the local maximum of the proxy function $p(t)$ is in the neighbourhood of t^h.

Under the assumptions 1–3, the optimality condition for the problem (2.15.1) at ε^h is that the gradient equals zero at that point. This means that $m_{\ell j}^h = \lambda_{\ell j}^h$ for $j = 1, \ldots k$, $j \neq \ell$. This is the background of the absolute value checking at step (4) (see also Section 1.7).

The consistency of the marginal rates of substitution is checked because it is important for the successful convergence of the algorithm. The consistency at a single point is tested by the chain rule and by limiting the discrepancy (the formula is given in [Sakawa, 1982]) by a given tolerance level. The consistency at successive points is tested by checking the concavity and monotonicity of the proxy function (the proxy function must fulfill the same assumptions as the value function). A theorem giving conditions for different types of proxy functions is presented in [Sakawa, 1982].

To ensure the convergence of the algorithm it must, at each iteration, be checked that sufficient improvement is obtained. If the decision maker prefers the new solution, the procedure may continue. Otherwise, a new step-size must be estimated.

It is remarked in [Sakawa, 1982] that the SPOT algorithm is nothing but a feasible direction method as for the convergence rate. The convergence can be demonstrated by the convergence of the modified feasible direction method. For this statement to be true an ideal (i.e., consistent with correct answers) decision maker must be assumed. SPOT can be classified to belong to methods of non ad hoc nature. If the value function is known, then the marginal rates of substitution can be computed directly.

Applications and Extensions

The functioning of the SPOT algorithm is demonstrated in [Sakawa, 1982] by an academic example. It is shown that even though the marginal rates of substitution are only approximations, it did not worsen the results remarkably. A problem concerning industrial pollution in Osaka City in Japan is solved by SPOT in [Sakawa, Seo, 1980, 1982(a), (b)]. The problem is defined as a large-scale problem in [Sakawa, Seo, 1980] and a dual decomposition method is used to solve the ε-constraint problems.

Fuzzy SPOT is presented in [Sakawa, Yano, 1985]. The decision maker is supposed to assess the marginal rates of substitution in a fuzzy form. In [Sakawa, Mori, 1983], a new method for nonconvex problems is proposed, where the weighted L_∞-metric is used to generate Pareto optimal solutions instead of the ε-constraint method, and trade-off rates are not used. A method related to the preceding one is presented in [Sakawa, Mori, 1984]. The difference is that a penalty scalarizing function is used in generating Pareto optimal solutions (see Section 2.19). This method is also applicable to nonconvex problems.

Concluding Remarks

Ideas of several methods have been combined in SPOT and several concepts are utilized. The difficulties in determining the marginal rates of substitution mentioned in Section 2.14 are still valid. The consistency of the marginal rates of substitution is in SPOT even more important than in the GDF method.

A positive feature in SPOT when compared to the GDF method is that only Pareto optimal solutions are handled. Also the burden on the decision maker is smaller because a proxy functions is employed when selecting the step-size.

Many assumptions are set to guarantee the proper functioning of the algorithm. Some of them are quite difficult to check in practice (see concluding remarks of the GDF method).

Because the multiobjective optimization problem was assumed to be convex, globally Pareto optimal solutions are obtained. If the convexity assumptions are relaxed, locally Pareto optimal solutions are to be considered.

2.16. Zionts-Wallenius Method

After presenting several multiobjective optimization methods based on the optimization of the underlying (implicitly known) value function, we shortly mention one more method based on the same idea. It is called a Zionts-Wallenius (ZW) method and it was originally presented in [Zionts, Wallenius, 1976] for maximizing MOLP problems with linear implicit value functions. In [Wallenius, Zionts, 1977], it was presented more thoroughly with some experiences of implementations. The method was developed in [Zionts, Wallenius, 1983] to handle certain types of nonlinear value functions. Here we present the method only briefly in the latter, more general form, and for minimization problems.

General Outline

It is assumed that

1. The underlying value function U exists and is implicitly known to the decision maker. In addition, U is a pseudoconcave, nonincreasing and continuously differentiable function on Z (possible are, e.g., linearly additive or concave functions).
2. The objective functions are linear.
3. The constraint functions are linear forming a closed and bounded feasible region S.

The assumption 2 can be generalized to convex objective functions by considering piecewise linearizations.

Pareto optimal solutions are generated in the ZW method by solving weighting problems with positive weights (see Section 2.2) by the simplex algorithm. This means that the (nonlinear) underlying value function is approximated locally by linear weighting problems. (Remember that the weighting problem corresponds to a linear value function.) In other words, an "optimal" weighting vector is tried to determine.

The basic ideas of the algorithm are the following. After obtaining a Pareto optimal criterion vector $\mathbf{z}^h \in Z$ by the weighting method, all its adjacent Pareto optimal extreme points (in the criterion space) are identified (see [Zionts, Wallenius, 1975, 1980]) and the decision maker is asked to compare each of them with $\mathbf{z}^h$ (i.e., express preference or indifference). According to the responses of the decision maker, the space of the weighting vectors is restricted. If two solutions are too close to each other to be compared, total trade-off rates (obtained from the reduced costs portion of the optimal simplex tableau) are utilized. Trade-off rates are also used in situations where the decision maker wants to move along a Pareto optimal edge leading from $\mathbf{z}^h$ to some adjacent Pareto optimal extreme point but not as far as the extreme point. The decision maker's opinions of the trade-off rates are also used to restrict the space of weighting vectors. Thus, the ZW method transforms the preference information of the decision maker into constraints on the weighting vector space. At the next iteration, the weighting vector for the weighting problem is selected from the contracted space. If the contracted weighting vector space is empty (this may happen if the decision maker is not consistent), then the oldest constraints are deleted until the space becomes nonempty. The algorithm continues until the decision maker does not prefer any of the adjacent Pareto optimal solutions.

The purpose of the method is to generate a sequence of improved extreme solutions. In [Zionts, Wallenius, 1983], it is proved that the solution $\mathbf{x}^* \in S$ and the corresponding $\mathbf{z}^* \in Z$ are globally optimal (with respect to the underlying U), if the decision maker does not prefer any Pareto optimal edge emanating from $\mathbf{z}^*$. If we only know that the decision maker does not prefer any adjacent Pareto optimal extreme point, then the solution can be regarded as a locally optimal solution (with respect to U) and better solutions may exist in the relative interior of some adjacent facet. Such a point cannot be found by the ZW method but some other algorithm must be applied. If the answers of the decision maker are consistent and correct, the method may converge rapidly. A positive feature is that earlier mistakes can be corrected during the solution process (by deleting old constraints on the weighting vector space). The ZW method can be characterized as a non ad hoc method. The questions posed by the algorithm can be answered deterministically if the value function is known.

Without taking any negative attitude, we mention that the ZW method was scored the worst of four interactive methods compared in [Buchanan, Daellenbach, 1987] (see Section 5). The problem was not in the easiness of using or understanding the method, but the decision makers felt that they were at the mercy of the method. Thus, the decision makers thought that they were not able to take enough responsibility of the solution process. Notice that all the interactive methods described so far share this property is common. They are algorithm-led rather than decision maker-led methods.

Applications and Extensions

A Fortran implementation of the ZW algorithm is outlined in [Breslawski, Zionts, 1985]. Another way of generalizing the original ZW method of [Zionts, Wallenius, 1976] for pseudoconcave value functions is suggested in [Korhonen, Wallenius, 1984]. This approach is based on so-called reference directions (see Section 2.21). A new variation is presented also in [Koksalan, Beseli, 1989], where the two versions of Zionts and Wallenius have been combined. This method does not require any trade-off information.

An extension of the ZW method, presented in [Ramesh, Karwan, Zionts, 1989(a)], pays attention to inconsistent answers of the decision maker. In the ZW method, the problem is handled by dropping old constraints of the weighting vector space. This may mean a loss of information on the preference structure, and more questions have to be asked to return that information. In the extension, convex cones are formed according to the answers of the decision maker representing the preferences. Even though the constraints on the weighting vector space are deleted, the cones are not changed. This extension is modified for integer programming in [Ramesh, Karwan, Zionts, 1989(b)].

The ZW method is extended for concave (maximization case) objective and value functions and convex feasible regions in [Roy, Wallenius, 1992]. A more general case of nonlinear objective functions, nonconvex feasible regions and concave value functions is also discussed. The generalized reduced gradient method is used for optimization since it is based on the ideas of the simplex method. Trade-off rates are still presented to the decision maker to obtain information but a proxy function is used to approximate the value function. In addition, the ZW method is adapted for multiple decision makers in [Korhonen, Moskowitz, Wallenius, Zionts, 1986].

The method in [White, 1980] is based on the same idea as the ZW method by trying to determine the optimal weights of a linear underlying value function. The realization is different but related to the ZW method.

Concluding Remarks

The role of the decision maker has been tried to keep rather simple in the ZW method. (S)he is only asked to compare two criterion vectors at a time. However, the number of comparisons per iteration depends on the problem to be solved, and, if the problem is large, then many questions have to be asked from the decision maker until every extreme point and adjacent facet has been explored. When comparing, for example, the ZW method and the GDF method, the key difference is that the ZW method requires more ordinal comparisons, whereas the GDF method requires fewer, but more complicated, indifference judgements.

The concept of trade-off rates must be clear to the decision maker in order to guarantee satisfactory progress. If the trade-offs cause problems, it may be positive that (unlike the ISWT method) the ZW method is not based completely on them. It is still quite a substantial requirement that the decision maker has a reasonable (implicit) value function. To assume that the value function is linear is a very serious restriction. In that case, the convergence is guaranteed better, though.

Employing the weighting method guarantees the solutions to be Pareto optimal. The weakness of the ZW method for MOLP problems is that it can only find optimal solutions which are at the edges of the feasible criterion region. Naturally, the seriousness of this weakness depends on the structure of the problem.

2.17. Interactive Weighted Tchebycheff Procedure

The interactive weighted Tchebycheff (IWT) procedure, proposed in [Steuer, Choo, 1983] and [Steuer, 1986] and refined in [Steuer, 1989(a)], is an interactive weighting vector space reduction method. Thus, it has something in common with the ZW method even though the two methods are of different structure. A remarkable difference when compared to the methods described thus far is that the value function plays no central role in the IWT method. In this presentation, we introduce the IWT algorithm according to the refined version but modified for minimization problems.

Introduction

The IWT method has been designed to be easy to use for the decision maker, and, thus, complicated information is not required. To start with, a utopian vector below the ideal criterion vector is established. Then the distance from the feasible criterion region to the utopian vector, measured by a weighted Tchebycheff metric, is minimized. Different solutions are obtained with different weighting vectors in the metric. The space of solutions is reduced by working with sequences of progressively smaller subsets of the weighting vector space. Thus, the idea is to develop a sequence of progressively smaller subsets of the Pareto optimal set until a final solution is located. Different alternative criterion vectors are presented to the decision maker and (s)he is asked to select the most preferred of them. The feasible region is then reduced and alternatives from the reduced space are presented to the decision maker for selection.

Contrary to many other interactive methods for multiobjective optimization, the IWT procedure does not presume many assumptions on the problem to be solved. It is assumed that

1. The underlying value function is monotone.
2. All the objective functions are bounded over the feasible region S.

The *utopian vector* $\mathbf{z}^{**}$ is defined to be an infeasible criterion vector that strictly dominates every Pareto optimal solution. The components are formed by setting

$$z_i^{**} = z_i^{\star} - \varepsilon_i$$

for all $i = 1, \ldots, k$, where $z_i^{\star}$ is a component of the ideal criterion vector and $\varepsilon_i > 0$ is a relatively small but computationally significant scalar.

The metric to be used for measuring the distances, the weighted Tchebycheff (L_∞-) metric (see Section 2.7) is

$$\max_{1 \le i \le k} [w_i(f_i(\mathbf{x}) - z_i^{**})],$$

where $\mathbf{w} \in W = \{\mathbf{w} \in \mathbf{R}^k \mid 0 < w_i < 1,\ \sum_{i=1}^k w_i = 1\}$. We have a family of metrics since $\mathbf{w} \in W$ can vary widely. The nondifferentiable problem of minimizing the distance between $f_i(\mathbf{x}) \in Z$ and $\mathbf{z}^{**}$ with the weighted Tchebycheff metric can be solving as a differentiable *weighted Tchebycheff problem* (2.7.11) (where the ideal criterion vector is replaced by the utopian vector) instead of the nondifferentiable min-max problem.

Adapting Theorem 2.7.15, we know that every Pareto optimal solution of any multiobjective optimization problem can be found by solving the problem (2.7.11) with $\mathbf{z}^{**}$. The negative aspect with the weighted Tchebycheff problem is that some of the solutions may be weakly Pareto optimal (though according to Theorem 2.7.14 at least one of the solutions is Pareto optimal). This weakness has been overcome in the IWT procedure by formulating the distance minimization problem as a *lexicographic weighted Tchebycheff problem*

$$(2.17.1) \qquad \begin{array}{ll} \text{lex minimize} & \alpha, \sum_{i=1}^{k}(f_i(\mathbf{x}) - z_i^{**}) \\ \text{subject to} & \alpha \ge w_i(f_i(\mathbf{x}) - z_i^{**}), \quad \text{for all } i = 1, \ldots, k, \\ & \mathbf{x} \in S. \end{array}$$

The functioning of the problem (2.17.1) is described in Figure 14 by a problem with two objective functions. The fat line illustrates the Pareto optimal set. The problem with α as the objective function has L-shaped contours (the thin solid line) whose vertices lie along the line emanating from $\mathbf{z}^{**}$ in the direction $(\frac{1}{w_1}, \frac{1}{w_2}, \ldots, \frac{1}{w_k})$. When minimizing α, such a contour is determined which is closest to $\mathbf{z}^{**}$ and intersects Z. If this problem does not have a unique solution, that is, there are several feasible points on the optimal contour intersecting Z, then some of them may not be Pareto optimal. (In practice, the uniqueness is usually difficult to check, and, to be on the safe side, the following step must be taken.) In this case, the sum term is minimized subject to the obtained points to determine which of them is closest to $\mathbf{z}^{**}$ according to the L_1-metric (the dashed line). Thus a unique solution, which is Pareto optimal (see Theorem 2.7.12), is obtained. It is proved in more detail in [Steuer, 1986] that every solution of the lexicographic weighted Tchebycheff problem is Pareto optimal and for every Pareto optimal solution there exists a weighting vector $\mathbf{w} \in W$ such that the solution is unique for the lexicographic weighted Tchebycheff problem.

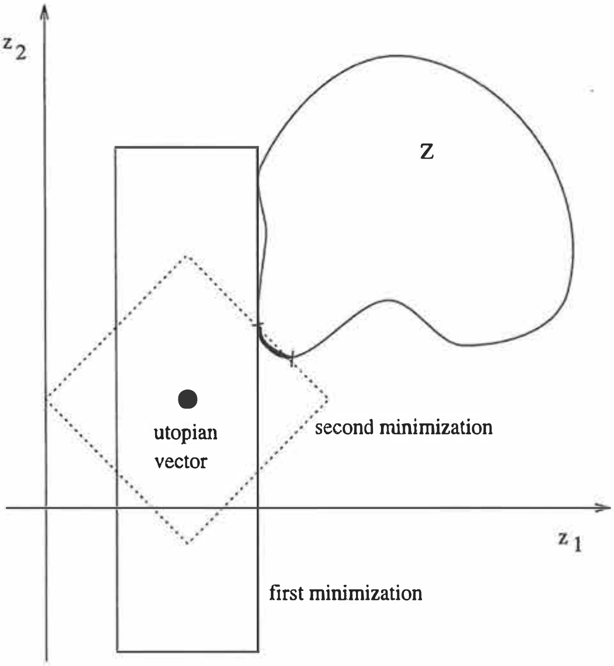


Figure 14. Lexicographic weighted Tchebycheff problem.

Different Pareto optimal solutions can be obtained by altering the weighting vector. At each iteration h, the weighting vector space $W^h = \{\mathbf{w}^h \in \mathbf{R}^k \mid l_i^h < w_i^h < u_i^h, \sum_{i=1}^k w_i^h = 1\}$ is reduced to W^{h+1}, where $W^{h+1} \subset W^h$. With a sequence of progressively smaller subsets of the weighting vector space a sequence of smaller subsets of the Pareto optimal set is sampled.

At the first iteration, a sample of the whole Pareto optimal set is generated by solving the lexicographic weighted Tchebycheff problem with well dispersed weighting vectors from $W = W^1$ (with $l_i^1 = 0$ and $u_i^1 = 1$). The reduction of W^h is done by tightening the upper and the lower bounds for the weighting vectors. Let $\mathbf{z}^h$ be the criterion vector that the decision maker chooses from the sample at the hth iteration and let $\mathbf{w}^h$ be the corresponding weighting vector in the problem (2.17.1). Now, a concentrated group of weighting vectors centred around $\mathbf{w}^h$ is formed. In this way, a sample of Pareto optimal solutions centred about $\mathbf{z}^h$ is obtained.

The number of the alternative criterion vectors to be presented to the decision maker is denoted here by P. The number is usually specified by the decision maker. It may be fixed or different at each iteration. The algorithm becomes more reliable, if as many alternatives as possible can be evaluated effectively at each iteration. Human capabilities are yet limited, and some kind of a compromise is desirable.

When reducing the weighting vector space at each iteration, a reduction factor r is needed. The larger the reduction factor is, the faster the weighting vector space is reduced and the smaller are the decision maker's possibilities for making errors and changing her or his mind about aspirations during the process. The correct selection of r is thus important. It is suggested in [Steuer, Choo, 1983] and [Steuer, 1986] that $(1/P)^{1/k} \lessapprox r \lessapprox v^{1/(H-1)}$, where v is the final interval length of the weighting vectors, $\frac{1}{2k} \lessapprox v \lessapprox \frac{3}{2k}$, H is the number of iterations to be carried out and $\lessapprox$ stands for "approximately equal or less".

IWT Algorithm

Now we can present the main features of the IWT algorithm.

(1) Specify values for the set size $P(\gtrapprox k)$, reduction factor r and an approximation for the number of iterations $H(\approx k)$. Set $l_i^1 = 0$ and $u_i^1 = 1$ for all $i = 1, \dots, k$. Construct the utopian vector. Set $h = 1$.

(2) Form the weighting vector space $W^h = \{\mathbf{w}^h \in \mathbf{R}^k \mid l_i^h < w_i^h < u_i^h, \ \sum_{i=1}^k w_i^h = 1\}$.

(3) Generate $2P$ dispersed weighting vectors $\mathbf{w}^h \in W^h$.

(4) Solve the lexicographic weighted Tchebycheff program (2.17.1) for each of the $2P$ weighting vectors.

(5) Present P most different of the resulting criterion vectors to the decision maker and let her or him choose the most preferred among them, denoting it by $\mathbf{z}^h$.

(6) If $h = H$ go to step (8). Otherwise, modify, if necessary, the weighting vector corresponding to $\mathbf{z}^h$ such that if the problem (2.17.1) was solved again, $\mathbf{z}^h$ would be a uniquely generated solution at the vertex of the intersecting new contour.

(7) Specify l_i^{h+1} and u_i^{h+1} for the reduced weighting vector space W^{h+1}, set $h = h + 1$ and go to step (2).

(8) The final solution is $\mathbf{x}^h$ corresponding to $\mathbf{z}^h$.

Dispersed weighting vectors are generated from W^h in step (3). In practice, this can be realized by generating randomly a large set (e.g., $50k$) of weighting vectors. Then the vectors are filtered (see [Steuer, 1986]) or clustered (see [Miettinen, 1990]). The clustering is practical since subroutines for it are available in many subroutine libraries. While we want to obtain $2P$ well dispersed weighting vectors, we form $2P$ clusters and choose one candidate from each of them either arbitrarily or near the centre.

Computationally, the following subroutine can be used to obtain random weighting vectors in W^h. The subroutine has been developed for the implementation described in [Miettinen, 1990]. We leave the index h for clearness. For $i = 1, \dots, k$ set

$$we_i = l_i + ra(u_i - l_i),$$

where ra is a random number between 0 and 1. Calculate the sums $we = \sum_{i=1}^k we_i$, $l = \sum_{i=1}^k l_i$ and $u = \sum_{i=1}^k u_i$. Finally, set

$$w_i = \begin{cases} we_i + \frac{1-we}{l-we}(l_i - we_i) & \text{if } we > 1 \\ we_i + \frac{1-we}{u-we}(u_i - we_i) & \text{if } we < 1 \\ we_i & \text{otherwise.} \end{cases}$$

The lexicographic weighted Tchebycheff problem is solved $2P$ times for $2P$ weighting vectors (instead of P) in case the same or a very similar solution is obtained with different weighting vectors. The $2P$ (or less) criterion vectors are again filtered or clustered to obtain P most different. In this way, it is guaranteed that P different criterion vectors can be presented to the decision maker. For graphical illustration of the alternatives, see Chapter 4.

The modification of the weighting vector in step (6) can be conducted by setting

$$w_i^h = \frac{1}{f_i(\mathbf{x}^h) - z_i^{**}} \left(\sum_{i=1}^{k} \frac{1}{f_i(\mathbf{x}^h) - z_i^{**}} \right)^{-1}$$

for all $i = 1, \ldots, k$. The modification is relevant because the weighting vector space is reduced with respect to $\mathbf{w}^h$. It is good to have an unbiased basis for the reduction.

Several possibilities for reducing the weighting vector space have been suggested. It is proposed in [Steuer, 1986] that we set

$$(l_i^{h+1}, u_i^{h+1}) = \begin{cases} (0, r^h) & \text{if } w_i^h - \frac{r^h}{2} \leq 0 \\ (1 - r^h, 1) & \text{if } w_i^h + \frac{r^h}{2} \geq 1 \\ (w_i^h - \frac{r^h}{2}, w_i^h + \frac{r^h}{2}) & \text{otherwise,} \end{cases}$$

where r^h means raising r to the hth power. In [Steuer, 1989(a)], an auxiliary scalar ω is determined so that the ratio of the volumes of W^{h+1} and W^h is r. Then ω is used in the reduction instead of the term $\frac{1}{2}r^h$.

The predetermined number of iterations is not necessarily conclusive. The decision maker can stop iterating when (s)he obtains a satisfactory solution or can continue the solution process further if necessary.

It is suggested in [Steuer, 1986, 1989(a)] that the sampling of the Pareto optimal set works in the most unbiased way if the ranges of the objective function values over the Pareto optimal set are approximately the same. This can be accomplished by re-scaling the objective functions in the similar way as presented in Section 2.2, when necessary. It is advisable to use the scaling only in the calculations and present the alternatives to the decision maker in the original form. More suggestions for modifications of the algorithm are presented in [Steuer, 1989(a)].

The convergence rate of the IWT procedure is very difficult to establish. It is stressed in [Steuer, 1989(a)] that the IWT procedure is able to converge to any Pareto optimal solution. The reduction factor r is comprehended as a convergence factor. The weighting vector space is reduced until a solution is obtained that is good enough to be a final solution (see [Steuer, Choo, 1983]). The IWT procedure can be characterized as a non ad hoc method. If the value function is known, it is easy to select from the set of P alternatives the one which maximizes U. It is worth noticing that the IWT procedure, unlike many other methods, is able to converge to a non-extreme solution in MOLP problems.

We do not here go into details of the original version of the IWT procedure. We only mention that the possibility of getting weakly Pareto optimal solutions is handled by introducing an augmentation term to give a slight slope to the contour of the metric. An augmented weighted Tchebycheff metric is defined by

$$\max_{1 \leq i \leq k} [w_i(f_i(\mathbf{x}) - z_i^{**})] + \rho \sum_{i=1}^{k} (f_i(\mathbf{x}) - z_i^{**}),$$

where ρ is a sufficiently small positive scalar. The metric is illustrated in Figure 15, where the dashed lines represent the contours of the augmented metric. The contour of the weighted Tchebycheff metric (solid line) has been added only to ease the comparison with Figure 14. In this version, the lexicographic optimization is avoided, but the IWT algorithm is more complicated in other ways (see [Steuer, 1986]). A numerical illustration of the algorithm is presented in [Steuer, 1986]. Another implementation of the IWT algorithm is presented in [Miettinen, 1990], where several different graphical illustrations of the alternatives are presented to the decision maker at each iteration (see also Chapter 4).

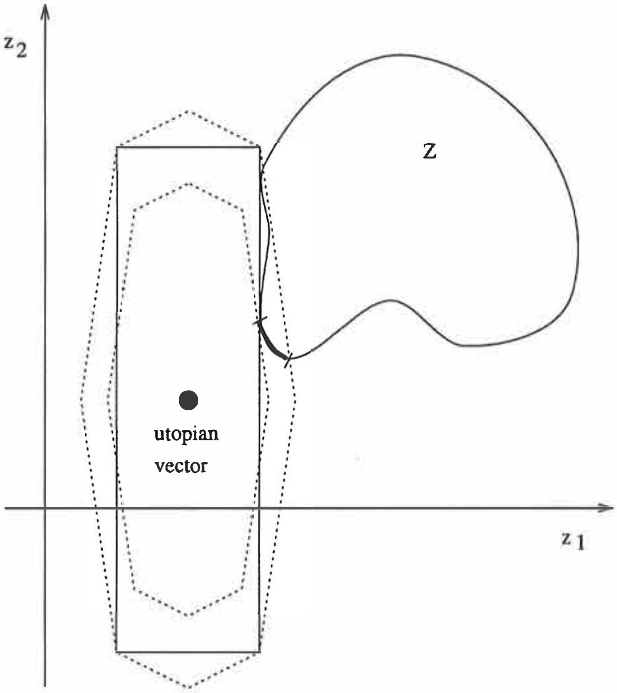


Figure 15. Augmented weighted Tchebycheff problem.

The IWT procedure in its original form is applied in [Wood, Greis, Steuer, 1982] to water allocation problems of a river basin and in [Silverman, Steuer, Whisman, 1988] to manpower supply forecasting. In [Olson, 1993], the IWT procedure is applied to a sausage blending problem. A method related to the IWT procedure is introduced in [Kaliszewski, 1987]. There, a so-called modified weighted Tchebycheff metric is used for measuring the distances.

Concluding Remarks

A positive feature in the IWT procedure is that the role of the decision maker is quite easy to understand. (S)he does not need to realize new concepts or specify numerical answers as, for example, in the ISWT and the GDF method. The easiness of comparing the alternative criterion vectors depends on the magnitude of P and on the number of the objective functions. Personal capabilities of the decision makers also play an important role. It is also good that all the alternatives are Pareto optimal.

The flexibility of the method is decreased by the fact that the discarded parts of the weighting vector space cannot be restored, if the decision maker changes her or his mind. Thus, some consistency is required.

The weakness of the IWT method is that very much calculation is needed at each iteration and many of the results are discarded. For large and complex problems, where the evaluation of the values of the objective functions may be laborious, the

IWT procedure is not a realistic choice. On the other hand, it is possible to utilize parallel computing since all the lexicographic problems can be solved independently.

Not that any absolute superiority could be concluded, we, however, mention that the IWT procedure performed best in the comparative evaluation of the ZW, the SWT and the IWT method, and a naive approach in [Buchanan, Daellenbach, 1987] (see Chapter 5). Anyhow, a difficulty encountered was comprehending the information provided. The test example had only three objective functions and six alternatives were presented at each iteration. And the burden nothing but becomes larger when the number of the objective functions is increased.

2.18. Step Method

The step method (STEM), presented in [Benayon, de Montgolfier, Tergny, Laritchev, 1971], contains elements somewhat similar to the IWT procedure, but is based on a different idea. STEM is one of the first interactive methods developed for multiobjective optimization problems. It was originally designed for MOLP problems to be maximized but can be extended for nonlinear problems (see, e.g., [Sawaragi, Nakayama, Tanino, 1985] or [Eschenauer, Osyczka, Schäfer, 1990]). It can be considered to aspire at finding satisficing solutions (as introduced in Section 1.4) instead of optimizing an underlying value function. The main features for minimizing MOLP problems are only briefly sketched in the following.

It is assumed that

1. The multiobjective optimization problem is linear.
2. The objective functions are bounded over the feasible region S.

General Outline

It is assumed in STEM that at a certain criterion point the hopes of the decision maker are fulfilled so that (s)he can give up a little in the value(s) of some objective function(s) in order to improve the values of some other objective functions. The ideal criterion vector $\mathbf{z}^\star$ is used as a reference point in the calculations and the distances are measured by the weighted Tchebycheff (L_∞-) metric.

The nadir point $\mathbf{z}^{\text{nad}}$ is approximated from the payoff table as explained in Section 1.5. Thus, the maximal element of the ith column is z_i^{nad}. Information about the ranges of the Pareto optimal set is used in generating the weighting vector for the metric. The weighting vector is calculated by the following formula

$$w_i = \frac{e_i}{\sum_{j=1}^k e_j}, \qquad i = 1, \dots, k,$$

where for every $i = 1, \dots, k$

$$e_i = \begin{cases} \dfrac{z_i^{\text{nad}} - z_i^\star}{z_i^{\text{nad}}} \left(\sum_{j=1}^n c_{ij}{}^2\right)^{-\frac{1}{2}} & \text{if } z_i^{\text{nad}} > 0 \\ \dfrac{z_i^{\text{nad}} - z_i^\star}{|z_i^\star|} \left(\sum_{j=1}^n c_{ij}{}^2\right)^{-\frac{1}{2}} & \text{otherwise.} \end{cases}$$

(The denominators are not allowed to be zero.) Here the terms c_{ij} are the coefficients of the linear objective functions. The first term in the multiplication gives more weight to such objective functions that are far from the ideal criterion vector and the latter term normalizes the gradients of the objective functions according to the L_2-norm.

The distance between the ideal criterion vector and the feasible criterion region is minimized by the weighted Tchebycheff metric (the weighting coefficients specified as above). The obtained solution and the ideal criterion vector are presented to the decision maker for comparison. Then the decision maker is asked to specify such objective function(s) whose value(s) (s)he is willing to relax (weaken) to decrease the values of some other objective functions. (S)he must also determine the amount(s) of acceptable relaxation. Ways of helping the decision maker in this phase are presented in [Benayon, de Montgolfier, Tergny, Laritchev, 1971]. The feasible region is restricted according to the information of the decision maker and the weights of the relaxed objective functions are set equal to zero. Then a new distance minimization problem is solved and the procedure continues until the decision maker does not want to change any component. If the decision maker is not satisfied with any of the components, then the procedure must also be stopped. In this case, STEM fails to find a satisfactory solution.

Unlike many other methods, STEM does not assume the existence of an underlying value function. Even if one were available, it could not help in answering the questions. Thus nothing can be said about the convergence of STEM with respect to a value function. It can be characterized as an ad hoc method. However, the developers of the method mention that the algorithm produces a final solution fast if the new constraints constructed during the solution process become ineligible for further relaxations. They state that fewer iterations are needed, if several objective functions are relaxed at the same time.

Applications and Extensions

STEM is applied in [Bare, Mendoza, 1988] to solve forest land management planning problems of Washington area, and in [Wäscher, 1990] to solve problems of cutting stock. In [Vijayalakshmi, 1987], it is applied to Indian sugar industry, and in [Rentz, Hanicke, 1988], it is used in planning parts of energy support systems.

A modification of STEM is presented in [Slowinski, Warczynski, 1984]. The L_2-metric is there utilized in the distance minimization to guarantee the Pareto optimality of the solutions. The simplex method and the ellipsoid method are compared as optimization routines.

A so-called exterior branching algorithm is presented in [Aubin, Naslund, 1972]. It is an extension of STEM into nonlinear problems. There are several differences when compared with the original method. For example, the decision maker does not need to specify amounts of change and an implicit value function is assumed to exist. Some extensions and modifications of STEM are also presented in [Chankong, Haimes, 1983(b)] and [Crama, 1983]. STRANGE, an extension of STEM to treat linear, stochastic multiobjective optimization problems is introduced in [Teghem, Dufrane, Thauvoye, Kunsch, 1986].

Concluding Remarks

The idea of specifying objective functions whose values should be decreased or can be increased seems quite simple and appealing. However, specifying the amounts of change may sometimes be more difficult.

According to the results presented in Section 2.7, the solutions of STEM are not necessarily Pareto optimal but weakly Pareto optimal solutions may be obtained. A positive feature for MOLP problems is that STEM is able to identify Pareto optimal solutions which are not extreme points (unlike, e.g., the Zionts-Wallenius method in Section 2.16).

STEM and the IWT method have a limitation in common. Namely, the decision maker must be somewhat consistent in her or his actions. It is not possible to withdraw the restrictions set on the feasible region.

2.19. Reference Point Method

The reference point method, presented in [Wierzbicki, 1980(a), (b), 1981, 1982], is based on a reference point. The reference point is a feasible or infeasible point in the criterion space being reasonable or desirable to the decision maker. The reference point is used to derive *achievement scalarizing functions* which have minimal solutions at Pareto optimal points (only). So, value functions or weighting vectors are not used as a basis when generating Pareto optimal solutions. Notice that no specific assumptions are set on the problem to be solved.

Introduction

The basic idea behind the reference point method is to reconsider the fact how decision makers make decisions. It is doubted in [Wierzbicki, 1980(a), (b)] that individuals do not make everyday decisions by maximizing some value function. Instead, it is claimed that decision makers want to attain certain aspiration levels (e.g., when making purchases according to a shopping list). Wierzbicki suggests that, while thousands of consumers behave on the average as if they maximized a value function, no individual behaves in that way. The basic idea is satisficing (introduced in Section 1.4) rather than optimizing. In addition, reference points are easy and intuitive for the decision maker to specify and their consistency is not required so essentially.

Before we can present the algorithm itself, we must present new concepts and new ways of approaching the multiobjective optimization problem. This section is based on the papers [Wierzbicki, 1977, 1980(b), 1981, 1982, 1986(a), (b)]. (Somewhat similar results for scalarizing functions are presented in [Jahn, 1984] and [Luc, 1986].) By a reference point method we here mean that of Wierzbicki's.

It is shown, for instance, in [Wierzbicki, 1980(b), 1986(a)] that Pareto optimal solutions can be characterized by achievement scalarizing functions if the functions satisfy certain requirements. An achievement scalarizing function is a function $s_{\bar{\mathbf{z}}}\colon Z \to \mathbf{R}$, where $\bar{\mathbf{z}} \in \mathbf{R}^k$ is an arbitrary reference point of aspiration levels. In the following, we shorten the name to an *achievement function*.

Because we do not know the feasible criterion region Z explicitly, we, in practice, minimize the function $s_{\bar{\mathbf{z}}}(\mathbf{f}(\mathbf{x}))$ subject to $\mathbf{x} \in S$ (see, e.g., Figure 1). Thus, we deal with the feasible region in the decision variable space. For notational convenience, we, however, present the problem in this section as if it were solved in the feasible criterion region.

First, we define some general properties applied to an achievement function $s_{\bar{\mathbf{z}}}$.

Definition 2.19.1. *A function $s_{\bar{\mathbf{z}}}\colon Z \to \mathbf{R}$ is strictly monotone (or strictly order preserving) if for $\mathbf{z}^1$ and $\mathbf{z}^2 \in Z$*

$$z_i^1 < z_i^2 \text{ for all } i = 1, \ldots, k \quad \text{imply} \quad s_{\bar{\mathbf{z}}}(\mathbf{z}^1) < s_{\bar{\mathbf{z}}}(\mathbf{z}^2).$$

Definition 2.19.2. *A function $s_{\bar{\mathbf{z}}} \colon Z \to \mathbf{R}$ is strongly monotone (or strongly order preserving) if for $\mathbf{z}^1$ and $\mathbf{z}^2 \in Z$*

$$z_i^1 \le z_i^2 \text{ for } i = 1, \ldots, k \text{ and } z_j^1 < z_j^2 \text{ for some } j \quad \text{imply} \quad s_{\bar{\mathbf{z}}}(\mathbf{z}^1) < s_{\bar{\mathbf{z}}}(\mathbf{z}^2).$$

We already defined a set $\mathbf{R}_\varepsilon^k = \{\mathbf{z} \in \mathbf{R}^k \mid \operatorname{dist}(\mathbf{z}, \mathbf{R}_+^k) \le \varepsilon \|\mathbf{z}\|\}$ in connection with the ε-proper Pareto optimality (see Definition 1.8.2).

Definition 2.19.3. *A function $s_{\bar{\mathbf{z}}} \colon Z \to \mathbf{R}$ is ε-strongly monotone if for $\mathbf{z}^1$ and $\mathbf{z}^2 \in Z$*

$$\mathbf{z}^1 \in \mathbf{z}^2 - \mathbf{R}_\varepsilon^k \setminus \{\mathbf{0}\} \quad \text{imply} \quad s_{\bar{\mathbf{z}}}(\mathbf{z}^1) < s_{\bar{\mathbf{z}}}(\mathbf{z}^2).$$

Definition 2.19.4. *A continuous function $s_{\bar{\mathbf{z}}} \colon Z \to \mathbf{R}$ is order representing if it is strictly monotone as a function of $\mathbf{z} \in Z$ for any $\bar{\mathbf{z}} \in \mathbf{R}^k$ and if*

$$\{\mathbf{z} \in \mathbf{R}^k \mid s_{\bar{\mathbf{z}}}(\mathbf{z}) < 0\} = \bar{\mathbf{z}} - \mathit{int}\, \mathbf{R}_+^k$$

(for all $\bar{\mathbf{z}} \in \mathbf{R}^k$).

Definition 2.19.5. *A continuous function $s_{\bar{\mathbf{z}}} \colon Z \to \mathbf{R}$ is order approximating if it is strongly monotone as a function of $\mathbf{z} \in Z$ for any $\bar{\mathbf{z}} \in \mathbf{R}^k$ and if*

$$\bar{\mathbf{z}} - \mathbf{R}_{\bar{\varepsilon}}^k \subset \{\mathbf{z} \in \mathbf{R}^k \mid s_{\bar{\mathbf{z}}}(\mathbf{z}) \le 0\} \subset \bar{\mathbf{z}} - \mathbf{R}_\varepsilon^k$$

(for all $\bar{\mathbf{z}} \in \mathbf{R}^k$) with $\varepsilon > \bar{\varepsilon} \ge 0$.

Remark. *For a continuous function $s_{\bar{\mathbf{z}}}$ which is order representing or order approximating, we have*

$$s_{\bar{\mathbf{z}}}(\bar{\mathbf{z}}) = 0.$$

Now we can present the following theorems according to [Wierzbicki, 1986(a), (b)]. The problem to be solved is

$$\text{(2.19.6)} \qquad \begin{array}{ll} \text{minimize} & s_{\bar{\mathbf{z}}}(\mathbf{z}) \\ \text{subject to} & \mathbf{z} \in Z. \end{array}$$

Theorem 2.19.7. *If the achievement function $s_{\bar{\mathbf{z}}} \colon Z \to \mathbf{R}$ is strictly monotone, then every solution of the problem (2.19.6) is weakly Pareto optimal. If the achievement function $s_{\bar{\mathbf{z}}} \colon Z \to \mathbf{R}$ is strongly monotone, then every solution of the problem (2.19.6) is Pareto optimal. Finally, if the achievement function $s_{\bar{\mathbf{z}}} \colon Z \to \mathbf{R}$ is ε-strongly monotone, then every solution of the problem (2.19.6) is ε-properly Pareto optimal.*

Proof. Here, we only prove the second statement because of the similarity of the proofs. We assume that $s_{\bar{\mathbf{z}}}$ is strongly monotone. Let $\mathbf{z}^* \in Z$ be a solution of the problem (2.19.6). Let us suppose that it is not Pareto optimal. In this case, there exists a criterion vector $\mathbf{z} \in Z$ such that $z_i \le z_i^*$ for all $i = 1, \ldots, k$ and $z_j < z_j^*$ for some j. According to the strong monotonicity of $s_{\bar{\mathbf{z}}}$, we know that $s_{\bar{\mathbf{z}}}(\mathbf{z}) < s_{\bar{\mathbf{z}}}(\mathbf{z}^*)$, which contradicts the assumption that $\mathbf{z}^*$ minimizes $s_{\bar{\mathbf{z}}}$. Thus $\mathbf{z}^*$ is Pareto optimal. ▮

The above result is now rewritten to be able to characterize Pareto optimal solutions with the help of order representing and order approximating achievement functions. The proof follows from the proof of Theorem 2.19.7.

Corollary 2.19.8. *If the achievement function $s_{\bar{\mathbf{z}}}: Z \to \mathbf{R}$ is order representing, then, for any $\bar{\mathbf{z}} \in \mathbf{R}^k$, every solution of the problem (2.19.6) is weakly Pareto optimal. If the achievement function $s_{\bar{\mathbf{z}}}: Z \to \mathbf{R}$ is order approximating with some ε and $\bar{\varepsilon}$ as in Definition 2.19.5, then, for any $\bar{\mathbf{z}} \in \mathbf{R}^k$, every solution of the problem (2.19.6) is Pareto optimal. If $s_{\bar{\mathbf{z}}}$ in addition is $\bar{\varepsilon}$-strongly monotone, then every solution of the problem (2.19.6) is $\bar{\varepsilon}$-properly Pareto optimal.*

The preceding theorem states sufficient conditions for a solution of an achievement function to be (weakly, ε-properly, or) Pareto optimal. The following theorem gives the corresponding necessary conditions.

Theorem 2.19.9. *If the achievement function $s_{\bar{\mathbf{z}}}: Z \to \mathbf{R}$ is order representing and $\mathbf{z}^* \in Z$ is weakly Pareto optimal or Pareto optimal, then the minimum of the problem (2.19.6) is attained at $\bar{\mathbf{z}} = \mathbf{z}^*$ and the value of the achievement function is zero. If the achievement function $s_{\bar{\mathbf{z}}}: Z \to \mathbf{R}$ is order approximating and $\mathbf{z}^* \in Z$ is ε-properly Pareto optimal, then the minimum of the problem (2.19.6) is attained at $\bar{\mathbf{z}} = \mathbf{z}^*$ and the value of the achievement function is zero.*

Proof. Here, we only prove the statement for Pareto optimality. The proofs of the other statements are very similar. (The proof of the necessary condition for ε-proper Pareto optimality can be found in [Wierzbicki, 1986(a)].)

Let $\mathbf{z}^* \in Z$ be Pareto optimal. This means that there does not exist any other point $\mathbf{z} \in Z$ such that $z_i \leq z_i^*$ for all $i = 1, \ldots, k$ and $z_j < z_j^*$ for some j. Let us assume that $\mathbf{z}^*$ is not a solution of the problem (2.19.6) when $\bar{\mathbf{z}} = \mathbf{z}^*$. So, there exists some vector $\mathbf{z}^\circ \in Z$ such that $s_{\bar{\mathbf{z}}}(\mathbf{z}^\circ) < s_{\bar{\mathbf{z}}}(\mathbf{z}^*) = s_{\bar{\mathbf{z}}}(\bar{\mathbf{z}}) = 0$ and $\mathbf{z}^\circ \neq \mathbf{z}^*$. Since $s_{\bar{\mathbf{z}}}$ was assumed to be order representing, we have $\mathbf{z}^\circ \in \bar{\mathbf{z}} - \operatorname{int} \mathbf{R}_+^k = \mathbf{z}^* - \operatorname{int} \mathbf{R}_+^k$. This means that $z_i^\circ < z_i^*$ for all i, which contradicts the assumption that $\mathbf{z}^*$ is Pareto optimal. ▮

Remark. *By the help of the results in Theorem 2.19.9 a certain point can be confirmed not to be weakly, ε-properly or Pareto optimal (if the optimal value of the achievement function differs from zero).*

Now we are able to completely characterize the set of weakly Pareto optimal solutions with the help of order representing achievement functions. The sets of ε-properly Pareto optimal solutions and Pareto optimal solutions are characterized almost completely (if the closure of the sets of solutions of the problem (2.19.6) for an order approximating achievement function is taken as $\varepsilon \to 0$). If the solutions of the problem (2.19.6) are supposed to be unique, the above theorems can make the characterization of Pareto optimal solutions complete.

If the reference point is feasible, or more specifically $\bar{\mathbf{z}} \in Z + \mathbf{R}_+^k$, then the minimization of the achievement function $s_{\bar{\mathbf{z}}}(\mathbf{z})$ subject to $\mathbf{z} \in Z$ must produce a solution which maximizes the surplus to the Pareto optimal set. If the reference point is infeasible, that is, $\bar{\mathbf{z}} \notin Z + \mathbf{R}_+^k$, then the minimization of the achievement function $s_{\bar{\mathbf{z}}}(\mathbf{z})$ subject to $\mathbf{z} \in Z$ must produce a solution which minimizes the distance to the Pareto optimal set.

The importance with the achievement functions is that the decision maker can obtain any arbitrary weakly Pareto optimal or Pareto optimal (or at least ε-properly Pareto optimal) solution by moving the reference point only. There are many achievement functions which satisfy the above-presented conditions. An example of order representing functions is

$$s_{\bar{\mathbf{z}}}(\mathbf{z}) = \max_{1 \le i \le k} [w_i(z_i - \bar{z}_i)],$$

where $\mathbf{w}$ is some fixed positive weighting vector. Let us briefly convince ourselves that the above-mentioned function really is order representing. The continuity of the function is obvious. If we have $\mathbf{z}^1$ and $\mathbf{z}^2 \in Z$ such that $z_i^1 < z_i^2$ for all $i = 1, \dots, k$, then $s_{\bar{\mathbf{z}}}(\mathbf{z}^1) = \max_i[w_i(z_i^1 - \bar{z}_i)] < \max_i[w_i(z_i^2 - \bar{z}_i)] = s_{\bar{\mathbf{z}}}(\mathbf{z}^2)$ and thus the function is strictly monotone. If the inequality $s_{\bar{\mathbf{z}}}(\mathbf{z}) = \max_i[w_i(z_i - \bar{z}_i)] < 0$ holds, then we must have $z_i < \bar{z}_i$ for all $i = 1, \dots, k$, that is, $\mathbf{z} \in \bar{\mathbf{z}} - \operatorname{int} \mathbf{R}_+^k$.

An example of order approximating functions is

$$s_{\bar{\mathbf{z}}}(\mathbf{z}) = \max_{1 \le i \le k} [w_i(z_i - \bar{z}_i)] + \rho \sum_{i=1}^{k} w_i(z_i - \bar{z}_i), \tag{2.19.10}$$

where $\mathbf{w}$ is as above and $\rho > 0$ is sufficiently small when compared with ε and large when compared with $\bar{\varepsilon}$. This function is also $\bar{\varepsilon}$-strongly monotone.

An example of so-called penalty scalarizing functions is

$$s_{\bar{\mathbf{z}}}(\mathbf{z}) = -\|\mathbf{z} - \bar{\mathbf{z}}\|^2 + \varrho\|(\mathbf{z} - \bar{\mathbf{z}})_+\|^2, \tag{2.19.11}$$

where $\varrho > 1$ is a scalar penalty coefficient and $(\mathbf{z} - \bar{\mathbf{z}})_+$ is a vector with components $\max[0, z_i - \bar{z}_i]$. This function is strictly monotone, strongly monotone for all norms in $\mathbf{R}^k$ except the L_∞-norm and order approximating with $\varepsilon \ge 1/\varrho$ (see [Wierzbicki, 1980(a), 1982]). More examples of order representing and order approximating functions are presented, for example, in [Wierzbicki, 1980(b), 1986(a), (b)].

Reference Point Algorithm

After presenting the necessary mathematical background, we can introduce the interactive multiobjective optimization technique of Wierzbicki. The method is very simple and practical. Before the solution process starts, some information is given to the decision maker about the problem. If possible, the ideal criterion vector and the (approximated) nadir point are presented to illustrate the ranges of the Pareto optimal set. Another possibility is to minimize and maximize the objective functions individually in the feasible region (if it is bounded). Both decision variable and criterion values are presented. An appropriate form for the achievement function must also be selected.

The basic steps of the reference point method are the following.

(1) Present information about the problem to the decision maker. Set $h = 1$.

(2) Ask the decision maker to specify a reference point $\bar{\mathbf{z}}^h \in \mathbf{R}^k$ (a level for every objective function).

(3) Minimize the achievement function and obtain a (weakly) Pareto optimal solution $\mathbf{z}^h$. Present the solution to the decision maker.

(4) Calculate a number of k other (weakly) Pareto optimal solutions by minimizing the achievement function with perturbed reference points

$$\bar{\mathbf{z}}(i) = \bar{\mathbf{z}}^h + d^h \mathbf{e}^i,$$

where $d^h = \|\bar{\mathbf{z}}^h - \mathbf{z}^h\|$ and $\mathbf{e}^i$ is the $i$th unit vector for $i = 1, \dots, k$.

(5) Present the alternatives to the decision maker. If (s)he finds one of the $k + 1$ solutions satisfactory, (s)he chooses it as a final solution. Otherwise, ask the decision maker to specify a new reference point $\bar{\mathbf{z}}^{h+1}$. Set $h = h + 1$ and go to step (3).

The reason for writing the word weakly in parentheses in the algorithm is that it depends on the achievement function selected whether the solutions are Pareto optimal or weakly Pareto optimal.

The advantage of perturbing the reference point is that the decision maker gets a better conception about possible solutions. If the reference point is far from the Pareto optimal set, the decision maker gets a wider description of the Pareto optimal set and if the reference point is near the Pareto optimal set, then a finer description of the Pareto optimal set is given. Affects of the perturbation and close and distant reference points are illustrated in Figure 16.

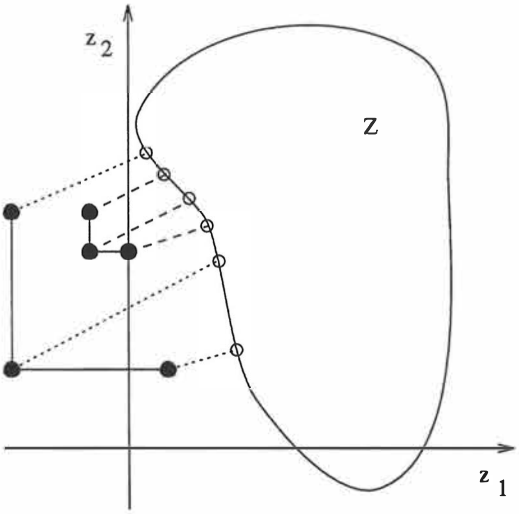


Figure 16. Altering the reference points.

The reference point method can be characterized as a non ad hoc method. Alternatives are easy to compare if the value function is known. On the other hand, a reference point cannot be directly defined with the help of the value function. However, it is possible to test that a new reference point has a higher value function value than the earlier solutions.

As to the infinite convergence of the algorithm the following result is stated in [Wierzbicki, 1980(b)].

Theorem 2.19.12. *If the solutions of the achievement function in the algorithm are unique and if the minimal value of* $\|\mathbf{z} - \bar{\mathbf{z}}\|$ *subject to Pareto optimal criterion vectors is equal to the minimal value of the achievement function* $s_{\bar{\mathbf{z}}}$ *subject to* Z *for* $\bar{\mathbf{z}} \in Z + \mathbf{R}^k_+$, *then for any norm in* $\mathbf{R}^k$, *the solution procedure is convergent. In other words,* $\lim_{h\to\infty} \|\mathbf{z}^h - \mathbf{z}^{h+1}\| = \lim_{h\to\infty} d^h = 0$.

Proof. See references in [Wierzbicki, 1980(b)].

A modification of the algorithm guaranteeing the convergence is presented in [Wierzbicki, 1980(b)].

Applications and Extensions

The reference point approach is applied to econometric models in [Olbrisch, 1986]. In cases when there exists a weighting vector such that the solution of the weighting problem is equal to the solution of the problem (2.19.6) the weighting vector can be obtained as a partial derivative of the achievement function. See [Wierzbicki, 1982] for details.

It is remarked in [Vetschera, 1991(b)] that an achievement function widely used with the reference point context is not stable with respect to changes in the set of objective functions. A subset of reference points, called dominating points, is considered in [Skulimowski, 1989]. A point is called a *dominating point* if it is not dominated by any feasible point and it dominates at least one of the feasible points. Skulimowski classifies the dominating points and studies their properties.

The reference point method is generalized for several decision makers or several reference points in [Song, Cheng, 1988] and [Vetschera, 1991(a)]. A software package based on the reference point ideas of Wierzbicki is presented in Section 3.2. This package is called DIDAS.

Concluding Remarks

The reference point method of Wierzbicki is quite easy for the decision maker to understand. The decision maker only has to specify appropriate aspiration levels and compare criterion vectors. What has been said about the comparison of alternatives in connection with the previous methods is also valid here. The solutions are Pareto optimal or weakly Pareto optimal depending on the achievement function employed.

The freedom of the decision maker has both positive and negative aspects. The decision maker can direct the solution process and is free to change her or his mind during the process. However, the convergence is not necessarily fast if the decision maker is not determined. There is no clear strategy to produce the final solution since the method does not help the decision maker to find improved solutions.

The method of Wierzbicki can be regarded as a generalization of goal programming (in Section 2.11). Aspiration levels are central in both methods, but the method of Wierzbicki is able to handle both feasible and infeasible aspiration levels, the contrary of goal programming.

2.20. Satisficing Trade-Off Method

The satisficing trade-off method, presented in [Nakayama, Sawaragi, 1984], [Nakayama, Furukawa, 1985] and [Nakayama, 1985(a)], is based on ideas similar to the reference point method of Wierzbicki. The method is presented here only briefly (and converted into minimization).

General Outline

Just like the reference point method of Wierzbicki, the satisficing trade-off method originates from aspiration levels forming a reference point and is based on satisficing decision making as can be deduced from its name. However, the achievement function for obtaining Pareto optimal or weakly Pareto optimal solutions is somewhat different.

The theoretical derivation of the method is based on the three requirements for scalarizing functions presented at the beginning of Chapter 2. As was then mentioned, there does not exist a scalarizing function which could satisfy all the three requirements. It was proved in Section 2.7 that every Pareto optimal solution can be found by the weighted L_∞-norm, whereas the same is not always true for weighted L_p-norms ($1 \leq p < \infty$) (depending on the degree of nonconvexity of the problem).

Thus, the weighted L_∞-norm seems to be a more promising scalarizing function than the others. However, its solutions are weakly Pareto optimal.

A Pareto optimal solution can be produced from a weakly Pareto optimal solution, for example, by the lexicographic ordering or by solving the problem (1.9.5). In some cases, it may be expensive to solve additional optimization problems. To avoid the additional optimizations, augmentation terms comparable to those in Sections 2.17 and 2.19 can be added. When augmented distance measures are used, the solutions obtained are guaranteed to be Pareto optimal but the drawback is that not all of them are necessarily found. This depends on how well the augmentation term is managed to be selected. In addition, the third requirement of Chapter 2 concerning satisficing decision making also fails (see [Nakayama, 1985(a)] and [Steuer, 1986]).

In the satisficing trade-off method the achievement function to be minimized is of the form

$$\max_{1 \le i \le k} \left[\frac{1}{\bar{z}_i - z_i^{**}} (f_i(\mathbf{x}) - z_i^{**}) \right],$$

where $\bar{\mathbf{z}}$ is the reference point and $\mathbf{z}^{**}$ is a utopian vector (see Section 2.17) so that $\bar{\mathbf{z}} > \mathbf{z}^{**}$. If some objective function f_j is not bounded in S, then some small scalar value is selected to be z_j^{**}. If the problem is bounded, then solutions obtained by this method are guaranteed to be weakly Pareto optimal (see Theorem 2.7.13) and every Pareto optimal solution can be found (see Theorem 2.7.15). It is proved in [Nakayama, 1985(a)] and [Sawaragi, Nakayama, Tanino, 1985] that the solution obtained is satisfactory (i.e., $f_i(\mathbf{x}^*) \le \bar{z}_i$ for all i) if the reference point is feasible.

Because the optimal Kuhn-Tucker multipliers of the differentiable counterpart problem (2.7.11) (see Section 2.7) multiplied by the weighting coefficients, are used to obtain trade-off information, it must be assumed that

1. All the objective and the constraint functions are continuously differentiable.

If, in addition, trade-off rates are desired to be available, more assumptions have to be fulfilled. The assumptions are parallel to those in Theorem 2.3.17, see [Yano, Sakawa, 1987]. This fact has not earlier been emphasized when introducing the method.

The functioning of the satisficing trade-off method is the following. After a (weakly) Pareto optimal solution (depending whether an augmentation term is included or not) has been obtained by optimizing the achievement function, it is presented to the decision maker. Based on this information (s)he is asked to classify the objective functions into three classes. The classes are the objective functions whose values (s)he wants to improve, the objective functions whose values (s)he agrees to relax (impair) and the objective functions whose values (s)he accepts as they are. New aspiration levels are required for the objective functions in the first two classes. The current criterion values are set as the aspiration levels for the objectives in the third class. If trade-off rates are obtainable from the Kuhn-Tucker multipliers and the weighting coefficients, they can be utilized to help in specifying the new aspiration levels. Trade-off information is used to check the feasibility of the reference point. If it is not feasible then the number of minimizations of the achievement function can be reduced by specifying higher aspiration levels directly (remember that satisficing solutions are obtained when the reference point is feasible). The procedure continues until the decision maker does not want to improve any of the objectives.

Because no specific assumptions are set on the underlying value function, convergence results based on it are not available. Even though a value function existed, it could not be directly used to determine the functions to be decreased and increased and the amounts of change. Thus the method is characterized to be an ad hoc method. On the other hand, one must remember that the aim of the method has particularly been in satisficing rather than optimizing some value function.

Applications and Extensions

Some theoretical specifications about the algorithm are presented in [Nakayama, Furukawa, 1985]. The method is also applied to the aseismic design of a tower-pier system of a long span suspension bridge. An application to a water quality control problem is presented in [Nakayama, Sawaragi, 1984] and to a river basin problem in [Nakayama, 1985(a)]. In [Olson, 1993], the method is employed to solve a sausage blending problem and, in [Nakayama, Nomura, Sawada, Nakajima, 1986], to solve a blending problem of industrial plastic materials. The satisficing trade-off method is adapted for linear fractional objective functions in [Nakayama, 1991(a)] with an application concerning material blending in cement production.

Sensitivity analysis of the algorithm for linear problems is investigated in [Nakayama, 1989]. The burden of the decision maker is reduced in the extension described in [Nakayama, 1991(b), 1992]. Then the decision maker does not have to specify aspiration levels for the objective functions whose values are to be impaired but they are calculated automatically by a specified formula. Also exact trade-off information by parametric analysis is introduced for linear and quadratic problems.

Concluding Remarks

The satisficing trade-off method contains identical elements with STEM and the reference point method of Wierzbicki. Therefore, comments given there are not repeated here. The role of the decision maker is easy to understand and the solutions obtained are Pareto optimal or weakly Pareto optimal depending on the scalarizing function selected.

Notice that, in practice, classifying the objective functions into three classes and specifying the amounts of increment and decrement for their values is exactly the same as specifying a new reference point. A new reference point is implicitly formed. Either the new aspiration levels are larger, smaller, or the same as in the current solution. Thus the same outcome can be obtained with different reasonings.

Because the method is based on satisficing decision making, the decision maker can freely search for a satisficing solution and change her or his mind, if necessary. No convergence based on value functions has even been intended.

2.21. Visual Interactive Approach

A visual interactive approach has been introduced in [Korhonen, Laakso, 1984, 1985, 1986(a)]. It contains ideas from the methods of Geoffrion, Dyer and Feinberg, Zionts and Wallenius, and Wierzbicki. The algorithm works best for MOLP problems if the optimality of the final solution is wanted to check. Otherwise, it can be applied also to nonlinear problems. The algorithm was originally designed for problems to be maximized but here it is presented in the form of minimization.

Introduction and Visual Interactive Algorithm

An interesting feature of the visual interactive approach is that no explicit knowledge is assumed on the properties of the value function during the solution process.

However, sufficient conditions for optimality can be established for the termination point of the algorithm, if the decision maker's underlying value function is supposed to be pseudoconcave (and differentiable) at that point (and several other assumptions to be listed later are fulfilled).

In the method of Wierzbicki in Section 2.19, a reference point consisting of aspiration levels for each objective function was projected onto the Pareto optimal set by an achievement function. This idea is extended here. A so-called *reference direction*, which is a vector from the current iteration point to the reference point, is projected onto the Pareto optimal set. The decision maker can examine this Pareto optimal curve with the means of computer graphics.

The algorithm is as follows. Once again, in the notations we handle criterion vectors for simplicity. Naturally, the actual calculations are performed in the decision variable space.

(1) Find an arbitrary starting criterion point $\mathbf{z}^1 \in \mathbf{R}^k$. Set $h = 1$.

(2) Ask the decision maker to specify a reference point $\bar{\mathbf{z}} \in \mathbf{R}^k$ and set $\mathbf{d}^{h+1} = \bar{\mathbf{z}} - \mathbf{z}^h$ as a new reference direction.

(3) Find the set Z^{h+1} of (weakly) Pareto optimal solutions $\mathbf{z}$ that solve the problem

$$\begin{aligned} \text{minimize} \quad & s_{\bar{\mathbf{z}},\mathbf{w}}(\mathbf{z}) \\ \text{subject to} \quad & \bar{\mathbf{z}} = \mathbf{z}^h + t\mathbf{d}^{h+1} \\ & \mathbf{z} \in Z \text{ is Pareto optimal,} \end{aligned}$$

where $s_{\bar{\mathbf{z}},\mathbf{w}}$ is an achievement function, $\mathbf{w}$ is a weighting vector and t increases from zero to infinity.

(4) Ask the decision maker to select the most preferred solution $\mathbf{z}^{h+1}$ in Z^{h+1}.

(5) If $\mathbf{z}^h \neq \mathbf{z}^{h+1}$, set $h = h + 1$ and go to step (2). Otherwise, check the optimality conditions. If the conditions are satisfied, stop with $\mathbf{z}^{h+1}$ as a final solution. Else, set $h = h + 1$ and set $\mathbf{d}^{h+1}$ to be a new search direction identified by the optimality checking procedure. Go to step (3).

The setting of the algorithm makes it possible for the starting point to be any point in the criterion space. It does not have to be feasible, much less Pareto optimal, since it is projected onto the (weakly) Pareto optimal set in step (3).

The straight line from the current iteration point (or its Pareto optimal projection at the first iteration) to the boundary of the Pareto optimal set is projected into a Pareto optimal curve traversing across the Pareto optimal set. The developers of the method suggest that the values of the objective functions along the curve are plotted on the computer screen as value paths with different colours illustrating each of the objectives. The decision maker can move the cursor back and forth and see the corresponding numerical values at each point.

The achievement function $s_{\bar{\mathbf{z}},\mathbf{w}}(\mathbf{z})$ is of the same form as presented in Section 2.19, namely

$$s_{\bar{\mathbf{z}},\mathbf{w}}(\mathbf{z}) = \max_{i \in I} \frac{z_i - \bar{z}_i}{w_i},$$

where $I = \{i \mid w_i > 0\} \subset \{1, \ldots, k\}$, $\mathbf{w}$ is a weighting vector, $\mathbf{z} \in Z$ is a criterion vector and $\bar{\mathbf{z}} \in \mathbf{R}^k$ is an arbitrary reference point. Notice that if weakly Pareto optimal solutions are desired to be avoided, then an augmentation term must be added to the achievement function as, for example, in Sections 2.17 and 2.19 (see also [Steuer, 1986]). However, originally it was not considered to be necessary because the purpose is just to produce different solutions effectively.

The minimization problem of $s_{\bar{\mathbf{z}},\mathbf{w}}$ is nondifferentiable but it can be transformed into an equivalent, differentiable (and solvable) form

$$\text{(2.21.1)} \qquad \begin{array}{ll} \text{minimize} & \alpha \\ \text{subject to} & f_i(\mathbf{x}) - \alpha w_i \leq z_i^h + t d_i^{h+1} \text{ for all } i \in I \\ & \mathbf{x} \in S, \end{array}$$

with $\mathbf{x} \in S$ as a variable. As a weighting vector one can use, for example, the reference point specified by the decision maker.

As the parameter t increases from zero towards infinity, a curve of weakly Pareto optimal solutions emanating from the point $\mathbf{z}^h$ towards the boundary of the Pareto optimal set is formed. If the original problem is linear, then the problem can be solved by using any code for parametric linear programming. The weighted Tchebycheff distance measure has been chosen to facilitate the parametric linear programming (even though the solutions are only guaranteed to be weakly Pareto optimal). If some of the constraint or the objective functions are nonlinear, an approximate solution can be obtained by solving the problem (2.21.1) with several different values for t.

Checking the optimality conditions in step (5) is the most complicated part of the algorithm. So far, no specific assumptions have been set on the value function. It may change during the solution process or it may not even exist at all. However, we can check whether a given criterion vector $\mathbf{z}^{h+1}$ (and the corresponding decision variable $\mathbf{x}^{h+1}$) is optimal to the value function, assuming that

1. The decision maker's underlying value function is pseudoconcave at $\mathbf{x}^{h+1}$.
2. The feasible region S is convex, compact and bounded.
3. The constraint functions are continuous and differentiable.

Let the feasible directions at $\mathbf{z}^{h+1}$ be denoted by $\mathbf{d}(j)$, $j = 1, \ldots, p$. We define a cone C containing all those feasible directions by

$$\text{(2.21.2)} \qquad C = \left\{ \mathbf{z} \in \mathbf{R}^k \mid \mathbf{z} = \mathbf{z}^{h+1} + \sum_{j=1}^{p} \beta_j \mathbf{d}(j), \ \ \beta_j \geq 0 \right\}.$$

If we have $\mathbf{z}^h = \mathbf{z}^{h+1}$ in the fifth step, we know that the projection of $\mathbf{d}^h$ on the weakly Pareto optimal set is not a direction of improvement. Then we can apply the following sufficient conditions for optimality.

Theorem 2.21.3. *Let the assumptions 1–3 be satisfied. Let $\mathbf{z}^{h+1} \in Z$ and let C be a cone containing all the feasible directions at $\mathbf{z}^{h+1}$ (as in (2.21.2)). Let us assume that*

$$U(\mathbf{z}^{h+1}) \geq U(\mathbf{z}^{h+1} + \beta_j \mathbf{d}(j)) \quad \textit{for all } \beta_j \geq 0 \textit{ and } j = 1, \ldots, p.$$

Then $\mathbf{z}^{h+1}$ is a globally optimal solution (with respect to U).

Proof. See [Korhonen, Laakso, 1986(a)].

For MOLP problems we know that if the current solution is not optimal, then one of the feasible directions of the cone C must be a direction of improvement. This direction is then used as a new reference direction in step (3). In other words, to be able to apply Theorem 2.21.3 at a certain point, the decision maker must first check every feasible direction at that point for improvement. This increases both the computational burden and the burden on the decision maker. It is demonstrated in [Korhonen, Laakso, 1986(a), (b)] and [Halme, Korhonen, 1989] how the amount of search directions can be reduced.

Notice that the termination condition of Theorem 2.21.3 is analogous to the Karush-Kuhn-Tucker optimality conditions. This is proved in [Halme, Korhonen, 1989]. If the value function is known, it is easy to compare alternative criterion vectors. Thus the visual interactive approach can be characterized as a non ad hoc method. However, what was said about the difficulty in determining new reference points in connection with the reference point method in Section 2.19 is also valid here.

The graphical illustration of the alternatives has been an important aspect in the development of multiobjective optimization methods which try to improve and facilitate the cooperation between the decision maker and the analyst (computer). Further developments of the graphical illustration of this method are presented in Section 3.1.

A similar interactive line search algorithm for MOLP problems is presented in [Benson, Aksoy, 1991]. The procedure generates only Pareto optimal points and is able to automatically correct possible errors in the decision maker's judgement.

Concluding Remarks

The role of the decision maker reminds the reference point method. (S)he has to both specify reference points and select the most preferred alternatives. In the reference point methods, there are, however, fewer choices to select from. By the visual interactive approach, the decision maker can explore a wider part of the weakly Pareto optimal set than by the reference point method and the satisficing trade-off method, even by providing similar reference point information. This possibility brings the the task of comparing the alternatives and selecting the most preferred of them.

The visual interactive approach works best for MOLP problems, as it has basically been designed for them. It is interesting that the method requires no additional assumptions on the problem and the underlying value function until the optimality of the final solution is to be examined. The optimality can be guaranteed under certain assumptions and with some effort.

The performance of the method depends greatly on how well the decision maker manages to specify the reference directions that lead to improved solutions. Korhonen and Laakso mention that particularly when the number of objective functions is large, the specification of reference points may be quite laborious for the decision maker. In this case, they suggest that random directions in conjunction with decision maker-defined reference directions should be used. See [Korhonen, Laakso, 1986(a)] for a discussion about other ways of specifying the reference directions.

The computation time for large problems can be reduced by presenting one piece at a time of the weakly Pareto optimal curve to the decision maker. If (s)he finds the end point to be the most satisfactory one, then the next piece is to be presented. If the number of the objective functions is large, the power of graphical illustration suffers. For this reason, it is advisable not to have more than ten objective functions at a time.

The consistency of the decision maker's answers is not important and it is not checked in the algorithm. Thus the algorithm may cycle. This can also be seen as a positive feature since the decision maker is able to return to such parts that (s)he already has examined, if (s)he changes her or his mind.

Adaptation into Goal Programming

The ideas of the visual interactive approach have been adapted to the goal programming environment in [Korhonen, Laakso, 1986(b)]. The problem to be solved is now assumed to be in the generalized goal programming form (see Section 2.11). Again, the original formulation for maximization problems is here transformed for minimization. Just as the original visual interactive approach, the adaptation works best for MOLP problems. However, more general problems can be solved if it is assumed that

1. All the objective functions (flexible goals) are convex and differentiable.
2. The feasible region restricted by the inflexible (rigid) goals is a convex set.

The optimality of the final solution can be checked also in this adaptation in an identical way, as described above.

The index set of the objective functions is denoted by G and the index set of the constraint functions is denoted by R. The feasible region restricted by the constraint functions (inflexible goals) is denoted by S_R to remind that the region alters when the roles of the goals are changed. As in Section 2.11, the aspiration levels are denoted by $\bar{z}_i$ and the deviational variables are denoted by δ_i. It can be said that if the ith goal is a flexible goal, then δ_i is unbounded, and, if the ith goal is an inflexible goal, then δ_i is bounded. The aspiration levels for flexible goals are not absolute. The decision maker wants to improve the values of the objective functions as much as possible. Notice that combining achievement functions into goal programming eliminates the problems caused by feasible aspiration levels (see Section 2.11).

The advantage of formulating the problem as a generalized goal programming problem is that the decision maker can easily convert flexible goals into inflexible ones and vice versa in an interactive fashion. This increases the freedom of the decision maker. Now we need a new concept, called unique Pareto optimality.

Definition 2.21.4. *A decision vector* $\mathbf{x}^* \in S_R$ *is uniquely Pareto optimal if it is Pareto optimal and for any* $\mathbf{x} \in S_R$ *the equalities* $f_i(\mathbf{x}) = f_i(\mathbf{x}^*)$ *for all* $i \in G$ *imply* $\mathbf{x} = \mathbf{x}^*$.

It is important to know that uniquely Pareto optimal solutions remain uniquely Pareto optimal when the sets of flexible and inflexible goals are altered. This result is proved in [Korhonen, Laakso, 1986(b)]

Again, optimality of a current solution is checked by assuming that the decision maker's underlying value function is pseudoconcave at that point in S_R and that (s)he always takes a move in the direction (s)he likes (i.e., the value function is not ill-conditioned). It is proved in [Korhonen, Laakso, 1986(b)] that assuming pseudoconcavity in S_R is not stronger an assumption than assuming that in Z.

For clarity, we here state the visual interactive approach adapted for generalized goal programming. (The reference direction is now $\bar{z}_i - f_i(\mathbf{x}^h)$).

(1) Ask the decision maker to specify the aspiration levels $\bar{z}_i$ and the initial partition into flexible and inflexible goals. Set $h = 1$. Find an initial Pareto optimal solution $\mathbf{x}^1$, for example, by solving the problem

$$\begin{aligned} \text{minimize} \quad & s(\boldsymbol{\delta}, \mathbf{w}) \\ \text{subject to} \quad & f_i(\mathbf{x}) + \delta_i = \bar{z}_i, \;\; i \in G \\ & f_j(\mathbf{x}) + \delta_j = \bar{z}_j, \;\; \delta_j \geq 0, \;\; j \in R. \end{aligned}$$

(2) If the decision maker is satisfied with $\mathbf{f}(\mathbf{x}^h)$ and is willing to check its optimality, go to step (5), otherwise, stop. If the decision maker wants to continue, (s)he can redefine the division of goals and give new aspiration levels for the flexible goals.

(3) Solve the problem

$$\begin{aligned} \text{minimize} \quad & s(\boldsymbol{\delta}, \mathbf{w}) \\ \text{subject to} \quad & f_i(\mathbf{x}) + \delta_i = f_i(\mathbf{x}^h) + td_i, \ \ i \in G \\ & \mathbf{x} \in S_R, \end{aligned}$$

for all $t \geq 0$, where $d_i = \bar{z}_i - f_i(\mathbf{x}^h)$.

(4) Present the Pareto optimal curve to the decision maker for evaluation and ask the most satisfactory solution $\mathbf{f}(\mathbf{x}^{h+1})$. If the decision maker finds the old solution better than the new alternatives, set $h = h + 1$ and go to step (5). Otherwise, set $h = h + 1$ and go to step (2).

(5) Construct a cone with vertex at $\mathbf{x}^h$ and which contains the whole feasible set. Present the directions spanning the cone to the decision maker as in step (4). If none of them is a direction of improvement, then stop with $\mathbf{x}^h$ as a final solution. Else, if the decision maker wants to continue, go to step (2). Otherwise, stop.

We can utilize a similar achievement function as earlier: $s(\boldsymbol{\delta}, \mathbf{w}) = \max_{i \in G}[\delta_i / w_i]$. In this case, the problem to be solved in step (3) is

$$\begin{aligned} \text{minimize} \quad & \alpha \\ \text{subject to} \quad & f_i(\mathbf{x}) - \alpha w_i \leq f_i(\mathbf{x}^h) + td_i \text{ for all } i \in G \\ & \mathbf{x} \in S_R. \end{aligned}$$

If (some) objective functions (or flexible goals) are nonlinear, they are supposed to be linearized at $\mathbf{x}^h$. In this case, all the solutions obtained during the solution process are not necessarily weakly Pareto optimal. However, the optimality checking procedure identifies improved solutions.

The following step in developing the visual interactive approach has been to diversify the graphical illustration into a form called *Pareto race*. It is described in Section 3.1.

So far, we have described quite a lot of different methods for multiobjective optimization. The question of differentiability has not been emphasized. Let us handle this aspect in the following. All the noninteractive methods, presented in Sections from 2.2 to 2.11, can also be employed with nondifferentiable objective functions whenever the single objective optimization routines utilized can handle nondifferentiable functions. The case is totally different with interactive methods. Most of the interactive methods described this far assume that the functions involved are differentiable (or even linear), with the exception of the IWT procedure (Section 2.17) and the reference point method (Section 2.19).

Few methods which have especially been designed to handle nondifferentiable problems have been proposed. In the sequel, we suggest such new methods. There the objective functions are assumed to be locally Lipschitzian.

Another aim in developing new methods has been in trying to overcome some of the weaknesses detected in the older methods. All the previously described methods have had an effect in the development. Either they have offered good ideas to adopt or unsatisfactory properties to avoid.

Trade-off rate information cannot be exploited in nondifferentiable problems as in the ISWT method and SPOT. The natural reason is that obtaining trade-off information from the Kuhn-Tucker multipliers of the ε-constraint problem necessitates twice continuous differentiability of the functions. How to obtain trade-off information in nondifferentiable cases needs and deserves more research.

While the ideas of the ISWT method and SPOT cannot be extended into nondifferentiable cases, it can be tried with the GDF method. In Section 2.22, we introduce a modification based on the subgradient method.

On the other hand, the ideas of reference points and satisficing decision making seem to be generalizable into nondifferentiable problems. We can adopt the ideas of classification of the objective functions and reference points of STEM, the reference point method, and the satisficing trade-off method and the idea of reference direction of the visual interactive method and mix them with some ideas of nondifferentiable analysis. The outcome is described in Section 2.23.

2.22. Subgradient GDF Method

The subgradient GDF method, presented in [Miettinen, Mäkelä, 1991, 1993, 1994], is a modification of the Geoffrion-Dyer-Feinberg (GDF) method, described in Section 2.14. Two serious drawbacks of the original method have been overcome in the subgradient GDF method. Firstly, the assumption 2 in Section 2.14 of the continuous differentiability of the objective functions is relaxed. Secondly, the solutions obtained are guaranteed to be Pareto optimal. The subgradient GDF method has been developed by the author in collaboration with M. Mäkelä at the Department of Mathematics of the University of Jyväskylä.

For completeness, we list all the assumptions.

1. The underlying value function $U\colon \mathbf{R}^k \to \mathbf{R}$ exists, is implicitly known to the decision maker and is continuously differentiable.
2. U is componentwise decreasing and strictly decreasing at least with respect to the reference function so that $\frac{dU(\mathbf{f}(\mathbf{x}))}{df_\ell} < 0$.
3. All the objective functions f_i are locally Lipschitzian.
4. The feasible region S is convex and compact.

Introduction

As mentioned earlier, the basic idea of the original GDF method is to maximize the decision maker's value function by the gradient method of Frank and Wolfe. The Frank–Wolfe (FW) method was selected instead of the steepest ascent (gradient) method for its faster convergence properties. It is known that the steepest ascent method, where the search direction is simply stated as the gradient $\nabla_{\mathbf{x}} U(\mathbf{f}(\mathbf{x}))$, is rather sensitive to the zig-zag phenomenon (since the consecutive search directions are orthogonal). However, the dependence on the geometry of the feasible region S is a disadvantage of the FW method (see [Miettinen, Mäkelä, 1991]).

Generalizing the FW method directly into the nondifferentiable case is in theory simple. When maximizing the linear approximation of U, the gradients of the functions f_i in the chain rule are replaced by their subdifferentials. However, in practice, this plan of action presumes that the whole subdifferential is known at each point, which is possible for few special problems. Usually, even the calculation of one arbitrary subgradient may be difficult. In order to maintain the generality of the method, we have to be content with one subgradient at each point (which is a standard assumption in nondifferentiable optimization).

Experiments of the generalization by replacing the gradient in the FW algorithm by one arbitrary subgradient have been reported in [Miettinen, Mäkelä, 1991]. The results proved that this was not a productive way. The search direction became so haphazard that the decision maker's desires had almost no affect at all in the solution process.

After these experiences, we return to the origin of nondifferentiable optimization instead of trying to generalize the FW method as such. In the sequel, we replace the FW method by a pioneering method in the nondifferentiable optimization, the (Kiev) subgradient method (see [Shor, 1985]).

The simple structure of the basic subgradient method has made it widely used, although it suffers from some serious drawbacks. Namely, in order to guarantee the convergence properties the step-sizes have to be chosen a priori. Also the lack of an implementable stopping criterion is a disadvantage of the subgradient method. However, in the multiobjective case, these difficulties can be overcome since the choice of the step-sizes is up to the decision maker. The decision maker is also the one who normally terminates the solution process (even if a feasible search direction exists).

The idea of the subgradient method is to choose the search direction to be the direction of an arbitrary subgradient $\xi \in \partial_{\mathbf{x}} U(\mathbf{f}(\mathbf{x}^h))$. In order to obtain as versatile selection of candidate solutions as possible we "stretch" the search direction $\mathbf{d}^h$ as far as possible (which actually is the case also in the FW method). This can be done as follows.

Let us first choose the search direction as in the subgradient method, that is, $\mathbf{d}_0^h = \xi \in \partial_{\mathbf{x}} U(\mathbf{f}(\mathbf{x}^h))$. Then we search for a scalar $\beta_0^h \geq 0$ which solves the problem

$$\text{(2.22.1)} \qquad \begin{aligned} &\text{maximize} \quad \beta \\ &\text{subject to} \quad \mathbf{x}^h + \beta \mathbf{d}_0^h \in S. \end{aligned}$$

If $\beta_0^h > 0$, then we obtain the actual search direction by $\mathbf{d}^h = \beta_0^h \mathbf{d}_0^h$. On the other hand, if $\beta_0^h = 0$, then $\mathbf{d}_0^h$ points out from the feasible region S, and we have to project the infeasible direction onto the feasible region. In this case, we derive $\mathbf{d}^h$ as a solution of the projection problem

$$\text{(2.22.2)} \qquad \begin{aligned} &\text{minimize} \quad \|\mathbf{d}_0^h - \mathbf{d}\| \\ &\text{subject to} \quad \mathbf{x}^h + \mathbf{d} \in S. \end{aligned}$$

It is obvious that, after the projection, the new iteration point $\mathbf{x}^h + \mathbf{d}^h$ lies on the boundary of S, and thus the search direction $\mathbf{d}^h$ extends as far as possible. It is also easy to show that, if we still have $\mathbf{d}^h = \mathbf{0}$ after the projection, then $\mathbf{d}^T \xi \leq 0$ for all $\mathbf{d} \in \mathbf{R}^n$ such that $\mathbf{x}^h + \mathbf{d} \in S$, that is, a necessary optimality condition for U to attain its maximum is fulfilled (see [Miettinen, Mäkelä, 1991]).

Notice that our aim is to keep the search direction as close to the subgradient direction as possible, first stretching and, not until secondly, projecting, while in the FW method both of them are done at the same time.

After obtaining the search direction the step-size is determined as in the GDF method. Discrete criterion vectors $\mathbf{f}(\mathbf{x}^h + t\mathbf{d}^h)$ $(0 \leq t \leq 1)$ calculated with different values of t are presented to the decision maker, who then selects the most preferred one among the alternatives.

Calculation of the Subgradient

In this subsection, we handle the evaluation of the subgradient $\xi \in \partial_{\mathbf{x}} U(\mathbf{f}(\mathbf{x}^h))$. Through the generalized Jacobian chain rule (see [Clarke, 1983]), the subdifferential at the point $\mathbf{x}^h$ is of the form

$$\partial_{\mathbf{x}} U(\mathbf{f}(\mathbf{x}^h)) = \nabla_{\mathbf{f}} U(\mathbf{f}(\mathbf{x}^h))^T \partial_{\mathbf{x}} \mathbf{f}(\mathbf{x}^h),$$

where $\partial_{\mathbf{x}} \mathbf{f}(\mathbf{x}^h) = \operatorname{conv} \{J \in \mathbf{R}^{k \times n} \mid J = \lim_{l \to \infty} \nabla \mathbf{f}(\hat{\mathbf{x}}^l);\ \hat{\mathbf{x}}^l \to \mathbf{x}^h,\ \hat{\mathbf{x}}^l \in \mathbf{R}^n \setminus \Omega_{\mathbf{f}}\}$ is the generalized Jacobian of $\mathbf{f}$ at $\mathbf{x}^h$ and $\nabla \mathbf{f}(\hat{\mathbf{x}}^l)$ denotes the usual $k \times n$ Jacobian matrix. A collection of points where some of the objective functions is not differentiable is denoted by $\Omega_{\mathbf{f}}$. Then a single subgradient $\xi \in \partial_{\mathbf{x}} U(\mathbf{f}(\mathbf{x}^h))$ may be calculated by

$$\xi = \sum_{i=1}^{k} \left(\frac{d\,U(\mathbf{f}(\mathbf{x}^h))}{d\,f_i} \right) \xi_i,$$

where ξ_i is the ith row of a representative from $\partial_{\mathbf{x}} \mathbf{f}(\mathbf{x}^h)$. From here we proceed as in the original GDF method. The assumption 2 implies that the partial derivative $\frac{d\,U(\mathbf{f}(\mathbf{x}^h))}{d\,f_\ell}$ is negative. We can divide ξ by the scalar $-\frac{d\,U(\mathbf{f}(\mathbf{x}^h))}{d\,f_\ell}$. The direction of the subgradient ξ is now in the form

$$\sum_{i=1}^{k} -m_i \xi_i,$$

where $m_i = \frac{d\,U(\mathbf{f}(\mathbf{x}^h))}{d\,f_i} / \frac{d\,U(\mathbf{f}(\mathbf{x}^h))}{d\,f_\ell}$ is the marginal rate of substitution involving f_ℓ and f_i at the point $\mathbf{x}^h$ for all $i = 1, \ldots, k$, $i \neq \ell$. As in the original GDF method, the marginal rates of substitution have to be asked from the decision maker either right off or using different auxiliary procedures (as in [Dyer, 1973(a)]).

Producing Pareto Optimal Solutions

One serious weakness of the original GDF method is that the generated solutions are not necessarily Pareto optimal. To improve this failing we adopt in the subgradient GDF method a modification where the solution candidates are "projected" onto the set of Pareto optimal solutions before they are presented to the decision maker. The purpose is to generate Pareto optimal solutions that are close to the solutions obtained in the algorithm. This can be carried out in many ways. Because the points to be projected are not necessarily even weakly Pareto optimal, for instance, lexicographic ordering is too robust. One practical way is to utilize achievement functions, presented in Section 2.19.

Let $\mathbf{z}^h$ be a solution obtained by the subgradient GDF method. We set it as a reference point ($\bar{\mathbf{z}} = \mathbf{z}^h$) and minimize some achievement function. If the selected achievement function is order approximating, the solution produced is guaranteed to be Pareto optimal (and weakly Pareto optimal for order representing functions) (see Section 2.19 and also [Wierzbicki, 1980(a), 1982]).

Achievement functions utilized most up to now are the augmented weighted Tchebycheff function

$$(2.22.3) \qquad \begin{aligned} &\text{minimize} \quad s_{\bar{\mathbf{z}}}(\mathbf{f}(\mathbf{x})) = \max_{1 \le i \le k} [w_i(f_i(\mathbf{x}) - \bar{z}_i)] + \rho \sum_{i=1}^{k} w_i(f_i(\mathbf{x}) - \bar{z}_i) \\ &\text{subject to} \quad \mathbf{x} \in S, \end{aligned}$$

where w_i:s are weighting coefficients (for example, $w_i = \frac{1}{z_i^{\text{nad}} - z_i^{\star}}$ if the problem is bounded) and $\rho > 0$ is a relatively small augmentation term, and the penalty scalarizing function

$$\text{(2.22.4)} \qquad \begin{array}{ll} \text{minimize} & s_{\bar{\mathbf{z}}}(\mathbf{f}(\mathbf{x})) = -\|\mathbf{f}(\mathbf{x}) - \bar{\mathbf{z}}\| + \varrho\|(\mathbf{f}(\mathbf{x}) - \bar{\mathbf{z}})_+\| \\ \text{subject to} & \mathbf{x} \in S, \end{array}$$

where $(\mathbf{f}(\mathbf{x}) - \bar{\mathbf{z}})_+$ denotes the positive part of $\mathbf{f}(\mathbf{x}) - \bar{\mathbf{z}}$ and $\varrho > 1$ is a scalar penalty coefficient.

Unlike the function (2.19.11) presented in Section 2.19, we employ in (2.22.4) an exact penalty scalarizing function with the L_2-norm. If the function (2.19.11) is used, it is well known that penalty coefficients of large order of magnitude are required. This may be computationally expensive and cause serious numerical difficulties as has been noticed in [Miettinen, Mäkelä, 1991]. When we use the exact penalty scalarizing function, there exists a finite limit for the penalty coefficient ϱ such that, when penalties greater than the limit are used, Pareto optimal solutions are obtained.

Notice that the possible nondifferentiability of the achievement function should be no problem because nondifferentiable optimization routines are used anyhow. Note, however, that the objective functions of (2.22.3) and (2.22.4) are compositions of the nondifferentiable map $\mathbf{f}$ and a nondifferentiable outer function (the norm). This may cause difficulties in the computation of the subgradient. Anyway, if the objective functions are subdifferentially regular (in the sense of Clarke, see [Clarke, 1983]), then the generalized chain rule is valid (with equality instead of inclusion).

In general, one can state that the problem (2.22.3) is computationally easier to solve than the problem (2.22.4). It must be noticed that different achievement functions produce different Pareto optimal solutions. The reason for this is that they move into different directions from the non-Pareto optimal reference point.

One more possibility to be employed in the projection phase is a black-box routine MPB. It will be introduced more closely in Section 2.23. Pareto optimality cannot be guaranteed for its solutions but weakly Pareto optimal solutions are usually obtained.

No matter which projection method is employed, it increases the computational burden. Anyway, it is the price to be paid for the certainty that the decision maker does not have to handle non-Pareto optimal solutions which mostly are highly unsatisfactory (at least while a componentwise decreasing value function is assumed).

Subgradient GDF Algorithm

Now we can present the algorithm of the subgradient GDF method.

(1) Ask the decision maker to specify a reference function f_ℓ. Choose a starting point $\mathbf{x}^1 \in S$. Set $h = 1$.

(2) Ask the decision maker to specify the marginal rates of substitution m_i^h between f_ℓ and the other objective functions f_i ($i = 1, \ldots, k$, $i \neq \ell$) at the point $\mathbf{x}^h$.

(3) Set $\mathbf{d}_0^h = \xi \in \partial_{\mathbf{x}} U(\mathbf{f}(\mathbf{x}^h))$ and get $\beta_0^h \geq 0$, which solves the problem (2.22.1). If $\beta_0^h > 0$, then set $\mathbf{d}^h = \beta_0^h \mathbf{d}_0^h$. Otherwise, that is, if $\beta_0^h = 0$, then set $\mathbf{d}^h$ to be the solution of the projection problem (2.22.2). If $\mathbf{d}^h = \mathbf{0}$, then go to step (7).

(4) Calculate P different criterion vectors $\mathbf{f}(\mathbf{x}^h + t_j \mathbf{d}^h)$, where $t_j = \left(\frac{j-1}{P-1}\right)$, $j = 1, \ldots, P$.

(5) Produce Pareto optimal solutions from the criterion vectors by minimizing the achievement function with $\bar{\mathbf{z}} = \mathbf{f}(\mathbf{x}^h + t_j \mathbf{d}^h)$ for $j = 1, \ldots, P$.

(6) Present P alternative criterion vectors to the decision maker who chooses the most preferred one among them. Denote the corresponding decision variable by $\mathbf{x}^{h+1}$. Set $h = h + 1$. If the decision maker wants to continue, then go to step (2).
(7) Stop. The final solution is $\mathbf{x}^h$.

As in the original GDF method, the convergence of the subgradient GDF method is rather robust in consequence of the approximation of the marginal rates of substitutions and the step-sizes. We have to notice that the convergence rate of the subgradient method is rather poor (see [Shor, 1985]) when compared with the Frank-Wolfe method in the differentiable case. However, as in nondifferentiable optimization in general, this kind of compromise has to be done when the assumption of differentiability is weakened.

On the other hand, the improvement of our method is that it guarantees the convergence to a Pareto optimal solution. In fact, all the alternatives presented to the decision maker are Pareto optimal. Thus the decision maker is not bothered with unsatisfactory solutions. In addition, since the underlying value function was assumed to be componentwise decreasing, it is assured that the solutions cannot get worse when they are projected into the Pareto optimal set in step (5). The subgradient GDF method is non ad hoc in nature as its origin.

Numerical experiments with the subgradient GDF method are presented in Chapters 6 and 7.

Concluding Remarks

Merits of the subgradient GDF method are the Pareto optimality of all the solutions and the ability to handle nondifferentiable objective functions. Notice that the underlying value function has to be still assumed to be continuously differentiable. However, guaranteeing the Pareto optimality increases the computational costs. Also the calculation of the element from the generalized Jacobian may be difficult.

A common drawback of the original and the subgradient GDF method is that it may be difficult for the decision maker to specify the marginal rates of substitution. This remains a remarkable fact because it is important that the direction of the gradient of the value function is approximated closely enough. The role and the selection of the reference function (denoted here by f_ℓ) is as important as in the original GDF method.

Note that the problems (2.22.2), (2.22.3) and (2.22.4) are nondifferentiable. Since the objective functions themselves are nondifferentiable, this property does not cause additional difficulties. The nondifferentiable optimization problems can be solved by effective bundle methods (see, e.g., [Mäkelä, Neittaanmäki, 1992]).

2.23. NIMBUS Method

NIMBUS is an interactive multiobjective optimization method designed especially to be able to handle nondifferentiable functions. The interaction phase has been aimed at being comparatively simple and easy to understand for the decision maker. NIMBUS contains a subroutine, called MPB, which is needed to produce (weakly) Pareto optimal solutions. The role of MPB is of a black-box nature. Therefore, we do not present it here in detail. NIMBUS has been developed by the author in collaboration with M. Mäkelä at the Department of Mathematics of the University of Jyväskylä.

The assumptions in the NIMBUS method are that

1. All the objective and the constraint functions are locally Lipschitzian.
2. The feasible region S is convex.
3. Less is preferred to more to the decision maker.

The assumption 3 could be formulated to require that the underlying value function is componentwise decreasing. The reason for avoiding this wording is that an underlying value function is not assumed to exist. The assumption only concerns the form of the general preference structure of the decision maker.

Introduction

The starting point in developing the NIMBUS method has been somewhat the opposite to the starting point in the subgradient GDF method. In the subgradient GDF method, the purpose was to develop a theoretically sound and logical method. This led into difficulties on the decision maker's side and more or less instable results, not to mention the computational costs. In the NIMBUS method, the basis has been to overcome difficulties encountered with the other interactive methods. The most important aspects appeared to be the effectiveness and the comfortableness of the decision maker.

In the NIMBUS method, the idea is that the decision maker examines the values of the objective functions calculated at a point $\mathbf{x}^h$ and divides the objective functions into up to five classes. Those classes are functions f_i whose values should be decreased ($i \in I^{<}$), functions f_i whose values should be decreased down till some aspiration level ($i \in I^{\leq}$), functions f_i whose values are satisfactory at the moment ($i \in I^{=}$), functions f_i whose values are allowed to increase up till some upper bound ($i \in I^{>}$), and functions f_i whose values are allowed to change freely ($i \in I^{\diamond}$), where $I^{<} \cup I^{\leq} \cup I^{=} \cup I^{>} \cup I^{\diamond} = \{1, \ldots, k\}$.

The decision maker is asked to specify the aspiration levels $\bar{z}_i$ for $i \in I^{\leq}$ and the upper bounds ε_i for $i \in I^{>}$. The difference between the classes $I^{<}$ and $I^{\leq}$ is that the functions in $I^{<}$ are to be minimized as far as possible but the functions in $I^{\leq}$ only till the aspiration level. Thus the functions in the latter class are called *aspiration functions*. Also weighting coefficients can be given to the functions in these two classes.

Notice that we have more classes than in STEM and the satisficing trade-off method. In this way, the classification part of the objective functions can be performed as if the ε-constraint method, the weighting method, lexicographic ordering or goal programming was used to produce (weakly) Pareto optimal solutions.

According to the classification and connected information we form a new problem

$$
\begin{aligned}
& \text{minimize} && \left\{ f_i(\mathbf{x})/w_i \ (i \in I^{<}),\ \max_{j \in I^{\leq}} \left[\max \left[f_j(\mathbf{x})/w_j - \bar{z}_j,\ 0 \right] \right] \right\} \\
(2.23.1) \quad & \text{subject to} && f_i(\mathbf{x}) \leq f_i(\mathbf{x}^h), \quad i \in I^{=} \\
& && f_i(\mathbf{x}) \leq \varepsilon_i, \qquad\quad i \in I^{>} \\
& && \mathbf{x} \in S,
\end{aligned}
$$

where the weighting coefficients w_i, $i \in I^{<} \cup I^{\leq}$, are positive and sum up to one, the aspiration levels $\bar{z}_i$ must satisfy the condition $\bar{z}_i < f_i(\mathbf{x}^h)$ for $i \in I^{\leq}$ and the upper bounds ε_i must satisfy $\varepsilon_i > f_i(\mathbf{x}^h)$ for $i \in I^{>}$.

Notice that (2.23.1) is still a (nondifferentiable) multiobjective optimization problem. It is solved with the help of the MPB routine. The solution is denoted by $\hat{\mathbf{x}}^h$. If the decision maker does not like $\mathbf{f}(\hat{\mathbf{x}}^h)$ for some reason, no dead end is at hand. (S)he can explore solutions between $\mathbf{x}^h$ and $\hat{\mathbf{x}}^h$. In this case, we form a search direction $\mathbf{d}^h = \hat{\mathbf{x}}^h - \mathbf{x}^h$.

The step-size is determined by the decision maker just as in the subgradient GDF method. Criterion vectors $\mathbf{f}(\mathbf{x}^h + t\mathbf{d}^h)$ are calculated with different values of $0 \leq t \leq 1$. Then (weakly) Pareto optimal solutions are produced from them (through MPB) and presented to the decision maker for selection.

Unlike with some other methods based on classification, the success of the solution process does not depend entirely on how well the decision maker manages in specifying the classification and the appropriate parameter values. It is important that the classification is not irreversible (as is the case, e.g., in STEM). Thus, no unrecallable damage is caused in NIMBUS if the solution $\mathbf{f}(\hat{\mathbf{x}}^h)$ is not what was expected. The decision maker is free to go back or explore intermediate points. (S)he can easily get to know the problem and its possibilities by specifying, for example, loose upper bounds and examining intermediate solutions. NIMBUS is indeed learning-oriented.

NIMBUS Algorithm

In the following, we describe the NIMBUS algorithm in detail. Because NIMBUS contains so many alternative ways to proceed in the solution process, the detailed algorithm becomes inevitably long and somewhat obscure. However, to the decision maker NIMBUS is much easier to use than one could think when looking at the algorithm.

(1) Choose a starting point $\mathbf{x}^0 \in S$ and calculate its weakly Pareto optimal counterpart $\mathbf{x}^1 \in S$, employing MPB with $I^< = \{1, \ldots, k\}$. Set $h = 1$.

(2) Ask the decision maker to divide the objective functions into the classes $I^<$, $I^\leq$, $I^=$, $I^>$, and $I^\diamond$ at the point $\mathbf{z}^h = \mathbf{f}(\mathbf{x}^h)$ such that $I^> \cup I^\diamond \neq \emptyset$ and $I^< \cup I^\leq \neq \emptyset$. If either of the unions is empty, goto step (9). Ask the aspiration levels $\bar{z}_i^h$ for $i \in I^\leq$ and the upper bounds ε_i^h for $i \in I^>$ from the decision maker. Ask also the possible positive weighting coefficients w_i^h for $i \in I^< \cup I^\leq$ summing up to one.

(3) Calculate $\hat{\mathbf{x}}^h$ by solving the problem (2.23.1) by the MPB routine. If $\hat{\mathbf{x}}^h = \mathbf{x}^h$, ask the decision maker whether (s)he wants to try another classification. If yes, set $\mathbf{x}^{h+1} = \mathbf{x}^h$, $h = h + 1$ and go to step (2). If no, go to step (9).

(4) Let us denote $\hat{\mathbf{z}}^h = \mathbf{f}(\hat{\mathbf{x}}^h)$. Present $\mathbf{z}^h$ and $\hat{\mathbf{z}}^h$ to the decision maker. If the decision maker wants to see different alternatives, set $\mathbf{d}^h = \hat{\mathbf{x}}^h - \mathbf{x}^h$ and go to step (6). If the decision maker prefers $\mathbf{z}^h$, set $\mathbf{x}^{h+1} = \mathbf{x}^h$, $h = h + 1$, and to go step (2).

(5) Now the decision maker wants to continue from $\hat{\mathbf{z}}^h$. If $I^< \neq \emptyset$ set $\mathbf{x}^{h+1} = \hat{\mathbf{x}}^h$, $h = h + 1$ and go to step (2). Otherwise ($I^< = \emptyset$), the weak Pareto optimality must be checked, setting $I^< = \{1, \ldots, k\}$ and employing MPB. Let the solution be $\check{\mathbf{x}}^h$ and set $\mathbf{x}^{h+1} = \check{\mathbf{x}}^h$. Set $h = h + 1$ and go to step (2).

(6) Calculate P different criterion vectors $\mathbf{f}(\mathbf{x}^h + t_j \mathbf{d}^h)$, where $t_j = \left(\frac{j-1}{P-1}\right)$, $j = 1, \ldots, P$.

(7) Produce weakly Pareto optimal solutions from the criterion vectors, employing MPB (with $I^< = \{1, \ldots, k\}$).

(8) Present P alternative criterion vectors to the decision maker and let her or him

choose the most preferred one among them. Denote the corresponding decision variable by $\mathbf{x}^{h+1}$. Set $h = h + 1$. If the decision maker wants to continue, go to step (2).

(9) Check the Pareto optimality of $\mathbf{x}^h$ by solving the problem (1.9.5) presented in Theorem 1.9.4 with $\mathbf{x}^h$ as $\mathbf{x}^*$. Let the solution be $(\tilde{\mathbf{x}},\tilde{\boldsymbol{\varepsilon}})$. Stop. The final solution is $\tilde{\mathbf{x}}$ with the corresponding $\tilde{\mathbf{z}}$.

Notice that the decision maker must be ready to give up something in order to attain improvement for some other objective functions. The search procedure stops in step (2) also if the decision maker does not want to improve any criterion value.

We must remember that the best we can guarantee is a local optimum for the problem (2.23.1). Because of the structure of the MPB method, its solutions are quite sensitive with respect to the starting point. If the solution obtained is not completely satisfactory, one can always solve the problem again with another starting point. Changing the starting point is also advised if the decision maker has to stop the search process with $\hat{\mathbf{x}}^h = \mathbf{x}^h$ in step (3). It is also possible to improve the algorithm in step (3) to avoid the case $\hat{\mathbf{x}}^h = \mathbf{x}^h$. If the upper bounds specified by the decision maker are too tight, one can use them as a reference point and project them onto the Pareto optimal set. Showing the new solution to the decision maker provides her or him information about the possibilities and the limitations of the problem, and some dead ends can be avoided, too.

The justification of step (5) is given in the form of Theorem 2.23.6. In the last step, the Pareto optimality of the final solution is guaranteed by solving an additional problem introduced in Theorem 1.9.4. For clarity of notations, it has not been mentioned in the algorithm that the decision maker may check Pareto optimality at any time during the solution process. Then the problem (1.9.5) is solved with the current solution as $\mathbf{x}^*$.

MPB Routine

Here we briefly sketch the MPB routine which is employed in the NIMBUS algorithm to produce (weakly) Pareto optimal solutions. For details see [Mäkelä, 1993]. The most advanced version of the bundle family for nondifferentiable single objective optimization, namely, the proximal bundle method is presented in [Kiwiel, 1990]. It has been generalized to handle nonconvex and constrained problems in [Mäkelä, Neittaanmäki, 1992]. The MPB routine is an extension into a multiobjective case. The strategy of handling several objective functions in MPB is based on the ideas presented in [Kiwiel, 1984, 1985(a), (b)] and [Wang, 1989]. The basic idea is to move into a direction where the values of all the current objective functions improve.

The problem to be solved is of the form

$$\begin{aligned} &\text{minimize} && \{f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_{k'}(\mathbf{x})\} \\ &\text{subject to} && (g_1(\mathbf{x}), g_2(\mathbf{x}), \ldots, g_{m'}(\mathbf{x}))^T \leq \mathbf{0}, \end{aligned} \tag{2.23.2}$$

where $k' \leq k$ and $m' \geq m$. If the problem has linear constraints or upper and lower bounds for the decision variables, in addition to nonlinear constraints, they are treated separately, each type in the most effective way. The possible linear constraints and bounds are taken into account in the minimization problems in the following.

The MPB routine is not directly based on employing any scalarizing function. Some kind of scalarization is, however, needed in deriving the minimization method for all the objective functions. Theoretically, we utilize an *improvement function* $H\colon \mathbf{R}^n \times \mathbf{R}^n \to \mathbf{R}$ defined by

$$H(\mathbf{x},\mathbf{y}) = \max\{f_i(\mathbf{x}) - f_i(\mathbf{y}),\ g_l(\mathbf{x}) \mid i = 1,\dots,k',\ l = 1,\dots,m'\}.$$

As a part of the proof of Theorem 1.11.9 (and Corollary 1.11.16) in Section 1.11 it has been shown that a necessary condition for a point $\mathbf{x}^*$ to be a weakly Pareto optimal solution of the problem (2.23.2) is that $\mathbf{x}^*$ minimizes the function $H(\cdot\,,\mathbf{x}^*)$.

In the following, the iteration counter h refers to the inner iterations of the MPB routine, not the iterations of NIMBUS. In the MPB routine, the solution is looked for iteratively, until some stopping criterion is fulfilled. Let $\mathbf{x}^h$ be the current approximation to the solution of (2.23.2) at the iteration h. Then we seek for the search direction $\mathbf{d}^h$ as a solution of the optimization problem

$$\text{(2.23.3)} \qquad \begin{aligned} &\text{minimize} && H(\mathbf{x}^h + \mathbf{d}, \mathbf{x}^h) \\ &\text{subject to} && \mathbf{d} \in \mathbf{R}^n. \end{aligned}$$

Since the problem (2.23.3) is still nondifferentiable, we must approximate it somehow.

Let us assume for a moment that the problem is convex. We suppose that, at the iteration h besides the iteration point $\mathbf{x}^h$, we have some auxiliary points $\mathbf{y}^j \in \mathbf{R}^n$ from the past iterations and subgradients $\xi^j_{f_i} \in \partial f_i(\mathbf{y}^j)$ for $j \in J^h$, $i = 1,\dots,k'$, and $\xi^j_{g_l} \in \partial g_l(\mathbf{y}^j)$ for $j \in J^h$, $l = 1,\dots,m'$, where J^h is a nonempty subset of $\{1,\dots,h\}$. We linearize the objective and the constraint functions at the point $\mathbf{y}^j$ and denote

$$\begin{aligned} \bar{f}_{i,j}(\mathbf{x}) &= f_i(\mathbf{y}^j) + (\xi^j_{f_i})^T(\mathbf{x} - \mathbf{y}^j) \quad \text{for all} \quad i = 1,\dots,k',\ j \in J^h \quad \text{and} \\ \bar{g}_{l,j}(\mathbf{x}) &= g_l(\mathbf{y}^j) + (\xi^j_{g_l})^T(\mathbf{x} - \mathbf{y}^j) \quad \text{for all} \quad l = 1,\dots,m',\ j \in J^h. \end{aligned}$$

Now we can define a convex piecewise linear approximation to the improvement function by

$$\hat{H}^h(\mathbf{x}) = \max\{\bar{f}_{i,j}(\mathbf{x}) - f_i(\mathbf{x}^h),\ \bar{g}_{l,j}(\mathbf{x}) \mid i = 1,\dots,k',\ l = 1,\dots,m',\ j \in J^h\}$$

and we get an approximation to (2.23.3) by

$$\text{(2.23.4)} \qquad \begin{aligned} &\text{minimize} && \hat{H}^h(\mathbf{x}^h + \mathbf{d}) + \tfrac{1}{2}u^h\|\mathbf{d}\|^2 \\ &\text{subject to} && \mathbf{d} \in \mathbf{R}^n, \end{aligned}$$

where $u^h > 0$ is some weighting coefficient. The penalty term $\frac{1}{2}u^h\|\mathbf{d}\|^2$ is added to guarantee that there exists a solution to the problem (2.23.4) and to keep the approximation local enough.

Notice that (2.23.4) is still a nondifferentiable problem, but due to the min-max-nature it is equivalent to the following (differentiable) quadratic problem with $\mathbf{d}$ and v as variables

$$\text{(2.23.5)} \qquad \begin{aligned} &\text{minimize} && v + \tfrac{1}{2}u^h\|\mathbf{d}\|^2 \\ &\text{subject to} && v \ge -\alpha^h_{f_i,j} + (\xi^j_{f_i})^T\mathbf{d},\ i = 1,\dots,k',\ j \in J^h \\ & && v \ge -\alpha^h_{g_l,j} + (\xi^j_{g_l})^T\mathbf{d},\ l = 1,\dots,m',\ j \in J^h, \end{aligned}$$

where

$$\alpha^h_{f_i,j} = f_i(\mathbf{x}^h) - \bar{f}_{i,j}(\mathbf{x}^h),\ i = 1,\dots,k',\ j \in J^h \quad \text{and}$$
$$\alpha^h_{g_l,j} = -\bar{g}_{l,j}(\mathbf{x}^h),\ l = 1,\dots,m',\ j \in J^h$$

are so-called linearization errors.

In the nonconvex case, we replace the linearization errors by so-called subgradient locality measures

$$\beta^h_{f_i,j} = \max\ \left[|\alpha^h_{f_i,j}|, \gamma_{f_i}\|\mathbf{x}^h - \mathbf{y}^j\|^2\right]$$
$$\beta^h_{g_l,j} = \max\ \left[|\alpha^h_{g_l,j}|, \gamma_{g_l}\|\mathbf{x}^h - \mathbf{y}^j\|^2\right],$$

where $\gamma_{f_i} \geq 0$ for $i = 1,\dots,k'$ and $\gamma_{g_l} \geq 0$ for $l = 1,\dots,m'$ are so-called distance measure parameters ($\gamma_{f_i} = 0$ if f_i is convex and $\gamma_{g_l} = 0$ if g_l is convex).

Let $(\mathbf{d}^h, v^h)$ be the solution of the problem (2.23.5). Next, we employ a line search algorithm described in [Mäkelä, Neittaanmäki, 1992], which detects discontinuities in the gradients of the objective functions. Roughly speaking, we try to find a step-size $0 < t^h \leq 1$ such that $H(\mathbf{x}^h + t^h\mathbf{d}^h, \mathbf{x}^h)$ is minimal when $\mathbf{x}^h + t^h\mathbf{d}^h \in S$. The iteration is terminated when $-\frac{1}{2}v^h < \varepsilon$, where $\varepsilon > 0$ is an accuracy parameter supplied by the user. The subgradient aggregation strategy due to [Kiwiel, 1985(c)] is used to restrict the storage requirements, and a modification of the weight updating algorithm described in [Kiwiel, 1990] is used to update the weight u^h.

On the Optimality of the Solutions

First, we state a theoretical result concerning the optimality of $\hat{\mathbf{x}}^h$ in step (3) of the NIMBUS algorithm.

Theorem 2.23.6. *If the set $I^<$ is nonempty in the NIMBUS algorithm, then the solution $\hat{\mathbf{x}}^h$ in step (3) is a weakly Pareto optimal solution of the original multiobjective optimization problem with k objective functions.*

Proof. Let $\hat{\mathbf{x}}^h$ be a solution of the problem (2.23.1) with some sets $I^<$, $I^\leq$, $I^=$, $I^>$ and $I^\diamond$, where $I^< \neq \emptyset$. Let us assume that $\hat{\mathbf{x}}^h$ is not weakly Pareto optimal. Then there exists a decision vector $\mathbf{x} \in S$ such that $f_i(\mathbf{x}) < f_i(\hat{\mathbf{x}}^h)$ for all $i = 1,\dots,k$.

Because $\hat{\mathbf{x}}^h$ is a feasible solution of the problem (2.23.1), we have $f_i(\mathbf{x}) < f_i(\hat{\mathbf{x}}^h) \leq f_i(\mathbf{x}^h)$ for $i \in I^=$ and $f_i(\mathbf{x}) < f_i(\hat{\mathbf{x}}^h) \leq \varepsilon_i$ for $i \in I^>$. Thus also $\mathbf{x}$ is a feasible solution of the problem (2.23.1).

For all $i \in I^\leq$ is valid $f_i(\mathbf{x})/w_i - \bar{z}_i < f_i(\hat{\mathbf{x}}^h)/w_i - \bar{z}_i$ with $w_i > 0$. It implies that $\max\ \left[f_i(\mathbf{x})/w_i - \bar{z}_i,\ 0\right] \leq \max\ \left[f_i(\hat{\mathbf{x}}^h)/w_i - \bar{z}_i,\ 0\right]$ for $i \in I^\leq$, and further

$$\max_{i \in I^\leq}\left[\max\ \left[f_i(\mathbf{x})/w_i - \bar{z}_i,\ 0\right]\right] \leq \max_{i \in I^\leq}\left[\max\ \left[f_i(\hat{\mathbf{x}}^h)/w_i - \bar{z}_i,\ 0\right]\right].$$

While, in addition,

$$f_i(\mathbf{x})/w_i < f_i(\hat{\mathbf{x}}^h)/w_i,$$

$w_i > 0$, for all $i \in I^<$, the point $\hat{\mathbf{x}}^h$ cannot be an optimal solution of the problem (2.23.1). This contradiction implies that $\hat{\mathbf{x}}^h$ must be weakly Pareto optimal. The proof is also valid if some of the classes $I^\leq$, $I^=$, $I^>$ or $I^\diamond$ is empty. ▮

Optimality of the solutions produced by the MPB routine is an important fact to consider. Here we only present some results without proofs, since proving would necessitate explicit expression of the MPB algorithm.

Theorem 2.23.7. *Let the multiobjective optimization problem be convex and Slater's constraint qualification (Definition 1.11.20) be satisfied. If the MPB routine stops with a finite number of iterations, then the solution is weakly Pareto optimal. On the other hand, any accumulation point of the infinite sequence of solutions generated by the MPB routine is weakly Pareto optimal.*

Proof. See [Kiwiel, 1985(a)] or [Wang, 1989].

If the convexity assumption is not satisfied, we obtain somewhat weaker results about substationary points. Let us first define a substationary point.

Definition 2.23.8. *A decision vector $\mathbf{x}^* \in S$ is called a substationary point if it satisfies the (necessary) optimality condition presented in Theorem 1.11.9.*

Now we can state the following general theorem.

Theorem 2.23.9. *If the MPB routine stops with a finite number of iterations, then the solution is a substationary point. On the other hand, any accumulation point of the infinite sequence of solutions generated by the MPB routine is a substationary point.*

Proof. See [Wang, 1989] and references therein.

Notice that only the substationarity of the solutions of the MPB routine is guaranteed for general multiobjective optimization problems. However, we have so far referred and shall continue referring to the solutions as (weakly) Pareto optimal. This practice has been adopted for the fluency of the presentation.

The NIMBUS method has not been developed to converge in the traditional sense. The aim has been to formulate a method where the decision maker can easily explore the (weakly) Pareto optimal set. When the solution process stops so that the decision maker does not want to change any objective function value, the solution is then optimal.

An important factor is that the final solution is always Pareto optimal because of the structure of the algorithm. In addition, all the intermediate points are at least substationary points and they can be projected into Pareto optimal points, if so desired.

Numerical experiments with the NIMBUS method are presented in Chapters 6 and 7.

User Interface

Because the classification phase in the NIMBUS algorithm may seem complicated at first sight, we examine it more closely in this subsection. A user interface can have a remarkable role in this important phase. We propose a snapshot of an implementation, which is under development. The interface has been created to help the decision maker in the classification. It has been depicted in Figure 17.

It is assumed that, as the problem has been inputted, a symbol has been attached to each of the objective functions. Only these symbols are dealt with in this phase. The function descriptions can be seen in the Edit menu, if necessary. The function symbols can be seen in the topmost row. Below is the criterion vector to be considered in the row titled Criterion values.

The decision maker can proceed in different ways. With a mouse (s)he can select a symbol of an objective function in the topmost row, drag it and drop it into some of the five boxes underneath, one box for each function class. No matter where the symbol is dropped inside a box, it will automatically be placed into a correct position.

Another way to specify the classification is to use the symbols of the function classes ($<, \leq, =, >, \diamond$) and enter an appropriate symbol to each objective function in the third topmost row titled Classes. The class symbols available can be seen in connection with the explanatory texts in the boxes underneath or in the Help menu.

If the decision maker likes to indicate preferences in the form of a reference point, it is also possible. In this case, only aspiration functions, functions to be fixed and functions to be relaxed are used. Then the components of the reference point are specified into the field called New levels in the fourth topmost row.

No matter which of these three ways is used in the classification, the information specificd is updated automatically to all the other appropriate fields. For example, if the class of some objective function is specified to be = in the row called Classes, then the function also appears in the box titled Functions fixed with the corresponding value. The value appears also in the row titled New levels. Naturally, if some function is to be minimized or is set free, then the corresponding field in the row called New levels is empty.

The weighting coefficients, the aspiration levels and the upper bounds can be specified with the help of graphical devices. In proportion as functions are classified into aspiration functions and functions to be minimized, their weighting coefficients are updated to sum up to one. If the decision maker does not want to employ weighting coefficients, they remain equal. On the other hand, (s)he can alter the coefficients by moving the bars or by giving numerical values. When some coefficient is altered, the other coefficients are updated so that they all sum up to one (and the others are equal). A coefficient can be fixed by locking. Then it remains unaltered when the other coefficients are handled. The weighting coefficients are also illustrated by vertical bars on the right.

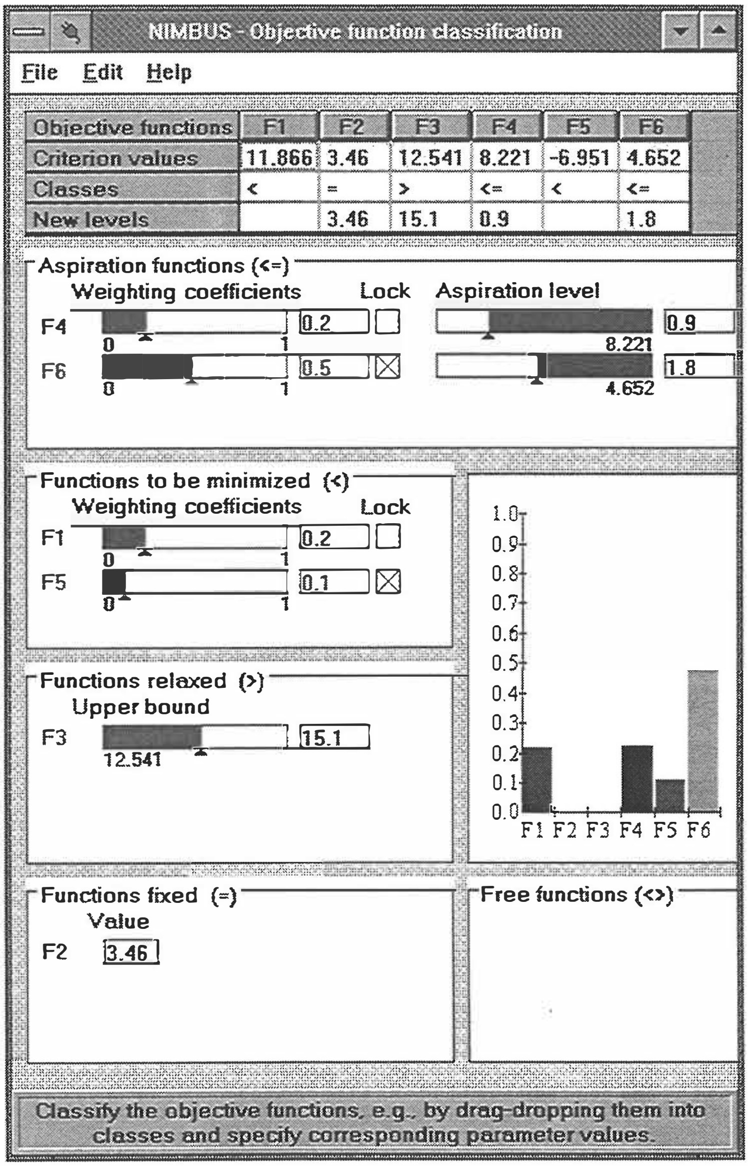
Specifying aspiration levels and upper bounds can also be realized by moving bars or by specifying numerical values in the appropriate boxes. Notice that the aspiration lcvcls must be smaller than the current criterion value. Thus the current value is the upper bound in the bar. The other end of the bar can be unrestricted or equal to the corresponding ideal criterion vector value (if it has been calculated).

The case with upper bounds is exactly the opposite. Here the lowest value in the bar is the current criterion value and the largest value may be unrestricted, equal to the corresponding (approximated) nadir point component, or equal to some other estimate.

The idea in the description above is that the decision maker can select the order of specifying the information as (s)he wishes. Either the functions are first classified and then the corresponding information is given, or the information is specified in proportion as the functions are classified. The program keeps checking the validity of all the inputs and gives error messages and advice, if necessary. Additional help can in every situation be found from the Help menu.

Such a user interface aims at flexibility. The decision maker is not forced to adjust oneself into one rigid manner but has different possibilities to provide the same information. Therefore, one can select a way that suits one's personal characteristics best.

In the example depicted in Figure 17, six objective functions have been classified and the corresponding parameter information has been provided.



Objective functions	F1	F2	F3	F4	F5	F6
Criterion values	11.866	3.46	12.541	8.221	-6.951	4.652
Classes	<	=	>	<=	<	<=
New levels		3.46	15.1	0.9		1.8

Aspiration functions (<=)

	Weighting coefficients	Lock	Aspiration level
F4	0.2	□	8.221 0.9
F6	0.5	☒	4.652 1.8

Functions to be minimized (<)

	Weighting coefficients	Lock
F1	0.2	□
F5	0.1	☒

Functions relaxed (>)

	Upper bound
F3	12.541 15.1

Functions fixed (=)

	Value
F2	3.46

Free functions (<>)

Classify the objective functions, e.g., by drag-dropping them into classes and specify corresponding parameter values.

Figure 17. Classification of the objective functions.

Concluding Remarks

A positive feature of the NIMBUS method compared with the subgradient GDF method (and many other interactive methods) is that the questions posed to the decision maker are not so demanding. The method aims at being flexible and the decision maker can select to what extent (s)he exploits the versatile possibilities of the method. The calculations are not too massive, either because the MPB method is computationally efficient. The reason for this is that, at the moment, bundle methods are the most efficient methods in nondifferentiable optimization (see [Mäkelä, 1993]). As to the MPB routine, NIMBUS stands out from the majority of existing interactive multiobjective optimization methods because it is not based on an explicit scalarizing function.

The decision maker is free to move around the (weakly) Pareto optimal set and also to change her or his mind if necessary. Previous acts do not limit the movements. The decision maker can also extract undesirable solutions from further consideration keeping some upper bounds fixed. Naturally, the decision maker does not have to employ all of the five classes if (s)he feels uncomfortable with some of them. However, it is important to provide different ways of action to the decision maker. The classification of the functions and the specification of the appropriate parameter information does not necessarily have to succeed as well as in the other methods based on the classification (such as STEM). The reason is that also intermediate solutions can be examined and, thus, more information of the problem can be obtained. On the other hand, the decision maker can also cancel some classification step because nothing is irreversible. Such a flexibility is not possible, for example, in STEM and the satisficing trade-off method.

Eventually, it is up to the user interface to take the most of the possibilities of the method and provide it to the user. In connection with this, we have given a proposal for realizing the function classification phase.

Even though the MPB routine cannot guarantee the Pareto optimality of the solutions, at least the final solution is Pareto optimal. If the user finds it important that the intermediate solutions are Pareto optimal, they can be projected. It adds computational costs, though.

The method is ad hoc in nature, since the existence of a value function does not directly advise the decision maker how to act to attain her or his desires. The intention has especially been to release the decision maker from the assumption of an underlying value function. What is important is that the method satisfies two widely agreed desirable properties of interactive methods: not to place too demanding assumptions on the decision maker or the information exchanged, and to be able to find (weakly) Pareto optimal solutions quickly and efficiently.

Unfortunately, it has not been possible to empirically test NIMBUS with real decision makers, so far. However, the main emphasis in this presentation is in the mathematical side of multiobjective optimization, not in the behavioural aspects.

2.24. Other Interactive Methods

There is a great number of interactive methods. It is not the purpose nor practical or possible to discuss all of them in this context. Nevertheless, in addition to those presented in the previous sections, some methods are listed in the following. Only basic concepts and ideas of the methods are mentioned and references are indicated. The methods have been roughly divided according to their basis on goal programming, on weighted L_p-metrics and reference points, and on miscellaneous ideas.

Methods Based on Goal Programming

A rather straightforward extension of goal programming into an interactive form has been presented in [Masud, Hwang, 1981]. The method is called interactive sequential goal programming (ISGP). Also an interactive multiple goal programming

(IMGP) method, described in [Nijkamp, Spronk, 1980] and [Spronk, 1990], has been created to combine the flexibility of goal programming and the robustness and learning characters of interactive approaches. At each iteration, the algorithm provides a lot of information about the problem to the decision maker, who can revise the aspiration levels of the goals.

A sequential multiobjective problem solving (SEMOPS) technique is briefly outlined in [Monarchi, Kisiel, Duckstein, 1973]. Five types of goal specifications in the form of points and intervals are allowed. A different measure of deviation is utilized for each type. For example, if the goal is of the form $f_i(\mathbf{x}) \leq \bar{z}_i$, then the corresponding measure of deviation is $\delta_i = f_i(\mathbf{x})/\bar{z}_i$. At each iteration, a subset of deviations is summed up and then minimized. The decision maker may change that subset and specify new aspiration levels. The solutions are not guaranteed to be Pareto optimal. A related method, called sequential information generator for multiple objective problems (SIGMOP), is introduced in [Monarchi, Weber, Duckstein, 1976]. SIGMOP is a flexible method where the decision maker can alter aspiration levels and weighting coefficients as (s)he separates attainable solutions from the desired ones. As an application, a pollution problem in water resources is solved by the SEMOPS and the SIGMOP methods in the references mentioned.

The ideas of goal programming, the ε-constraint problem and trade-offs are combined in a direction-searching method proposed in [Masud, Zheng, 1989]. The method aims at reducing the cognitive burden on the decision maker while not increasing the computational complexity. The algorithm is illustrated by a numerical example. The properties of the method are also compared with those of several other interactive methods.

A method combining features from goal programming and the method of global criterion is suggested in [Hallefjord, Jörnsten, 1986]. After the decision maker has specified the reference point, the distance between it and the feasible criterion region is minimized by an entropy function. The mathematical background of the method is widely provided in the reference.

Methods Based on Weighted L_p-Metrics and Reference Points

The idea of the method in [Moldavskiy, 1981] is to form a grid in the space of weighting vectors and to map this grid onto the Pareto optimal set. Weighted L_p-metrics are used as scalarizing functions. The decision maker can contract the space of the weighting vectors until the most satisfactory solution is obtained.

Diaz presents in [Diaz, 1987] a method based on sensitivity analysis and weighted Tchebycheff metrics. The distance between a reference point and the Pareto optimal set is minimized by the weighted Tchebycheff metric and the effects of changing some aspiration level are studied by sensitivity analysis. The method in [Sunaga, Mazeed, Kondo, 1988] utilizes the weighted Tchebycheff metric and transforms the constrained (min-max) problem into a series of (differentiable) unconstrained problems by penalty functions.

In [Loganathan, Sherali, 1987], an interactive cutting-plane algorithm with applications is presented. The idea is to maximize the underlying value function. The weighted Tchebycheff metric is utilized with marginal rates of substitution as weighting coefficients. The convergence of the algorithm is also treated.

The method proposed in [M'silti, Tolla, 1993] combines features from the ε-constraint method and the augmented weighted Tchebycheff metric. The global Pareto optimality of the solution obtained is checked.

The method of the displaced ideal for MOLP problems, described in [Zeleny, 1973, 1974, 1976], can be characterized as an interactive extension of the method of global criterion. A subset of the Pareto optimal set is obtained by minimizing the distance between the ideal criterion vector and the feasible criterion region by the weighted L_p-metrics with altered exponents p. The subset is reduced by moving the reference point towards the feasible criterion region until the subset of Pareto optimal solution is small enough for the decision maker to select the most preferred solution. The method is based on empirical studies of the decision maker's behaviour.

The method of Wierzbicki is a basis of the interactive reference point methods introduced in [Bogetoft, Hallefjord, Kok, 1988]. The multiobjective optimization problem is supposed to be convex. Kuhn-Tucker multiplier information is presented to the decision maker to guide the specification of new reference points. Several different modifications are presented in the paper and their convergence properties are studied.

In the method presented in [Tapia, Murtagh, 1989], the decision maker is asked to express preferential desires to attain her or his reference point. So-called preference criteria are formed from this information. These preference criteria are then used as a reference point in the achievement function. The authors also report some encouraging numerical experiments.

The method presented in [Weistroffer, 1982] assumes that the decision maker specifies required values or maximum-achievement levels. Then the surplus is maximized to the Pareto optimal set. The methods in [Weistroffer, 1984, 1987] and [Narula, Weistroffer, 1989(b)] expect that the decision maker provides required and desired values for every objective function. Then an achievement function is optimized. The required and the desired values are modified until a most preferred solution is obtained. Some convergence results are handled.

The method in [Skulimowski, 1992] is also based on several reference points. The feasible region is limited according to the reference point information and some value function values are developed.

Ways of approaching discrete multiobjective optimization problems have been included in the method introduced in [Kok, Lootsma, 1985]. The ideal criterion vector is used as a reference point. Pairwise-comparison methods are applied between the reference point and the (possibly approximated) nadir point. The distances are measured by solving the augmented weighted Tchebycheff problem.

In the method described in [Kirilov, 1991], a reference direction is formed with the help of the reference point. Then, several (weakly) Pareto optimal solutions are generated along the reference direction to be presented to the decision maker.

Methods Based on Miscellaneous Ideas

An interactive extension of the weighting method is presented in [Steuer, 1986]. Many of its ideas are related to those of the IWT method (in Section 2.17). The set of the weighting vectors is reduced according to the choices of the decision maker. The weighting vectors are generated randomly from the reduced space and filtered to obtain a well dispersed set.

The bi-reference procedure presented in [Michalowski, Szapiro, 1992] has been developed for MOLP problems. The decision maker is asked to specify the worst acceptable criterion vector, and the search direction is obtained as the difference between the worst and the ideal criterion vectors. As long a step as possible is taken into that direction and the decision maker is asked to partition the objective functions into three classes (to be improved, to be kept unchanged and to be relaxed). Then the worst and the ideal criterion vectors are replaced and the procedure continues until no significant improvement is achieved. The performance of the bi-reference procedure is compared with other interactive procedures by solving some test examples from the literature. At least in those examples the procedure has managed quite well.

Two different interactive relaxation methods are put forward in [Nakayama, Tanino, Sawaragi, 1980] and [Lazimy, 1986(b)]. The latter is applicable to both continuous and integer problems. The methods are based on the maximization of an underlying value function in a new but equivalent form with additional constraints. Marginal rates of substitution and other estimates of the value function are required from the decision maker. Similar ingredients are utilized in the decomposition method, presented in [Lazimy, 1986(a)]. It is based on the duality theory for nonlinear programming. The original problem is decomposed into a series of linear subproblems and two-attribute problems. A relaxation-projection technique, especially for bi-objective problems with an application to scheduling, is proposed in [Ferreira, Geromel, 1990].

An interactive algorithm with several alternative subproblems is proposed in [Mukai, 1980]. Every objective function is treated equally in the beginning, and the subproblems generate feasible directions where the values of all the objective functions improve. The decision maker can then indicate what objective functions to improve at the expense of the others, and a new direction is generated. Tools for extending the algorithms of Mukai to be applicable to nondifferentiable objective functions are presented in [Kiwiel, 1984, 1985(a), (b)] and [Wang, 1989]. The ideas were applied in the NIMBUS method in Section 2.23.

The method submitted in [Roy, Mackin, 1991] is based on a sequence of pairwise questions and it tries to approximate the parameters of a proxy value function. An example of including ideas from other research areas in interactive multiobjective optimization is presented in [Tapia, Murtagh, 1992]. The authors analyze the preferences of a decision maker in MOLP problems with Markovian processes. Cardinal priority ranking of Pareto optimal solutions is part of the method proposed.

Methods for discrete and continuous multiobjective optimization problems have been combined in [Slowinski, 1991]. A finite set of Pareto optimal points is generated and then the ordinal regression is applied. The method is intended to be practical in the situations where the decision maker wants to focus on a subset of Pareto optimal points at early stages of the process. Similar ideas are utilized in [Bard, 1986]. A set of Pareto optimal solutions is generated by the ε-constraint method and ranked by means of multiattribute decision analysis. The method is demonstrated by an example on the selection of automation options for an upcoming Space Station.

An interactive step trade-off method combining ideas from the SWT method and STEM is presented in [Yang, Chen, Zhang, 1990]. It utilizes trade-off rates and the division of objectives into those to be improved, those that should maintain their values and those to be impaired. A method for complex problems with high dimensionality is proposed in [Baba, Takeda, Miyake, 1988]. The method utilizes a random optimization method and is also applicable to nondifferentiable objective functions. Possibilities of multiobjective optimization in structural mechanics are presented in [Eschenauer, Schäfer, Bernau, 1989]. Two interactive methods are briefly described and applied to the optimization of a conical shell.

The method in [Kim, Gal, 1993] has been intended for MOLP problems. It is based on a recently developed concept, a maximally changeable dominance cone, and marginal rates of substitution. The effectiveness of the method is illustrated by a numerical example. Ideas for reducing the burden on the decision maker in interactive methods are introduced in [Korhonen, Wallenius, Zionts, 1984] and further developed in [Ramesh, Karwan, Zionts, 1988]. An underlying quasiconcave value function is assumed to exist. Convex cones are formed according to the preference relations of the decision maker. The cones are formed such that the solutions in the cones can be dropped from further considerations, because they are dominated by some other solutions. Thus, fewer questions have to be asked from the decision maker in charting the preferences. These ideas of convex cones can be applied as well to multiobjective optimization as multiattribute decision analysis. The ideas have been applied, for example, in [Ramesh, Karwan, Zionts, 1989(a), (b)].

3. Software for Solving Multiobjective Optimization Problems

The recent development of computers and the improvement in the speed, storage capacities, and flexibility of computing facilities has made it possible to produce more sophisticated and demanding software for solving multiobjective optimization problems. Efficient computers enable the implementation of interactive algorithms, since they can produce sufficiently fast responses for the decision maker without the user getting frustrated in waiting. Nevertheless, taking into account the large amount of methods developed for solving multiobjective optimization problems, the number of widely tested and user-friendly computer programs that are widely available is small.

Most of the software packages developed for multiobjective optimization problems can be called multiobjective decision support systems forming a class of decision support systems. *Decision support systems* (often shortened to DSS) can be defined to be interactive computer-based systems designed for helping in the decision making process. The main parts of a decision support system are a model, an optimizer (solver) and an interface between the model, the optimizer and the user. By an interface we mean the input language and style, exchange of information and presentation of the results. It is to be remembered that the human-computer interface must be designed with at least as much care and effort as the other parts of the system.

The role and the requirements of the model, the optimizer and the interface in the multiobjective optimization environment are outlined in [Jelassi, Jarke, Stohr, 1985]. A decision support system is worthwhile to have capabilities of self-learning and model updating. The interface is a dominating factor when the user-friendliness of the system is regarded. One can state that developing software for multiobjective optimization problems is once again a multiobjective optimization problem in itself and proper planning is essential. Several (conflicting) objectives to be taken into consideration in multiobjective software design are mentioned in [Olkucu, 1989].

Features to be taken into consideration when designing decision support systems are also handled in [Lewandowski, 1986]. Different definitions of user-friendliness and rules for dialogue design are given. One must point out that plenty of effort has been made in developing methodological and computational aspects of the systems but the interface, between the system and its user is often of poor quality. This is a serious weakness, since no matter how brilliant the methodology and its implementation is, it will be discarded if the interface does not suit the user. In any case, the algorithms must be implemented in such a manner that computer-technical requirements do not overshadow the real problem and non-skilled persons can use the algorithms, too. One way to try to improve the situation is to provide different interface possibilities for the same system for computer specialists, trained users and average users.

Existing software packages by the year 1980 are listed in [Hwang, Paidy, Yoon, Masud, 1980]. The programs have mainly been developed for linear and goal programming problems. We do not present them here more closely, because they are quite primitive when compared with modern computer facilities. The state of decision support systems developed to aid in the multiobjective optimization and multiattribute decision analysis problems up till the year 1988 has been gathered in [Eom, 1989]. The presentation is only cursory. A classification of the system applications is provided. Some software implementations are also mentioned in [Weistroffer, Narula, 1991].

Most of the software implementing the extensive amount of existing multiobjective optimization methods is neither commonly available nor widely known. However, some implementations are introduced in the following. No detailed information about the realizations is described, since the implementations are under continuous development and the details may be out-of-date at any moment. Only those computer programs are presented whose implementations have been at the author's disposal. They are VIG, DIDAS, CAMOS, ADBASE and TRIMAP. On this occasion it is in order to thank the developers of the programs for giving a copy to the author. At the end of the chapter some other programs are briefly mentioned.

By a *user* we in the following mean either a decision maker or an analyst who uses the solution program. If the user is a decision maker, it is usually assumed that the problem has been formed earlier (and perhaps loaded in the system) so that the decision maker can concentrate on the solution process.

After describing each software product, some practical user experiences are mentioned. It is difficult to compare the programs with each other because they have been developed for so different purposes. Unfortunately, many of the programs are capable of handling MOLP problems only.

3.1. Visual Interactive Approach to Goal Programming

The visual interactive approach to goal programming (VIG), described in [Korhonen, 1987, 1990, 1991(a)] and [Korhonen, Wallenius, 1989(c), 1990]), is a dynamic, visual and interactive solution system for MOLP problems with emphasis on graphical illustration. VIG is based on the visual interactive approach and its adaptation to generalized goal programming, both of which were presented in Section 2.21. What is new is a so-called Pareto race (see [Korhonen, Wallenius, 1988]), which develops reference directions in a dynamic way.

VIG is a commercial product. It has been implemented by P. Korhonen in Helsinki, Finland, on IBM compatible microcomputers (in Turbo Pascal) into a menu-driven system. The user's guide is available (see [Korhonen, 1987]). This description handles version 2.27.

General Outline

VIG follows the principles of the algorithm for the visual interactive approach adapted to goal programming. However, the graphical illustration is quite different. The developers of Pareto race felt that producing one picture at each iteration may make the user feel somewhat at a mercy of the system. The aim of the Pareto race is to transform the solution process into a dynamic form where the user can feel that (s)he is in control. Instead of looking at a certain weakly Pareto optimal curve at a time, the user can move to any direction in the weakly Pareto optimal set. There are no restrictive assumptions limiting the user's behaviour. Using Pareto race in VIG is like driving a car. An accelerator, a steering wheel, gears and brakes are available in the function keys of the keyboard.

In practice, the Pareto race has been implemented by varying two variables (which were already introduced in Section 2.21). The variables are the reference direction and the step-size. Their values are updated according to the actions of the user. If the user pushes the accelerator key, one more step of predetermined size is taken into the reference direction. If the user pushes the gears-forwards key, the step-size is increased and if (s)he pushes the gears-backwards key, the step-size is increased into the opposite direction. If the user pushes the brakes key, the step-size is decreased. If the user wants to change the direction, (s)he pushes the ordinal number of the goal whose significance (s)he wants to increase. This causes a change in the reference direction. The values of the objective functions can also be fixed and relaxed. The values of the objective functions are displayed as a bar chart, where the lengths of the bars keep changing according to the instructions of the user. The objective function values are on view also numerically.

Every time the parameters are changed, the system minimizes the corresponding achievement function and shows the solution. All this happens so quickly that the lengths of the bar graphs seem to change dynamically in real-time as the user "travels" on the weakly Pareto optimal set. The functioning of the system reminds a simple video game. Several solutions can be saved for further analysis.

Because of the achievement function employed, VIG produces weakly Pareto optimal solutions. It is, however, suggested in [Korhonen, Halme, 1994] that lexicographic ordering can be used to guarantee the Pareto optimality of the solutions.

Pareto Race (and VIG) can be characterized as an ad hoc method. No value function is now assumed to exist and, thus, the optimality of the final solution is not checked. How satisfactory a solution the user manages to find depends on her or his patience. A more profound presentation of VIG has been collected in [Miettinen, 1990].

VIG is used in [Korhonen, Soismaa, 1988] for pricing alcoholic beverages in Finland. Pareto race is used in locating ocean waste disposal sites in [Leschine, Wallenius, Verdini, 1992]. Other problems where VIG has been applied are presented in [Korhonen, 1990]. The performance of VIG is compared with four other methods in [Korhonen, Wallenius, 1989(b)]. An MOLP problem with five objective functions was solved by 65 decision makers. VIG was found to be superior to the other methods. The main reason was that the decision makers found the aspiration levels to be a comfortable way of expressing preference relations.

Features from VIG (Pareto race) and the visual interactive approach (see Section 2.21) have later on been combined into a computer graphics-based decision support system in a novel way. This system is described in [Korhonen, Wallenius, Zionts, 1992]. The new method is especially useful for large-scale linear problems. Graphical presentations from both of the basis methods have been included.

Practical Experiences

The main benefit of VIG is its easiness to use. No difficult questions are asked from the user. The user can feel that (s)he is in command instead of the computer or the analyst. The graphical illustration is the dominating part of the system. Unfortunately, only linear problems can be solved. The maximum size of the problem is 96 variables and 100 functions, of which up to 10 may be objective functions.

The fact that no parts of the weakly Pareto optimal set can be eliminated from the examination, can be seen as a drawback. It may also be difficult to find again some solution discovered earlier, if it was not saved. If there is a large number of objective functions, the user may have difficulties in finding improved solutions.

3.2. DIDAS

DIDAS (Dynamic Interactive Decision Analysis and Support) system (earlier DIDASS) is a family of decision support systems developed at the International Institute for Applied Systems Analysis (IIASA) in Austria in collaboration between IIASA and several Polish scientific institutions. There is a lot of literature describing different steps of the development work (see [Lewandowski, Grauer, 1982], [Grauer, 1983(a), (b)], [Grauer, Lewandowski, Wierzbicki, 1984], [Lewandowski, Kreglewski, Rogowski, Wierzbicki, 1987], [Rogowski, Sobczyk, Wierzbicki, 1987], [Kreglewski, Paczynski, Wierzbicki, 1987], [Kreglewski, 1989] and [Kreglewski, Granat, Wierzbicki, 1991]).

DIDAS is a commercial product and it can be inquired from IIASA. Originally, DIDAS was designed for mainframe computers, but nowadays microcomputer versions exist. Some versions are in Fortran 77 (see, e.g., [Grauer, 1983(a)]) and some in Pascal (see [Kreglewski, Paczynski, Wierzbicki, 1987]). There are special packages for dynamic problems, linear problems (called IAS-DIDAS-L, see [Rogowski, Sobczyk, Wierzbicki, 1987]) and nonlinear problems. Here we mainly outline the nonlinear version, called IAC-DIDAS-N (for short DIDAS-N) for IBM compatible microcomputers up till version 4.0.

General Outline

DIDAS is an implementation of the reference point method, presented in Section 2.19. It is a dynamic decision support system and it aims at helping in achieving better decisions. Users of DIDAS may be decision makers, analysts, or decision makers and analysts together. The forms of the achievement functions are varied in different versions of the system.

DIDAS is based on satisficing decision making. The user gives a reference point and a corresponding Pareto optimal (or weakly Pareto optimal) solution is generated by optimizing an achievement function. Then the user can specify new aspiration levels or let the system help in determining them.

Three kinds of objective functions can be given. They can be functions to be minimized, to be maximized, or to be stabilized (i.e., minimized if their values are above stabilization levels and maximized if their values are below stabilization levels). Both equality constraints and inequality constraints can be used. All the functions are assumed to be differentiable. In addition to the aspiration levels $\bar{z}_i$, the user is asked in some versions (e.g., in IAC-DIDAS-N 4.0) to specify *reservation levels* $\tilde{z}_i$ for the objective functions. The reservation levels are interpreted as "soft" upper bounds (or lower bounds in maximization case) for the objective functions. The reservation levels should be more pessimistic than the aspiration levels; for example, for objective functions to be minimized, they should be greater than the aspiration levels.

In the beginning, the ranges of the Pareto optimal set are presented to the user to give an overview of the problem. In some versions, these ranges are also used to scale the objective functions. In different versions of DIDAS, the ranges are determined in different ways. Some versions utilize the means described in Section 1.5 (see [Kreglewski, 1989]) and some versions optimize the achievement function with systematically varied aspiration levels and collect the best and worst objective function values obtained (see [Lewandowski, Kreglewski, Rogowski, Wierzbicki, 1987]). A *neutral Pareto optimal solution* which is situated "in the middle" of the Pareto optimal set is usually calculated as a starting point. Also so-called relative achievement factors and conflict coefficients may be calculated at the beginning of the solution process to give information about the difficulty of the problem (see [Kreglewski, 1989]).

A vital factor in the successful functioning of DIDAS is that the user can easily affect the selection of Pareto optimal solutions by changing the aspiration levels. Because of this, the objective functions are in many versions scaled by the difference between a slightly displaced ideal criterion vector and the current reference point. The user is assumed to specify aspiration levels between the ideal criterion vector and the nadir point. Now the user can implicitly attach more importance in attaining some aspiration level by placing it near the ideal criterion value. In that case, the corresponding objective function is weighted stronger in the achievement function.

Now we give an example of achievement functions utilized in DIDAS. If all the objective functions are to be minimized, an order-approximating achievement function to be maximized can be of the form

$$\min_{1\le i\le k}\Big[\min\left[(\tilde{z}_i - f_i(\mathbf{x}))/s'_i,\ \ 1+(\bar{z}_i - f_i(\mathbf{x}))/s''_i\right]\Big]+$$
$$\varepsilon\left(\sum_{i=1}^{k}\min\left[(\tilde{z}_i - f_i(\mathbf{x}))/s'_i,\ \ 1+(\bar{z}_i - f_i(\mathbf{x}))/s''_i\right]\right),$$

where $\varepsilon > 0$ and the coefficients $s'_i > 0$ and $s''_i > 0$ are scaling units to be determined automatically according to a complicated formula (see [Kreglewski, 1989] and [Kreglewski, Granat, Wierzbicki, 1991]). Achievement functions are typically nondifferentiable at the reference point. So is this function. However, it can be solved in an equivalent differentiable form if the objective functions are differentiable. In practice, the achievement functions are replaced by differentiable approximate functions which are also order-approximating (see, e.g., [Kreglewski, Granat, Wierzbicki, 1991]).

The developers of DIDAS have also considered the problems of gradients. They have come to the conclusion that only gradient-based nonlinear programming algorithms for the optimization of the achievement function are efficient and robust enough to be employed in interactive decision support systems. An important question is how to obtain the gradients of the objective and the constraint functions. For two reasons it is not advisable to ask the derivatives from the user of the system. Firstly, the formulation of the derivatives is a time-consuming and laborious task, and secondly, errors and mistakes are likely to occur. Mistakes have been found to be a main reason for nonlinear optimization methods to fail in convergence. However, numerical estimation of the gradients is considered to be very time-consuming and not accurate enough. A potential possibility is to formulate the gradients symbolically inside the model. The implementation of this gradient formulation is briefly handled in [Kreglewski, 1989]. One more alternative is to use automatic differentiation. However, such results have not been reported.

Some experiences in applying DIDAS to macroeconomics planning are reported in [Grauer, Lewandowski, Wierzbicki, 1984]. DIDAS is used in empirical tests in [Bischoff, 1985] to experiment with different scalarizing functions. A problem of determining the optimal temperature in a greenhouse is solved by a nonlinear version of DIDAS in [Udink ten Cate, 1985]. In [Stam, Kuula, Cesar, 1992], DIDAS-N is used in analysing the acid rain problem in Europe. A trajectory-oriented extension of DIDAS is described and applied in [Lewandowski, Rogowski, Kreglewski, 1985(a), (b)]. IMPROVE, a package for MOLP problems, which is based on experiences with DIDAS, is introduced in [Barnikow, Gollmer, Grauer, 1986]. Some applications to strategic planning of carbochemical industry are also reported.

Practical Experiences of the Version IAC-DIDAS-N 4.0

Even though the facilities of microcomputers have been utilized, the interface between the user and the system is still quite cumbersome. The screen contains cells where the problem information is to be input. The system is controlled by pushing *alt* or *ctrl* keys and letters or function keys. The solution process is divided into three phases and to be able to continue to the next phase, the current problem must be saved. The user must be acquainted with the system to be able to use it. The system itself does not give any guidance how to proceed or what one is expected to do. Once the problem with reservation and aspiration levels and some other information has been successfully stored, the problem may be solved. The reservation and the aspiration levels can be altered and the obtained results can be saved. Then the results which have been saved can be displayed in a graphical bar form.

What is positive with IAC-DIDAS-N is that many kinds of problems can be solved with it. In addition, the user does not have to give derivatives of the functions if (s)he does not want to. One must acknowledge that a lot of attention has been paid to developing efficient achievement functions and other computational tunings. On the other hand, the interface has to develop remarkably to become a system that can be called user-friendly. When the system is like this, the user needs a lot of practice and experience to be able to exploit the possibilities of the system effectively.

3.3. CAMOS

CAMOS (Computer Aided Multicriterion Optimization System) is an optimization system developed by A. Osyczka in Cracow, Poland, to treat especially (nonlinear) computer aided optimal design problems. It has been written in standard Fortran to run on IBM compatible microcomputers (see [Osyczka, 1989(b), 1992] and for an earlier version [Osyczka, 1984]). However, there are no obstacles for the system to be run on other computers. CAMOS is a commercial product distributed through International Software Publishers (see [Osyczka, 1992]).

General Outline

CAMOS does not contain any highly developed interactive solution method. The system has been meant for producing Pareto optimal solutions with different generating methods. The user can solve the problem with different single and (noninteractive) multiobjective optimization methods, with different parameters and input data, with different starting points, etc.

Using CAMOS requires the user to know Fortran since the problem to be solved must be given as a Fortran subroutine. The functions are supposed to be minimized and equality and inequality constraints can be used. The user can also supply general data which can be altered inside CAMOS. In addition, the input subroutine can contain code for printing the results of the problem. Since the system has been intended for general usage to be compatible between different computer systems, no graphical capabilities of microcomputers have been utilized in the interface.

CAMOS is able to solve nonlinear single and multiobjective optimization problems with continuous, integer, discrete and mixed decision variables. The methods for identifying (weakly) Pareto optimal solutions are the weighting method with or without normalizing the objective function (see Section 2.2), the method of global criterion, the min-max method (which means the method of global criterion with L_∞-norm) and the weighted min-max method (see Section 2.7).

Parameters for the methods are asked from the user. The weighting coefficients are assumed to sum up to one in the weighting methods. The normalized weighting method means that every objective function is divided by its ideal criterion value. In the method of global criterion, the user must specify the exponent p. The norms with ideal criterion values as denominators and without the exponent $1/p$ are used (when compared with Section 2.7). The min-max method has been briefly described by the formula (2.7.3). See more details, for example, in [Osyczka, 1984, 1992]. If the user does not want the ideal criterion vector to be used in the methods, (s)he may specify another reference point. In this case, solutions generated are not necessarily Pareto optimal.

In addition, there is a method called "generating a set of Pareto optimal solutions". This means that feasible decision vectors are generated by a random search method. Each new point is compared with the stored points. If the new point is dominated by the older points, it is discarded. On the other hand, if the new point dominates some of the older points, they are discarded and the new point is stored.

The user can choose the optimization method for the newly developed single objective optimization problem among the random search (Monte Carlo) method, the direct search method of Hooke and Jeeves, the nonlinear simplex method of Nelder and Mead, the variable metric method of Davidon, Fletcher and Powell, and the flexible tolerance method of Himmelblau. The user may also include new solution methods. Also some combinations of the methods are permitted.

To illustrate the functioning of CAMOS the results of solving two practical problems are reported in [Osyczka, 1992]. They are the optimal design of multiple clutch brakes and the optimal counterweight balancing of robot arms.

Practical Experiences

The interface in CAMOS is system-driven and the user must answer a lot of questions, which may be rather frustrating. Information about expected reply alternatives can be obtained by the HELP command. There does not, anyway, exist any default replies. This is sometimes laborious, for example, when specific data for the selected single objective optimization routine is given. HELP command shows standard data as a guideline and the user must give the values even though (s)he does not want to change the standard input. To be able to change the problem to be solved, CAMOS must be quit and another Fortran program must be linked with it.

The user of CAMOS has to be active. When one solution has been obtained, the system does not advise how to proceed. The user must know what to do next.

What is positive in using Fortran programs to input the problem to be solved is that also more complicated problems can be solved. By complication we mean problems where the values of the objective functions are determined via calling subroutines and not by short expressions. The same language feature can also be seen as a negative part of the system. Knowing Fortran is a threshold to using CAMOS. On the other hand, one can think that an analyst inputs the Fortran parts and the decision maker only uses the program which is ready for use. One can also state that CAMOS has not been intended to be an easy tool for every average user.

When using CAMOS, it must be kept in mind that it may be quite sensitive to the starting point specified by the user. This can be overcome by solving the same problem with several different starting points or with the random search method to generate a starting point.

A professional user may find CAMOS a practical tool for solving problem if (s)he has got enough patience, time and know-how and (s)he does not expect highly developed interaction.

3.4. ADBASE

ADBASE is a widely distributed package of programs for solving MOLP problems. ADBASE has been developed by R. E. Steuer in Georgia, USA (see [Steuer, 1989(c)]), though some of the programs have been made by others. This short summary is about the version 9/89. All the programs are written in low level Fortran IV and run on both mainframe and personal computers (with or without a math-coprocessor). ADBASE is a commercial product. It can be inquired from R. Steuer.

General Outline

In addition to the package itself, ADBASE is also the name of a program which is able to generate all the Pareto optimal extreme solutions and all the unbounded Pareto optimal edges of an MOLP problem. ADBASE operates by solving a family of weighting problems. Subsets of Pareto optimal extreme points and unbounded Pareto optimal edges can be generated by reducing the set of weighting vectors. Also preemptive goal programming problems can be solved. ADBASE uses a multiobjective analogue of the revised simplex method of single objective optimization. One can also generate linear random test problems by ADBASE.

All the programs in the ADBASE package have been integrated so that output from one program can be used as input to some other program. FILTER is a program for finding a dispersed subset from a set of vectors. LAMBDA is a program for gathering representatives from the set of weighting vectors specified by lower and upper bounds. UTILITY computes the value of a specified value function at given criterion vectors. VECTOR computes an updated weighting vector utilized in the older version of the IWT procedure and CONVERT converts input formats.

The use of ADBASE package is demonstrated in [Steuer, 1986]. It is shown, for example, how the programs of the package can be used to solve MOLP problems by the IWT procedure. Some computational testing with a revised version is reported in [Steuer, Gardiner, 1991]. It is demonstrated how programs of ADBASE can be exploited in solving MOLP problems by Wierzbicki's reference point method as well as by the IWT method.

Practical Experiences

ADBASE is rather a collection of tools than a program. The user must know what (s)he must and can do with the different subroutines to obtain desired solutions. Thus it is not meant for decision makers as such. Instead, from the subroutines available, an analyst must first collect a program which implements some method.

The weakness of ADBASE is its stiffness and thus relatively high threshold of usage. The input and the output formats are fixed and inflexible and no graphical properties are available. Two input files of prespecified contents and layout are required for a multiobjective optimization problem to be solved by ADBASE. Because of versatile possibilities of the problem, there are quite many parameters whose values have to be set.

Getting accustomed to using ADBASE takes time and effort. Likewise is the case with the flexible usage of the auxiliary programs. But after the user has familiarized oneself with the programs, MOLP problems can be solved and analysed in many ways.

3.5. TRIMAP

TRIMAP is an interactive program for solving MOLP problems with three objective functions. It has been developed at the University of Coimbra, Portugal (see [Clímaco, Antunes, 1987, 1989]). The program has been written in Pascal and it runs in the Apple Macintosh environment utilizing its graphical capabilities. The version to be handled here is 2.d1. The program can be inquired from the developers.

General Outline

The starting point of TRIMAP is the weighting method with the possibility to decompose the set of the weighting vectors into subsets where the weighting method results in the same solution. Computer graphics plays an important part. Also the ε-constraint method has been included so that it would be possible to eliminate regions of the criterion space. The program is based on progressive and selective "learning" of the Pareto optimal set where the user can eliminate subsets of the Pareto optimal solutions as they become uninteresting. The convergence of the solutions is not of interest in TRIMAP.

What the user can do with TRIMAP is to study the changes when (s)he alters the weighting coefficients or the upper bounds. The user can also consider different projections of the criterion space and the weighting space. The limitation to three objective functions allows the effective use of computer graphics. The fact that there are only two independent weighting coefficients is utilized in the graphical presentations. A new version of TRIMAP for three-objective transportation problems has been introduced in [Clímaco, Antunes, Alves, 1993].

The ZW method, STEM and the computer program TRIMAP have been compared in [Clímaco, Antunes, 1990]. A problem of power generation system expansion planning with three linear objective functions was solved. All the programs were implemented in the same (Macintosh II) environment. The authors conclude that the TRIMAP was easiest to use since the decision maker does not have to answer difficult questions.

Practical Experiences

TRIMAP is at its best when analyzing problems of a relatively small dimension. The interface is quite user-friendly and TRIMAP is suitable also for users with little experience with computers. The assumption of three objective functions limits the number of practical applications. The developers of TRIMAP have mostly used it for educational purposes (see [Clímaco, 1989]).

3.6. Other Software Packages

Software comparisons reported in the literature concern mainly programs for multiattribute decision analysis. We just mention that seven microcomputer implementations in C are presented and compared in [Colson, De Bruyn, 1987]. Five of them are intended for multiattribute decision analysis. Also an implementation of STEM is reported. Main features and requirements of eight microcomputer software packages are introduced in [Lotfi, Teich, 1989]. One of them is VIG and the other seven are for discrete alternatives.

There exist several software packages for general single objective optimization problems that also contain some possibilities for multiobjective optimization. Here we briefly indicate some of them.

EASY_OPT is an optimization system which has been intended to be easy to use, as can be concluded from its name. It has been developed by K. Schittkowski in Bayreuth, Germany, and it works in IBM compatible microcomputers. The system contains methods for both single and multiobjective linear and nonlinear optimization problems. A Fortran-type language PCOMP is used for specifying nonlinear functions.

Multiobjective optimization means in EASY_OPT that Pareto optimal solutions are generated with different methods. Then a graphical projection into the (z_1, z_2)-space is presented for evaluation. Thus no actual interaction is possible.

Several generation methods are available. The basic methods, the weighting method and the ε-constraint method, are available. Four different variants of the method of global criterion have also been included. Other possibilities are minimizing the maximum of the objective functions, of the weighted objective functions or of the absolute values of the objective functions. At hand is also a kind of Archimedian goal programming with five different distance measures.

Expert System for Mathematical Programming (EMP), created by K. Schittkowski in Bayreuth, Germany, and described in [Schittkowski, 1988], is a general expert system. Both mainframe and microcomputer versions are available. Here, as in the EASY_OPT, the main emphasis has been in other algorithms (e.g., data fitting and single objective optimization) than multiobjective optimization. Thus the possibilities for solving (differentiable) multiobjective optimization problems are restricted into the weighting method, the weighting method with additional upper bounds for the objective functions and minimizing one objective function and suppressing all the others. In addition, it is possible to handle solutions in an interactive mode. New solutions can be generated and saved and also deleted from the store.

EMP has been implemented by a SUSY language. The idea is that a pile of questions is asked and according to the answers, an executable Fortran program is produced. Still, the user is assumed to know some Fortran because the objective and the constraint functions are to be given as Fortran expressions. The outcoming single objective optimization problem is solved by an NLPQL method based on the SQP (sequential quadratic programming) ideas. Unfortunately, EMP is not an expert system from the point of view of multiobjective optimization. Choosing the method is up to the user, and, as mentioned earlier, there are not so many methods to choose from.

It has been reported in [Miettinen, 1990] how a method based on Kanniappan's ideas (see Section 2.2) has been appended into EMP. Instead of the weighting coefficients, their upper or lower bounds or both of them are asked from the user.

NOA, a collection of Fortran subroutines for minimizing nondifferentiable functions subject to linear and nonlinear (nondifferentiable) constraints, is described in [Kiwiel, Stachurski, 1989]. NOA is applicable to multiobjective optimization problems since the single objective function to be minimized is supposed to be a maximum of several functions. Thus, for example, reference point problems with the achievement function (2.19.10) can be solved.

The subgradient GDF and the NIMBUS algorithms, introduced in Sections 2.22 and 2.23, respectively, have been implemented by Fortran 77 at the Department of Mathematics of the University of Jyväskylä. However, the main interest has been in testing purposes. Thus, the user interface side is still under development. For this reason, the programs are not presented here. A snapshot of the user interface of NIMBUS, which is under development, was proposed in Section 2.23.

4. Graphical Illustration

Graphical illustration is an essential part when designing modern user interfaces. Graphics may be used to assist the decision maker in specifying values for problem parameters or to illustrate the contents and the meaning of questions posed by the algorithms. In such realizations, the upper limit lies in one's imagination. One proposal into that direction was suggested in Figure 17 in Section 2.23.

In spite of the more general possibilities, we restrict our treatment in this chapter. By graphical illustration we here mean the ways of presenting several criterion vectors to the decision maker. One can notice the need for such illustration when examining the interactive methods described in Chapter 2.

As computers have developed, more attention has been paid towards the role and the possibilities of computer graphics in building human-computer interfaces. Nevertheless, utilizing graphical illustration does not invalidate the limits of human capacity for processing information. So there is no sense in trying to offer too many criterion vectors for evaluation, no matter how clear the illustrations are. It is claimed in [Miller, 1956] that the limit usually is seven plus or minus two. (Seven ways of decreasing the number of alternatives are presented in [Graves, Ringuest, Bard, 1992].)

Naturally, many different ways for illustrating criterion vectors can be thought of. However, elegant graphics must not be an end in itself. The graphics must be easy to comprehend for the decision maker. On one hand, not too much information is allowed to be lost and, on the other hand, no extra unintentional information should be included in the presentation.

If there are two objective functions, the graphical illustration of the criterion space is effective. The feasible criterion region and, especially, its Pareto optimal subspace can be sketched on a plane. It is suggested in [Meisel, 1973] that when there are three objective functions, the Pareto optimal set can be expressed by three projections on a plane. The interpretation of such information is far more difficult for the decision maker. This idea of projections is utilized in TRIMAP.

In the following, we present some ways of graphical illustration. Some of the ways are clarified by applying them to an example of three alternative criterion vectors of a problem with three objective functions.

A widely used way of representing sets of criterion vectors is to use value paths, as suggested in [Schilling, ReVelle, Cohon, 1983]. Then, horizontal lines of different colours or of different line styles represent the values of the objective functions at different alternatives. It means that one line displays one alternative. This is depicted in Figure 18. The bars in the figure show the ranges of the objective function in the Pareto optimal set. If the ranges are known, they give additional information about the available possibilities. Notice that each objective can have a scale of its own. Examples are suggested in [Hwang, Masud, 1979] and [Törn, 1983] how to display the scales of the objective functions.

The roles of the lines and the bars can also be interchanged. Then bars denote alternatives and lines denote objective functions. In this case, possible different scales of the objective functions have to be interpreted in a different way. This way has been utilized, for instance, in the first implementations of the visual interactive approach (described in Section 2.21), and its counterpart for discrete problems, called VIMDA, see [Korhonen, 1986, 1991(a)]. The idea in VIMDA is that when the user horizontally moves the cursor on a bar representing an alternative, the numerical criterion values are presented.

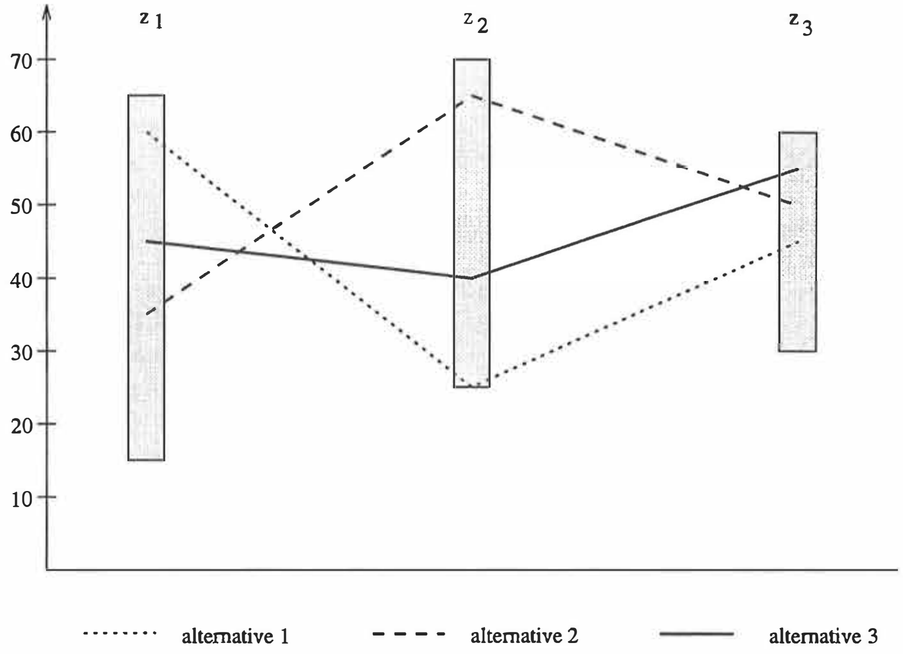


Figure 18. Value paths.

Value paths are an effective means of presenting information to the decision maker without overloading her or him. Another general way of illustration is to use bar charts. Then a group of bars represents the values of one objective function at different alternatives, as in Figure 19. The bars of the same colour form one criterion vector. Also here, separate ranges for objective functions are possible. Parallel ideas have been realized, for example, in DIDAS.

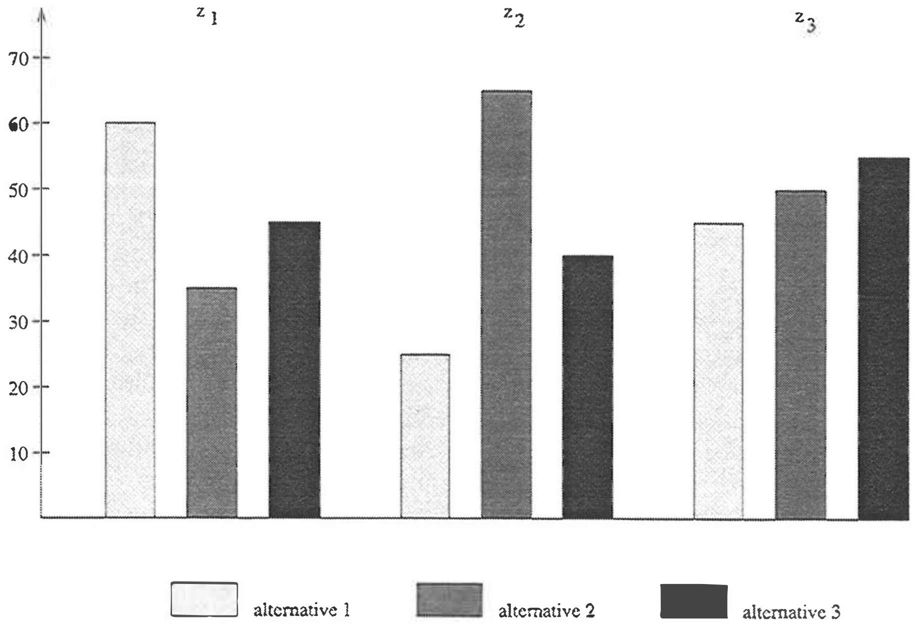


Figure 19. Bar chart.

The problem how we can determine a priori whether the graphical formats used will aid rather than hinder the decision making has been studied in [Jarvenpaa, 1989] by comparative studies. The concluding result is that knowledge about the relationship between the presentation format and the decision strategy can facilitate the selection of the presentation format. Especially, benefits of bar charts and grouped bar charts have been studied in [Jarvenpaa, 1989]. See the reference for further details.

Literature describing graphic presentation of data has been summarized in [Lewandowski, Granat, 1991]. It is stated that the research done does not provide clear answers regarding what types of data presentation to favour in the decision making context. Lewandowski and Granat suggest a technique for graphical presentation of matrices of rank 2, called BIPLOT. The set of Pareto optimal criterion vectors forms a matrix. This matrix is factorized into a product of two matrices. The vectors in the two matrices are of order two and can be plotted on a plane giving a representation of the original criterion vectors. Dynamic BIPLOT in aspiration-based decision support systems is also described. Another thing is how much experience one must have to be able to interpret representations like these.

It is suggested in [Mañas, 1982] that criterion vectors can be represented in a star coordinate system. For example, an alternative of five criteria is represented as an irregular pentagon. This requires the ideal criterion vector and the (possibly approximated) nadir point to be known. An example is given in Figure 20. Each circle represents one alternative criterion vector. The ideal criterion value is at the centre and the component of the nadir point is at the circle. Each ray represents one objective function. See details in [Mañas, 1982].

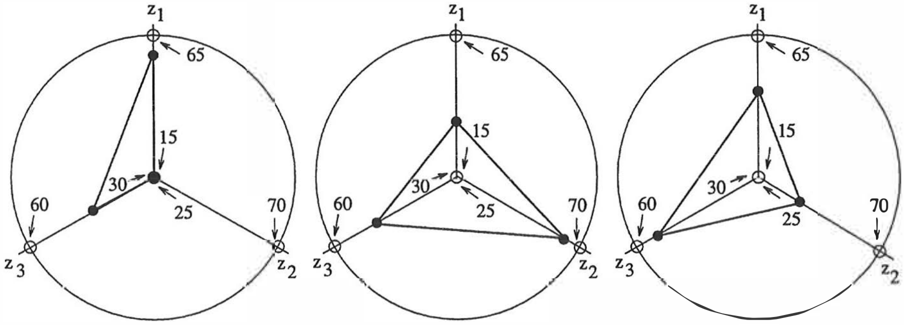


Figure 20. Star coordinate system.

Similar ideas are exploited in [Kasanen, Östermark, Zeleny, 1991]. An example of them is presented in Figure 21. This illustration form can be called a spider-web because of its shape. Each apex represents one objective function. The outer triangle shows the (possibly approximated) nadir point, the inner triangle (the darkest one) stands for the ideal criterion vector and the middle triangle (the grey one) presents one alternative criterion vector. Thus, only the middle triangle is different in different alternatives. The ideas are further developed in the reference.

Somewhat parallel ideas are utilized in [Angehrn, 1990(a), (b)] when illustrating discrete alternatives in a program called Triple C (Circular Criteria Comparison). A circle is divided into k (the number of objective functions) sectors. The size (radius) of each slice indicates the magnitude of the criterion value. One of the ways mentioned in [Klimberg, 1992] is to transform criterion vectors into two-dimensional curves with the aid of Fourier series. All the vectors can be plotted on the same coordinate system for comparison.

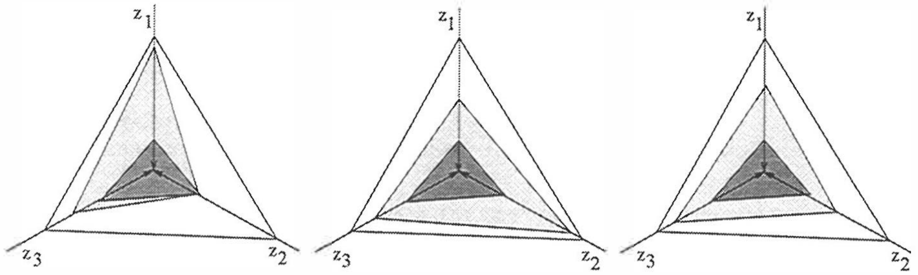


Figure 21. Spider-web illustration.

Different ideas of graphical illustration are presented in [Korhonen, 1991(b)]. Korhonen reminds that, for example, Chernoff's faces have been developed to illustrate numerical information. The reasoning is that the values of each objective function are parametrized to represent some feature of an icon. The icon used must be such that the user can see the icon becoming "better" as the value of the objective function improves. This is why concepts like symmetry and harmony are important. An icon which people have been used to seeing in a harmonious and symmetric form is a house. Thus, Korhonen suggests so-called harmonious houses to be used as icons. Objective functions are associated with the corner points of the house, the door, the windows, or the roof. The aim is that when the values of the objective functions are near the ideal criterion vector, the house is quite harmonious and symmetric. This type of illustration has especially been intended for pairwise comparison.

Using colours in illustration has its good and weak points. The colours must be easy to discriminate. Another important fact is that some colours may have specific connotations to the user. Such colours should be avoided as far as possible. An experimental evaluation of graphical and colour-enhanced information presentation is given in [Benbasat, Dexter, 1985]. It has been found out that a tabular representation is the best when a simple retrieval of data is important and a graphical representation is the best when relationships among the data have to be illustrated. Graphs are visually appealing but sometimes tables are easier to read since they provide exact values. Benbasat and Dexter conclude that colours improve the readability and understandability of both the symbolic and graphical displays. Colours make it easier for the subject to associate visually the promotion and profit figures (or lines) belonging to the same territory, since the promotion-profit pairs are coded in the same colour. Benbasat and Dexter have obtained encouraging results with multi-colour reports.

Other proposals for the graphical illustration of alternatives have been given, for example, in [Vetschera, 1992] (for discrete problems). The method is based on indifference regions and linear underlying value functions.

A recommended way of presenting information to the decision maker is to offer the same data in different forms. In this way, the decision maker can choose the most illustrative and informative representations. The illustrations may also supplement each other. The decision maker can change her or his attention from one figure to another and possibly skip undesirable alternatives before the final selection. A simple tabular format may be one of the figures. Corresponding ideas are suggested, for instance, in [Silverman, Steuer, Whisman, 1985] and [Steuer, 1986] and their implementation is described in [Miettinen, 1990]. In the last-mentioned reference, tabular information, value paths and bar charts are used to illustrate the IWT procedure.

Finally, one must concede that if there is a great number of alternatives, the decision maker may get confused no matter how the alternatives are illustrated.

5. Comparing the Methods

As one can be convinced from all that has been said so far, there is a large variety of methods for multiobjective optimization problems and none of them can be claimed to be superior to the others in every aspect. Thus, it is in order to consider some matters of comparison and selection between the methods.

In addition to theoretical properties, also practical applicability plays an important role in selecting an appropriate method. Some matters of interest may come up in the computational applications available to support the method selection. There have only been quite a few actual computational applications of the multiobjective optimization techniques in the literature. Many methods have been presented without computational experiences. It is tenably remarked in [Bischoff, 1986] that even most of the applications presented are merely proposals for applications or they deal with highly idealized problems. For most (interactive) methods a natural reason is the difficulty (in finding and) in testing with real decision makers. A complicating fact is also the diversity of the decision makers.

A few comparisons of multiobjective optimization methods and software packages have been reported in the literature, and here we mention some of them. Then we consider some matters in selecting a method. We also propose a decision tree for the selection. Finally, we compile some features of the interactive methods handled into a comparative table.

5.1. Comparisons Available in the Literature

Here we briefly mention some comparisons available and bring about a few results and conclusions obtained. For more detailed information we cite the references indicated. An interesting matter to notice is that most of the multiobjective optimization problems solved when testing the methods have been linear. It is true that complex nonlinear functions cause difficulties of their own and the characteristics of the solution methods may be disturbed. On the other hand, features concerning nonlinear problems may remain unnoticed. On the whole, the comparisons available are not of too much help if one is looking for a method for a nonlinear problem.

Some caution is in order when trying to judge something from the comparisons. The comparisons have been performed according to different criteria and under varied circumstances. Thus they are not fully proportional. Which method is the most suitable for a certain problem depends highly on the personality of the decision maker and on the problem to be solved.

Instead of a human decision maker one can sometimes employ value functions in the comparisons, but such tests do not fully reflect the real usefulness of the methods. One can try to compensate the lack of a real decision maker by employing several different value functions. If, for example, marginal rates of substitution are desired, the decision maker's inconsistency and inaccurate responses can be imitated by multiplying them with different random numbers. These means have been employed in [Shin, Ravindran, 1992].

On the other hand, value functions cannot help in testing ad hoc methods. Notice also that practical experience is especially important in evaluating the techniques with respect to criteria related to the decision makers. It is important to compare a method under a variety of circumstances so that the conclusions can be generalized.

One critical factor that can affect the performance of the methods in the comparisons is the user interface. Nothing is usually mentioned about the realization of the user interface in the comparisons reported. It is important to remember that one can damage a "good" method with a poor user interface or to support a poor method with a good interface. A good user interface, in addition to the illustration of the (intermediate) results, also means a clear and intelligible input phase. (It is natural that when someone has developed a method, (s)he may be (more or less) tempted to form the testing conditions to favour her or his own method.)

An MOLP problem for determining the most economical combination of grape growing and wine production in Hungary has been solved by the weighting method, the ε-constraint method, the lexicographic ordering and the weighted L_1- and L_2-metrics with normalized objective functions in [Szidarovszky, Szenteleki, 1987]. It is noticed that different solutions are obtained with each method. It is also stated that the weighted L_1- and L_2-metrics with normalized objective functions produce the most uniform distribution of criterion vectors. Finally, the weighted L_1-metric is concluded to be the most convenient way for generating Pareto optimal solutions in large-scale MOLP problems.

A linear problem of mining industry has been solved in [Peterson, 1984] by the weighting method, the ε-constraint method, the weighted method of global criterion with and without denominators, and by the lexicographic ordering. The solutions from the other methods are utilized in the methods of global criterion and all the solutions are analysed. The conclusion is that the solution methods should be applied by complementing each other.

Notice that no interactive methods were included in the comparisons mentioned so far. The following comparisons involve interactive methods.

It is described in [Dyer, 1973(b)] how nine (student) decision makers were given an MOLP problem on choosing an engine for a car. They were first asked to suggest a solution approach and then compare it with the GDF method and a trial-and-error procedure. (The trial-and-error procedure was simply such that the decision maker was asked to enter a criterion vector and the program informed whether it was feasible or not. The decision makers were supposed to explore the feasible criterion region until they were unable to find more preferred solutions.)

The criteria in the evaluation were the easiness of using the procedure and the confidence in the solution obtained. Dyer obtained results favouring the GDF method. Thus, he concludes that the GDF method can successfully be used by untrained decision makers.

The performance of the GDF method, STEM and the trial-and-error procedure (the same as used by Dyer) is compared from the point of view of a decision maker in [Wallenius, 1975]. A total of 36 business school students and managers from industry were employed as decision makers. The following aspects of the methods were compared: the decision maker's confidence in the solution obtained, ease of use and understanding of the method, usefulness of the information provided, and rapidity of convergence. The linear management problem to be solved contained three objective functions.

The results are analysed statistically in [Wallenius, 1975]. One interesting conclusion is how well the trial-and-error procedure competed with the more sophisticated methods. Nevertheless, Wallenius points out that its performance could weaken if the problems were more complex. Difficulties in estimating the marginal rates of substitution deteriorated the overall performance of the GDF method. Wallenius suggests that a logical direction of research would be to attempt to better adjust the methods to match the characteristics of a human decision maker, and vice versa.

There was a remarkable difference in the results of Dyer and Wallenius concerning the GDF method. Some trials of analysing the reasons are presented by Wallenius.

The capabilities of the ZW, the SWT and the IWT method and a naive solution method are compared in [Buchanan, Daellenbach, 1987] from the point of view of the user in solving a linear three-objective optimization problem. The naive method here means producing a weakly Pareto optimal point from a point suggested by the decision maker with the help of the L_∞-norm.

The problem concerned producing electrical components of lamps. A total of 24 decision makers (students and academic staff) were employed. The criteria in the comparison were partly the same as those of Wallenius. In addition to confidence in the final solution, ease of use and ease of understanding the logic of the method, CPU and elapsed time were compared. The most important criterion was the relative preference of using each method. The conclusions were that the IWT procedure was clearly preferred to the other methods and the ZW method was the worst under the first four criteria. The SWT method was in the middle. Once again, the naive method performed surprisingly well. The authors conclude that the decision makers seem to prefer such solution methods where they can feel being in control.

Steuer's method (see Section 2.24 and [Steuer, 1986]) and the ZW method are compared in [Michalowski, 1987]. Five decision makers from the planning department of a factory were employed. A linear production planning problem with three objective functions was solved and the evaluation criteria were not fixed in advance, though the main interest was in the decision phase. The decision makers had critical comments about both the methods, and each of them obtained a different final solution. One can say that the decision processes by the ZW method terminated slightly faster than those by Steuer's method.

The method of Steuer and STEM are tested in [Brockhoff, 1985]. An amount of 147 decision makers were employed to solve six problems involving buying cars. The results and progress were analysed according to several criteria leaving the method of Steuer with the best outcomes on the average.

A more detailed review of the above-described and some other empirical studies involving real decision makers is presented in [Olson, 1992]. It can be stated that no final conclusions can be made from the experiments. The reason is that the test settings and the samples are not similar enough.

An MOLP problem with three objective functions concerning natural gas business has been solved with several methods in [Mote, Olson, Venkataramanan, 1988]. A nonlinear value function was employed instead of human decision makers. The problem was solved by the GDF, the SWT and the ZW method, STEM, goal programming, and the method of Steuer. Only standard LP codes were utilized in the calculations. No superiority of any technique could be indicated. The methods had differences concerning the burden upon the decision maker and ad hoc and non ad hoc properties.

A characteristic in common with the evaluations that will be described in this paragraph is that they are based on intuition and insight rather than practical experiences and tests. A collection of features of five nonlinear interactive methods is presented in [Masud, Zheng, 1989]. The methods are compared with regard to eleven items, for example, certainty of obtaining a Pareto optimal solution, optimization technique used, type of information required from the decision maker, computational complexity compared to the GDF method, and the number of iterations needed with the decision maker compared to the GDF method. A similar table comparing the decision maker's burden, ease in actual use, effectiveness and handling of inconsistency has been gathered in [Shin, Ravindran, 1991] for ten methods. A classification and evaluation of methods according to 21 criteria is given in [Rietveld, 1980]. The number of items a decision maker has to assess simultaneously and per iteration for eight different methods in a medium size linear decision problem are tabulated in [Kok, 1985]. Kok concludes that the displaced ideal method, the interactive multiple goal programming method and STEM are most promising because their presumptions are realistic. A total of 19 interactive methods for MOLP problems have been listed according to three characteristics in [Larichev, Polyakov, Nikiforov, 1987]. The characteristics are the reliability of information elicitation from the decision maker, insignificant sensitivity to random decision maker's errors and good speed of convergence. The basic principles of the methods are also introduced. Features of STEM, the GDF, the ZW and the IWT method, the reference point method and the visual interactive approach, among others, have been tabulated in [Vanderpooten, Vincke, 1989] and [Vincke, 1992]. The criteria were, for instance, prior assumptions of a value function, applicability, trial and error support, mathematical convergence, the number of questions posed and the computational burden.

Finally, we mention some other comparative studies of the methods. Characteristic values in optimizing the multiobjective layout of a conical shell by the GDF method, STEM and three other methods are reported in [Eschenauer, Osyczka, Schäfer, 1990]. The comparison of the performances of the satisficing trade-off method and the IWT method when solving a linear sausage blending problem in [Olson, 1993] is mainly presented to emphasize the power of the weighted L_∞-metrics in multiobjective optimization. Several methods for quadratic multiobjective optimization problems are surveyed and compared in [Helbig, 1990(b)]. Some comparisons of continuous and discrete methods are presented in [Korhonen, Wallenius, 1989(b)].

5.2. Selecting a Method

Choosing an appropriate solution method for a certain multiobjective optimization problem is not easy, as has been assured. None of the existing methods can be labelled as the best for every situation, since there are a lot of aspects to consider and many of the comparison criteria are of somewhat fuzzy character. The features of the problem to be solved and the capabilities of the decision maker have to be charted before a solution method can be chosen. Some method may suit some problem and some decision maker better than the other.

Hobbs has written down in [Hobbs, 1986] some of the criteria to consider when methods are evaluated. The selection criteria are appropriateness, ease of use, validity and sensitivity of results to choosing the method. Appropriateness means that the method is appropriate to the problem to be solved, the people who will use it and the institutional setting in which it will be implemented. Ease of use means the effort and the knowledge required from the analyst and the decision maker. Validity means that the method measures what is is supposed to and the assumptions set are consistent with reality. Sensitivity of results to choosing the method means that solutions obtained by the method do not significantly differ from those of other methods. If the choice of method affects decisions significantly, then the relative validity of different methods should be considered. If the form of the method does not matter, then the most important criteria are the ease of use and appropriateness.

Stewart reduces the number of the critical criteria in selecting a solution method into three in [Stewart, 1992]. The input required from the decision maker must be meaningful and unequivocal, the method must be as transparent as possible and it must be simple and efficient.

The role of the decision maker must be considered very important. Many experiments have shown that decision makers prefer simpler methods because they can understand them better and they feel more in control. An important fact to keep in mind is that theoretically irrelevant aspects, such as question phasing, may affect the confidence that the decision maker feels in some method. The concept of the decision maker's confidence is analysed further in [Bischoff, 1986].

Other important criteria for the decision maker in selecting the solution method are, for example, the simplicity of the concepts involved, possibilities of interaction, easiness of interpreting the results and the chances of choosing the most preferred solution from a wide enough set of alternatives. The method must also fit the decision maker's way of thinking. The communication language between the decision maker and the method (system) must be understandable to the decision maker. (S)he wants also to see that the information (s)he provides has some (desirable) effect on the solutions obtained. One more element, not mentioned thus far, in the selection is how well the decision maker knows the problem to be solved. If one does not know its limitations, possibilities and potentialities well, one needs a method that supports the user in getting acquainted with the problem. In the opposite case, a method which makes it possible to directly focus on some interesting sector is advisable.

An attempt at aiding in the selection of a solution method is presented in [Gershon, Duckstein, 1983]. The selection problem is modelled as a multiobjective optimization problem. A set of 28 criteria for the selection are suggested and they are divided into four groups. Only the criteria in the last group have to be considered every time the selection algorithm is applied. The criteria take into account the characteristics of the problem, the decision maker and the methods. Many types of problems are taken into consideration in the criteria (e.g., discrete and continuous variables). The model contains 13 solution methods from which to select. The set of methods can naturally be modified. The amount of selection criteria can also be altered to include only such criteria that are relevant for the problem to be solved. Finally, after the methods have been evaluated according to the selection criteria, the resulting multiobjective optimization problem is solved by the method of global criterion (e.g., L_1-metric).

Different decision trees and taxonomies for providing assistance in selecting a method for multiattribute decision analysis problems are described in [Hwang, Yoon, 1981] and [Teghem, Delhaye, Kunsch, 1989]. However, it is criticized in [Ozernoy, 1992] that to design a covering and versatile decision tree usually results in an explosion of the number of the nodes. Another problem with decision tree diagrams is what to do when the user answers "I do not know".

An expert system for advising in the selection of solution methods for problems with discrete alternatives in proposed in [Jelassi, Ozernoy, 1989]. The reference mainly describes the development of the expert system. Steps in the development of another expert system for selecting the most appropriate method for discrete problems are described in [Ozernoy, 1992]. The questions posed by the system are based on if/then rules. They lead into recommending some method or stating that no method can be recommended. The user of the system can also always ask why some question is posed.

Few advice exists for selecting a method for nonlinear and continuous problems. Therefore, despite the above mentioned pitfalls and faults in decision tree diagrams, we still here present one in Figure 22. The tree is mostly based on plain facts about the assumptions set by the methods on the problem to be solved and preferences of the decision maker. Because of space limitations not all the properties have been able to be included. The tree contains eighteen methods described in Chapter 2. Only such methods have been included which have been presented in more detail or whose main features have been introduced.

The starting node is situated at the lower left corner. The tree diagram has been created in such a way that only the answers "yes" or "no" are available. Whenever the answer should be "I do not know", the answer "no" can be given. Of course, for example, when it is inquired whether the problem is an MOLP problem one can always answer "no" also for linear problems. Then more general methods, which do not utilize the linearity, are employed. The nodes containing only capital letters are reached when no method can be found along that path. Then one can try some other path. The endeavour has been that as many previous answers as possible are exploited. Thus, some dead ends may be avoided.

As repeated several times, selecting the solution method is a difficult and important task. In the subsection titled "concluding remarks" in connection with each method described here, we have tried to mention favouring and criticizing aspects. The matters are always more or less subjective.

Finally, we present in Table 23 a comparison of eleven interactive multiobjective methods described here. It can be regarded as a brief summary of the methods. However, one must always take up a sceptical attitude towards such attempts to compress matters to the utmost extreme. The table is subjective and there is no reason even to try to deny it.

Different problems arise when one tries to collect a table like this. Among them are, for example, which matter is important enough to be included, how it should be formulated, and whether the matter is positive or negative.

Table 23 contains some properties described when introducing the methods. They are related to the general features of the methods and their solutions. Properties concerning the assumptions set to the problem to be solved have not been included in the table. They have been handled in the decision tree in Figure 22.

The table is by no means self-contained. However, we do not explain the table here in detail but refer to the corresponding sections were the methods have been presented. For clarity, explanatory comments on some properties listed are in order.

The property "final solution Pareto optimal" has not been ticked in the columns corresponding to the reference point method and the satisficing trade-off method. The reason is that it is up to the form of the achievement function used whether the solutions are Pareto optimal or weakly Pareto optimal.

The property "implementation available" in the fourth row means such methods whose computer implementation has been available to the author. However, the implementations of the subgradient GDF method and the NIMBUS method have been done for testing purposes. An implementation of NIMBUS with special interest in the user interface is under development.

In the sixth row the question "gradient or subgradient of functions needed" refers to such methods, where the algorithm itself necessitates the availability of gradients

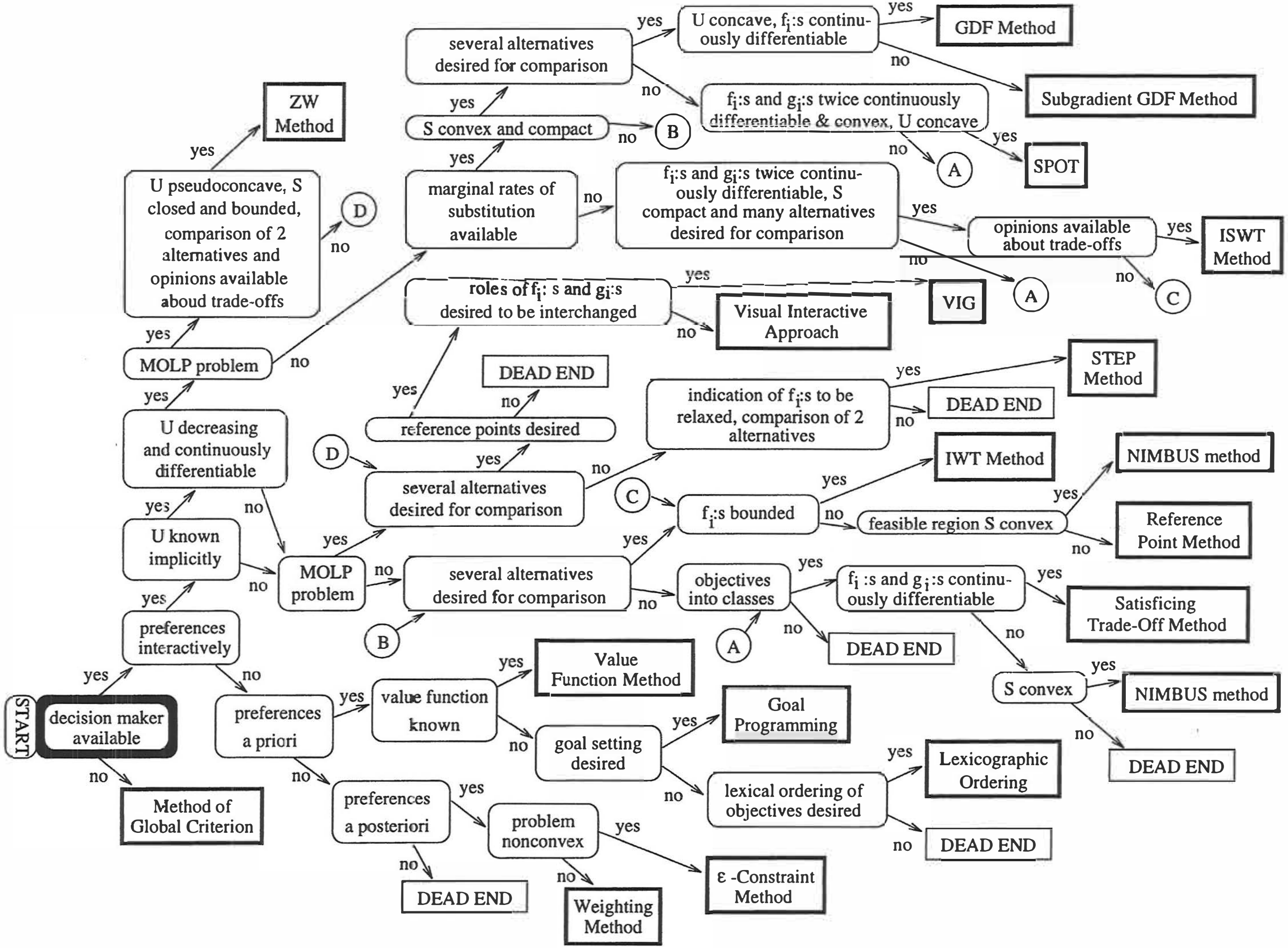

START
decision maker available
Method of Global Criterion
preferences interactively
preferences a priori
U known implicitly
U decreasing and continuously differentiable
MOLP problem
U pseudoconcave, S closed and bounded, comparison of 2 alternatives and opinions available aboud trade-offs
ZW Method
D
marginal rates of substitution available
S convex and compact
B
several alternatives desired for comparison
U concave, f_i:s continuously differentiable
GDF Method
Subgradient GDF Method
f_i:s and g_i:s twice continuously differentiable & convex, U concave
SPOT
A
f_i:s and g_i:s twice continuously differentiable, S compact and many alternatives desired for comparison
opinions available about trade-offs
ISWT Method
C
roles of f_i: s and g_i:s desired to be interchanged
VIG
Visual Interactive Approach
reference points desired
DEAD END
MOLP problem
several alternatives desired for comparison
indication of f_i:s to be relaxed, comparison of 2 alternatives
STEP Method
DEAD END
IWT Method
f_i:s bounded
feasible region S convex
NIMBUS method
Reference Point Method
several alternatives desired for comparison
objectives into classes
f_i :s and g_i :s continuously differentiable
Satisficing Trade-Off Method
DEAD END
S convex
NIMBUS method
DEAD END
value function known
Value Function Method
goal setting desired
Goal Programming
lexical ordering of objectives desired
Lexicographic Ordering
DEAD END
preferences a posteriori
DEAD END
problem nonconvex
ε -Constraint Method
Weighting Method
yes
no

Figure 22. Tree diagram.

or subgradients no matter which method is used to solve the resulting single objective optimization problem. Thus, if (sub)gradients for some reason cannot be calculated, such methods cannot be used. In connection with the ISWT and the satisficing trade-off method, the mark (x) means that the answer is positive or negative depending on the problem to be solved (whether the Kuhn-Tucker multipliers are all positive or not).

The feature "only for linear problems" means that the formulation of the method described in this presentation has been for MOLP problems. Other formulations for nonlinear problems may exist but they are not original.

The properties have been subjectively classified into positive, neutral and negative ones. The neutral properties are so much up to subjective opinions that they have had to be left unsigned. Some types of information required from the decision maker have been collected at the end of the table.

		ISWT method	GDF method	SPOT	ZW method	IWT method	STEP method	reference point method	satisficing trade-off method	VIG	subgradient GDF method	NIMBUS method
+	final solution Pareto optimal	×		×	×	×					×	×
+	final solution weakly Pareto optimal	×		×	×	×	×	×	×	×	×	×
+	suitable also for nondifferentiable problems					×		×			×	×
+	implementation available					×		×		×	×	×
~	ad hoc nature						×		×	×		×
~	(sub)gradients of functions needed	(×)	×						(×)		×	×
-	sensitive needing consistent answers	×	×	×	×	×	×				×	
-	computationally expensive					×						
-	only for linear problems				×		×			×		
-	difficult questions posed	×	×	×							×	
	only comparisons required				×	×						
	classification of objective functions required						×		×			×
	marginal rates of substitution required		×	×							×	

Table 23. Properties of interactive methods.

6. Results on Numerical Test Examples

In this chapter, we present results of some numerical experiments with the subgradient GDF and the NIMBUS method, described in Sections 2.22 and 2.23, respectively. The aim is to illustrate the possibilities and limitations of these methods with some examples of different nature.

The implementations have been written in Fortran 77 and the test runs have been performed in an HP9000/735 (99MHz) computer. The intermediate possibly nondifferentiable single objective optimization problems have been solved by the code PBNCBC, which is an implementation of the proximal bundle (PB) algorithm derived in [Mäkelä, Neittaanmäki, 1992]. PBNCBC calls the quadratic solver QPDF4 derived in [Kiwiel, 1986]. The implementation of the black-box routine MPB, which optimizes several objective functions at a time subject to nonlinear constraints, is called MPBNGC. It has been introduced in [Mäkelä, 1993]. MPBNGC is a genuine extension of PBNCBC. Here we briefly present some of the parameter values utilized. For a closer description of their role we cite the references mentioned above. The line search parameter has been set to RL=0.1, the upper bound for the size of the bundle has been set to JMAX=10 and the distance measure parameter has been set to GAM=0.0 or GAM=0.1.

The accuracy in the calculations has been 10^{-6} unless stated otherwise. For clarity, the solutions are here presented with much fewer decimals than in the actual solution processes. Whenever alternatives are presented to the decision maker, their number has been chosen to be $P = 5$.

One way to measure the efficiency of a solution process is to count the number of the function value calculations. For this reason, we count how many times the subroutine evaluating the objective function values is called. This number is called an SC (subroutine call) value and it will be associated with every solution process that is described here. Notice that the SC values of different methods cannot be compared objectively and absolutely. However, they can be used to get an impression of the general complexity of the methods.

As an exception to the practice followed so far in this presentation, we assume that all the vectors appearing are row vectors. Thus we avoid the need to transpose all the vectors and the readability of the text improves.

In the following, we solve two academic problems presented in the literature. The functions involved are differentiable. It is natural that the methods must also work in such cases. In the next chapter, we solve (nondifferentiable) optimal control problems describing the deflection of a string and the continuous casting of steel.

Before we continue it must be stressed that there has been no real decision maker involved in the solution processes. Thus the decisions and the selections are almost arbitrary. The intention has been to introduce the methods and to give some impression of the methods in general.

6.1. First Problem

The first problem to be solved has been presented in [Nakayama, Sawaragi, 1984] and [Nakayama, 1985(a)]. It contains three objective functions

$$\begin{array}{llll} & \text{minimize} & \{36.03-(8.05x_1+1.04x_2+24.0x_3), \\ (6.1.1) & & 287.58+2295.59(x_1-0.45)^2+404.46(x_2-0.45)^2, \\ & & 1050.73+10035.34(x_3-0.45)^2\} \\ & \text{subject to} & 0.45 \leq x_1,\ x_2,\ x_3 \leq 1.0. \end{array}$$

The problem describes a hypothetical water quality control problem modelling a river basin. The river has three upper streams, two of which have a treatment plant. They are financed by one local government. In addition, there is a treatment plant in the lower stream and it is financed by another local government. The aim is to minimize treatment costs and environmental pollution. The pollution rate is measured by a biological oxygen demand (BOD) concentration. The objective functions stand for the BOD concentration at the inflow point of the river basin into the sea, treatment costs in the upper reach and treatment costs in the lower reach, respectively. Three decision variables describe the percent of water treated in each of the three treatment plants.

The starting point has been arbitrarily selected to be $\mathbf{x}^1 = (0.8, 0.8, 0.8)$ and the corresponding criterion vector is $\mathbf{z}^1 = (9.558, 618.336, 2280.059)$. First, we solve the problem by the subgradient GDF method.

Subgradient GDF Method

In the solution process to be described here we employ the augmented weighted Tchebycheff function (2.22.3) with $\rho = 0.001$ to produce Pareto optimal solutions. Because the problem has box-constraints, it is bounded, and an estimate to the ranges of the Pareto optimal set can be obtained from the payoff table. The weighting coefficients utilized in the augmented weighted Tchebycheff function are thus the inverses of the differences between the components of the (approximated) nadir point and the ideal criterion vector. The first objective function has been selected to be the reference function.

At the starting point $\mathbf{z}^1$, the marginal rates of substitution are specified as $m_2^1 = 0.5$, $m_3^1 = 0.2$. Five Pareto optimal alternatives obtained by the method have been listed in Table 24.

	f_1	f_2	f_3
$\boxed{1}$	9.549	617.944	2278.602
2	12.058	540.017	1739.715
3	14.566	473.784	1354.382
4	17.078	419.466	1123.420
5	19.618	373.301	1050.730

Table 24.

The first alternative is selected to continue with because the BOD concentration at the inflow point (the first objective function) should not be increasing. We set $\mathbf{z}^2 = (9.549, 617.944, 2278.602)$. Thus the first iteration was used for learning. It was needed to see how small marginal rates of substitution are needed to obtain desired solutions. The user (decision maker) must realize the different scales of the objective function values.

In the second iteration, the marginal rates of substitution are set as $m_2^2 = 0.001$ and $m_3^2 = 0.001$. Again, five alternatives are presented for evaluation. They can be seen in Table 25.

	f_1	f_2	f_3
1	9.549	617.944	2278.602
2	8.194	650.604	2655.118
3	6.838	684.885	3081.874
$\boxed{4}$	5.482	720.786	3558.866
5	4.126	758.308	4086.102

Table 25.

Alternatives are now obtained from the right direction. The fourth alternative is selected for continuation, that is, $\mathbf{z}^3 = (5.482, 720.786, 3558.866)$. The BOD concentration is at that point at a tolerable level and, on the other hand, the treatment costs (the second and the third objective function) are too high in the fifth alternative.

Next, the marginal rates of substitution $m_2^3 = 0.0004$ and $m_3^3 = 0.001$ are specified. The intention is to get a more concentrated set of alternatives. They have been tabulated in Table 26.

	f_1	f_2	f_3
1	5.482	720.786	3558.866
$\boxed{2}$	5.128	733.541	3686.036
3	4.774	746.490	3816.345
4	4.421	759.634	3949.786
5	4.067	772.974	4086.385

Table 26.

From this selection the second alternative, $\mathbf{z}^4 = (5.128, 733.541, 3686.036)$, is picked, because, especially, the value of the third objective function is too high in the rest of the alternatives. The solution process could be stopped here because the solution obtained is quite satisfactory. In this case, the SC value would be 218.

Anyway, we present one more iteration and set the marginal rates of substitution $m_2^4 = 0.0012$ and $m_3^4 = 0.0014$. The alternatives obtained are shown in Table 27.

	f_1	f_2	f_3
1	5.128	733.541	3686.036
$\boxed{2}$	4.857	744.629	3783.477
3	4.586	755.859	3882.685
4	4.315	767.234	3983.656
5	4.044	778.752	4086.392

Table 27.

In the second alternative of Table 27, the BOD concentration has still been managed to decrease with treatment costs that can be tolerated. Thus it is selected as the final solution and $\mathbf{z}^5 = (4.857, 744.629, 3783.477)$. The corresponding decision variable vector is $\mathbf{x}^5 = (0.876, 0.764, 0.972)$. The SC value of this solution process with four iterations is 277 (with the augmented weighted Tchebycheff function).

If the penalty scalarizing function (2.22.4) is used (with the penalty term $\varrho = 1000.0$), the projection phase is very laborious and the alternatives differ remarkably from those presented above. The reason is that the penalty scalarizing function is too sensitive to the scales of the objective functions and it strives for minimizing the values of the second and the third objective function more willingly than the first objective function. Naturally, the achievable relative improvement in their values is larger. The result is that the decision maker cannot see the desired affects in the alternatives obtained. The penalty scalarizing function should be modified to better take into consideration the different scales of the objective functions. This is not a straightforward task to do. In addition, it is not here worth the trouble because the computational burden would still remain. To give an image of the burden we mention that if the problem is solved with the help of the same marginal rates of substitution as in the previously described solution process with four iterations, the SC value is 925!

If the black-box routine MPB is used to produce (weakly) Pareto optimal solutions, the alternatives obtained are naturally not exactly the same but fairly similar to those presented above. The same marginal rates of substitution and the same choices are valid. The first iteration proved out to cause some troubles. The accuracy in the calculations was relaxed to be 10^{-4}, and still a lot of iterations were needed. So, the SC value of the first iteration is 269. In the following iterations the difference in the decimals of the criterion values resulted in such alternatives where no projection was necessary. Thus, the SC value of the latter three iterations is in all 18.

As can be seen from the solution process described, the difficulty of the subgradient GDF method lies in specifying the marginal rates of substitution. In this example, the marginal rates of substitution were more or less arbitrary or occasional because there was no actual decision maker involved.

NIMBUS Method

In this section, we solve the problem (6.1.1) by the NIMBUS method. The starting point is still $\mathbf{x}^0 = (0.8, 0.8, 0.8)$ and the corresponding criterion vector is $\mathbf{z}^0 = (9.558, 618.336, 2280.059)$. We start by minimizing all the objective functions simultaneously. This guarantees that we can begin the actual solution process from a (weakly) Pareto optimal solution. We obtain $\mathbf{z}^1 = (9.543, 618.328, 2280.044)$. Notice the difference between $\mathbf{z}^1$ and the first alternative in Table 24, which is the projected one obtained by the augmented weighted Tchebycheff function. The difference in which direction the criterion values change to become Pareto optimal originates from different projection philosophies. In this case, the distinction is not remarkable because the starting point is situated so close to the Pareto optimal set.

At $\mathbf{z}^1$, the first two objective functions, namely, the BOD concentration and the treatment costs in the upper reach, are desired to be minimized, $I^< = \{1, 2\}$ without any special weighting, and an upper bound is given to the third, $I^> = \{3\}$ as $\varepsilon_3^1 = 3000.0$. The solution obtained is $\hat{\mathbf{z}}^1 = (7.375, 616.162, 3000.0)$. It is selected for continuation without considering any intermediate alternatives, so $\mathbf{z}^2 = \hat{\mathbf{z}}^1$.

It can be noticed that the treatment costs must be relaxed to decrease the BOD concentration. Thus $I^{<} = \{1\}$ and $I^{>} = \{2,3\}$ with $\varepsilon_2^2 = 800.0$ and $\varepsilon_3^2 = 4000.0$. Usually, it is advisable to specify loose enough upper bounds so that feasible solutions are more likely to be found. This causes no irreversible damage since one can always select some intermediate point between the previous and the new solution.

The solution obtained is $\hat{\mathbf{z}}^2 = (4.149, 800.0, 4000.0)$. Even though the BOD concentration is good, the treatment costs are after all too high at $\hat{\mathbf{z}}^2$. A set of five candidates has been listed in Table 28.

	f_1	f_2	f_3
1	7.375	616.162	3000.000
2	6.569	658.307	3230.658
3	5.762	702.995	3474.211
$\boxed{4}$	4.956	750.226	3730.658
5	4.149	800.000	4000.000

Table 28.

From this table the fourth alternative is selected. Some adjustment could naturally be done and the solution process could be continued, but we stop here with $\mathbf{z}^3 = (4.956, 750.226, 3730.658)$ as the final solution. The corresponding decision variable vector is $\mathbf{x}^3 = (0.879, 0.763, 0.967)$.

The SC value of this solution process is 550! The reason for this enormous number is the instability in minimizing all the objective functions before the solution process itself. As can be seen, the starting point and its (weakly) Pareto optimal counterpart are almost the same. So all this trouble was unnecessary. This kind of occurrence can be avoided by setting a maximum number for the function value evaluations at each projection phase. Another possibility is to relax the accuracy in the calculations. If only the actual iterations are considered, the SC value is 99. Making sure that the final solution is Pareto optimal adds the SC value by 14.

One must keep in mind that the SC values presented here are very relative. When the same problem is solved with a different accuracy or from a different starting point the SC value may vary considerably. If the accuracy in the computations above is reduced from 10^{-6} to 10^{-4}, then no projection takes place in the beginning and the SC value of the first projection decreases from 451 to 1.

This water quality control problem has been solved in [Nakayama, Sawaragi, 1984] by the satisficing trade-off method. To illustrate the fact that the NIMBUS method can be used like the methods of reference point-type, we solve the problem again. At this time, we set the components of the reference point specified in [Nakayama, Sawaragi, 1984] (in the continuation briefly referred as the paper) to be either aspiration levels or upper bounds as need arises. The original reference points of the paper are denoted by $\mathbf{a}$.

The starting point used in the original paper does not become apparent in the text. We select $\mathbf{x}^0 = (0.5, 0.5, 0.5)$ as a starting point instead of the starting point used earlier. The reason is that we want to demonstrate that NIMBUS can as easily find solutions that are situated farther from the starting point as those situated nearer.

The criterion vector corresponding to $\mathbf{x}^0$ is $\mathbf{z}^0 = (19.485, 294.330, 1075.818)$. When we select $I^{<} = \{1,2,3\}$, we obtain $\mathbf{z}^1 = (19.483, 294.328, 1075.816)$. The first reference point in the paper is $\mathbf{a}^1 = (5.0, 700.0, 3000.0)$. Thus we set $I^{\leq} = \{1\}$ with $\bar{z}_1^1 = 5.0$ and $I^{>} = \{2, 3\}$ with $\varepsilon_2^1 = 700.0$ and $\varepsilon_3^1 = 3000.0$. The resulting vector is $\hat{\mathbf{z}}^1 = (6.992, 700.0, 3000.0)$. It is selected for continuation, and $\mathbf{z}^2 = \hat{\mathbf{z}}^1$. (Naturally, the solutions obtained by the satisficing trade-off method and by NIMBUS are somewhat different because of their different solution philosophies.)

The second reference point in the paper is $\mathbf{a}^2 = (5.0, 800.0, 3500.0)$. The function classification is the following: $I^{\leq} = \{1\}$ with $\bar{z}_1^2 = 5.0$ and $I^{>} = \{2, 3\}$ with $\varepsilon_2^2 = 800.0$ and $\varepsilon_3^2 = 3500.0$. With this setting we get $\hat{\mathbf{z}}^2 = (5.303, 800.0, 3500.0)$ and $\mathbf{z}^3 = \hat{\mathbf{z}}^2$.

The BOD concentration is still too high and we must relax the aspirations of the treatment costs. For this reason, the third reference point in the paper is $\mathbf{a}^3 = (5.0, 800.0, 3700.0)$. Now we can use three classes of objectives, $I^{\leq} = \{1\}$ with $\bar{z}_1^3 = 5.0$, $I^{=} = \{2\}$ and $I^{>} = \{3\}$ with $\varepsilon_3^3 = 3700.0$. With this classification we obtain $\hat{\mathbf{z}}^3 = \mathbf{z}^4 = (5.0, 799.697, 3627.466)$, the corresponding decision vector being $\mathbf{x}^4 = (0.902, 0.779, 0.957)$. This is the final solution, because the BOD concentration has attained its aspiration level. The SC value of this solution process is 222. The final solution is Pareto optimal, and this checking means an increment in the SC value by 14 to 236.

Notice that because of the emptiness of the set $I^{<}$ the algorithm had to check the (weak) Pareto optimality of the solutions at each iteration. Every $\hat{\mathbf{z}}$ vector was (weakly) Pareto optimal, and for this reason this auxiliary step was not mentioned above. Anyway, it was performed at every iteration.

The final criterion vector obtained by the satisficing trade-off method was $(4.949, 791.83, 3662.2)$. Nothing is said in the paper about the computational complexity.

When examining the solution processes with the subgradient GDF method and the NIMBUS method and the amount of information asked from the decision maker, one may think that the subgradient GDF method is easier to use than NIMBUS. Only a couple of marginal rates of substitution are required per iteration, while NIMBUS needs the classification of the objective functions plus values for different parameters. Nevertheless, it is not always the amount of information that counts most but the nature of information. As has been stated already in Section 2.14, marginal rates of substitution are difficult to specify for most of the people. As to the NIMBUS method, its flexibility allows the user to decide how much information to supply. Eventually, it is the task of the user interface to make NIMBUS easy to use by supporting the user. A good user interface can be created by careful planning. No aid, however, can completely eliminate the difficulties in specifying marginal rates of substitution.

6.2. Second Problem

The second problem to be solved has been handled, for example, in [Chankong, Haimes, 1978] when introducing the ISWT method. The problem is the following

$$\text{(6.2.1)} \qquad \begin{aligned} &\text{minimize} && \{(x_1 - 3)^2 + (x_2 - 2)^2,\ x_1 + x_2,\ x_1 + 2x_2\} \\ &\text{subject to} && x_1,\ x_2 \geq 0. \end{aligned}$$

The value function has been given as $U(z_1, z_2, z_3) = 1 - \frac{z_1}{30} - \frac{z_2}{15} - \frac{z_3}{30}$ whenever $0 \leq z_1 \leq 10$, $0 \leq z_2 \leq 5$ and $0 \leq z_3 \leq 10$, and, otherwise, $U(z_1, z_2, z_3) = 0$. This is a nonincreasing function. Because of the setting of the value function, we can give upper bounds 5.0 for the decision variables. We have set the starting point to be the same as in [Chankong, Haimes, 1978], that is, $\mathbf{x}^1 = (2.75, 1.625)$, the corresponding criterion vector being $\mathbf{z}^1 = (0.203, 4.375, 6.0)$.

Subgradient GDF Method

Because of the linearity of the value function, the subgradient GDF (just as the original GDF) method converges in one iteration (because a linear approximation of a linear function is exact). Nevertheless, we present the solution process with the subgradient GDF method.

We have selected the first objective function as the reference function. Here we present five Pareto optimal alternatives obtained by using the augmented weighted Tchebycheff function in the projection. Because the problem is bounded, weighting coefficients have been obtained from the payoff table. The augmentation term is $\rho = 0.01$.

The marginal rates of substitution are easy to calculate from the value function ($m_i = \frac{d\,U(\mathbf{z})}{d\,z_i} / \frac{d\,U(\mathbf{z})}{d\,z_1}$). At the point $\mathbf{z}^1$ they are $m_2^1 = 2.0$ and $m_3^1 = 1.0$. As a matter of fact, the marginal rates of substitution are constant whenever $0 \leq z_1 \leq 10$, $0 \leq z_2 \leq 5$ and $0 \leq z_3 \leq 10$. Otherwise, they cannot be calculated. The set of five alternatives obtained can be seen in Table 29.

	f_1	f_2	f_3
1	0.203	4.375	6.000
2	0.927	3.656	4.875
3	2.176	2.937	3.750
4	3.950	2.219	2.625
$\boxed{5}$	6.250	1.500	1.500

Table 29.

In the fifth alternative, the value of the value function is maximal (i.e., $U = 0.641667$), and so $\mathbf{z}^2 = (6.25, 1.5, 1.5)$ is selected as the final solution. The solution process stops because no movements are desired. The corresponding decision variable vector is $\mathbf{x}^2 = (1.5, 0.0)$.

Essentially, subroutine calls result from the projections. All the alternatives happen to be Pareto optimal this time, so the projection phase would not be needed. Because of numerical instabilities, however, the SC value is 49.

If the penalty scalarizing function is used in the projection (with the penalty coefficient $\varrho = 1000.0$), then the alternatives are naturally somewhat different. An exception is the fifth alternative, which is of course the same and which is selected. The penalty term adds computational difficulty and instability. For this reason, the SC value is as much as 84.

On the other hand, if the black-box routine MPB is used, it notices directly that no projections are needed. Thus, it produces solutions most efficiently and the SC value is only 7.

NIMBUS Method

In this section, we solve the problem (6.2.1) by the NIMBUS method from the same starting point $\mathbf{x}^0 = (2.75, 1.625)$ and the corresponding criterion vector $\mathbf{z}^0 = (0.203, 4.375, 6.0)$ as above. We start by minimizing all the objective functions simultaneously (i.e., $I^< = \{1, 2, 3\}$). This time no improvement is possible because the starting point is already Pareto optimal. Therefore, we set $\mathbf{z}^1 = \mathbf{z}^0$.

We continue by minimizing the second and the third objective function because of their large values, $I^{<} = \{2, 3\}$. Weighting coefficients are attached to them to give different stresses with respect to their ranges, so $w_2^1 = 0.3$ and $w_3^1 = 0.7$. Reciprocally, something must be allowed to increase. There is no other alternative than to relax the first objective function, $I^{>} = \{1\}$, and we set $\varepsilon_1^1 = 5.2$. The resulting criterion vector is $\hat{\mathbf{z}}^1 = (5.2, 1.807, 1.984)$, which is accepted for continuation ($\mathbf{z}^2 = \hat{\mathbf{z}}^1$).

Aspiration functions are used in the following iteration, that is, $I^{\leq} = \{2, 3\}$ with $\bar{z}_2^2 = 1.4$ and $\bar{z}_3^2 = 1.4$ without any special weighting coefficients. The first objective function must again be relaxed and $I^{>} = \{1\}$ with $\varepsilon_1^2 = 6.7$. This setting produces $\hat{\mathbf{z}}^2 = (6.603, 1.387, 1.387)$, which is not acceptable. In NIMBUS, it is typical for aspiration functions that the solution process does not necessarily stop at the aspiration value. For this reason, it is good to have a look at the five alternatives shown in Table 30.

	f_1	f_2	f_3
1	5.200	1.807	1.984
2	5.534	1.702	1.835
3	5.879	1.597	1.685
$\boxed{4}$	6.235	1.492	1.536
5	6.603	1.387	1.387

Table 30.

Now we can employ the value function available to select the most preferred candidate from the table. The value function has the largest value in the fourth alternative ($U = 0.64151$). Thus, $\mathbf{z}^3 = (6.235, 1.492, 1.536)$ is the one that is selected.

Finally, some refinement is attempted by setting $I^{\leq} = \{3\}$, $\bar{z}_3^3 = 1.49$, $I^{=} = \{2\}$ and $I^{>} = \{1\}$ with $\varepsilon_1^3 = 6.4$. This produces a point $\hat{\mathbf{z}}^3 = (6.397, 1.446, 1.467)$. Five alternatives between $\mathbf{z}^3$ and $\hat{\mathbf{z}}^3$ can be seen in Table 31.

	f_1	f_2	f_3
$\boxed{1}$	6.235	1.492	1.536
2	6.275	1.480	1.519
3	6.316	1.469	1.502
4	6.356	1.457	1.484
5	6.397	1.446	1.467

Table 31.

No improvement can be found in this direction, and the first alternative is selected as the final solution, $\mathbf{z}^4 = (6.235, 1.492, 1.536)$. The corresponding decision variable vector is $\mathbf{x}^4 = (1.448, 0.043)$. Of course, we could continue and try some other classification but we stop with the current solution. As to the efficiency of the NIMBUS method, very little calculation is needed. The SC value of this solution process is 45. The final solution is Pareto optimal. The checking increases the SC value to 68.

Notice that all the value function calculations have been carried out with more accurate criterion vectors than shown here.

When the same problem (6.2.1) was solved by the ISWT method in [Chankong, Haimes, 1978], much less satisfactory results were obtained. Even though the explicit value function was employed, there are always some uncertainties involved with the surrogate worth values. For this reason, 22 iterations were needed to produce a satisfactory solution (with U=0.64053). Thus, a great number of function value calculations could not have been avoided.

The test results of this problem give a more or less unrealistic image of the potentialities of the methods. Most misleading is the case with the subgradient GDF method. The marginal rates of substitution are "too easy" to determine from an explicit value function. While, in addition, the value function happened to be linear, only one iteration was needed. When the value function is nothing more than just a set of preference relations in the mind of the decision maker, it is very difficult to express exact numerical values. However, the correctness of the marginal rates of substitution is crucial for the convergence of the method.

The case is different with the NIMBUS method. Even though we had the explicit value function at our disposal, it did not directly reveal exactly which objective function values should be decreased, which ones relaxed, and how much. The value function could be exploited in the selection phase when comparing the alternatives. The NIMBUS method is not based so strictly on the idea of maximizing the underlying value function but on exploring the potentialities of the problem and a sort of aspiration-based satisficing decision making.

Notice that as well as knowing the explicit value function makes the subgradient GDF method converge fast, the NIMBUS method can also be stopped after one iteration if the appropriate upper bounds and aspiration levels are known. However, neither the explicit value function nor the exact aspiration levels and upper bounds are realistic to be assumed in practical problems.

After introducing some basic properties of the subgradient GDF method and the NIMBUS method with simple examples, we can continue with more complex problems. In the following, there are also nondifferentiable functions involved.

7. Applications to Optimal Control Problems

In the following, we present new ways to solve certain state-constrained complex optimal control problems that have been widely solved and treated in different connections at the University of Jyväskylä. They are of nondifferentiable and multiobjective nature.

Originally, they have been solved by first scalarizing the objective functions by the weighting method and then regularizing the nondifferentiabilities into a differentiable form. After discretization and employing the finite element method, the problems have been able to be solved by traditional, differentiable single objective optimization methods. However, both scalarization and regularization simplify the problem and cause errors.

Later, the regularization has been given up by employing nondifferentiable analysis, but the weighting method has still been utilized (see, e.g., [Haslinger, Neittaanmäki, 1988], [Laitinen, 1989], [Mäkelä, 1990], [Mäkelä, Neittaanmäki, 1992] and [Neittaanmäki, Tiba, 1994]). The weighting method has been used in the a priori form. Some weighting coefficients have been specified, and one has had to be content with the solution obtained. However, the relative importances of the objective functions are not usually known in advance and the weighting method is artificial. As some of the functions originate from technological constraints, the method may bring about inaccuracies and the solution may be irrelevant in a technological sense. For this reason, interactive methods are recommendable. Then the decision maker is involved in the solution process and can revise preferences, if necessary.

We handle a model of an elastic string and the continuous casting process of steel. We formulate the problems to be solved and apply the subgradient GDF method and the NIMBUS method to solve them. Because the derivation and the numerical treatment of the problems into implementable forms have been treated in several contexts, we do not repeat them here. Instead, we settle for introducing the main problems briefly and indicating appropriate references.

7.1. Elastic String

The problem to be handled describes an elastic string which is deflected by some vertical force. The force is bounded from above and the deflection of the string is limited from below by some rigid obstacle. The aim is to maximize the contact area between the string and the obstacle with minimal total force. Thus, we have two conflicting objective functions. The general setting of the problem has been illustrated in Figure 32.

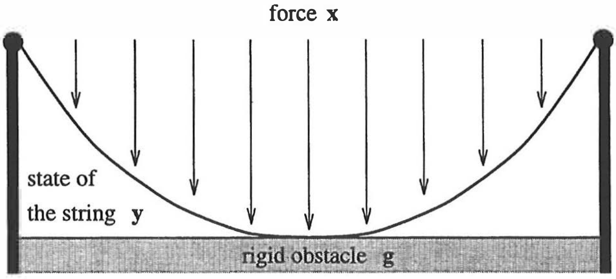


Figure 32. Elastic string deflected by a force towards a rigid obstacle.

The formulation of the problem involves variational inequalities and employing the finite element method (FEM). The solution of the problem by the subgradient GDF method has been described in [Miettinen, Mäkelä, 1991, 1993].

Setting of the Problem

Let $\Omega = (0, 10) \subset \mathbf{R}$ be the domain considered and let $\mathbf{y}$ be the deflection of the string *(state variable)* belonging to the Sobolev space $H_0^1(\Omega)$. The string is deflected by the force $\mathbf{x}$ *(control variable)* belonging to the space of the square integrable functions $L^2(\Omega)$. The string cannot overpass the obstacle $\mathbf{g}$, which belongs to the space $H^1(\Omega)$ such that $\mathbf{g}(0) \leq 0$ and $\mathbf{g}(10) \leq 0$. The set of admissible forces (controls) is defined by

$$\mathbf{S} = \{\mathbf{x} \in L^2(\Omega) \mid 0 \leq \mathbf{x}(s) \leq \beta \text{ almost everywhere in } \Omega\}.$$

We denote by $\langle \cdot, \cdot \rangle_{0,\Omega}$ the usual inner product on $L^2(\Omega)$. The dependence between a given force $\mathbf{x} \in \mathbf{S}$ and the deflection $\mathbf{y}(\mathbf{x}) \in \mathbf{K}$ can be described by the following variational inequality

$$\text{(7.1.1)} \qquad \langle \mathbf{y}(\mathbf{x})', \mathbf{v}' - \mathbf{y}(\mathbf{x})' \rangle_{0,\Omega} \geq \langle \mathbf{x}, \mathbf{v} - \mathbf{y}(\mathbf{x}) \rangle_{0,\Omega}$$

for all $\mathbf{v} \in \mathbf{K}$, where

$$\mathbf{K} = \{\mathbf{v} \in H_0^1(\Omega) \mid \mathbf{v} \geq \mathbf{g} \text{ almost everywhere in } \Omega\}.$$

The first objective function to be minimized is the utilized total force

$$f_1(\mathbf{x}) = \frac{1}{2} \int_\Omega \mathbf{x}^2 \, ds.$$

The second objective function is maximizing the contact area. It has been modelled as minimizing the difference between the string and the obstacle

$$f_2(\mathbf{x}) = \int_\Omega (\mathbf{y}(\mathbf{x}) - \mathbf{g}) \, ds.$$

Obviously, the objective function f_1 is differentiable but the function f_2 is nonconvex and nondifferentiable. Thus the multiobjective optimization problem to be solved is nondifferentiable.

The state problem (7.1.1) has a unique solution, and thus the optimal control problem has at least one optimal solution. For details of the existence and the uniqueness results, see [Haslinger, Neittaanmäki, 1988].

The variational formulation is discretized by linear finite elements for approximating $\mathbf{y}$ and $\mathbf{x}$. For details of this finite element (FE) handling and calculating the subgradients, see [Mäkelä, 1990] and [Miettinen, Mäkelä, 1993]. In addition to the subroutines mentioned at the beginning of Chapter 6, Powell's quadratic solver ZQPCVX is needed in the numerical implementation.

We have chosen the value $n = 40$ for the discretization parameter (number of variables) and $g = (-1.0, \ldots, -1.0)$ for the discrete approximation of the rigid obstacle to be used. The bounds for the decision variables are $0.0 \leq x_i \leq 20.0$ $(i = 1, \ldots, n)$ (see [Miettinen, Mäkelä, 1991]). The starting point has been arbitrarily chosen to be $\mathbf{x}^1 = (1.0, \ldots, 1.0)$.

Subgradient GDF Method

As in the examples of the previous chapter, we here present one solution process by the subgradient GDF method where the augmented weighted Tchebycheff function is used to produce Pareto optimal solutions (with the augmentation term $\rho = 0.0001$ and the weighting coefficients from the payoff table). The corresponding solution processes, employing the penalty scalarizing function and the black-box routine MPB, are described only briefly. The first objective function has been selected as the reference function.

The criterion vector corresponding to the starting point $\mathbf{x}^1 = (1.0, \dots, 1.0)$ is $\mathbf{z}^1 = (5.0, 0.953)$. The first marginal rate of substitution is set to be $m_2^1 = 3.0$. It produces the alternatives listed in Table 33.

	f_1	f_2
1	2.548	0.766
2	1.610	0.884
3	1.077	1.004
$\boxed{4}$	0.693	1.159
5	0.019	4.376

Table 33.

The trend in the alternatives is convenient, because the force utilized (f_1) was far too large in the beginning. However, in the fifth alternative, the string is too little in contact, and so the fourth alternative is selected for continuation.

At the point $\mathbf{z}^2 = (0.693, 1.159)$, the marginal rate of substitution is set as $m_2^2 = 0.5$. The set of five alternatives obtained can be seen in Table 34.

	f_1	f_2
1	0.693	1.159
$\boxed{2}$	0.298	1.529
3	0.140	1.957
4	0.077	2.387
5	0.047	2.811

Table 34.

The second alternative is selected from this table, because the total force is quite small. In the rest of the alternatives, the string (f_2) looses too much contact.

At the point $\mathbf{z}^3 = (0.298, 1.529)$, the marginal rate of substitution is set again as $m_2^3 = 0.5$. The resulting five candidates have been tabulated in Table 35.

In the second alternative, the total force (f_1) has been managed to decrease a little and the contact area does not diminish too much. Thus it is the point to be selected. The next iterations are only a fine adjustment. The solution $\mathbf{z}^4 = (0.252, 1.613)$ could as well be selected as the final solution.

To illustrate how almost identical marginal rates of substitution may produce alternatives from different directions, we proceed from here by setting the marginal rate

	f_1	f_2
1	0.298	1.529
$\boxed{2}$	0.252	1.613
3	0.135	1.984
4	0.062	2.566
5	0.033	3.151

Table 35.

	f_1	f_2
$\boxed{1}$	0.252	1.613
2	0.291	1.540
3	0.326	1.483
4	0.363	1.431
5	0.396	1.392

Table 36.

of substitution at the point $\mathbf{z}^4$ as $m_2^4 = 0.6$. This gives alternatives presented in Table 36.

As can be seen, the alternatives move into the direction where the total force and the contact area increase. This is not what was desired, and thus the first alternative is selected. We set $\mathbf{z}^5 = \mathbf{z}^4$. When we set $m_2^5 = 0.4$ (instead of 0.6), we get the alternatives of Table 37 for evaluation.

	f_1	f_2
1	0.252	1.613
2	0.240	1.640
$\boxed{3}$	0.202	1.736
4	0.148	1.923
5	0.112	2.112

Table 37.

We stop the solution process by selecting the third solution $\mathbf{z}^6 = (0.202, 1.736)$ as the final one. The total force employed and the deflection of the string at the original starting point and at the final solution have been illustrated in Figures 38 and 39, respectively. Notice that the starting point is not Pareto optimal.

The SC value of the above-described solution process with five iterations is 1628. This means that 76 subroutine calls are needed on the average for projecting one criterion vector onto the Pareto optimal set. If the last two iterations are abandoned, then the SC value of the first three iterations is in all 1023.

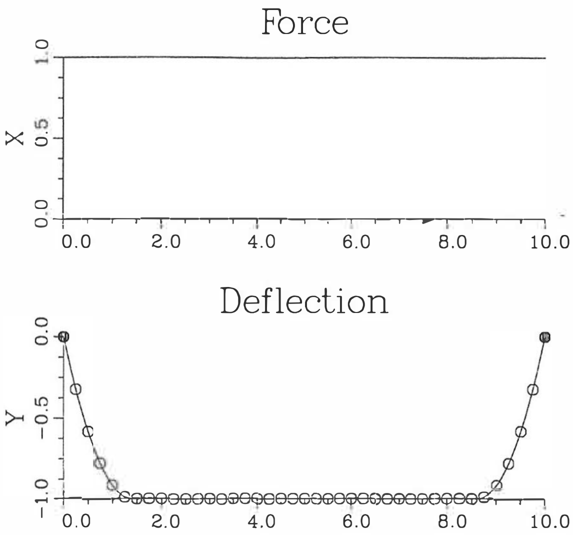


Figure 38. Force and deflection at the starting point.

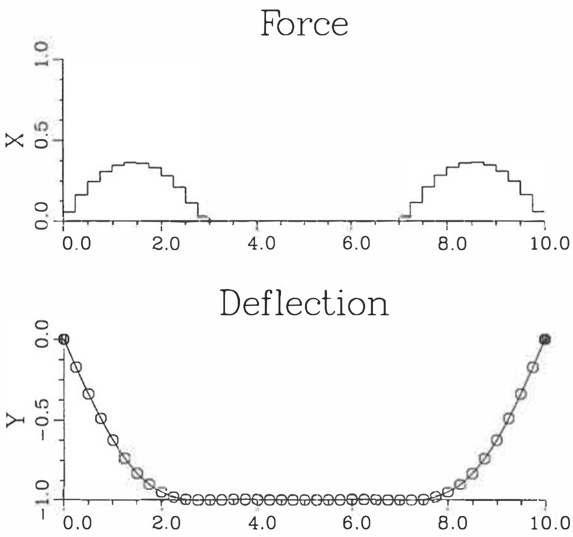


Figure 39. Final force and deflection with the subgradient GDF method.

If the penalty scalarizing function is used in the projection phase, the alternatives obtained are remarkably different from those presented above. If we, despite this fact, repeat the solution process with the same choices and marginal rates of substitution, the corresponding SC value is 2718. If we choose the best alternative at each iteration for five iterations, the SC value is 1629. (As a curiosity, we can state that the above-used marginal rates of substitution are usable almost as such and only the number of the selected alternative is different.) From this solution process we conclude that the SC value can vary greatly depending on the alternatives.

As to employing the black-box routine MPB, we obtain a significantly smaller SC value than with the other projection types. The SC value for five iterations is only 665. Thus, producing one alternative onto the (weakly) Pareto optimal set only takes

31 subroutine calls on the average. The alternatives obtained by the MPB routine and the augmented weighted Tchebycheff function are within limits so much alike that the same marginal rates of substitution and selections are valid.

We do not here present the total force employed and the deflection of the string at the final solutions obtained with the penalty scalarizing function and the MPB routine because of their similarity to the solution in Figure 39. Instead, we have a look at the difference between the projection methods.

To illustrate how different solutions are obtained with the augmented weighted Tchebycheff function, the penalty scalarizing function and the MPB routine, we consider the starting point $\mathbf{x}^1 = (1.0, \dots, 1.0)$. The original force employed and the deflection can be seen in Figure 38. In the following, we show what the starting point looks like when it has been projected onto the (weakly) Pareto optimal set with those three methods.

The Pareto optimal counterpart of $\mathbf{x}^1$ obtained by the augmented weighted Tchebycheff method can be seen in Figure 40. The corresponding criterion vector is (2.548, 0.766). It is the first alternative in Table 33.

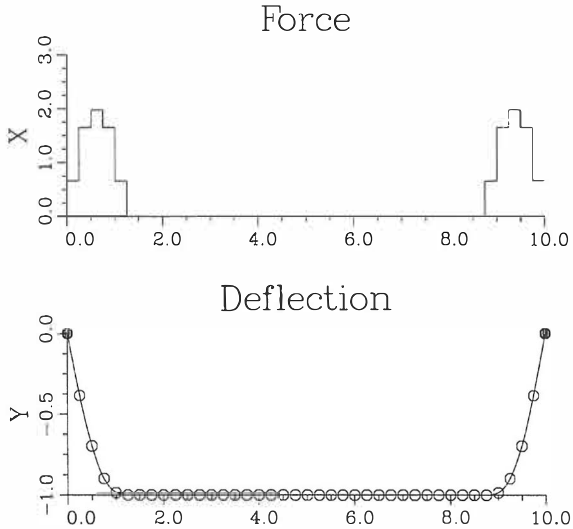


Figure 40. Starting point by the augmented weighted Tchebycheff function.

In Figure 41, the Pareto optimal analogue of the starting point produced by the penalty scalarizing function is illustrated. The corresponding criterion vector is $(1.273, 0.953)$. In this figure, the total force needed and the deflection of the string are quite different from the previous figure.

Finally, a third state of the string is depicted in Figure 42. It has been obtained from the starting point by the MPB routine. The corresponding criterion vector is $(4.821, 0.631)$.

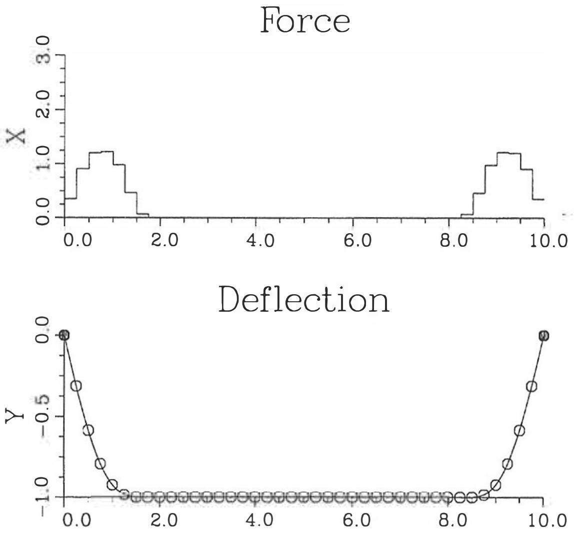


Figure 41. Starting point by the penalty scalarizing function.

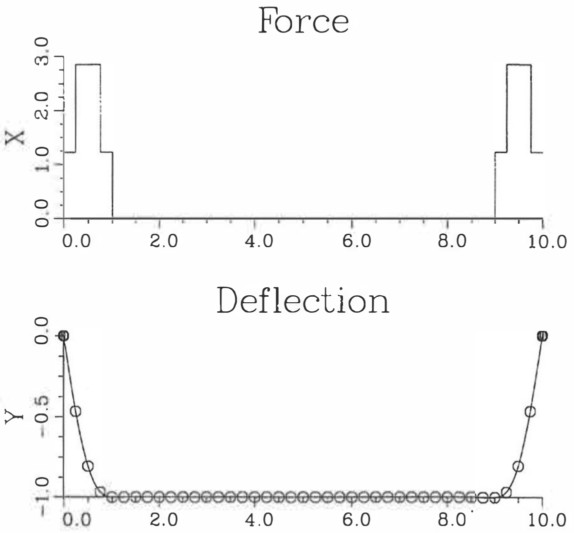


Figure 42. Starting point by the MPB routine.

These figures prove that even though producing Pareto optimal solutions is quite laborious and many function value calculations are needed, it is, however, worthwhile. As can be seen above, the original solution is considerably worse than the Pareto optimal counterparts. Another matter is the difference of the Pareto optimal solutions obtained by different functions.

NIMBUS Method

We start solving the string problem by the NIMBUS method from the same starting point $\mathbf{x}^0 = (1.0, \dots, 1.0)$. The corresponding criterion vector is still $\mathbf{z}^0 = (5.0, 0.953)$. When we minimize both of the objective functions by setting $I^< = \{1, 2\}$, we obtain

$\mathbf{z}^1 = (4.821, 0.631)$. This solution is the same as depicted in Figure 42, because the MPB routine has been used to produce both of them.

The total force must be decreased in $\mathbf{z}^1$. Therefore, we want to minimize the first objective function and we must relax the second objective function. If the upper bound is here set, for example, as 1.0, the solution process needs remarkably many inner iterations (and function value calculations) to produce a solution. This may sometimes happen because of some numerical instabilities and the structure of the problem. Such bottlenecks can usually be best overcome by loosening the upper bound.

Because the user does not necessarily know the potentialities of the problem, (s)he can decide to minimize one objective function and see what happens to the other ones. Likewise, we set here $I^{<} = \{1\}$ and $I^{\diamond} = \{2\}$. This produces the solution $\hat{\mathbf{z}}^1 = (0.0, 10.0)$. Naturally, this is too extreme, and five alternatives are in order. They have been listed in Table 43.

	f_1	f_2
1	4.821	0.631
$\boxed{2}$	1.200	0.971
3	0.026	3.472
4	0.005	7.003
5	0.000	10.00

Table 43.

The second alternative is selected from this set, because the string (f_2) looses too much contact in the latter alternatives. We set $\mathbf{z}^2 = (1.2, 0.971)$. Still the total force (f_1) is too big and it must be decreased.

We set the first objective function as an aspiration function $I^{\leq} = \{1\}$ with $\bar{z}_1^2 = 0.5$. The second objective function must then be relaxed, $I^{>} = \{2\}$ with $\varepsilon_2^2 = 1.5$. This settlement produces $\hat{\mathbf{z}}^2 = (0.481, 1.351)$. This seems to be a good solution, and we accept it for continuation. Notice that at this iteration the set of functions to be minimized was empty. Thus the (weak) Pareto optimality of the solution is not guaranteed. For this reason, the algorithm minimizes both of the objective functions further simultaneously, and we obtain $\mathbf{z}^3 = (0.458, 1.326)$ to continue from.

At this point, it seems that it would still be desirable to decrease the total force, and we set $I^{<} = \{1\}$ and $I^{>} = \{2\}$ with $\varepsilon_2^3 = 2.0$. We get $\hat{\mathbf{z}}^3 = (0.131, 2.0)$. Five candidates of this iteration can be seen in Table 44.

	f_1	f_2
1	0.458	1.326
2	0.333	1.472
$\boxed{3}$	0.240	1.640
4	0.175	1.820
5	0.131	2.000

Table 44.

The third alternative is the best, because the total force (f_1) is quite small but the contact (f_2) is still pretty good. Therefore, we continue from $\mathbf{z}^4 = (0.24, 1.64)$. To tune the solution a little bit we continue for one more iteration.

We allow only slight movements, setting $I^{\leq} = \{1\}$ with $\bar{z}_1^4 = 0.2$ and $I^{>} = \{2\}$ with $\varepsilon_2^4 = 1.75$. The solution obtained is $\hat{\mathbf{z}}^4 = (0.2, 1.745)$. Five alternatives have been tabulated in Table 45.

	f_1	f_2
1	0.240	1.640
2	0.229	1.665
3	0.219	1.691
$\boxed{4}$	0.209	1.717
5	0.199	1.745

Table 45.

We stop here, and the fourth alternative $\mathbf{z}^5 = (0.209, 1.717)$ is selected as the final solution. The SC value of the above-described solution process is 447. The final solution is Pareto optimal. This additional testing adds the total SC value by 10 to be 457. It is remarkably smaller than any of the SC values reported in connection with the subgradient GDF method. The final solution obtained by the NIMBUS method can be seen in Figure 46. It is so close to the solution obtained by the subgradient GDF method, presented in Figure 39, that one can hardly tell the difference.

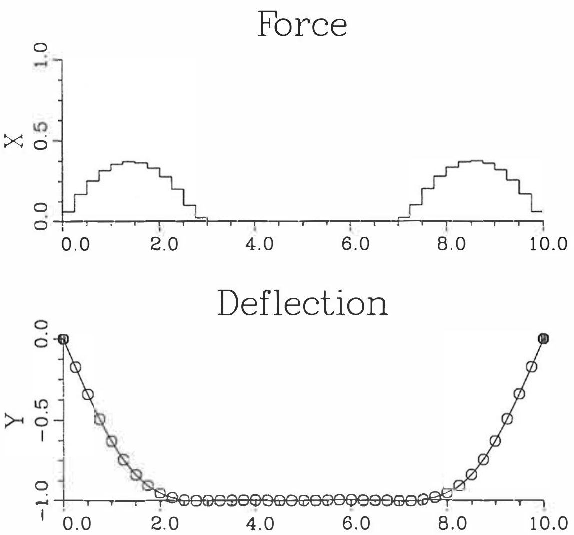


Figure 46. Force and deflection at the final solution with the NIMBUS method.

The direct comparison between the results obtained here and those obtained earlier (e.g., in [Mäkelä, 1990]), when multiobjective capabilities were not utilized, is quite difficult. The reason is that the problem settings are different. A more complicated discretized model, which describes the original problem better, has been able to be solved here. Thus, the results are in that sense more satisfactory.

7.2. Continuous Casting of Steel

The second problem of optimal control is, both mathematically and computationally, substantially more complicated than the previous one. The model described is used to simulate the continuous casting process of steel. Here we present only the main parts of the problem. A more detailed description can be found in [Mäkelä, Männikkö, 1992, to appear]. As a multiobjective optimization problem, the continuous casting problem has been handled in [Miettinen, Mäkelä, 1994].

Processing steel by continuous casting has increased its popularity in industry during the last few years. Thus, a considerable amount of steel is manufactured in the world by the continuous casting. Numerical simulation models have been developed to avoid expensive and timeconsuming full-scale tests on production machines. The model to be presented here is used for controlling cooling water sprays.

The main parts of the casting process are depicted in Figure 47. Molten steel is poured from a tundish into a water cooled mould. After the molten steel has formed a solid shell, the steel strand is drawn down with a constant speed (the solid shell surrounding the liquid pool). After the mould (from the point p^1 on), the strand is supported by rollers and cooled down by water sprays until the steel has solidified completely. Our simulation model has been intended for controlling this cooling process by adjusting the intensity of the water sprays. The water sprays end at the point p^2, and the maximum length of the liquid pool is p^3. The strand is straightened at the point p^4 and cut up at the point p^5.

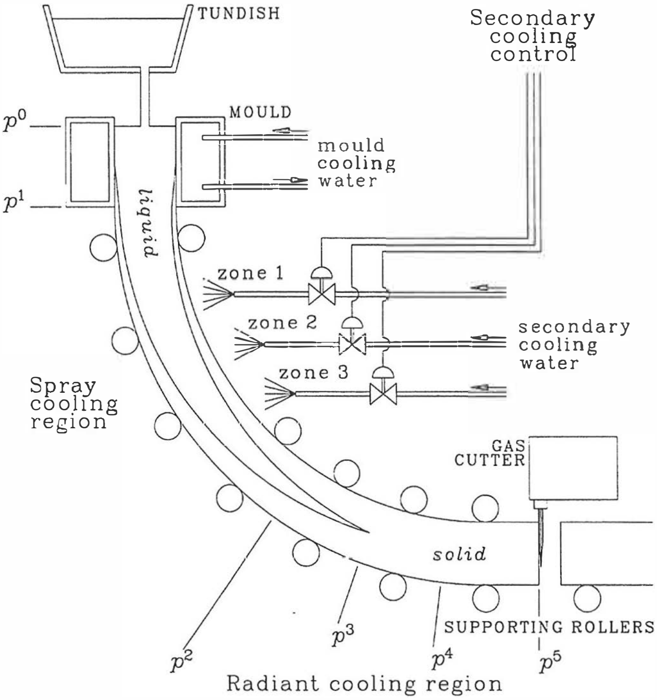


Figure 47. Schematic representation of the continuous casting process.

As to the quality of the product, it is important that both overcooling and undercooling are avoided. Thus, the surface temperature must transform smoothly as the steel proceeds between the cooling regions. The main purpose of the problem is to minimize the defects in the final products.

Setting of the Problem

Because it is difficult to measure the surface temperature directly, it is better to calculate the temperature distribution numerically. Notice that the point where the steel does not contain any molten parts depends on the adjustment of the cooling and is thus not known in advance. In addition, there exists a so-called mushy zone where the steel is neither solid nor liquid.

The temperature distribution of the strand is calculated by solving a nonlinear heat transfer equation with free boundaries between solid and liquid phases. Due to the piecewise linear approximation of nonlinear terms, the objective functions are nondifferentiable.

We here handle the casting process from the point p^1 till the point p^4. The control variable $\mathbf{x}$ represents a heat transfer coefficient, which has an effect on the temperature distribution $\mathbf{y}(\mathbf{x})$ (the state) of the steel strand.

Let $\Omega \subset \mathbf{R}^2$ denote the cross-section of the strand (e.g., a square or a circle) and Γ its boundary. Time variables are now denoted by t. Let t_i $(i = 1, \ldots, 4)$ be the time events when Ω passes the points p^i. Thus, we consider the time period between t_1 and t_4 $(t_1 = 0$ and $t_4 = T)$. Moreover, we denote $Q = (0,T) \times \Omega$, $\Sigma = (0,T) \times \Gamma$, $Q_1 = (t_3, T) \times \Omega$ and $\Sigma_1 = (0, t_2) \times \Gamma$. The set of admissible control variables is of the form

$$S = \{\mathbf{x} \in L^2(\Sigma) \mid 0 < \alpha(t) \leq \mathbf{x}(t,s) \leq \beta(t),\ t \in (0,T),\ s \in \Gamma\},$$

where $L^2(\Sigma)$ denotes square integrable functions on Σ. For any point $\mathbf{x} \in S$, the temperature distribution $\mathbf{y} = \mathbf{y}(\mathbf{x})$ is obtained by solving the state system

$$\begin{aligned}
&\frac{d}{dt}H(\mathbf{y}) - \Delta K(\mathbf{y}) = 0 && \text{in } Q \\
(7.2.1) \qquad &\frac{d}{dn}K(\mathbf{y}) = \begin{cases} \mathbf{x} \cdot (\mathbf{y}^{wat} - \mathbf{y}) + c \cdot ((\mathbf{y}^{ext})^4 - \mathbf{y}^4) & \text{on } \Sigma_1 \\ c \cdot ((\mathbf{y}^{ext})^4 - \mathbf{y}^4) & \text{on } \Sigma \setminus \Sigma_1 \end{cases} \\
&\mathbf{y}(0,s) = \mathbf{y}^0(s) && s \in \Omega,
\end{aligned}$$

where Δ denotes the two-dimensional Laplace operator, $\frac{d}{dn}$ denotes the normal derivative, the enthalpy function H and Kirchhoff's transformation K are piecewise linear functions, the constants $\mathbf{y}^0$, $\mathbf{y}^{wat}$ and $\mathbf{y}^{ext}$ denote the initial, the spray water and the surrounding environment temperature, respectively, and c is a physical constant. Notice that between the points p^1 and p^2 the steel is cooled down by both the water sprays and normal radiation. After p^2 only the radiation is left. We assume that we know the temperature distribution at the end of the mould, $\mathbf{y}^0$. From there we obtain the initial condition.

Next, we formulate the first objective function. We assume that we have some technologically desirable temperature distribution $\mathbf{y}^d = \mathbf{y}^d(t,s)$ on the boundary of the strand. Naturally, we want the actual surface temperature to be as close to $\mathbf{y}^d$ as possible. Thus we have the objective function

$$f_1(\mathbf{x}) = \varepsilon_1 \int_0^T \tfrac{1}{2}\|\mathbf{y}(\mathbf{x}) - \mathbf{y}^d\|_{0,\Gamma}^2 \, dt.$$

In addition, we have the following technological constraints

$$\begin{aligned} &\mathbf{y}^{min} \leq \mathbf{y}(\mathbf{x}) \leq \mathbf{y}^{max} && \text{on } \Sigma \\ &\mathbf{y}^{min'} \leq \frac{d}{dt}\mathbf{y}(\mathbf{x}) \leq \mathbf{y}^{max'} && \text{on } \Sigma \\ &0 \leq \mathbf{y}(\mathbf{x}) \leq \mathbf{y}^{sol} && \text{in } Q_1 \\ &\mathbf{y}^{duc} \leq \mathbf{y}(\cdot, T; \mathbf{x}) \leq \mathbf{y}^{sol} && \text{in } \Omega, \end{aligned} \tag{7.2.2}$$

where the constants $\mathbf{y}^{min}$, $\mathbf{y}^{max}$, $\mathbf{y}^{min'}$, $\mathbf{y}^{max'}$, $\mathbf{y}^{sol}$ and $\mathbf{y}^{duc}$ denote some minimum and maximum bounds (also for derivatives), the solidus and the ductility temperatures, respectively. The state constraints in (7.2.2) have been defined in this approach to be so tight that the feasible region is empty. For this reason, the constraints in (7.2.2) are translated into objective functions $f_2, \ldots, f_5$, respectively, using projection operators and exact penalty functions. Thus, our aim is to find thc solution violating the "constraints" as little as possible. For details see [Mäkelä, Männikkö, 1992, to appear].

The physical meanings of the objective functions to be minimized are the following

f_1 – keep the surface temperature near the desired temperature,
f_2 – keep the surface temperature between some upper and lower bounds,
f_3 – avoid excessive cooling or reheating on the surface,
f_4 – restrict the length of the liquid pool, and
f_5 – avoid too low temperatures at the yield point.

The system (7.2.1) is discretized by the finite element method (FEM). We do not here present the discretization nor the exact formulation of the four latter objective functions. What has been stated in [Mäkelä, Männikkö, 1992, to appear], as to these parts, is valid also here. The difference is that, earlier, all of the five objective functions were summed up together. Here every single one of them is treated as a separate objective function.

In the computational experiments, when solving the optimal control problem of the continuous casting of steel, the dimension of the problem is $n = 325$. The rest of the numerical data for the problem can be found in [Mäkelä, Männikkö, 1992].

Subgradient GDF Method

The first impression of the computational complexity of the problem was strengthened during the solution process. The solutions obtained depend, for instance, on the starting point of the whole process, the starting point at the projection phase, the computational accuracy (of the auxiliary scalar projection function) required and, of course, on the marginal rates of substitution provided by the decision maker.

The reference function has been selected to be f_1. Here we present a solution process where the augmented weighted Tchebycheff function is used to produce Pareto optimal solutions. The augmentation term employed is $\rho = 0.0001$. The weighting coefficients are all set to be equal to one. The reason for this is the complicated nature of the problem. The nadir point is very difficult to be approximated. On the other hand, it is known that the scales of the objective functions do not differ from each other remarkably. Thus it is safer to use equal weighting.

Exceptionally, when compared with earlier examples, the accuracy in the calculations had to be tightened to 10^{-8}. In addition, the maximum number of function value calculations at each projection phase was restricted to 150.

We do not present the decision variable values because of the great dimension of the problem ($n = 325$). The objective functions have been scaled so that their values are equal to one at the starting point. Thus we have $\mathbf{z}^1 = (1.0, 1.0, 1.0, 1.0, 1.0)$.

In addition to the iterations described here, there were several iterations when the specified marginal rates of substitution produced alternatives from an opposite direction to what was expected. Reasons for this are the difficulty of specifying marginal rates of substitution and the unstable behaviour of the complicated problem.

Now we can go into the solution process. At $\mathbf{z}^1$, the length of the liquid pool is too long in the fourth objective function. Also the temperature at the yield point (f_5) is too low. Both of these facts bring about the possibility that the strand gets defects when it is straightened. So the marginal rates of substitution are chosen at the first iteration as $m_2^1 = 1.5$, $m_3^1 = 2.5$, $m_4^1 = 3.5$ and $m_5^1 = 4.5$. After the projections, five ($P = 5$) Pareto optimal alternatives shown in Table 48 are obtained.

	f_1	f_2	f_3	f_4	f_5
1	0.5179	0.0000	0.5179	0.5179	0.5179
$\boxed{2}$	0.5992	0.0000	0.5960	0.6652	0.3403
3	0.6765	0.0000	0.6432	0.7855	0.2182
4	0.7796	0.0000	0.6885	0.9067	0.1015
5	0.9446	0.0000	0.7668	1.0984	0.0203

Table 48.

None of the alternatives obtained is particularly good, but the second one, $\mathbf{z}^2 = (0.5992, 0.0, 0.596, 0.6652, 0.3403)$, is selected because the value of the fifth objective has been considerably decreased. In the latter alternatives, the values of the other objective functions are too high.

At the second iteration, the marginal rates of substitution, $m_2^2 = 1.0$, $m_3^2 = 50.0$, $m_4^2 = 40.0$, and $m_5^2 = 40.0$, are specified to emphasize the relative importance of the third, the fourth and the fifth objective function. The alternatives obtained are presented in Table 49.

	f_1	f_2	f_3	f_4	f_5
1	0.5992	0.0000	0.5960	0.6652	0.3403
2	0.5614	0.0000	0.5915	0.5269	0.4285
3	0.5358	0.0000	0.5875	0.3895	0.5168
4	0.5220	0.0000	0.5840	0.2552	0.6054
$\boxed{5}$	0.5193	0.0000	0.5795	0.1217	0.7008

Table 49.

From this set the fifth candidate is selected for continuation. The reason is that even though the value of the fifth objective function increased, the values of the third and especially the fourth objective function were managed to decrease remarkably. Thus, we continue from $\mathbf{z}^3 = (0.5193, 0.0, 0.5795, 0.1217, 0.7008)$.

Now the importance of the third and the fourth objective function, avoiding excessive cooling or reheating on the surface and restricting the length of the liquid pool, respectively, has become more evident. The marginal rates of substitution at the third iteration are the following, $m_2^3 = 1.0$, $m_3^3 = 70.0$, $m_4^3 = 40.0$ and $m_5^3 = 20.0$. Five alternatives produced by this setting can be seen in Table 50.

	f_1	f_2	f_3	f_4	f_5
1	0.5193	0.0000	0.5795	0.1217	0.7008
2	0.5079	0.0000	0.5355	0.0000	1.0335
3	0.5943	0.0000	0.4714	0.0000	0.9283
4	0.7211	0.0000	0.3019	0.0000	1.0013
$\boxed{5}$	0.9047	0.0000	0.2385	0.0000	0.7067

Table 50.

From these alternatives the fifth one, $\mathbf{z}^4 = (0.9047, 0.0, 0.2385, 0.0, 0.7067)$, is the best, because the value of the fourth objective function decreased down to zero and the value of the third objective function decreased also nicely. The fact that the value of the first objective function increased is not so dangerous. Keeping the surface temperature near some desired temperatures is not as important as the other objective functions. For briefness, we do not here present all the iterations in the complete extent.

We just state that at the fourth iteration the marginal rates of substitution are set as $m_2^4 = 1.0$, $m_3^4 = 70.0$, $m_4^4 = 40.0$ and $m_5^4 = 40.0$ and the solution selected is $\mathbf{z}^5 = (0.9814, 0.0, 0.1749, 0.3339, 0.3925)$. The reason for this selection is that even though the value of the fourth objective function increased, the values of the third and the fifth objective function decreased in proportion.

At the fifth iteration, the marginal rates of substitution are specified to be $m_2^5 = 1.0$, $m_3^5 = 70.0$, $m_4^5 = 80.0$ and $m_5^5 = 10.0$. The intention is to emphasize the importance of the fourth objective function. At this iteration, the solution $\mathbf{z}^6 = (1.1103,\ 0.0, 0.0747, 0.0, 0.7848)$ is selected because the values of the third and the fourth objective function are quite satisfactory.

The solution process continues by setting $m_2^6 = 1.0$, $m_3^6 = 80.0$, $m_4^6 = 60.0$ and $m_5^6 = 20.0$. This time the vector $\mathbf{z}^7 = (1.1286, 0.0, 0.0289, 0.0043, 0.6389)$ is considered the best. The reason is that the values of the third and the fifth objective function decreased, at the cost of the fourth objective function, though. Anyway, this is quite a good solution.

At the seventh iteration, the marginal rates of substitution are $m_2^7 = 1.0$, $m_3^7 = 5.0$, $m_4^7 = 3.0$ and $m_5^7 = 2.0$. The alternatives to be considered in this phase have been listed in Table 51.

	f_1	f_2	f_3	f_4	f_5
1	1.1286	0.0000	0.0289	0.0043	0.6389
2	1.0850	0.0000	0.0233	0.0000	0.7533
$\boxed{3}$	1.0918	0.0000	0.0252	0.0000	0.7079
4	1.0779	0.0000	0.0187	0.0000	1.0300
5	1.0737	0.0000	0.0349	0.0000	0.9447

Table 51.

The third alternative is the most preferred one from this list. It seems that the value of the fourth objective function (restricting the length of the liquid pool) can only be decreased by increasing the value of the fifth objective function (allowing lower temperatures at the yield point). The solution could not be improved, and so the solution process is stopped with $\mathbf{z}^8 = (1.0918, 0.0, 0.0252, 0.0, 0.7079)$ as the final solution.

The SC value of this above-described solution process with seven iterations is 3435. It is a large number but the problem to be solved is complicated. This number contains only those iterations when some progress was accomplished.

Next, we outline the solution of the continuous casting problem, employing the penalty scalarizing function in the subgradient GDF method. The penalty coefficient utilized is $\varrho = 10000.0$. If the same marginal rates of substitution and choices as in the solution process above are used, the SC value is 2635. However, the result obtained ($\mathbf{z}^8 = (0.8521, 0.0, 0.2766, 0.0, 0.9471)$) is not particularly satisfactory. The best possible solution obtained by the penalty scalarizing function is $\mathbf{z}^8 = (1.2609, 0.0, 0.074, 0.0, 0.71)$. This could not be improved. When this solution and the one obtained with the augmented weighted Tchebycheff function are compared, this solution is not even Pareto optimal. The reason for this phenomenon is the fact that all the solutions obtained are only locally Pareto optimal. Thus, such situations may occur.

The SC value of the solution process with the penalty scalarizing function is 2679. Also this process took seven iterations. The number is somewhat less than with the augmented weighted Tchebycheff function but the result obtained is worse. Thus, the choice of the method by which Pareto optimal solutions are produced has its effect on the solution obtained.

When the black-box routine MPB is used (instead of the augmented weighted Tchebycheff function or the penalty scalarizing function) to produce (weakly) Pareto optimal solutions, its weakness shows up. After the first iteration, all the alternatives produced by the subgradient GDF method are already weakly Pareto optimal and the MPB routine cannot improve them at all. At this problem setting, the MPB routine does not appear to advantage. This proves the point that no method can be good at every context. For some problems the MPB routine produces solutions effectively, but for other problems its weak sides are emphasized.

NIMBUS Method

When the solution process is started by the NIMBUS method from the point $\mathbf{z}^0 = (1.0, 1.0, 1.0, 1.0, 1.0)$ by setting $I^< = \{1, 2, 3, 4, 5\}$, the resulting vector is $\mathbf{z}^1 = (0.5043, 0.0, 0.4548, 0.7371, 0.5922)$. At this point, there is too excessive cooling or reheating on the surface (f_3), the liquid pool is too long (f_4) and the temperature is too low at the yield point (f_5). In other words, the values of the third, the fourth and the fifth objective function are too high.

So, we continue with the specification $I^< = \{3, 4, 5\}$ with $w_3^1 = 0.35$, $w_4^1 = 0.5$ and $w_5^1 = 0.15$. Something must be allowed to increase, and we select the first objective into this class. We set $I^> = \{1\}$ and $\varepsilon_1^1 = 1.0$. The value of the second objective function is good, and we set $I^= = \{2\}$.

The result of this classification is that no improvement can be found and $\mathbf{z}^2 = \mathbf{z}^1$. When such cases occur, the reason often lies in too tight bounds. Therefore, we can try to continue by allowing the value of the second objective function to increase. The problem is so complicated that the possibility of jamming in some local optimum is great. To avoid the jamming we relax the second objective function relatively much. This allows improvement in the other functions and if the value of the second objective function should increase too much, one can always take a shorter step.

After this reasoning we leave the classification $I^< = \{3,4,5\}$ with $w_3^2 = 0.35$, $w_4^2 = 0.5$ and $w_5^2 = 0.15$ unchanged. Then we set $I^> = \{1,2\}$ with $\varepsilon_1^2 = 1.0$ and $\varepsilon_2^2 = 1.0$.

The solution obtained is $\hat{\mathbf{z}}^2 = (0.9938, 0.0, 0.0, 0.3147, 0.4403)$. As we can see, the value of the second objective function did not actually increase from 0.0. Allowing the increment was necessary only for computational reasons. This point is selected for continuation without considering any intermediate solutions; thus $\mathbf{z}^3 = \hat{\mathbf{z}}^2$.

At the point $\mathbf{z}^3$, the values of the fourth and the fifth objective function are still too high. Thus, their values are to be decreased and $I^< = \{4,5\}$. We also attach weighting coefficients to them, $w_4^3 = 0.8$ and $w_5^3 = 0.2$. The first objective function has less importance than the other ones and it is thus allowed to increase. Keeping in mind the earlier experiences, we give a loose upper bound also for the second objective function. We set $I^> = \{1,2\}$ with $\varepsilon_1^3 = 1.5$ and $\varepsilon_2^3 = 1.0$. The zero value for the third objective function is good, and we set $I^= = \{3\}$.

However, no improvement is possible with this specification either, and $\mathbf{z}^4 = \mathbf{z}^3$. Something must again be changed. Therefore, we revise the classification by leaving the class $I^=$ and relaxing the third objective function a little. It is so important that we cannot let it increase significantly. We have $I^< = \{4,5\}$ with $w_4^4 = 0.8$ and $w_5^4 = 0.2$ and $I^> = \{1,2,3\}$ with $\varepsilon_1^4 = 2.0$, $\varepsilon_2^4 = 1.0$ and $\varepsilon_3^4 = 0.01$.

This setting gives a solution $\hat{\mathbf{z}}^4 = (1.6059, 0.0, 0.0029, 0.0002, 0.353)$. For certainty, we take a glance at five alternatives. They have been tabulated in Table 52.

	f_1	f_2	f_3	f_4	f_5
1	0.9938	0.0000	0.0000	0.3147	0.4403
2	1.1058	0.0000	0.0004	0.2305	0.4238
3	1.2455	0.0000	0.0009	0.1524	0.4036
4	1.4124	0.0000	0.0017	0.0808	0.3801
$\boxed{5}$	1.6059	0.0000	0.0029	0.0002	0.3530

Table 52.

The fifth alternative is the best from this set and we have $\mathbf{z}^5 = \hat{\mathbf{z}}^4$. This seems to be quite a good solution, but we still would like the third objective function to decrease to the zero value. In other words, we would like to better avoid excessive cooling or reheating on the surface. In addition, it would be interesting to see whether the values of the fourth and the fifth objective function can still be decreased.

Thus, we specify $I^< = \{3,4,5\}$ without any special weighting coefficients and, as before, $I^> = \{2\}$ with $\varepsilon_2^5 = 1.0$. For curiosity, we let the values of the first objective function change freely, $I^\diamond = \{1\}$.

Once again, no improved solutions can be found and $\mathbf{z}^6 = \mathbf{z}^5$. Now, too tight upper bounds cannot be the excuse. Nevertheless, there is no reason to give up yet. We have here a chance to utilize the versatile possibilities of the NIMBUS method, where we have different means to strive for the same goal. Instead of using functions to be minimized, we can use aspiration functions.

Therefore, we set $I^{\leq} = \{3, 4, 5\}$ without any special weighting. The aspiration levels of the third and the fourth objective function are naturally set to zero, $\bar{z}_3^6 = \bar{z}_4^6 = 0.0$ and for the fifth objective function we set $\bar{z}_5^6 = 0.2$. The rest of the classification need not be changed and $I^{>} = \{2\}$ with $\varepsilon_2^6 = 1.0$ and $I^{\diamond} = \{1\}$.

The solution obtained is $\bar{\mathbf{z}}^6 = (2.1479, 0.0157, 0.0, 0.0, 0.1997)$. The decrement in the values of the third, the fourth and the fifth objective function could be obtained only at the expense of increasing the second (and the first) objective function. Especially, the increment of the second objective function is not desirable because it is important to keep the surface temperature between some upper and lower bounds. Therefore, we do not take this step but return to the previous solution, $\mathbf{z}^7 = \mathbf{z}^6$.

We keep the classification similar to the previous one but increase the aspiration level of the fifth objective function. Thus, we have $I^{\leq} = \{3, 4, 5\}$ with $\bar{z}_3^7 = \bar{z}_4^7 = 0.0$ and $\bar{z}_5^7 = 0.23$, $I^{>} = \{2\}$ with $\varepsilon_2^7 = 1.0$, and $I^{\diamond} = \{1\}$.

After this specification, the solution generated is $\hat{\mathbf{z}}^7 = (2.0383, 0.0, 0.0, 0.0, 0.2286)$. The value of the first objective function is tolerable, and no intermediate solutions are desired as far as it is concerned. The values of the other objective functions are also satisfactory and so we stop the solution process here. In the end, the method checks that the final solution $\mathbf{z}^8 = (2.0383, 0.0, 0.0, 0.0, 0.2286)$ is Pareto optimal.

The SC value of the above-presented seven iterations is 508, where the checking of Pareto optimality takes 51 function calls. Thus, in this problem, the difference between the computational complexities of the subgradient GDF method and the NIMBUS method is manifested. The SC value of the subgradient GDF method is more than six times larger than that of the NIMBUS method, not to mention that the final solution obtained by NIMBUS is far more desirable.

The solution of the continuous casting problem by the weighting method has been described in [Mäkelä, Männikkö, 1992]. The decision maker was used only to give weighting coefficients to the objective functions. When all the coefficients were equal, the best criterion vector that could be obtained was $\mathbf{z}^{\circ} = (1.0378, 0.0, 0.0772, 0.002, 0.537)$. Even the solution obtained by the subgradient GDF method ($\mathbf{z} = (1.0918, 0.0, 0.0252, 0.0, 0.7079)$) can be considered to be better because of the smaller values in the important third and fourth objective function, despite the opposite difference in the fifth objective.

The second solution reported in [Mäkelä, Männikkö, 1992] was obtained by setting the weighting coefficient of the first objective function to be remarkably small and and the rest of the coefficients to be equal with each other. By this setting the best solution obtained was $\mathbf{z}^{\circ} = (2.1714, 0.0, 0.0019, 0.0, 0.1981)$. When it is compared with $\mathbf{z} = (2.0383, 0.0, 0.0, 0.0, 0.2286)$ attained by the NIMBUS method, the latter can be said to be better. There the value of the third objective function has been managed to decrease to zero by paying the price of a slight increment in the fifth objective function value.

8. Future Directions

In this chapter, we outline some challenging topics for the future development of multiobjective optimization, mainly from a mathematical point of view. They all deserve further research and examination.

Even though multiobjective optimization methods have been applied to solve problems from many sides of life, like design problems in engineering, production problems in economics, and environmental control problems in ecology, there still exist many new problem types where multiobjective optimization may bring along new possibilities. Encouraging experiences were obtained in the previous chapter when solving state-constrained optimal control problems. Especially challenging are the practical problems. Remarkably few real-life applications have been reported so far.

One interesting problem type is so-called multidisciplinary re-engineering. It means that old engineering problems, for example, in optimal design, whose solutions have been revised one feature at a time in the course of years, are solved again from the very beginning, taking different aspirations and aspects into consideration at the same time.

The methodology of multiobjective optimization must also be improved. The idea of using different methods in different phases of the solution process is quite new. In this way, positive features of various methods are taken into account in such phases of the solution process when they can best be benefited. In addition, some weaknesses of the methods may be possible to overcome.

A meta algorithm endeavouring at consolidating different methods of multiobjective optimization has been proposed in [Steuer, Whisman, 1986]. The idea is that the same meta program can be transformed into different methods by varying its controlling parameters. The GDF, the IWT and the reference point method with the visual interactive approach, STEM, the ε-constraint method and two interactive versions of the weighting method are available. This idea has been further developed in [Steuer, Gardiner, 1990]. An important fact to consider, when switching from one method to another in the middle of the solution process, how to maintain the convergence properties, needs still more examination.

An attempt to realize this meta algorithm has been named MCOP. It is a system for IBM compatible microcomputers, implemented at the Department of Mathematics of the University of Jyväskylä. The weighting method, the ε-constraint method and the reference point method have been included in the pilot version. Bar charts and value paths are available for graphical illustration. The graphical user interface makes it possible to expand the human-computer dialogue. The user can input the required weights, upper bounds and reference points in a graphical form.

Similar ideas of combining several methods have been proposed in [Clímaco, Antunes, 1991]. The system (only for MOLP problems) contains, for example, the ZW method, STEM, VIG and TRIMAP. A restriction caused by TRIMAP is that only problems with three objective functions can be handled. The system has been implemented in the Apple Macintosh environment with graphical illustrations. A further developed implementation of the above-mentioned ideas is described in [Antunes, Alves, Silva, Clímaco, 1992]. The method base package has been named TOMMIX. Means for supporting the decision maker in deciding when and how to change from one method to another have still to be explicated.

Another way to be elaborated is combining methods for continuous and discrete problems. It may, for example, mean that a set of solutions is generated to the continuous problem, and, then this set is ranked with the means of discrete methods. Examples of this have been presented in [Kok, Lootsma, 1985], [Bard, 1986] and [Slowinski, 1991]. One can also combine methods of global optimization to multiobjective optimization methods. In this way, one can aim at being able to handle globally Pareto optimal solutions, instead of locally Pareto optimal ones, also in nonconvex problems. Such ideas have been proposed in [Törn, 1983].

In [Arbel, Korhonen, 1994], the authors have developed a new method in the spirit of interior-point methods (of linear programming). The idea is to wander in the interior of the feasible criterion region and only in the end to come up to the Pareto optimal surface. In this way, they call into question the generally adopted idea that the decision makers should handle only (weakly) Pareto optimal solutions.

An important area of development is software implementing different methods and, especially, the user interface. As more and more advanced computers and graphical devices are created, the more tools are available when striving for easiness and even enjoyment of use. As far as problems of large scale are concerned, the possibilities of parallel computing are worth examining in making the solution processes more efficient.

One potentiality not to be forgotten are expert systems. They can be utilized in both selecting a solution method and in the solution process itself. As an example, interactive MOLP methods and expert system techniques have been integrated in [Antunes, Melo, Clímaco, 1992]. The system described includes five methods, for example, STEM and the ZW method. When the user of the system expresses her or his hopes for further actions (such as: "I want to know what is around the current solution"), the system suggests one of the interactive methods to be utilized. The program has been developed on an IBM/PS2 computer in C and the expert system components in Prolog. Computer graphics is also available. There is a lot of features that deserve further research and development, but this is certainly an interesting path to follow.

9. Conclusions

We have presented a self-contained survey of the state-of-the-art of multiobjective optimization with a great number of further references. We have begun by handling several significant concepts and their relations and then continued by considering some theoretical results and connections.

We have demonstrated the methodology of multiobjective optimization by describing several methods and by giving references to a large number of other methods. The methods have been classified into four groups according to the contribution of the decision maker in the solution process. Because the group of interactive methods has been developed most, the main emphasis has been in it. We have endeavoured to characterize the methods by some comments on their positive and negative features.

Existing software packages have been demonstrated by giving short overviews. Some practical experiences have also been expressed. Related to software, several possibilities of the graphical illustration of alternatives have been introduced.

Selected experiences and observations of the comparison of the methods have been put forward. Certain features of the interactive methods handled have been collected in a comparative table. In addition, some attempts to aid in the selection of a solution method have been made. A decision tree containing both interactive and noninteractive methods has been suggested.

While two of the methods presented are new, their functioning has been clarified by numerical examples. Those methods have also been applied to solve two benchmark-type problems of optimal control. The problems have earlier been solved by less developed methods. Improved results compared to the previous ones were obtained.

Many things can be concluded from the numerical experiments with the subgradient GDF method and the NIMBUS method on a more general basis. The computational expense is not distinguished with relatively simple problems. But when the complexity of the problem increases, so does also the difference in the computational costs. Naturally, there are profound differences in the solution philosophies of these two methods. However, one can say that NIMBUS seems to be computationally more efficient.

One reason for this is that there is no explicit scalarizing function employed. On the other hand, the algorithm cannot make any difference between weakly Pareto optimal (or substationary, to be exact) and Pareto optimal solutions. Thus, weakly Pareto optimal solutions are not projected anywhere. This (in itself, negative) feature may also have influence by decreasing the computational burden.

In addition to the computational expense, there are also differences in the easiness of use between the subgradient GDF method and the NIMBUS method. For many people it is easier to think of desired changes in the objective function values than to specify indifference relations. This may, especially, be the case if the criterion vector at which the marginal rates of substitution are to be specified is not particularly desirable. Then it may be frustrating to think of indifferent solutions instead of improvements wanted. A remarkable reason for obtaining more satisfactory solutions by the NIMBUS method in the examples reported is its more intelligible communication style.

One more interesting factor is the accuracy in the computation. It can be considered in a more extensive meaning as a separating factor between scalarizing functions and treating the objective functions as themselves. If some scalarizing function is employed, then it is the accuracy of that additional function that can be followed along the solution process. It may happen that when the accuracy of the scalarizing function has reached the desired level, the values of the actual objective functions can still change considerably. For this reason, the accuracy had to be tightened when solving the continuous casting problem with the subgradient GDF method and the augmented weighted Tchebycheff function. When no auxiliary scalarizing function is used, then the accuracies of the objective functions themselves can be followed directly during the solution process.

Many scalarizing functions have positive features whose importance is not to be underestimated, like producing Pareto optimal solutions only. However, employing some scalarizing function usually brings along extra parameters and the difficulty of specifying their values. This causes additional stability concern. To put it shortly, scalarizing functions add extra characteristics to the problem.

Scalarization cannot completely be avoided even in the MPB routine. However, the scalarization is carried out under the surface, invisible to the user. Whatever additional parameters or phases are needed, they cannot be seen and the user does not have to be bothered with them. The weakness of the MPB routine is that the Pareto optimality of the solutions obtained cannot be guaranteed. In theory, only the substationarity of the solutions is sure. In practice, it is, however, very likely that the solutions are at least weakly Pareto optimal. As a matter of fact, in the numerical experiments performed, the final solutions obtained have usually proved out to be Pareto optimal at the final testing.

Even though the solution processes with the subgradient GDF method and the NIMBUS method cannot be directly compared, we can still conclude that NIMBUS seems to be the more efficient and flexible of these two. Naturally, there are many challenges in the further development of NIMBUS. One of the challenges, also in general for software development, is creating illustrative and easy-to-use user interfaces. If the interface is able to adapt in the decision maker's style of making decisions and is of help in analyzing the alternatives and results, and can perhaps give suggestions or advice, then the interface may even overcome some lacks of the method itself.

In general, one can say that the theory and the methods of multiobjective optimization have been widely developed during the last few decades. Software implementations are considerably more infrequent. Even more exceptional are documentations on solving real-life multiobjective optimization problems. Reasons for this may be ignorance of all the extensive possibilities of existing methods or also the lack of suitable methods. For our part, we have filled a gap in nondifferentiable multiobjective optimization.

In the method development one can conclude that it is important to continue in the direction of user-friendliness. The methods must even better be able to correspond the characteristics of the decision maker. If the aspirations of the decision maker change during the solution process, the algorithm must be able to cope with it. Computational tests have confirmed the thought that the decision makers want to feel being in control in the solution process and they must understand what is happening. The decision maker must be the starting point in developing new interactive methods.

References

Aksoy Y., *Interactive Multiple Objective Decision Making: A Bibliography (1965–1988)*, Management Research News **13**, No. 2 (1990), 1–8.

Alekseichik M. I., G. E. Naumov, *Existence of Efficient Points in Infinite-Dimensional Vector Optimization Problem*, Automation and Remote Control **42**, No. 5, Part 2 (1981), 666–670.

Angehrn A. A., *Supporting Multicriteria Decision Making: New Perspectives & New Systems*, Working Paper, INSEAD, Fontainebleau (1990(a)).

——————, *"Triple C" A Visual Interactive MCDSS*, Working Paper, INSEAD, Fontainebleau (1990(b)).

Antunes C. H., M. J. Alves, A. L. Silva, J. N. Clímaco, *An Integrated MOLP Method Base Package – A Guided Tour of TOMMIX*, Computers and Operations Research **19**, No. 7 (1992), 609–625.

Antunes C. H., J. N. Clímaco, *Sensitivity Analysis in MCDM Using the Weight Space*, Operations Research Letters **12** (1992), 187–196.

Antunes C. H., M. P. Melo, J. N. Clímaco, *On the Integration of an Interactive MOLP Procedure Base and Expert System Technique*, European Journal of Operational Research **61**, No. 1–2 (1992), 135–144.

Arbel A., P. Korhonen, *Using Aspiration Levels in an Interactive Interior Multiobjective Linear Programming Algorithm*, Manuscript via private communication (1994).

Armann R., *Solving Multiobjective Programming Problems by Discrete Representation*, Optimization **20**, No. 4 (1989), 483–492.

Aubin J.-P., B. Naslund, *An Exterior Branching Algorithm*, Working Paper 72-42, European Institute for Advanced Studies in Management, Brussels (1972).

Baba N., H. Takeda, T. Miyake, *Interactive Multi-Objective Programming Technique Using Random Optimization Method*, International Journal of Systems Science **19**, No. 1 (1988), 151–159.

Balachandran M., J. S. Gero, *The Noninferior Set Estimation (NISE) Method for Three Objective Problem*, Engineering Optimization **9**, No. 2 (1985), 77–88.

Ballestero E., C. Romero, *A Theorem Connecting Utility Function Optimization and Compromise Programming*, Operations Research Letters **10**, No. 7 (1991), 421–427.

Bard J. F., *A Multiobjective Methodology for Selecting Subsystem Automation Options*, Management Science **32**, No. 12 (1986), 1628–1641.

Bárdossy A., I. Bogárdi, L. Duckstein, *Composite Programming as an Extension of Compromise Programming*, in "Mathematics of Multi Objective Optimization," Edited by P. Serafini, Springer-Verlag, Wien, New York, 1985, 375–408.

Bare B. B., G. Mendoza, *Multiple Objective Forest Land Management Planning: An Illustration*, European Journal of Operational Research **34** (1988), 44–55.

Barnikow A. B., R. Gollmer, M. Grauer, *The Interactive Multiobjective Package IMPROVE with Application to Strategic Planning of Carbochemical Industry*, in "Large-Scale Modelling and Interactive Decision Analysis," Edited by G. Fandel, M. Grauer, A. Kurzhanski, A. P. Wierzbicki, Lecture Notes in Economics and Mathematical Systems 273, Springer-Verlag, 1986, 242–250.

Batishchev D. I., V. F. Anuchin, D. E. Shaposhnikov, *The Use of the Qualitative Information on the Importance of Particular Criteria for the Computation of Weighting Coefficients*, in "Multiobjective Problems of Mathematical Programming," Edited by A. Lewandowski, V. Volkovich, Lecture Notes in Economics and Mathematical Systems 351, Springer-Verlag, 1991, 2–7.

Belkeziz K., M. Pirlot, *Proper Efficiency in Nonconvex Vector-Maximization-Problems*, European Journal of Operational Research **54** (1991), 74–80.

Bell D. E., *Double-Exponential Utility Functions*, Mathematics of Operations Research **11**, No. 2 (1986), 351–361.

Benayon R., J. de Montgolfier, J. Tergny, O. Laritchev, *Linear Programming with Multiple Objective Functions: Step Method (STEM)*, Mathematical Programming **1**, No. 3 (1971), 366–375.

Benbasat I., A. S. Dexter, *An Experimental Evaluation of Graphical and Color-Enhanced Information Presentation*, Management Science **31**, No. 11 (1985), 1348–1364.

Benito-Alonso M. A., P. Devaux, *Location and Size of Day Nurseries – a Multiple Goal Approach*, European Journal of Operational Research **6** (1981), 195–198.

Benson H. P., *Existence of Efficient Solutions for Vector Maximization Problems*, Journal of Optimization Theory and Applications **26**, No. 4 (1978), 569–580.

——————, *Vector Maximization with Two Objective Functions*, Journal of Optimization Theory and Applications **28**, No. 2 (1979), 253–257.

———————, *On a Domination Property for Vector Maximization with Respect to Cones*, Journal of Optimization Theory and Applications **39**, No. 1 (1983), 125–132.

———————, *Complete Efficiency and the Initialization of Algorithms for Multiple Objective Programming*, Operations Research Letters **10**, No. 8 (1991), 481–487.

Benson H. P., Y. Aksoy, *Using Efficient Feasible Directions in Interactive Multiple Objective Linear Programming*, Operations Research Letters **10**, No. 4 (1991), 203–209.

Benson H. P., T. L. Morin, *The Vector Maximization Problem: Proper Efficiency and Stability*, SIAM Journal on Applied Mathematics **32**, No. 1 (1977), 64–72.

Benson H. P., S. Sayin, *Optimization over the Efficient Set: Four Special Cases*, Journal of Optimization Theory and Applications **80**, No. 1 (1994), 3–18.

Ben-Tal A., *Characterization of Pareto and Lexicographic Optimal Solutions*, in "Multiple Criteria Decision Making Theory and Application," Edited by G. Fandel, T. Gal, Lecture Notes in Economics and Mathematical Systems 177, Springer-Verlag, Berlin, Heidelberg, 1980, 1–11.

Bhatia D., S. Aggarwal, *Optimality and Duality for Multiobjective Nonsmooth Programming*, European Journal of Operational Research **57**, No. 3 (1992), 360–367.

Bhatia D., N. Datta, *Necessary Conditions and Subgradient Duality for Non-Differentiable and Non-Convex Multi-Objective Programming Problem*, Cahiers du Centre d'Etudes de Recherche Operationelle **27**, No. 1–2 (1985), 131–139.

Bischoff E., *Two Empirical Tests with Approaches to Multiple-Criteria Decision Making*, in "Plural Rationality and Interactive Decision Processes," Edited by M. Grauer, M. Thompson, A. P. Wierzbicki, Lecture Notes in Mathematics and Economical Systems 248, Springer-Verlag, Berlin, Heidelberg, 1985, 344–347.

————, *Multi-Objective Decision Analysis – The Right Objectives?*, in "Large-Scale Modelling and Interactive Decision Analysis," Edited by G. Fandel, M. Grauer, A. Kurzhanski, A. P. Wierzbicki, Lecture Notes in Economics and Mathematical Systems 273, Springer-Verlag, 1986, 155–160.

Bitran G. R., *Duality for Nonlinear Multiple-Criteria Optimization Problems*, Journal of Optimization Theory and Applications **35**, No.3 (1981), 367–401.

Bogetoft P., Å. Hallefjord, M. Kok, *On the Convergence of Reference Point Methods in Multiobjective Programming*, European Journal of Operational Research **34** (1988), 56–68.

Bolintineanu S., B. D. Craven, *Linear Multicriteria Sensitivity and Shadow Costs*, Optimization **26**, No. 1–2 (1992), 115–127.

Borwein J. M., *On the Existence of Pareto Efficient Points*, Mathematics of Operations Research **8**, No. 1 (1983), 64–73.

Borwein J. M., D. M. Zhuang, *Super Efficiency in Vector Optimization*, Manuscript via private communication (1989).

————————, *Super Efficiency in Convex Vector Optimization*, ZOR – Methods and Models of Operations Research **35**, No. 3 (1991), 175–184.

Bowman V. J. Jr., *On the Relationship of the Tchebycheff Norm and the Efficient Frontier of Multiple-Criteria Objectives*, in "Multiple Criteria Decision Making," Edited by H. Thiriez, S. Zionts, Lecture Notes in Economics and Mathematical Systems 130, Springer-Verlag, Berlin, Heidelberg, 1976, 76–85.

Brauer D. C., G. Naadimuthu, *A Goal Programming Model for Aggregate Inventory and Distribution Planning*, Mathematical and Computer Modelling **16**, No. 3 (1992), 81–90.

Breslawski S., S. Zionts, *An Interactive Multiple Criteria Programming Package*, in "Decision Making with Multiple Objectives," Edited by Y. Y. Haimes, V. Chankong, Lecture Notes in Economics and Mathematical Systems 242, Springer-Verlag, Berlin, Heidelberg, 1985, 282–286.

Brockhoff K., *Experimental Test of MCDM Algorithms in a Modular Approach*, European Journal of Operational Research **22** (1985), 159–166.

Buchanan J. T., *Multiple Objective Mathematical Programming: A Review*, New Zealand Operational Research **14**, No. 1 (1986), 1–27.

Buchanan J. T., H. G. Daellenbach, *A Comparative Evaluation of Interactive Solution Methods for Multiple Objective Decision Models*, European Journal of Operational Research **29**, No. 3 (1987), 353–359.

Cambini A., L. Martein, *On the Existence of Efficient Points*, Optimization **28**No. 3-4 (1994), 283–290.

Carmichael D. G., "Structural Modelling and Optimization," Ellis Horwood Ltd., 1981.

Censor Y., *Pareto Optimality in Multiobjective Problems*, Applied Mathematics and Optimization **4**, No. 1 (1977), 41–59.

Chankong V., Y. Y. Haimes, *The Interactive Surrogate Worth Trade-Off (ISWT) Method for Multiobjective Decision-Making*, in "Multiple Criteria Problem Solving," Edited by S. Zionts, Lecture Notes in Economics and Mathematical Systems 155, Springer-Verlag, Berlin, Heidelberg, 1978, 42–67.

________________, *On the Characterization of Noninferior Solutions of the Vector Optimization Problem*, Automatica **18**, No. 6 (1982), 697–707.

________________, *Optimization-Based Methods for Multiobjective Decision-Making: An Overview*, Large Scale Systems **5**, No. 1 (1983(a)), 1–33.

________________, "Multiobjective Decision Making Theory and Methodology," Elsevier Science Publishing Co., Inc., New York, 1983(b).

Charnes A., W. W. Cooper, "Management Models and Industrial Applications of Linear Programming," John Wiley & Sons, Inc., New York, 1961.

________________, *Goal Programming and Multiple Objective Optimization; Part 1*, European Journal of Operational Research **1**, No. 1 (1977), 39–54.

Charnes A., W. W. Cooper, R. O. Ferguson, *Optimal Estimation of Executive Compensation by Linear Programming*, Management Science **1**, No. 2 (1955), 138–151.

Chen G.-Y., *Necessary Conditions of Nondominated Solutions in Multicriteria Decision Making*, Journal of Mathematical Analysis and Applications **104**, No. 1 (1984), 38–46.

Chen T., B.-I. Wang, *An Interactive Method for Multiobjective Decision Making*, in "Large Scale Systems: Theory and Applications 1983," Edited by A. Straszak, Pergamon Press, 1984, 277–282.

Chew K. L., E. U. Choo, *Pseudolinearity and Efficiency*, Mathematical Programming **28**, No. 2 (1984), 226–239.

Choo E. U., D. R. Atkins, *Proper Efficiency in Nonconvex Multicriteria Programming*, Mathematics of Operations Research **8**, No. 3 (1983), 467–470.

Choo E. U., K. P. Chew, *Optimal Value Functions in Parametric Programming*, Zeitschrift für Operations Research **29** (1985), 47–57.

Chou J. H., W.-S. Hsia, T.-Y. Lee, *On Multiple Objective Programming Problems with Set Functions*, Journal of Mathematical Analysis and Applications **105**, No. 2 (1985), 383–394.

Clarke F. H., "Optimization and Nonsmooth Analysis," John Wiley & Sons, Inc., 1983.

Clímaco J., *The TRIMAP Package as a Powerful Teaching Tool*, in "Multiple Criteria Decision Making and Risk Analysis Using Microcomputers," Edited by B. Karpak, S. Zionts, Springer-Verlag, Berlin, Heidelberg, 1989, 205–233.

Clímaco J. N., C. H. Antunes, *TRIMAP – An Interactive Tricriteria Linear Programming Package*, Foundations of Control Engineering **12**, No. 3 (1987), 101–119.

________________, *Implementation of a User-friendly Software Package – A Guided Tour of TRIMAP*, Mathematical and Computer Modelling **12**, No. 10/11 (1989), 1299–1309.

________________, *A Comparison of Microcomputer Implemented Interactive MOLP Methods Based on a Case Study*, in "Readings in Multiple Criteria Decision Aid," Edited by C. A. Bana e Costa, Springer-Verlag, Berlin, Heidelberg, 1990, 445–470.

________________, *Flexible Method Bases and Man-Machine Interfaces as Key Features in Interactive MOLP Approaches*, in "Multiple Criteria Decision Support," Edited by P. Korhonen, A. Lewandowski, J. Wallenius, Lecture Notes in Economics and Mathematical Systems 356, Springer-Verlag, 1991, 207–216.

Clímaco J. N., C. H. Antunes, M. J. Alves, *Interactive Decision Support for Multiobjective Transportation Problems*, European Journal of Operational Research **65**, No. 1 (1993), 58–67.

Clinton R. J., M. D. Troutt, *Toward Optimal Scaling in the Method of Abstract Forces for Interactive Multiple Criteria Optimization*, Mathematical and Computer Modelling **11** (1988), 589–594.

Cohon J. L., "Multiobjective Programming and Planning," Academic Press Inc., New York, 1978.

__________, *Multicriteria Programming: Brief Review and Application*, in "Design Optimization," Edited by J. S. Gero, Academic Press, Inc., 1985, 163–191.

Colson G, C. De Bruyn, *Multiple Criteria Supports on Micro-Computers*, Belgian Journal of Operations Research, Statistics and Computer Science **27**, No. 1 (1987), 29–58.

Corley H. W., *A New Scalar Equivalence for Pareto Optimization*, IEEE Transactions on Automatic Control **ac-25**, No. 4 (1980), 829–830.

Crama Y., *Analysis of STEM-Like Solutions to Multi-Objective Programming Problems*, in "Multi-Objective Decision Making," Edited by S. French, R. Hartley, L. C. Thomas, D. J. White, Academic Press, 1983, 208–213.

Craven B. D., *On Sensitivity Analysis for Multicriteria Optimization*, Optimization **19** (1988), 513–523.

——————, *Nonsmooth Multiobjective Programming*, Numerical Functional Analysis and Optimization **10**, No. 1&2 (1989), 49–64.

Current J., M. Marsh, *Multiobjective Transportation Network Design and Routing Problems: Taxonomy and Annotation*, European Journal of Operational Research **65**, No. 1 (1993), 4–19.

Current J., H. Min, *Multiobjective Design of Transportation Networks: Taxonomy and Annotation*, European Journal of Operational Research **26** (1986), 187–201.

Da Cunha N. O., E. Polak, *Constrained Minimization Under Vector-Valued Criteria in Finite Dimensional Spaces*, Journal of Mathematical Analysis and Applications **19**, No. 1 (1967), 103–124.

Dauer J. P., M. S. A. Osman, *Decomposition of the Parametric Space in Multiobjective Convex Programs Using the Generalized Tchebycheff Norm*, Journal of Mathematical Analysis and Applications **107**, No. 1 (1985), 156–166.

Dauer J. P., W. Stadler, *A Survey of Vector Optimization in Infinite-Dimensional Spaces, Part 2*, Journal of Optimization Theory and Applications **51**, No. 2 (1986), 205–241.

De Kluyver C. A., *An Exploration of Various Goal Programming Formulations – with Application to Advertising Media Selection*, Journal of the Operational Research Society **30** (1979), 167–171.

deNeufville R., M. McCord, *Unreliable Measurement of Utility: Significant Problems for Decision Analysis*, in "Operational Research '84," Edited by J. P. Brans, Elsevier Science Publishers B.V. (North-Holland), Amsterdam, 1984, 464–476.

Deshpande D. V., S. Zionts, *Sensitivity Analysis in Multiple Objective Linear Programming: Changes in the Objective Function Matrix*, in "Multiple Criteria Decision Making Theory and Application," Edited by G. Fandel, T. Gal, Lecture Notes in Economics and Mathematical Systems 177, Springer-Verlag, Berlin, Heidelberg, 1980, 26–39.

Despontin M., J. Moscarola, J. Spronk, *A User-Oriented Listing of Multiple Criteria Decision Methods*, Revue Belge de Statistique, d'Informatique et de Recherche Operationnelle **23**, No. 4 (1983), 4–110.

Dessouky M. I., M. Ghiassi, W. J. Davis, *Estimates of the Minimum Nondominated Criterion Values in Multiple-Criteria Decision-Making*, Engineering Costs and Production Economics **10** (1986), 95–104.

Diaz A., *Interactive Solution to Multiobjective Optimization Problems*, International Journal for Numerical Methods in Engineering **24** (1987), 1865–1877.

Dinkelbach W., *Multicriteria Decision Models with Specified Goal Levels*, in "Multiple Criteria Decision Making Theory and Application," Edited by G. Fandel, T. Gal, Lecture Notes in Economics and Mathematical Systems 177, Springer-Verlag, Berlin, Heidelberg, 1980, 52–59.

Doležal J., *Necessary Conditions for Pareto Optimality in Nondifferentiable Problems*, Problems of Control and Information Theory **14**, No. 2 (1985), 131–140.

Durier R., *Weighting Factor Results in Vector Optimization*, Journal of Optimization Theory and Applications **58**, No. 3 (1988), 411–430.

Dyer J. S., *Interactive Goal Programming*, Management Science **19**, No. 1 (1972), 62–70.

————, *A Time-Sharing Computer Program for the Solution of the Multiple Criteria Problem*, Management Science **19**, No. 12 (1973(a)), 1379–1383.

————, *An Empirical Investigation of a Man-Machine Interactive Approach to the Solution of the Multiple Criteria Problem*, in "Multiple Criteria Decision Making," Edited by J. L. Cochrane, M. Zeleny, University of South Carolina Press, Columbia, South Carolina, 1973(b), 202–216.

————, *The Effects of Errors in the Estimation of the Gradient on the Frank-Wolfe Algorithm, with Implications for Interactive Programming*, Operations Research **22**, No. 1 (1974), 160–174.

Dyer J. S., P. C. Fishburn, R. E. Steuer, J. Wallenius, S. Zionts, *Multiple Criteria Decision Making, Multiattribute Utility Theory: The Next Ten Years*, Management Science **38**, No. 5 (1992), 645–654.

Dyer J. S., R. K. Sarin, *Multicriteria Decision Making*, in "Mathematical Programming for Operations Researchers and Computer Scientists," Edited by A. G. Holzman, Marcel Dekker Inc., 1981, 123–148.

Ecker J. G., I. A. Kouada, *Finding Efficient Points for Linear Multiple Objective Programs*, Mathematical Programming **8**, No. 3 (1975), 375–377.

Ecker J. G., N. E. Shoemaker, *Multiple Objective Linear Programming and the Tradeoff-Compromise Set*, in "Multiple Criteria Decision Making Theory and Application," Edited by G. Fandel, T. Gal, Lecture Notes in Economics and Mathematical Systems 177, Springer-Verlag, Berlin, Heidelberg, 1980, 60–73.

————————, *Selecting Subsets from the Set of Nondominated Vectors in Multiple Objective Linear Programming*, SIAM Journal on Control and Optimization **19**, No. 4 (1981), 505–515.

Edgeworth F. Y., "Mathematical Psychics," University Microfilms International (Out-of-Print Books on Demand), 1987 (the original edition in 1881).

Eiduks J., *Control of Complicated Systems in Case of Vectorial Indicator of Quality*, in "Fourth Formator Symposium on Mathematical Methods for the Analysis of Large-Scale Systems," Edited by J. Bebeš, L. Bakule, Academia, Prague, 1983, 165–180.

Eiselt H. A., G. Pederzoli, C.-L. Sandblom, "Continuous Optimization Models," Operations Research: Theory, Techniques, Applications, Walter de Gruyter & Co., Berlin, New York, 1987.

El Abdouni B., L. Thibault, *Lagrange Multipliers for Pareto Nonsmooth Programming Problems in Banach Spaces*, Optimization **26**, No. 3-4 (1992), 277–285.

Eom H. B., *The Current State of Multiple Criteria Decision Support Systems*, Human Systems Management **8** (1989), 113–119.

Eschenauer H. A., *Multicriteria Optimization Procedures in Application on Structural Mechanics Systems*, in "Recent Advances and Historical Developments of Vector Optimization," Edited by J. Jahn, W. Krabs, Lecture Notes in Economics and Mathematical Systems 294, Springer-Verlag, Berlin, Heidelberg, 1987, 345–376.

Eschenauer H., J. Koski, A. Osyczka (Eds.), "Multicriteria Design Optimization Procedures and Applications," Springer-Verlag, Berlin, Heidelberg, 1990.

Eschenauer H. A., A. Osyczka, E. Schäfer, *Interactive Multicriteria Optimization in Design Process*, in "Multicriteria Design Optimization Procedures and Applications," Edited by H. Eschenauer, J. Koski, A. Osyczka, Springer-Verlag, Heidelberg, 1990, 71–114.

Eschenauer H. A., E. Schäfer, H. Bernau, *Application of Interactive Vector Optimization Methods with Regard to Problems in Structural Mechanics*, in "Discretization Methods and Structural Optimization – Procedures and Applications," Edited by H. A. Eschenauer, G. Thierauf, Lecture Notes in Engineering, Springer-Verlag, Berlin, Heidelberg, 1989, 95–101.

Ester J., *About the Stability of the Solution of Multiple Objective Decision Making Problems with Generalized Dominating Sets*, Systems Analysis Modelling Simulation **1**, No. 4 (1984), 287–291.

Ester J., R. Holzmüller, *About Some Methods and Applications in Multicriteria Decision Making*, Systems Analysis Modelling Simulation **3**, No. 5 (1986), 425–438.

Ester J., F. Tröltzsch, *On Generalized Notions of Efficiency in Multicriteria Decision Making*, Systems Analysis Modelling Simulation **3**, No. 2 (1986), 153–161.

Evans G. W., *An Overview of Techniques for Solving Multiobjective Mathematical Programs*, Management Science **30**, No. 11 (1984), 1268–1282.

Evans J. P., R. E. Steuer, *Generating Efficient Extreme Points in Linear Multiple Objective Programming: Two Algorithms and Computing Experience*, in "Multiple Criteria Decision Making," Edited by J. L. Cochrane, M. Zeleny, University of South Carolina Press, Columbia, South Carolina, 1973, 349–365.

Federer H., "Geometric Measure Theory," Springer-Verlag, Berlin, Heidelberg, 1969.

Ferreira P. A. V., J. C. Geromel, *An Interactive Projection Method for Multicriteria Optimization Problems*, IEEE Transactions on Systems, Man, and Cybernetics **20**, No. 3 (1990), 596–605.

Fiala P., *Duality and Stability in Linear Vector Optimization*, Foundations of Control Engineering **8**, No. 3-4 (1983), 155–163.

Fishburn P. C., *Lexicographic Orders, Utilities and Decision Rules: A Survey*, Management Science **20**, No. 11 (1974), 1442–1471.

Frank M., P. Wolfe, *An Algorithm for Quadratic Programming*, Naval Research Logistics Quarterly **3**, No. 1-2 (1956), 95–110.

Friedman L., A. Mehrez, *Towards a Theory of Determining Size in Multi-Objective Service Systems*, European Journal of Operational Research **63**, No. 3 (1992), 398–408.

Friesz T. L., *Multiobjective Optimization in Transportation: The Case of Equilibrium Network Design*, in "Organizations: Multiple Agents with Multiple Criteria," Edited by N. N. Morse, Lecture Notes in Economics and Mathematical Systems 190, Springer-Verlag, Berlin, Heidelberg, 1981, 116–127.

Gal T., *On Efficient Sets in Vector Maximum Problems – A Brief Survey*, European Journal of Operational Research **24** (1986), 253–264.

Gal T., H. Leberling, *Relaxation Analysis in Linear Vectorvalued Maximization*, European Journal of Operational Research **8** (1981), 274–282.

Gass S., T. Saaty, *The Computational Algorithm for the Parametric Objective Function*, Naval Research Logistics Quarterly **2** (1955), 39–45.

Geoffrion A. M., *Proper Efficiency and the Theory of Vector Maximization*, Journal of Mathematical Analysis and Applications **22**, No. 3 (1968), 618–630.

Geoffrion A. M., J. S. Dyer, A. Feinberg, *An Interactive Approach for Multi-Criterion Optimization, with an Application to the Operation of an Academic Department*, Management Science **19**, No. 4 (1972), 357-368.

Geromel J. C., P. A. V. Ferreira, *An Upper Bound on Properly Efficient Solutions on Multiobjective Optimization*, Operations Research Letters **10**, No. 2 (1991), 83–86.

Gershon M., L. Duckstein, *An Algorithm for Choosing of a Multiobjective Technique*, in "Essays and Surveys on Multiple Criteria Decision Making," Edited by P. Hansen, Lecture Notes in Economics and Mathematical Systems 209, Springer-Verlag, Berlin, Heidelberg, 1983, 53–62.

Ghosh D., B. B. Pal, M. Basu, *Implementation of Goal Programming in Long-Range Resource Planning in University Management*, Optimization **24**, No. 3–4 (1992), 373–383.

Giokas D., M. Vassiloglou, *A Goal Programming Model for Bank Assets and Liabilities Management*, European Journal of Operational Research **50**, No. 1 (1991), 48–60.

Göpfert A., *Multicriterial Duality, Examples and Advances*, in "Large-Scale Modelling and Interactive Decision Analysis," Edited by G. Fandel, M. Grauer, A. Kurzhanski, A. P. Wierzbicki, Lecture Notes in Economics and Mathematical Systems 273, Springer-Verlag, 1986, 52–58.

Grauer M., *A Dynamic Interactive Decision Analysis and Support System (DIDASS), User's Guide*, Working Paper WP-83-60, IIASA, Laxenburg (1983(a)).

————, *Reference Point Optimization – The Nonlinear Case*, in "Essays and Surveys on Multiple Criteria Decision Making," Edited by P. Hansen, Lecture Notes in Economics and Mathematical Systems 209, Springer-Verlag, Berlin, Heidelberg, 1983(b), 126–135.

Grauer M., A. Lewandowski, A. Wierzbicki, *DIDASS – Theory, Implementation and Experiences*, in "Interactive Decision Analysis," Edited by M. Grauer, A. P. Wierzbicki, Lecture Notes in Economics and Mathematical Systems 229, Springer-Verlag, 1984, 22–30.

Graves S. B., J. L. Ringuest, J. F. Bard, *Recent Developments in Screening Methods for Nondominated Solutions in Multiobjective Optimization*, Computers and Operations Research **19**, No. 7 (1992), 683–694.

Guddat J., F. Guerra Vasquez, K. Tammer, K. Wendler, "Multiobjective and Stochastic Optimization Based on Parametric Optimization," Akademie-Verlag, Berlin, 1985.

Gulati T. R., M. A. Islam, *Proper Efficiency in a Linear Fractional Vector Maximum Problem with Generalized Convex Constraints*, European Journal of Operational Research **36**, No. 3 (1988), 339–345.

————————, *Efficiency and Proper Efficiency in Nonlinear Vector Maximum Problems*, European Journal of Operational Research **44** (1990), 373–382.

Haimes Y. Y., *The Surrogate Worth Trade-off (SWT) Method and Its Extensions*, in "Multiple Criteria Decision Making Theory and Application," Edited by G. Fandel, T. Gal, Lecture Notes in Economics and Mathematical Systems 177, Springer-Verlag, Berlin, Heidelberg, 1980, 85–108.

Haimes Y. Y., V. Chankong, *Kuhn-Tucker Multipliers as Trade-Offs in Multiobjective Decision-Making Analysis*, Automatica **15**, No. 1 (1979), 59–72.

Haimes Y. Y., W. A. Hall, *Multiobjectives in Water Resource Systems Analysis: The Surrogate Worth Trade Off Method*, Water Resources Research **10**, No. 4 (1974), 615–624.

Haimes Y. Y., W. A. Hall, H. T. Freedman, "Multiobjective Optimization in Water Resources Systems," Elsevier Scientific Publishing Company, Amsterdam, 1975.

Haimes Y. Y., L. S. Lasdon, D. A. Wismer, *On a Bicriterion Formulation of the Problems of Integrated System Identification and System Optimization*, IEEE Transactions on Systems, Man, and Cybernetics **smc-1** (1971), 296–297.

Haimes Y. Y., D. Li, *Hierarchical Multiobjective Analysis for Large-Scale Systems: Review and Current Status*, Automatica **24**, No. 1 (1988), 53–69.

Haimes Y. Y., K. Tarvainen, T. Shima, J. Thadathil, "Hierarchical Multiobjective Analysis of Large-Scale Systems," Hemisphere Publishing Corporation, New York, 1990.

Hall W. A., Y. Y. Haimes, *The Surrogate Worth Trade-Off Method with Multiple Decision-Makers*, in "Multiple Criteria Decision Making Kyoto 1975," Edited by M. Zeleny, Lecture Notes in Economics and Mathematical Systems 123, Springer-Verlag, Berlin, Heidelberg, 1976, 207–233.

Hallefjord Å., K. Jörnsten, *An Entroty Targent-Point Approach to Multiobjective Programming*, International Journal of Systems Science **17**, No. 4 (1986), 639–653.

Halme M., P. Korhonen, *Nondominated Tradeoffs and Termination in Interactive Multiple Objective Linear Programming*, in "Improving Decision Making in Organisations," Edited by A. G. Lockett, G. Islei, Lecture Notes in Economics and Mathematical Systems 335, Springer-Verlag, Berlin, Heidelberg, 1989, 410–423.

Hansen P., M. Labbé, R. E. Wendell, *Sensitivity Analysis in Multiple Objective Linear Programming: The Tolerance Approach*, European Journal of Operational Research **38** (1989), 63–69.

Harrison T. P., R. E. Rosenthal, *Optimization of Utility and Value Functions*, Naval Research Logistics **35**, No. 3 (1988), 411–418.

Hartley R., "Linear and Nonlinear Programming: An Introduction to Linear Methods in Mathematical Programming," Ellis Horwood Limited, 1985.

Haslinger J., P. Neittaanmäki, "Finite Element Approximation for Optimal Shape Design: Theory and Applications," John Wiley & Sons, Chichester, 1988.

Hazen G. B., *Differential Characterizations of Nonconical Dominance in Multiple Objective Decision Making*, Mathematics of Operations Research **13**, No. 1 (1988), 174–189.

Helbig S., *On the Connectedness of the Set of Weakly Efficient Points of a Vector Optimization Problem in Locally Convex Spaces*, Journal of Optimization Theory and Applications **65**, No. 2 (1990(a)), 257–270.

———, *On Some Interactive Methods for Quadratic Vector Optimization Problems*, Talk at the Conference "300 Jahre Math. Gesellschaft in Hamburg" (1990(b)).

Hemming T., "Multiobjective Decision Making under Certainty," Doctoral Thesis, The Economic Research Institute at the Stockholm School of Economics, Stockholm, 1978.

———, *Some Modifications of a Large Step Gradient Method for Interactive Multicriterion Optimization*, in "Organizations: Multiple Agents with Multiple Criteria," Edited by N. N. Morse, Lecture Notes in Economics and Mathematical Systems 190, Springer-Verlag, Berlin, Heidelberg, 1981, 128–139.

Henig M. I., *Existence and Characterization of Efficient Decisions with Respect to Cones*, Mathematical Programming **23**, No. 1 (1982(a)), 111–116.

———, *Proper Efficiency with Respect to Cones*, Jounal of Optimization Theory and Applications **36**, No. 3 (1982(b)), 387–407.

———, *The Domination Property in Multicriteria Optimization*, Journal of Mathematical Analysis and Applications **114** (1986), 7–16.

———, *Value Functions, Domination Cones and Proper Efficiency in Multicriteria Optimization*, Mathematical Programming **46** (1990), 205–217.

Hobbs B. F., *What Can We Learn From Experiments in Multiobjective Decision Analysis?*, IEEE Transactions on Systems, Man, and Cybernetics **smc-16**, No. 3 (1986), 384–394.

Hu Y., *Theory and Application of Multiobjective Optimization in China – A Survey*, Annals of Operations Research **24** (1990), 45–51.

Hwang C. L., A. S. M. Masud, "Multiple Objective Decision Making – Methods and Applications: A State-of-the-Art Survey," Lecture Notes in Economics and Mathematical Systems 164, Springer-Verlag, Berlin, Heidelberg, 1979.

Hwang C. L., S. R. Paidy, K. Yoon, A. S. M. Masud, *Mathematical Programming with Multiple Objectives: A Tutorial*, Computers and Operations Research **7** (1980), 5–31.

Hwang C. L., K. Yoon, "Multiple Attribute Decision Making Methods and Applications: A State-of-the-Art Survey," Lecture Notes in Economics and Mathematical Systems 186, Springer-Verlag, Berlin, Heidelberg, 1981.

Ignizio J. P., "Goal Programming and Extensions," Lexington Books, D.C. Heath and Company, 1976.

———, *Generalized Goal Programming*, Computers and Operations Research **10** (1983(a)), 277–289.

———, *An Approach to the Modeling and Analysis of Multiobjective Generalized Networks*, European Journal of Operational Research **12**, No. 4 (1983(b)), 357–361.

———, "Introduction to Linear Goal Programming," Sage Publications, Sage University Press, Beverly Hills, 1985(a).

———, *Multiobjective Mathematical Programming via the MULTIPLEX Model and Algorithm*, European Journal of Operational Research **22** (1985(b)), 338–346.

Inuiguchi M., Y. Kume, *Goal Programming Problems with Interval Coefficients and Target Intervals*, European Journal of Operational Research **52**, No. 3 (1991), 345–360.

Isermann H., *Proper Efficiency and the Linear Vector Maximization Problem*, Operations Research **22**, No. 1 (1974), 189–191.

Iserman H., R. E. Steuer, *Computational Experience Concerning Payoff Tables and Minimum Criterion Values over the Efficient Set*, European Journal of Operational Research **33**, No. 1 (1988), 91–97.

Ishizuka Y., K. Shimizu, *Necessary and Sufficient Conditions for the Efficient Solutions of Nondifferentiable Multiobjective Problems*, IEEE Transactions on Systems, Man, and Cybernetics **smc-14**, No. 4 (1984), 625–629.

Jahn J., *Scalarization in Vector Optimization*, Mathematical Programming **29** (1984), 203–218.

______, *Scalarization in Multi Objective Optimization*, in "Mathematics of Multi Objective Optimization," Edited by P. Serafini, Springer-Verlag, Wien, New York, 1985, 45–88.

______, "Mathematical Vector Optimization in Partially Ordered Linear Spaces," Methoden und Verfahren der mathematischen Physik, Band 31, Verlag Peter Lang GmbH, Frankfurt am Main, 1986(a).

______, *Existence Theorems in Vector Optimization*, Journal of Optimization Theory and Applications **50**, No. 3 (1986(b)), 397–406.

______, *Parametric Approximation Problems Arising in Vector Optimization*, Journal of Optimization Theory and Applications **54**, No. 3 (1987), 503–516.

Jarvenpaa S. L., *The Effect of Task Demands and Graphical Format on Information Processing Strategies*, Management Science **35**, No. 3 (1989), 285–303.

Jedrzejowicz P., L. Rosicka, *Multicriterial Reliability Optimization Problem*, Foundations of Control Engineering **8**, No. 3–4 (1983), 165–173.

Jelassi M. T., M. Jarke, E. A. Stohr, *Designing a Generalized Multiple Criteria Decision Support System*, in "Decision Making with Multiple Objectives," Edited by Y. Y. Haimes, V. Chankong, Lecture Notes in Economics and Mathematical Systems 242, Springer-Verlag, 1985, 214–235.

Jelassi M. T., V. M. Ozernoy, *A Framework for Building an Expert System for MCDM Models Selection*, in "Improving Decision Making in Organisations," Edited by A. G. Lockett, G. Islei, Lecture Notes in Economics and Mathematical Systems 335, Springer-Verlag, Berlin, Heidelberg, 1989, 553–562.

Jendo S., *Multicriteria Optimization of Single-Layer Cable System*, Archives of Mechanics **38**, No. 3 (1986), 219–234.

Jeyakumar V., X. Q. Yang, *Convex Composite Multi-Objective Nonsmooth Programming*, Mathematical Programming **59**, No. 3 (1993), 325–343.

Jurkiewicz E., *Stability of Compromise Solution in Multicriteria Decision-Making Problems*, Journal of Optimization Theory and Applications **40**, No. 1 (1983), 77–83.

Kacprzyk J., S. A. Orlovski (Eds.), "Optimization Models Using Fuzzy Sets and Possibility Theory," Theory and Decision Library, Series B: Mathematical and Statistical Methods, D. Reidel Publishing Company, 1987.

Kaliszewski I., *A Characterization of Properly Efficient Solutions by an Augmented Tchebycheff Norm*, Bulletin of the Polish Academy of Sciences – Technical Sciences **33**, No. 7–8 (1985), 415–420.

__________, *Norm Scalarization and Proper Efficiency in Vector Optimization*, Foundations of Control Engineering **11**, No. 3 (1986), 117–131.

__________, *A Modified Weighted Tchebycheff Metric for Multiple Objective Programming*, Computers and Operations Research **14**, No. 4 (1987), 315–323.

Kanniappan P., *Necessary Conditions for Optimality of Nondifferentiable Convex Multiobjective Programming*, Journal of Optimization Theory and Applications **40** (1983), 167–173.

___________, *A Procedure for Solving a Multiobjective Programming*, Manuscript via private communication (1988).

Kasanen E., R. Östermark, M. Zeleny, *Gestalt System of Holistic Graphics: New Management Support View of MCDM*, Computers and Operations Research **18**, No. 2 (1991), 233–239.

Kaul R. N., V. Lyall, S. Kaur, *Semilocal Pseudolinearity and Efficiency*, European Journal of Operational Research **36**, No. 3 (1988), 402–409.

Keeney R. L., H. Raiffa, "Decisions with Multiple Objectives: Preferences and Value Tradeoffs," John Wiley & Sons, Inc., 1976.

Kim S. H., T. Gal, *A New Interactive Algorithm for Multi-Objective Linear Programming Using Maximally Changeable Dominance Cone*, European Journal of Operational Research **64**, No. 1 (1993), 126–137.

Kirilov L., *The Reference Direction Approach*, Manuscript via private communication (1991).

Kiwiel K. C., *An Aggregate Subgradient Descent Method for Solving Large Convex Nonsmooth Multiobjective Minimization Problems*, in "Large Scale Systems: Theory and Applications 1983," Edited by A. Straszak, Pergamon Press, 1984, 283–288.

__________, *A Descent Method for Nonsmooth Convex Multiobjective Minimization*, Large Scale Systems **8**, No. 2 (1985(a)), 119–129.

__________, *An Algorithm for Linearly Constrained Nonsmooth Convex Multiobjective Minimization*, in "Systems Analysis and Simulation 1985 Part I: Theory and Foundations," Edited by A. Sydow, S. M. Thoma, R. Vichnevetsky, Akademie-Verlag, Pergamon Press, Berlin, 1985(b), 236–238.

__________, "Methods of Descent for Nondifferentiable Optimization," Lecture Notes in Mathematics 1133, Springer-Verlag, Berlin, Heidelberg, 1985(c).

__________, *A Method for Solving Certain Quadratic Programming Problems Arising in Nonsmooth Optimization*, IMA Journal of Numerical Analysis **6** (1986), 137–152.

__________, *Proximity Control in Bundle Methods for Convex Nondifferentiable Optimization*, Mathematical Programming **46** (1990), 105-122.

Kiwiel K. C., A. Stachurski, *Issues of Effectiveness Arising in the Design of a System of Nondifferentiable Optimization Algorithms*, in "Aspiration Based Decision Support Systems: Theory, Software and Applications," Edited by A. Lewandowski, A. Wierzbicki, Lecture Notes in Economics and Mathematical Systems 331, Springer-Verlag, 1989, 180–192.

Klimberg R., *GRADS: A New Graphical Display System for Visualizing Multiple Criteria Solutions*, Computers and Operations Research **19**, No. 7 (1992), 707–711.

Klinger A., *Improper Solutions of the Vector Maximum Problem*, Operations Research **15**, No. 3 (1967), 570–572.

Kok M., *Scalarization and the Interface with Decision Makers in Interactive Multi Objective Linear Programming*, in "Mathematics of Multi Objective Optimization," Edited by P. Serafini, Springer-Verlag, Wien, New York, 1985, 433–438.

Kok M., F. A. Lootsma, *Pairwise-Comparison Methods in Multiple Objective Programming, with Applications in a Long-Term Energy-Planning Model*, European Journal of Operational Research **22**, No. 1 (1985), 44–55.

Koksalan M. M., C. Beseli, *A Variation of the Methods of Zionts and Wallenius*, in "Multiple Criteria Decision Making and Risk Analysis Using Microcomputers," Edited by B. Karpak, S. Zionts, Springer-Verlag, Berlin, Heidelberg, 1989, 283–291.

Koopmans T. C., *Analysis and Production as an Efficient Combination of Activities*, in "Activity Analysis of Production and Allocation," Edited by T. C. Koopmans, Yale University Press, New Haven, London, 1971 (originally published in 1951), 33–97.

Korhonen P. J., *Solving Discrete Multiple Criteria Problems by Using Visual Interaction*, in "Large-Scale Modelling and Interactive Decision Analysis," Edited by G. Fandel, M. Grauer, A. Kurzhanski, A. P. Wierzbicki, Lecture Notes in Economics and Mathematical Systems 273, Springer-Verlag, 1986, 176–185.

____________, *VIG (A Visual Interactive Approach to Goal Programming), User's Guide*, Num-Plan (1987).

____________, *The Multiobjective Linear Programming Decision Support System VIG and Its Applications*, in "Readings in Multiple Criteria Decision Aid," Edited by C. A. Bana e Costa, Springer-Verlag, Berlin, Heidelberg, 1990, 471–491.

____________, *Two Decision Support Systems for Continuous and Discrete Multiple Criteria Decision Making: VIG and VIMDA*, in "Methodology, Implementation and Applications of Decision Support Systems," Edited by A. Lewandowski, P. Serafini, M. G. Speranza, International Centre for Mechanical Sciences, Courses and Lectures No. 320, Springer-Verlag, Wien, New York, 1991(a), 85–103.

____________, *Using Harmonious Houses for Visual Pairwise Comparison of Multiple Criteria Alternatives*, Decision Support Systems **7**, No. 1 (1991(b)), 47–54.

Korhonen P., M. Halme, *Using Lexicographic Parametric Programming for Searching a Nondominated Set in Multiple Objective Linear Programming*, Manuscript via private communication (1994).

Korhonen P., J. Laakso, *A Visual Interactive Method for Solving the Multiple-Criteria Problem*, in "Interactive Decision Analysis," Edited by M. Grauer, A. P. Wierzbicki, Lecture Notes in Economics and Mathematical Systems 229, Springer-Verlag, 1984, 146–153.

______________, *On Developing a Visual Interactive Multiple Criteria Method – An Outline*, in "Decision Making with Multiple Objectives," Edited by Y. Y. Haimes, V. Chankong, Lecture

Notes in Economics and Mathematical Systems 242, Springer-Verlag, Berlin, Heidelberg, 1985, 272–281.

_______________, *A Visual Interactive Method for Solving the Multiple Criteria Problem*, European Journal of Operational Research **24** (1986(a)), 277–287.

_______________, *Solving Generalized Goal Programming Problems Using a Visual Interactive Approach*, European Journal of Operational Research **26** (1986(b)), 355–363.

Korhonen P., H. Moskowitz, J. Wallenius, *Choice Behaviour in Interactive Multiple-Criteria Decision Making*, Annals of Operations Research **23** (1990), 161–179.

_______________, *Multiple Criteria Decision Support – A Review*, European Journal of Operational Research **63**, No. 3 (1992), 361–375.

Korhonen P., H. Moskowitz, J. Wallenius, S. Zionts, *An Interactive Approach to Multiple Criteria Optimization with Multiple Decision-Makers*, Naval Research Logistics Quarterly **33**, No. 4 (1986), 589–602.

Korhonen P., M. Soismaa, *A Multiple Criteria Model for Pricing Alcoholic Beverages*, European Journal of Operational Research **37** (1988), 165–175.

Korhonen P., J. Wallenius, *A Modification of the Zionts-Wallenius Multiple Criteria Method for Nonlinear Utility Functions*, Manuscript via private communication (1984).

_______________, *A Pareto Race*, Naval Research Logistics **35** (1988), 615–623.

_______________, *A Careful Look at Efficiency and Utility in Multiple Criteria Decision Making: A Tutorial*, Asia–Pacific Journal of Operational Research **6**, No. 1 (1989(a)), 46–62.

_______________, *Observations Regarding Choice Behaviour in Interactive Multiple Criteria Decision-Making Environments: An Experimental Investigation*, in "Methodology and Software for Interactive Decision Support," Edited by A. Lewandowski, I. Stanchev, Lecture Notes in Economics and Mathematical Systems 337, Springer-Verlag, 1989(b), 163–170.

_______________, *VIG – A Visual and Dynamic Decision Support System for Multiple Objective Linear Programming*, in "Multiple Criteria Decision Making and Risk Analysis Using Microcomputers," Edited by B. Karpak, S. Zionts, Springer-Verlag, Berlin, Heidelberg, 1989(c), 251–281.

_______________, *A Multiple Objective Linear Programming Decision Support System*, Decision Support Systems **6**, No. 3 (1990), 243–251.

Korhonen P., J. Wallenius, S. Zionts, *Solving the Discrete Multiple Criteria Problem Using Convex Cones*, Management Science **30**, No. 11 (1984), 1336–1345.

_______________, *A Computer Graphics-Based Decision Support System for Multiple Objective Linear Programming*, European Journal of Operational Research **60**, No. 3 (1992), 280–286.

Kornbluth J. S. H., *A Survey of Goal Programming*, OMEGA **1**, No. 2 (1973), 193–205.

Koski J., R. Silvennoinen, *Norm Methods and Partial Weighting in Multicriterion Optimization of Structures*, International Journal for Numerical Methods in Engineering **24**, No. 6 (1987), 1101–1121.

Kostreva M. M., T. J. Ordoyne, M. Wiecek, *Multiple-Objective Programming with Polynomial Objectives and Constraints*, European Journal of Operational Research **57**, No. 3 (1992), 381–394.

Kreglewski T., *Nonlinear Optimization Techniques in Decision Support Systems*, in "Aspiration Based Decision Support Systems: Theory, Software and Applications," Edited by A. Lewandowski, A. Wierzbicki, Lecture Notes in Economics and Mathematical Systems 331, Springer-Verlag, 1989, 158–171.

Kreglewski T., J. Granat, A. P. Wierzbicki, *IAS-DIDAS-N: A Dynamic Interactive Decision Analysis and Support System for Multicriteria Analysis of Nonlinear Models, v. 4.0*, Collaborative Paper CP-91-010, IIASA, Laxenburg (1991).

Kreglewski T., J. Paczynski, A. P. Wierzbicki, *IAC-DIDAS-N A Dynamic Interactive Decision Analysis and Support System for Multicriteria Analysis of Nonlinear Models on Professional Microcomputers*, in "Theory, Software and Testing Examples for Decision Support Systems," Edited by A. Lewandowski, A. Wierzbicki, Working Paper WP-87-26, IIASA, Laxenburg, Austria, 1987, 177–192.

Kuhn H. W., A. W. Tucker, *Nonlinear Programming*, in "Proceedings of the Second Berkeley Symposium on Mathematical Statistics and Probability," Edited by J. Neyman, University of California Press, Berkeley, Los Angeles, 1951, 481–492.

Kumar P., N. Singh, N. K. Tewari, *A Nonlinear Goal Programming Model for Multistage, Multiobjective Decision Problems with Application to Grouping and Loading Problem in a Flexible Manufacturing System*, European Journal of Operational Research **53**, No. 2 (1991), 166–171.

Laitinen E. J., "On the Simulation and Control of the Continuous Casting Process," PhD Thesis, University of Jyväskylä, Department of Mathematics, Jyväskylä, 1989.

Larichev O. I., O. A. Polyakov, A. O. Nikiforov, *Multicriterion Linear Programming Problems*, Journal of Economic Psychology **8** (1987), 389–407.

Lazimy R., *Solving Multiple Criteria Problems by Interactive Decomposition*, Mathematical Programming **35** (1986(a)), 334–361.

————, *Interactive Relaxation Method for a Broad Class of Integer and Continuous Nonlinear Multiple Criteria Problems*, Journal of Mathematical Analysis and Applications **116** (1986(b)), 553–573.

Lee G. M., *On Efficiency in Nonlinear Fractional Vector Maximization Problem*, Optimization **25**, No. 1 (1992), 47–52.

Lee S. M., *Goal Programming*, in "Mathematical Programming for Operations Researchers and Computer Scientists," Edited by A. G. Holzman, Marcel Dekker Inc., 1981, 101–122.

Leschine T. M., H. Wallenius, W. A. Verdini,*Interactive Multiobjective Analysis and Assimilative Capasity-Based Ocean Disposal Decisions*, European Journal of Operational Research **56**, No. 2 (1992), 278–289.

Levary R. R., *Optimal Control Problems with Multiple Goal Objectives*, Optimal Control Applications & Methods **7** (1986), 201–207.

Lewandowski A., *Man-Machine Dialogue Interfaces in Decision Support Systems*, in "Large-Scale Modelling and Interactive Decision Analysis," Edited by G. Fandel, M. Grauer, A. Kurzhanski, A. P. Wierzbicki, Lecture Notes in Economics and Mathematical Systems 273, Springer-Verlag, 1986, 161–175.

Lewandowski A., J. Granat, *Dynamic BIPLOT as the Interaction Interface for Aspiration Based Decision Support Systems*, in "Multiple Criteria Decision Support," Edited by P. Korhonen, A. Lewandowski, J. Wallenius, Lecture Notes in Economics and Mathematical Systems 356, Springer-Verlag, 1991, 229–241.

Lewandowski A., M. Grauer, *The Reference Point Optimization Approach – Methods of Efficient Implementation*, Working Paper WP-82-26, IIASA, Laxenburg (1982).

Lewandowski A., T. Kreglewski, T. Rogowski, A. P. Wierzbicki, *Decision Support Systems of DIDAS Family (Dynamic Interactive Decision Analysis & Support)*, in "Theory, Software and Testing Examples for Decision Support Systems," Edited by A. Lewandowski, A. Wierzbicki, Working Paper WP-87-26, IIASA, Laxenburg, 1987, 4–26.

Lewandowski A., T. Rogowski, T. Kreglewski, *A Trajectory-Oriented Extensions of DIDASS and Its Applications*, in "Plural Rationality and Interactive Decision Processes," Edited by M. Grauer, M. Thompson, A. P. Wierzbicki, Lecture Notes in Mathematics and Economical Systems 248, Springer-Verlag, Berlin, Heidelberg, 1985(a), 261–268.

————————, *Application of DIDASS Methodology to a Flood Control Problem – Numerical Experiments*, in "Plural Rationality and Interactive Decision Processes," Edited by M. Grauer, M. Thompson, A. P. Wierzbicki, Lecture Notes in Mathematics and Economical Systems 248, Springer-Verlag, Berlin, Heidelberg, 1985(b), 325–328.

Li D., Y. Y. Haimes, *The Envelope Approach for Multiobjective Optimization Problems*, IEEE Transactions on Systems, Man, and Cybernetics **smc-17**, No. 6 (1987), 1026–1038; *Correction to "The Envelope Approach for Multiobjective Optimization Problems"*, IEEE Transactions on Systems, Man, and Cybernetics **18**, No. 2 (1988), p. 332.

————————, *Multiobjective Dynamic Programming: The State of the Art*, Control-Theory and Advanced Technology **5**, No. 4 (1989), 471–483.

Lieberman E. R., *Soviet Multi-Objective Programming Methods: An Overview*, in "Multiobjective Problems of Mathematical Programming," Edited by A. Lewandowski, V. Volkovich, Lecture Notes in Economics and Mathematical Systems 351, Springer-Verlag, 1991(a), 21–31.

———————, *Soviet Multi-Objective Mathematical Programming Methods: An Overview*, Management Science **37**, No. 9 (1991(b)), 1147–1165.

Lin J. G., *Multiple-Objective Problems: Pareto-Optimal Solutions by Methods of Proper Equality Constraints*, IEEE Transactions on Automatic Control **ac-21**, No. 5 (1976(a)), 641–650.

————, *Three Methods for Determining Pareto-Optimal Solutions of Multiple-Objective Problems*, in "Directions in Large-Scale Systems, Many-Person Optimization and Decentralized Control," Edited by Y. C. Ho, S. K. Mitter, Plenum Press, New York, 1976(b), 117–138.

Loganathan G. V., H. D. Sherali, *A Convergent Interactive Cutting-Plane Algorithm for Multiobjective Optimization*, Operation Research **35**, No. 3 (1987), 365–377.

Lotfi V., J. E. Teich, *Multiple Criteria Decision Making Using Personal Computers*, Working Paper No. 727, State University of New York at Buffalo, Buffalo, New York (1989).

Lowe T. J., J.-F. Thisse, J. E. Ward, R. E. Wendell, *On Efficient Solutions to Multiple Objective Mathematical Programs*, Management Science **30**, No. 11 (1984), 1346–1349.

Luc D. T., *On the Domination Property in Vector Optimization*, Journal of Optimization Theory and Applications **43**, No. 2 (1984(a)), 327–330.

————, *On Duality Theory in Multiobjective Programming*, Journal of Optimization Theory and Applications **43**, No. 4 (1984(b)), 557–582.

————, *On Scalarizing Method in Vector Optimization*, in "Large-Scale Modelling and Interactive Decision Analysis," Edited by G. Fandel, M. Grauer, A. Kurzhanski, A. P. Wierzbicki, Lecture Notes in Economics and Mathematical Systems 273, Springer-Verlag, 1986, 46–51.

————, *About Duality and Alternative in Multiobjective Optimization*, Journal of Optimization Theory and Applications **53**, No. 2 (1987), 303–307.

————, "Theory of Vector Optimization," Lecture Notes in Economics and Mathematical Systems 319, Springer-Verlag, Berlin, Heidelberg, 1989.

————, *Recession Cones and the Domination Property in Vector Optimization*, Mathematical Programming **49**, No. 1 (1990), 113–122.

Lucchetti R., *Some Existence Results and Stability in Multi Objective Optimization*, in "Mathematics of Multi Objective Optimization," Edited by P. Serafini, Springer-Verlag, Wien, New York, 1985, 179–187.

Luenberger D. G., "Linear and Nonlinear Programming," Second Edition, Addison-Wesley Publishing Company, Inc., 1984.

MacCrimmon K. R., *An Overview of Multiple Objective Decision Making*, in "Multiple Criteria Decision Making," Edited by J. L. Cochrane, M. Zeleny, University of South Carolina Press, Columbia, South Carolina, 1973, 18–44.

Mäkelä M. M., "Nonsmooth Optimization: Theory and Algorithms with Applications to Optimal Control," PhD Thesis, University of Jyväskylä, Department of Mathematics, Jyväskylä, 1990.

————, *Issues of Implementing a Fortran Subroutine Package NSOLIB for Nonsmooth Optimization*, Report 5/1993, University of Jyväskylä, Department of Mathematics, Laboratory of Scientific Computing, Jyväskylä (1993).

Mäkelä M. M., T. Männikkö, *Numerical Solution of Nonsmooth Optimal Control Problems with an Application to the Continuous Casting Process*, Preprint 147, University of Jyväskylä, Department of Mathematics, Jyväskylä (1992).

————, *Numerical Solution of Nonsmooth Optimal Control Problems with an Application to the Continuous Casting Process*, Advances in Mathematical Sciences and Applications (to appear).

Mäkelä M. M., P. Neittaanmäki, "Nonsmooth Optimization: Analysis and Algorithms with Applications to Optimal Control," World Scientific Publishing Co., Singapore, 1992.

Mañas M., *Graphical Methods of Multicriterial Optimization*, Zeitschrift für Angewandte Mathematik und Mechanik **62**, No. 5 (1982), 375–377.

Mangasarian O. L., "Nonlinear Programming," McGraw-Hill, Inc., 1969.

Martein L., *Some Results on Regularity in Vector Optimization*, Optimization **20** (1989), 787–798.

Martel J.-M., B. Aouni, *Incomporating the Decision-Maker's Preferences in the Goal-Programming Model*, Journal of the Operational Research Society **41**, No. 12 (1990), 1121–1132.

Martínez-Legaz J.-E., *Lexicographical Order and Duality in Multiobjective Programming*, European Journal of Operational Research **33** (1988), 342–348.

Martínez-Legaz J.-E., I. Singer, *Surrogate Duality for Vector Optimization*, Numerical Functional Analysis and Optimization **9** (1987), 547–568.

Maruściac I., *On Fritz John Type Optimality Criterion in Multi-Objective Optimization*, Revue d'Analyse Numérique et de Théorie de l'Approximation **11**, No. 1–2 (1982), 109–114.

Masud A. S. M., C. L. Hwang, *Interactive Sequential Goal Programming*, Journal of the Operational Research Society **32** (1981), 391–400.

Masud A. S. M., X. Zheng, *An Algorithm for Multiple-Objective Non-Linear Programming*, Journal of the Operational Research Society **40**, No. 10 (1989), 895–906.

Meisel W. S., *Tradeoff Decision in Multiple Criteria Decision Making*, in "Multiple Criteria Decision Making," Edited by J. L. Cochrane, M. Zeleny, University of South Carolina Press, Columbia, South Carolina, 1973, 461–476.

Michalowski W., *Evaluation of a Multiple Criteria Interactive Programming Approach: An Experiment*, INFOR **25**, No. 2 (1987), 165–173.

Michalowski W., T. Szapiro, *A Bi-Reference Procedure for Interactive Multiple Criteria Programming*, Operations Research **40**, No. 2 (1992), 247–258.

Miettinen K., "On Multiobjective Optimization," PhLic Thesis, (in Finnish), University of Jyväskylä, Department of Mathematics, Jyväskylä, 1990.

Miettinen K., M. M. Mäkelä, *Nonsmooth Multicriteria Optimization Applied to Optimal Control*, Reports on Applied Mathematics and Computing, No. 6, University of Jyväskylä, Jyväskylä (1991).

——————, *An Interactive Method for Nonsmooth Multiobjective Optimization with an Application to Optimal Control*, Optimization Methods and Software **2** (1993), 31–44.

——————, *A Nondifferentiable Multiple Criteria Optimization Method Applied to Continuous Casting Process*, in "Proceedings of the Seventh European Conference on Mathematics in Industry," Edited by A. Fasano, M. Primicerio, B. G. Teubner, Stuttgart, 1994, 255–262.

Miller G. A., *The Magical Number Seven, Plus or Minus Two: Some Limits on Our Capacity for Processing Information*, Psychological Review **63**, No. 2 (1956), 81–87.

Minami M., *Weak Pareto Optimality of Multiobjective Problem in a Banach Space*, Bulletin of Mathematical Statistics **19**, No. 3–4 (1980–81), 19–23.

——————, *Weak Pareto Optimality of Multiobjective Problcms in a Locally Convex Linear Topological Space*, Journal of Optimization Theory and Applications **34**, No. 4 (1981), 469–484.

——————, *Weak Pareto-Optimal Necessary Conditions in a Nondifferentiable Multiobjective Program on a Banach Space*, Journal of Optimization Theory and Applications **41**, No. 3 (1983), 451–461.

Mitra A., J. G. Patankar, *A Multi-Objective Model for Warranty Estimation*, European Journal of Operational Research **45**, No. 2–3 (1990), 347–355.

Miyaji I., K. Ohno, H. Mine, *Solution Method for Partitioning Students into Groups*, European Journal of Operational Research **33**, No. 1 (1988), 82–90.

Moldavskiy M. A., *Singling Out a Set of Undominated Solutions in Continuous Vector Optimization Problems*, Soviet Automatic Control **14**, No. 5 (1981), 47–53.

Monarchi D. E., C. C. Kisiel, L. Duckstein, *Interactive Multiobjective Programming in Water Resources: A Case Study*, Water Resources Research **9**, No. 4 (1973), 837–850.

Monarchi D. E., J. E. Weber, L. Duckstein, *An Interactive Multiple Objective Decision-Making Aid Using Nonlinear Goal Programming*, in "Multiple Criteria Decision Making Kyoto 1975," Edited by M. Zeleny, Lecture Notes in Economics and Mathematical Systcms 123, Springer-Verlag, Berlin, Heidelberg, 1976, 235–253.

Mote J., D. L. Olson, M. A. Venkataramanan, *A Comparative Multiobjective Programming Study*, Mathematical and Computer Modelling **10**, No. 10 (1988), 719–729.

M'silti A., P. Tolla, *An Interactive Multiobjective Nonlinear Programming Procedure*, European Journal of Operational Research **64**, No. 1 (1993), 115–125.

Mukai H., *Alqorithms for Multicriterion Optimization*, IEEE Transactions on Automatic Control **ac-25**, No. 2 (1980), 177–186.

Musselman K., J. Talavage, *A Tradeoff Cut Approach to Multiple Objective Optimization*, Operations Research **28**, No. 6 (1980), 1424–1435.

Nakayama H., *Geometric Consideration of Duality in Vector Optimization*, Journal of Optimization Theory and Applications **44**, No. 4 (1984), 625–655.

——————, *On the Components in Interactive Multiobjective Programming Methods*, in "Plural Rationality and Interactive Decision Processes," Edited by M. Grauer, M. Thompson, A. P. Wierzbicki, Lecture Notes in Mathematics and Economical Systems 248, Springer-Verlag, Berlin, Heidelberg, 1985(a), 234–247.

——————, *Lagrange Duality and Its Geometric Interpretation*, in "Mathematics of Multi Objective Optimization," Edited by P. Serafini, Springer-Verlag, Wien, New York, 1985(b), 105–127.

——————, *Duality Theory in Vector Optimization: An Overview*, in "Decision Making with Multiple Objectives," Edited by Y. Y. Haimes, V. Chankong, Lecture Notes in Economics and Mathematical Systems 242, Springer-Verlag, 1985(c), 109–125.

——————, *Sensitivity and Trade-Off Analysis in Multiobjective Programming*, in "Methodology and Software for Interactive Decision Support," Edited by A. Lewandowski, I. Stanchev, Lecture Notes in Economics and Mathematical Systems 337, Springer-Verlag, 1989, 86–93.

——————, *Satisficing Trade-Off Method for Problems with Multiple Linear Fractional Objectives and Its Applications*, in "Multiobjective Problems of Mathematical Programming," Edited by A. Lewandowski, V. Volkovich, Lecture Notes in Economics and Mathematical Systems 351, Springer-Verlag, 1991(a), 42–50.

———————, *Trade-off Analysis Based upon Parametric Optimization*, in "Multiple Criteria Decision Support," Edited by P. Korhonen, A. Lewandowski, J. Wallenius, Lecture Notes in Economics and Mathematical Systems 356, Springer-Verlag, 1991(b), 42–52.

———————, *Trade-Off Analysis Using Parametric Optimization Techniques*, European Journal of Operational Research **60**, No. 1 (1992), 87–98.

Nakayama H., K. Furukawa, *Satisficing Trade-Off Method with an Application to Multiobjective Structural Design*, Large Scale Systems **8** (1985), 47–57.

Nakayama H., J. Nomura, K. Sawada, R. Nakajima, *An Application of Satisficing Trade-Off Method to a Blending Problem of Industrial Materials*, in "Large-Scale Modelling and Interactive Decision Analysis," Edited by G. Fandel, M. Grauer, A. Kurzhanski, A. P. Wierzbicki, Lecture Notes in Economics and Mathematical Systems 273, Springer-Verlag, 1986, 303–313.

Nakayama H., Y. Sawaragi, *Satisficing Trade-Off Method for Multiobjective Programming*, in "Interactive Decision Analysis," Edited by M. Grauer, A. P. Wierzbicki, Lecture Notes in Economics and Mathematical Systems 229, Springer-Verlag, 1984, 113–122.

Nakayama H., T. Tanino, Y. Sawaragi, *An Interactive Optimization Method in Multicriteria Decisionmaking*, IEEE Transactions on Systems, Man, and Cybernetics **smc-10**, No. 3 (1980), 163–169.

Narula S. C., H. R. Weistroffer, *Algorithms for Multi-Objective Nonlinear Programming Problems: An Overview*, in "Improving Decision Making in Organisations," Edited by A. G. Lockett, G. Islei, Lecture Notes in Economics and Mathematical Systems 335, Springer-Verlag, Berlin, Heidelberg, 1989(a), 434–443.

———————, *A Flexible Method for Nonlinear Multicriteria Decisionmaking Problems*, IEEE Transactions on Systems, Man, and Cybernetics **19**, No. 4 (1989(b)), 883–887.

Neittaanmäki P., D. Tiba, "Optimal Control of Nonlinear Parabolic Systems: Theory, Algorithms, and Applications," Marcel Dekker, Inc., New York, 1994.

Ng K. Y., *A Multicriteria Optimization Approach to Aircraft Loading*, Operations Research **40**, No. 6 (1992), 1200–1206.

Nijkamp P., P. Rietveld, J. Spronk, *Open Problems in the Operationalization of Multiple Criteria Decision Making*, Systems Analysis Modelling Simulation **5**, No. 4 (1988), 311–322.

Nijkamp P., J. Spronk, *Interactive Multiple Goal Programming: An Evaluation and Some Results*, in "Multiple Criteria Decision Making Theory and Applications," Edited by G. Fandel, T. Gal, Lecture Notes in Economics and Mathematical Systems 177, Springer-Verlag, Berlin, Heidelberg, 1980, 278–293.

Olbrisch M., *The Interactive Reference Point Approach as a Solution Concept for Econometric Decision Models*, in "X. Symposium on Operations Research," Part 1, Sections 1-5, Edited by M. J. Beckmann, K.-W. Gaede, K. Ritter, H. Schneeweiss, Verlag Anton Hain Meisenheim GmbH, Königstein/Ts., 1986, 611–619.

Olkucu A., *Conflicting Objectives in Software System Design*, in "Multiple Criteria Decision Making and Risk Analysis Using Microcomputers," Edited by B. Karpak, S. Zionts, Springer-Verlag, Berlin, Heidelberg, 1989, 357–394.

Olson D. L., *Review of Empirical Studies in Multiobjective Mathematical Programming: Subject Reflection of Nonlinear Utility and Learning*, Decision Sciences **23**, No. 1 (1992), 1–20.

———————, *Tchebycheff Norms in Multi-Objective Linear Programming*, Mathematical and Computer Modelling **17**, No. 1 (1993), 113–124.

Oppenheimer K. R., *A Proxy Approach to Multi-Attribute Decision Making*, Management Science **24**, No. 6 (1978), 675–689.

Osman M. S. A., A. M. Ragab, *A Unified Approach for Characterizing the Noninferior Solutions of Multiobjective Nonlinear Programming Problems*, in "X. Symposium on Operations Research," Part 1, Sections 1-5, Edited by M. J. Beckmann, K.-W. Gaede, K. Ritter, H. Schneeweiss, Verlag Anton Hain Meisenheim GmbH, Königstein/Ts., 1986(a), 133–141.

———————, *An Algorithm for Solving Multiobjective Nonlinear Programming Problems*, in "X. Symposium on Operations Research," Part 1, Sections 1-5, Edited by M. J. Beckmann, K.-W. Gaede, K. Ritter, H. Schneeweiss, Verlag Anton Hain Meisenheim GmbH, Königstein/Ts., 1986(b), 143–147.

Osyczka A., "Multicriterion Optimization in Engineering with FORTRAN programs," Ellis Horwood Limited, 1984.

——————, *Multicriterion Decision Making with Min-Max Approach*, in "Optimization Methods in Structural Design," Edited by H. Eschenauer, N. Olhoff, Euromech-Colloquium 164, Wissenschaftsverlag, 1989(a).

——————, *Computer Aided Multicriterion Optimization System (CAMOS)*, in "Discretization Methods and Structural Optimization – Procedures and Applications," Edited by H. A. Eschenauer, G. Thierauf, Lecture Notes in Engineering, Springer-Verlag, Berlin, Heidelberg, 1989(b), 263–270.

——————, "Computer Aided Multicriterion Optimization System (CAMOS): Software Package in FORTRAN," International Software Publishers, 1992.

Osyczka A., J. Koski, *Selected Works Related to Multicriterion Optimization Methods for Engineering Design*, in "Optimization Methods in Structural Design," Edited by H. Eschenauer, N. Olhoff, Euromech-Colloquium 164, Wissenschaftsverlag, 1989.

Osyczka A., J. Zajac, *Multicriteria Optimization of Computationally Expensive Functions and Its Application to Robot Spring Balancing Mechanism Design*, in "Multicriteria Design Optimization Procedures and Applications," Edited by H. Eschenauer, J. Koski, A. Osyczka, Springer-Verlag, Heidelberg, 1990, 168–183.

Ozernoy V. M., *Choosing the "Best" Multiple Criteria Decision-Making Method*, INFOR **30**, No. 2 (1992), 159–171.

Papageorgiou N. S., *Pareto Efficiency in Locally Convex Spaces II: Stability Results*, Numerical Functional Analysis and Optimization **8**, No. 1 (1985), 117–136.

Pareto V., "Cours d'Economie Politique," Libraire Droz, Genève, 1964 (the first edition in 1896).

——————, "Manual of Political Economy," The MacMillan Press Ltd, 1971 (the original edition in French in 1927).

Pascoletti A., P. Serafini, *Scalarizing Vector Optimization Problems*, Journal of Optimization Theory and Applications **42**, No. 4 (1984), 499–524.

Passy U., Y. Levanon, *Analysis of Multiobjective Decision Problems by the Indifference Band Approach*, Journal of Optimization Theory and Applications **43**, No. 2 (1984), 205–235.

Peterson E. S., *Applications of Multiobjective Decision Making in the Mining Industry*, in "Papers on Nonlinear Analysis and Mathematical Systems Theory," Edited by J. Szép, P. Tallós, Department of Mathematics, Karl Marx University of Economics, Budapest, 1984, 37–65.

Philip J., *Algorithms for the Vector Maximization Problem*, Mathematical Programming **2** (1972), 207–229.

Polak E., A. N. Payne, *On Multicriteria Optimization*, in "Directions in Large-Scale Systems, Many-Person Optimization and Decentralized Control," Edited by Y. C. Ho, S. K. Mitter, Plenum Press, New York, 1976, 77–94.

Preda V., *On Efficiency and Duality for Multiobjective Programs*, Journal of Mathematical Analysis and Applications **166**, No. 2 (1992), 365–377.

Ramesh R., M. H. Karwan, S. Zionts, *Theory of Convex Cones in Multicriteria Decision Making*, Annals of Operations Research **16** (1988), 131–148.

——————, *Interactive Multicriteria Linear Programming: An Extension of the Method of Zionts and Wallenius*, Naval Research Logistics **36** (1989(a)), 321–335.

——————, *Preference Structure Representation Using Convex Cones in Multicriteria Integer Programming*, Management Science **35**, No. 9 (1989(b)), 1092–1105.

Rao S. S., "Optimization Theory and Applications," Second Edition, Wiley Eastern Limited, 1984.

——————, *Game Theory Approach for Multiobjective Structural Optimization*, Computers and Structures **25**, No. 1 (1987), 119–127.

Rarig H. M., Y. Y. Haimes, *Risk/Dispersion Index Method*, IEEE Transactions on Systems, Man, and Cybernetics **smc-13**, No. 3 (1983), 317–328.

Reeves G. R., R. C. Reid, *Minimum Values over the Efficient Set in Multiple Objective Decision Making*, European Journal of Operational Research **36**, No. 3 (1988), 334–338.

Rentz O., T. Hanicke, *Application of Vector Optimization Processes in Large-Scale Models*, ZOR – Methods and Models of Operations Research **32** (1988), 395–419.

ReVelle C., *Equalizing Superpower Force Disparities with Optimized Arms Control Choices: A Multi-Objective Approach*, European Journal of Operational Research **33**, No. 1 (1988), 46–53.

Rietveld P., "Multiple Objective Decision Methods and Regional Planning," North-Holland Publishing Company, 1980.

Ringuest J. L., "Multiobjective Optimization: Behavioral and Computational Considerations," Kluwer, Academic Publisher, 1992.

Rogowski T., J. Sobczyk, A. Wierzbicki, *IAC-DIDAS-L: A Dynamic Interactive Decision Analysis and Support System for Multicriteria Analysis of Linear and Dynamic Linear Models of Professional Microcomputers*, in "Theory, Software and Testing Examples for Decision Support Systems," Edited by A. Lewandowski, A. Wierzbicki, Working Paper WP-87-26, IIASA, Laxenburg, 1987, 106–124.

Romero C., "Handbook of Critical Issues in Goal Programming," Pergamon Press, 1991.

Rosenthal R. E., *Goal Programming – A Critique*, New Zealand Operational Research **11**, No. 1 (1983), 1–7.

____________, *Principles of Multiobjective Optimization*, Decision Sciences **16**, No. 2 (1985), 133–152.

Rosinger E. E., *Interactive Algorithm for Multiobjective Optimization*, Journal of Optimization Theory and Applications **35**, No. 3 (1981), 339–365; *Interactive Algorithm for Multiobjective Optimization*, Errata Corrige, Journal of Optimization Theory and Applications **38**, No. 1 (1982), 147–148.

____________, *Aids for Decision Making with Conflicting Objectives*, in "Mathematics of Multi Objective Optimization," Edited by P. Serafini, Springer-Verlag, Wien, New York, 1985, 275–315.

Rothermel M. A., D. A. Schilling, *Conjoint Measurement in Multiple Objective Decision Making: A New Approach*, European Journal of Operational Research **23** (1986), 310–319.

Roy A., P. Mackin, *Multicriteria Optimization (Linear and Nonlinear) Using Proxy Value Functions*, in "Multiple Criteria Decision Support," Edited by P. Korhonen, A. Lewandowski, J. Wallenius, Lecture Notes in Economics and Mathematical Systems 356, Springer-Verlag, 1991, 128–134.

Roy A., J. Wallenius, *Nonlinear Multiobjective Optimization: An Algorithm and Some Theory*, Mathematical Programming **55**, No. 2 (1992), 235–249.

Sadagopan S., A. Ravindran, *Interactive Algorithms for Multiple Criteria Nonlinear Programming Problems*, European Journal of Operational Research **25** (1986), 247–257.

Sadek I. S., J. C. Bruch Jr., J. M. Sloss, S. Adali, *Structural Control of a Variable Cross-Section Beam by Distributed Forces*, Mechanics of Structures and Machines **16**, No. 3 (1988–89), 313–333.

Sakawa M., *Interactive Multiobjective Decision Making by the Sequential Proxy Optimization Technique: SPOT*, European Journal of Operational Research **9**, No. 4 (1982), 386–396.

Sakawa M., N. Mori, *Interactive Multiobjective Decisionmaking for Nonconvex Problems Based on the Weighted Tchebycheff Norm*, Large Scale Systems **5**, No. 1 (1983), 69–82.

____________, *Interactive Multiobjective Decisionmaking for Nonconvex Problems Based on the Penalty Scalarizing Functions*, European Journal of Operational Research **17** (1984), 320–330.

Sakawa M., F. Seo, *Interactive Multiobjective Decisionmaking for Large-Scale Systems and Its Application to Environmental Systems*, IEEE Transactions on Systems, Man, and Cybernetics **smc-10**, No. 12 (1980), 796–806.

____________, *Interactive Multiobjective Decision Making by the Sequential Proxy Optimization Technique (SPOT) and Its Application to Environmental Systems*, in "Control Science and Technology for the Progress of Society," Edited by H. Akashi, IFAC, Pergamon Press, 1982(a), 1527–1532.

____________, *Interactive Multiobjective Decision Making in Environmental Systems Using Sequential Proxy Optimization Techniques (SPOT)*, Automatica **18**, No. 2 (1982(b)), 155–165.

Sakawa M., H. Yano, *Interactive Multiobjective Reliability Design of a Standby System by the Fuzzy Sequential Proxy Optimization Technique (FSPOT)*, International Journal of Systems Science **16**, No. 2 (1985), 177–195.

____________, *Trade-Off Rates in the Hyperplane Method for Multiobjective Optimization Problems*, European Journal of Operational Research **44**, No. 1 (1990), 105–118.

____________, *Generalized Hyperplane Methods for Characterizing Λ-Extreme Points and Trade-Off Rates for Multiobjective Optimization Problems*, European Journal of Operational Research **57**, No. 3 (1992), 368–380.

Sankaran S., *Multiple Objective Decision Making Approach to Cell Formation: A Goal Programming Model*, Mathematical and Computer Modelling **13**, No. 9 (1990), 71–81.

Sawaragi Y., H. Nakayama, T. Tanino, "Theory of Multiobjective Optimization," Academic Press, Inc., Orlando, Florida, 1985.

Schilling D. A., C. ReVelle, J. Cohon, *An Approach to the Display and Analysis of Multiobjective Problems*, Socio-Economic Planning Sciences **17**, No. 2 (1983), 57–63.

Schittkowski K., *EMP: An Expert System for Mathematical Programming*, in "Optimization, Parallel Processing and Applications," Edited by A. Kurzhanski, K. Neumann, D. Pallaschke, Lecture Notes in Economics and Mathematical Systems 304, Springer-Verlag, Berlin, Heidelberg, 1988, 125–157.

Schniederjans M. J., J. Hoffman, *Multinational Acquisition Analysis: A Zero-One Goal Programming Model*, European Journal of Operational Research **62**, No. 2 (1992), 175–185.

Sherali H. D., *Equivalent Weights for Lexicographic Multi-Objective Programs: Characterizations and Computations*, European Journal of Operational Research **11**, No. 4 (1982), 367–379.

Sherali H. D., A. L. Soyster, *Preemptive and Nonpreemptive Multi-Objective Programming: Relationships and Counterexamples*, Journal of Optimization Theory and Applications **39**, No. 2 (1983), 173–186.

Shin W. S., A. Ravindran, *Interactive Multiple Objective Optimization: Survey I – Continuous Case*, Computers and Operations Research **18**, No. 1 (1991), 97–114.

________________, *A Comparative Study of Interactive Tradeoff Cutting Plane Methods for MOMP*, European Journal of Operational Research **56**, No. 3 (1992), 380–393.

Shor N. Z., "Minimization Methods for Non-Differentiable Functions," Springer-Verlag, Berlin, 1985.

Silverman J., R. E. Steuer, A. W. Whisman, *Computer Graphics at the Multicriterion Computer/ User Interface*, in "Decision Making with Multiple Objectives," Edited by Y. Y. Haimes, V. Chankong, Lecture Notes in Economics and Mathematical Systems 242, Springer-Verlag, Berlin, Heidelberg, 1985, 201–213.

________________, *A Multi-Period, Multiple Criteria Optimization System for Manpower Planning*, European Journal of Operational Research **34** (1988), 160–170.

Simon C. P., *Scalar and Vector Maximization: Calculus Techniques with Economics Applications*, in "Studies in Mathematical Economics," Edited by S. Reiter, MAA Studies in Mathematics 25, The Mathematical Association of America, 1986, 62–159.

Sinha S. B., S. V. C. Sastry, R. P. Misra, *Management of a Community Storage System: A Goal Programming Model*, European Journal of Operational Research **34** (1988), 92–98.

Skulimowski A. M. J., *Classification and Properties of Dominating Points in Vector Optimization*, in "Methods of Operations Research 58," Edited by P. Kleinschmidt, F. J. Radermacher, W. Schweitzer, H. Wildermann, Athenäum Verlag, Frankfurt am Main, 1989, 99–112.

________________, *Multicriteria Decision Support Based on Multiple Reference Points*, Paper presented at the 10th International Conference on Multicriteria Decision-Making, Taipci, Taiwan (1992).

Slowinski R., *Interactive Multiobjective Optimization Based on Ordinal Regression*, in "Multiobjective Problems of Mathematical Programming," Edited by A. Lewandowski, V. Volkovich, Lecture Notes in Economics and Mathematical Systems 351, Springer-Verlag, 1991, 93–100.

Slowinski R., J. Warczynski, *Application of the Ellipsoid Method in an Interactive Procedure for Multicriteria Linear Programming*, Zeitschrift für Operations Research **28**, No. 3 (1984), 89–100.

Soland R. M., *Multicriteria Optimization: A General Characterization of Efficient Solutions*, Decision Sciences **10**, No. 1 (1979), 26–38.

Soloveychik D., *The Multiple Criteria Decision-Making Process Using the Methods of Multivariate Statistical Analysis*, in "Fifth International Conference on Collective Phenomena," Edited by J. L. Lebowitz, The New York Academy of Sciences, New York, 1983, 205–212.

Song A., W.-M. Cheng, *A Method for Multihuman and Multi-Criteria Decision Making*, in "Systems Analysis and Simulation 1988 I: Theory and Foundations," Edited by A. Sydow, S. G. Tzafestas, R. Vichnevetsky, Akademie-Verlag, Berlin, 1988, 213–216.

Sounderpandian J., *Value Functions When Decision Criteria Are Not Totally Substitutable*, Operations Research **39**, No. 4 (1991), 592–600.

Soyibo A., *Goal Programming Methods and Applications: A Survey*, Journal of Information and Optimization Sciences **6**, No. 3 (1985), 247–264.

Spronk J., *Interactive Multifactorial Planning: State of the Art*, in "Readings in Multiple Criteria Decision Aid," Edited by C. A. Bana e Costa, Springer-Verlag, Berlin, Heidelberg, 1990, 512–534.

Stadler W., *A Survey of Multicriteria Optimization or the Vector Maximum Problem, Part I: 1776-1960*, Journal of Optimization Theory and Applications **29**, No. 1 (1979), 1–52.

————————, *Fundamentals of Multicriteria Optimization*, in "Multicriteria Optimization in Engineering and in the Sciences," Edited by W. Stadler, Plenum Press, New York, 1988(a), 1–25.

——————(Ed.), "Multicriteria Optimization in Engineering and in the Sciences," Plenum Press, New York, 1988(b).

Staib T., *On Necessary and Sufficient Optimality Conditions for Multicriterial Optimization Problems*, ZOR – Methods and Models of Operations Research **35**, No. 3 (1991), 231–248.

Stam A., M. Kuula, H. Cesar, *Transboundary Air Pollution in Europe: An Interactive Multicriteria Tradeoff Analysis*, European Journal of Operational Research **56**, No. 2 (1992), 263–277.

Stam A., Y.-R. Lee, P. L. Yu, *Value Functions and Preference Structures*, in "Mathematics of Multiobjective Optimization," Edited by P. Serafini, CISM Courses and Lectures 289, 1985, 1–22.

Stancu-Minasian I. M., "Stochastic Programming with Multiple Objective Functions," Mathematics and Its Applications (East European Series), D. Reidel Publishing Company; Editura Academiei, 1984.

Sterna-Karwat A., *Continuous Dependence of Solutions on a Parameter in a Scalarization Method*, Journal of Optimization Theory and Applications **55**, No. 3 (1987), 417–434.

Steuer R. E., "Multiple Criteria Optimization: Theory, Computation, and Applications," John Wiley & Sons, Inc., 1986.

————————, *The Tchebycheff Procedure of Interactive Multiple Objective Programming*, in "Multiple Criteria Decision Making and Risk Analysis Using Microcomputers," Edited by B. Karpak, S. Zionts, Springer-Verlag, Berlin, Heidelberg, 1989(a), 235–249.

————————, *Trends in Interactive Multiple Objective Programming*, in "Methodology and Software for Interactive Decision Support," Edited by A. Lewandowski, I. Stanchev, Lecture Notes in Economics and Mathematical Systems 337, Springer-Verlag, 1989(b), 107–119.

————————, *ADBASE: Multiple Objective Linear Programming Package*, Operating Manual, Version 9/89, Department of Management Science and Information Technology, University of Georgia, Athens, Georgia (1989(c)).

Steuer R. E., E.-U. Choo, *An Interactive Weighted Tchebycheff Procedure for Multiple Objective Programming*, Mathematical Programming **26**, No. 3 (1983), 326–344.

Steuer R. E., L. R. Gardiner, *Interactive Multiple Objective Programming: Concepts, Current Status, and Future Directions*, in "Readings in Multiple Criteria Decision Aid," Edited by C. A. Bana e Costa, Springer-Verlag, Berlin, Heidelberg, 1990, 413–444.

————————————, *On the Computational Testing of Procedures for Interactive Multiple Objective Linear Programming*, in "Operations Research," Edited by G. Fandel, H. Gehring, Springer-Verlag, Berlin, Heidelberg, 1991, 121–131.

Steuer R. E., A. W. Whisman, *Toward the Consolidation of Interactive Multiple Objective Programming Procedures*, in "Large-Scale Modelling and Interactive Decision Analysis," Edited by G. Fandel, M. Grauer, A. Kurzhanski, A. P. Wierzbicki, Lecture Notes in Economics and Mathematical Systems 273, Springer-Verlag, 1986, 232–241.

Stewart T. J., *A Critical Survey on the Status of Multiple Criteria Decision Making Theory and Practice*, OMEGA **20**, No. 5/6 (1992), 569–586.

Sunaga T., M. A. Mazeed, E. Kondo, *A Penalty Function Formulation for Interactive Multiobjective Programming Problems*, in "System Modelling and Optimization," Edited by M. Iri, K. Yajima, Lecture Notes in Control and Information Sciences 113, Springer-Verlag, 1988, 221–230.

Szidarovszky F., K. Szenteleki, *A Multiobjective Optimization Model for Wine Production*, Applied Mathematics and Computation **22**, No. 2–3, (1987), 255–275.

Tabucanon M. T., "Multiple Criteria Decision Making in Industry," Elsevier Science Publishers B. V., Amsterdam, 1988.

Tamura K., S. Arai, *On Proper and Improper Efficient Solution of Optimal Problems with Multicriteria*, Journal of Optimization Theory and Applications **38**, No. 2 (1982), 191–205.

Tanino T., *Sensitivity Analysis in Multiobjective Optimization*, Journal of Optimization Theory and Applications **56**, No. 3 (1988(a)), 479–499.

————————, *Stability and Sensitivity Analysis in Convex Vector Optimization*, SIAM Journal on Control and Optimization **26**, No. 3 (1988(b)), 521–536.

————————, *Stability and Sensitivity Analysis in Multiobjective Nonlinear Programming*, Annals of Operations Research **27** (1990), 97–114.

Tanino T., Y. Sawaragi, *Stability of Nondominated Solutions in Multicriteria Decision-Making*, Journal of Optimization Theory and Applications **30**, No. 2 (1980), 229–253.

Tapia C. G., B. A. Murtagh, *The Use of Preference Criteria in Interactive Multiobjective Mathematical Programming*, Asia-Pacific Journal of Operational Research **6**, No. 2 (1989), 131–147.

————————, *A Markovian Process in Interactive Multiobjective Decision-Making*, European Journal of Operational Research **57**, No. 3 (1992), 421–428.

Tarvainen K., *On the Implementation of the Interactive Surrogate Worth Trade-Off (ISWT) Method*, in "Interactive Decision Analysis," Edited by M. Grauer, A. P. Wierzbicki, Lecture Notes in Economics and Mathematical Systems 229, Springer-Verlag, Berlin, Heidelberg, 1984, 154–161.

Teghem J. Jr., C. Delhaye, P. L. Kunsch, *An Interactive Decision Support System (IDSS) for Multicriteria Decision Aid*, Mathematical and Computer Modelling **12**, No. 10/11 (1989), 1311–1320.

Teghem J. Jr., D. Dufrane, M. Thauvoye, P. Kunsch, *STRANGE: An Interactive Method for Multi-Objective Linear Programming Under Uncertainty*, European Journal of Operational Research **26**, No. 1 (1986), 65–82.

Tell R., J. Wallenius, *A Survey of Multiple-Criteria Decision Methods and Applications: Utility Theory and Mathematical Programming*, The Finnish Journal of Business Economics **1**, No. 1 (1979), 3–22.

Thore S., A. Nagurney, J. Pan, *Generalized Goal Programming and Variational Inequalities*, Operations Research Letters **12** (1992), 217–226.

Törn A., *A Sampling-Search-Clustering Approach for Solving Scalar (Local, Global) and Vector Optimization Problems*, in "Theory and Practice of Multiple Criteria Decision Making," Edited by C. Carlsson, Y. Kochetkova, North-Holland Publishing Company, 1983, 119–141.

Udink ten Cate A., *On the Determination of the Optimal Temperature for the Growth of an Early Cucumber Crop in a Greenhouse*, in "Plural Rationality and Interactive Decision Processes," Edited by M. Grauer, M. Thompson, A. P. Wierzbicki, Lecture Notes in Economics and Mathematical Systems 248, Springer-Verlag, Berlin, Heidelberg, 1985, 311–318.

Vanderpooten D., *The Use of Preference Information in Multiple Criteria Interactive Procedures*, in "Improving Decision Making in Organisations," Edited by A. G. Lockett, G. Islei, Lecture Notes in Economics and Mathematical Systems 335, Springer-Verlag, Berlin, Heidelberg, 1989(a), 390–399.

———————, *The Interactive Approach in MCDA: A Technical Framework and Some Basic Conceptions*, Mathematical and Computer Modelling **12**, No. 10/11 (1989(b)), 1213–1220.

Vanderpooten D., P. Vincke, *Description and Analysis of Some Representative Interactive Multicriteria Procedures*, Mathematical and Computer Modelling **12**, No. 10/11 (1989), 1221–1238.

Vetschera R., *Feedback-Oriented Group Decision Support in a Reference Point Framework*, in "Multiobjective Problems of Mathematical Programming," Edited by A. Lewandowski, V. Volkovich, Lecture Notes in Economics and Mathematical Systems 351, Springer-Verlag, 1991(a), 309–314.

——————, *A Note on Scalarizing Functions under Changing Sets of Criteria*, European Journal of Operational Research **52**, No. 1 (1991(b)), 113–118.

——————, *A Preference-Preserving Projection Technique for MCDM*, European Journal of Operational Research **61**, No. 1–2 (1992), 195–203.

Vijayalakshmi, *An Application of Multi-Objective Modeling: The Case of the Indian Sugar Industry*, European Journal of Operational Research **28** (1987), 146–153.

Vincke P., "Multicriteria Decision-aid," John Wiley & Sons, Chichester, 1992.

Wallenius J., *Comparative Evaluation of Some Interactive Approaches to Multicriterion Optimization*, Management Science **21**, No. 12 (1975), 1387–1396.

Wallenius J., S. Zionts, *A Research Project on Multicriteria Decision Making*, in "Conflicting Objectives in Decision," Edited by D. E. Bell, R. Keeney, H. Raiffa, John Wiley & Sons, New York, 1977.

Wang S., *Lagrange Conditions in Nonsmooth and Multiobjective Mathematical Programming*, Mathematics in Economics **1** (1984), 183–193.

————, *Algorithms for Multiobjective and Nonsmooth Optimization*, in "Methods of Operations Research 58," Edited by P. Kleinschmidt, F. J. Radermacher, W. Schweitzer, H. Wildermann, Athenäum Verlag, Frankfurt am Main, 1989, 131–142.

————, *Second-Order Necessary and Sufficient Conditions in Multiobjective Programming*, Numerical Functional Analysis and Application **12**, No. 1–2 (1991), 237–252.

Wang S., Z. Li, *Scalarization and Lagrange Duality in Multiobjective Optimization*, Optimization **26**, No. 3–4 (1992), 315–324.

Wang S., X. Zhou, *Multiobjective Optimization in Spillway Profile*, in "Approximation, Optimization and Computing: Theory and Applications," Edited by A. G. Law, C. L. Wang, Elsevier Science Publishers B. V., 1990, 309–312.

Warburton A., *Quasiconcave Vector Maximization: Connectedness of the Sets of Pareto-Optimal and Weak Pareto-Optimal Alternatives*, Journal of Optimization Theory and Applications **40**, No. 4 (1983), 537–557.

Wäscher G., *An LP-Based Approach to Cutting Stock Problems with Multiple Objectives*, European Journal of Operational Research **44**, No. 2 (1990), 175–184.

Weber M., *Decision Making with Incomplete Information*, European Journal of Operational Research **28** (1987), 44–57.

Weck M., F. Förtsch, *Application of Multicriteria Optimization to Structural Systems*, in "System Modelling and Optimization," Edited by M. Iri, K. Yajima, Lecture Notes in Control and Information Sciences 113, Springer-Verlag, 1988, 471–483.

Weidner P., *On the Characterization of Efficient Points by Means of Monotone Functionals*, Optimization **19**, No. 1 (1988), 53–69.

__________, *Complete Efficiency and Interdependencies Between Objective Functions in Vector Optimization*, ZOR – Methods and Models of Operations Research **34** (1990), 91–115.

Weir T., *Proper Efficiency and Duality for Vector Valued Optimization Problems*, Journal of the Australian Mathematical Society, Series A **43**, No. 1 (1987), 21–34.

________, *On Efficiency, Proper Efficiency and Duality in Multiobjective Programming*, Asia-Pacific Journal of Operational Research **7** (1990), 46–54.

Weistroffer H. R., *Multiple Criteria Decision Making with Interactive Over-Achievement Programming*, Operations Research Letters **1**, No. 6 (1982), 241–245.

______________, *A Combined Over- and Under-Achievement Programming Approach to Multiple Objective Decision-Making*, Large Scale Systems **7**, No. 1 (1984), 47–58.

______________, *Careful Usage of Pessimistic Values is Needed in Multiple Objective Optimization*, Operations Research Letters **4**, No. 1 (1985), 23–25.

______________, *A Flexible Model for Multi-Objective Optimization*, in "Recent Advances and Historical Developments of Vector Optimization," Edited by J. Jahn, W. Krabs, Lecture Notes in Economics and Mathematical Systems 294, Springer-Verlag, Berlin, Heidelberg, 1987, 311–316.

Weistroffer H. R., S. Narula, *The Current State of Nonlinear Multiple Criteria Decision Making*, in "Operations Research," Edited by G. Fandel, H. Gehring, Springer-Verlag, Berlin, Heidelberg, 1991, 109–119.

Wendell R. E., D. N. Lee, *Efficiency in Multiple Objective Optimization Problems*, Mathematical Programming **12**, No. 3 (1977), 406–414.

White D. J., *Multi-Objective Interactive Programming*, Journal of the Operational Research Society **31**, No. 6 (1980), 517–523.

__________, *Concepts of Proper Efficiency*, European Journal of Operational Research **13**, No. 2 (1983(a)), 180–188.

__________, *A Selection of Multi-Objective Interactive Programming Methods*, in "Multi-Objective Decision Making," Edited by S. French, R. Hartley, L. C. Thomas, D. J. White, Academic Press, 1983(b), 99–126.

__________, *A Bibliography on the Applications of Mathematical Programming Multiple-Objective Methods*, Journal of the Operational Research Society **41**, No. 8 (1990), 669–691.

Wierzbicki A. P., *Basic Properties of Scalarization Functionals for Multiobjective Optimization*, Mathematische Operationsforschung und Statistik, Ser. Optimization **8**, No. 1 (1977), 55–60.

______________, *A Methodological Guide to Multiobjective Optimization*, in "Optimization Techniques, Part 1," Edited by K. Iracki, K. Malanowski, S. Walukiewicz, Lecture Notes in Control and Information Sciences 22, Springer-Verlag, Berlin, Heidelberg, 1980(a), 99–123.

______________, *The Use of Reference Objectives in Multiobjective Optimization*, in "Multiple Criteria Decision Making Theory and Applications," Edited by G. Fandel, T. Gal, Lecture Notes in Economics and Mathematical Systems 177, Springer-Verlag, Berlin, Heidelberg, 1980(b), 468–486.

______________, *A Mathematical Basis for Satisficing Decision Making*, in "Organizations: Multiple Agents with Multiple Criteria," Edited by N. N. Morse, Lecture Notes in Economics and Mathematical Systems 190, Springer-Verlag, Berlin, Heidelberg, 1981, 465–485.

______________, *A Mathematical Basis for Satisficing Decision Making*, Mathematical Modelling **3**, No. 25 (1982), 391–405.

——————, *On the Completeness and Constructiveness of Parametric Characterizations to Vector Optimization Problems*, OR Spektrum **8** (1986(a)), 73–87.

——————, *A Methodological Approach to Comparing Parametric Characterizations of Efficient Solutions*, in "Large-Scale Modelling and Interactive Decision Analysis," Edited by G. Fandel, M. Grauer, A. Kurzhanski, A. P. Wierzbicki, Lecture Notes in Economics and Mathematical Systems 273, Springer-Verlag, 1986(b), 27–45.

Winkels H.-M., M. Meika, *An Integration of Efficiency Projections into the Geoffrion Approach for Multiobjective Linear Programming*, European Journal of Operational Research **16**, No. 1 (1984), 113–127.

Wood E. F., N. P. Greis, R. E. Steuer, *Linear and Nonlinear Applications of the Tchebycheff Metric to the Multi Criteria Water Allocation Problem*, in "Environmental Systems Analysis and Management," Edited by S. Rinaldi, North-Holland Publishing Company, 1982, 363–376.

Yang J.-B., C. Chen, Z.-J. Zhang, *The Interactive Step Trade-Off Method (ISTM) for Multiobjective Optimization*, IEEE Transactions on Systems, Man, and Cybernetics **20**, No. 3 (1990), 688–695.

Yano H., M. Sakawa, *Trade-Off Rates in the Weighted Tchebycheff Norm Method*, Large Scale Systems **13**, No. 2 (1987), 167–177.

——————, *A Unified Approach for Characterizing Pareto Optimal Solutions of Multiobjective Optimization Problems: The Hyperplane Method*, European Journal of Operational Research **39** (1989), 61–70.

Yoshikawa K., M. Haruna, K. Kobayashi, *Multi-Objective Multi-Level Systems Approach to Planning of Public Works*, in "Control Science and Technology for the Progress of Society," Edited by H. Akashi, IFAC, Pergamon Press, 1982, 1545–1551.

Yu P. L., *A Class of Solutions for Group Decision Problems*, Management Science **19**, No. 8 (1973), 936–946.

———, *Cone Convexity, Cone Extreme Points, and Nondominated Solutions in Decision Problems with Multiobjectives*, Journal of Optimization Theory and Applications **14**, No. 3 (1974), 319–377.

———, "Multiple-Criteria Decision Making Concepts, Techniques, and Extensions," Plenum Press, New York, 1985.

———, *Habitual Domains*, Operations Research **39**, No. 6 (1991), 869–876.

Yu P. L., M. Zeleny, *The Set of All Nondominated Solutions in Linear Cases and a Multicriteria Simplex Method*, Journal of Mathematical Analysis and Applications **9**, No. 2 (1975), 430–468.

Zadeh, *Optimality and Non-Scalar-Valued Performance Criteria*, IEEE Transactions on Automatic Control **ac-8** (1963), 59–60.

Zangwill W. I., "Nonlinear Programming: A Unified Approach," Prentice-Hall Inc., 1969.

Zeleny M., *Compromise Programming*, in "Multiple Criteria Decision Making," Edited by J. L. Cochrane, M. Zeleny, University of South Carolina Press, Columbia, South Carolina, 1973, 262–301.

———, "Linear Multiobjective Programming," Lecture Notes in Economics and Mathematical Systems 95, Springer-Verlag, Berlin, Heidelberg, 1974.

———, *The Theory of Displaced Ideal*, in "Multiple Criteria Decision Making Kyoto 1975," Edited by M. Zeleny, Lecture Notes in Economics and Mathematical Systems 123, Springer-Verlag, Berlin, Heidelberg, 1976, 153–206.

———, "Multiple Criteria Decision Making," McGraw-Hill Inc., 1982.

———, *Stable Patterns from Decision-Producing Networks: New Interfaces of DSS and MCDM*, MCDM WorldScan **3** (1989), 6–7.

Zhou X., F. S. Mokhtarian, S. Zlobec, *A Simple Constraint Qualification in Convex Programming*, Mathematical Programming **61**, No. 3 (1993), 385–397.

Zionts S., *Methods for Solving Management Problems Involving Multiple Objectives*, in "Multiple Criteria Decision Making Theory and Application," Edited by G. Fandel, T. Gal, Lecture Notes in Economics and Mathematical Systems 177, Springer-Verlag, Berlin, Heidelberg, 1980, 540–558.

———, *Multiple Criteria Mathematical Programming: An Updated Overview and Several Approaches*, in "Multiple Criteria Decision Making and Risk Analysis Using Microcomputers," Edited by B. Karpak, S. Zionts, Springer-Verlag, Berlin, Heidelberg, 1989, 7–60.

Zionts S., J. Wallenius, *On Finding the Subset of Efficient Vectors of an Arbitrary Set of Vectors*, Working Paper 75-4, European Institute for Advanced Studies in Management, Brussels (1975).

————————, *An Interactive Programming Method for Solving the Multiple Criteria Problem*, Management Science **22**, No. 6 (1976), 652–663.

————————, *Identifying Efficient Vectors: Some Theory and Computational Results*, Operations Research **28**, No. 3 (1980), 785–793.

————————, *An Interactive Multiple Objective Linear Programming Method for a Class of Underlying Nonlinear Utility Functions*, Management Science **29**, No. 5 (1983), 519–529.

Zlobec S., *Two Characterizations of Pareto Minima in Convex Multicriteria Optimization*, Aplikace Matematiky **29**, No. 5 (1984), 342–349.

Zubiri J., *Scalarization of Vector Optimization Problems via Generalized Chebyshev Norm*, in "Mathematical Analysis and Systems Theory V.," Edited by J. Szép, P. Tallós, Department of Mathematics, Karl Marx University of Economics, Budapest, 1988, 39–42.