VISTA : A VISUAL INTERACTIVE METHOD FOR SOLVING MCDM PROBLEMS

A THESIS

SUBMITTED TO THE DEPARTMENT OF INDUSTRIAL ENGINEERING

AND THE INSTITUTE OF ENGINEERING AND SCIENCES

OF BILKENT UNIVERSITY

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

FOR THE DEGREE OF

MASTER OF SCIENCE

By

Aslıhan Tabanoğlu

September, 1994

VISTA: A VISUAL INTERACTIVE METHOD FOR SOLVING MCDM PROBLEMS

A THESIS

SUBMITTED TO THE DEPARTMENT OF INDUSTRIAL ENGINEERING

AND THE INSTITUTE OF ENGINEERING AND SCIENCE

OF BİLKENT UNIVERSITY

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

FOR THE DEGREE OF

MASTER OF SCIENCE

By

Aslıhan Tabanoğlu

September 1994

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Halim Doğrusöz, Ph. D. (Supervisor)

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Selim Aktürk, Ph. D.

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Serpil Sayın, Ph. D.

Approved for the Institute of Engineering and Science:

Mehmet Baray, Ph. D.
Director of Institute of Engineering and Science

Abstract

VISTA: A VISUAL INTERACTIVE METHOD FOR SOLVING MCDM PROBLEMS

Aslıhan Tabanoğlu
M.S. in Industrial Engineering
Supervisor:
Prof. Dr. Halim Doğrusöz
September 1994

In this thesis, recognizing the need of interaction with DM (Decision Maker) in solving MCDM (Multiple Criteria Decision Making) problems, a practical interactive algorithm called VISTA (Visual Interactive Sequential Tradeoffs Algorithm) is developed, and a DSS (Decision Support System) is designed to assist DM to use judgement effectively. The algorithm operates by successively optimizing a chosen objective function while the remaining objectives are converted to constraining objectives by setting their satisficing values, one of which is parametrically varied. By plotting the maximum value of the main objective function versus the parameter varied, a tradeoff curve is constructed between the optimized and the parametrized objective, while assuring constraining objectives (satisficing values guaranteed). This tradeoff curve is presented to the DM, and the DM is asked to choose a compromise solution between these two objectives. This chosen point is used as the new satisficing value of the parametrized objective, and a new tradeoff curve is generated by parametrizing another constraining objective function's right hand side and so on. This interactive procedure is continued until the DM is satisfied with the current decision or some other termination criterion is met. Special features to facilitate the DM's judgement (MRS (Marginal Rate of Substitution) Curve, Multiple Comparison Plots, Convergence Plots), and the start and the termination (Start, Terminate, a Hybrid Approach) of the algorithm are provided. Two example problems are worked out with VISTA to demonstrate the practicality of the algorithm. The model and the entire procedure are validated.

Keywords: Multiple Criteria Decision Making, Decision Support System, Visual Interactive Method.

Özet

VISTA: ÇOK AMAÇLI KARAR VERME PROBLEMLERİNİN ÇÖZÜMÜNDE GÖRSEL ETKİLEŞİMLİ BİR YÖNTEM

Aslıhan Tabanoğlu
Endüstri Mühendisliği Yüksek Lisans
Tez Yöneticisi:
Prof. Dr. Halim Doğrusöz
Eylül 1994

Bu tezde Çok Amaçlı Karar Verme Problemlerininin çözümünde, yargı kullanımının gereği dikkate alınarak, VISTA (Visual Interactive Sequential Tradeoffs Algorithm - Görsel Etkileşimli Ardışık Değiş-Tokuş Algoritması) adını verdiğimiz pratik ve etkileşimli bir algoritma geliştirilmiş, ve karar vericinin (KV) yargı kullanımına hizmet edecek bir Karar Destek Sistemi tasarlanmıştır. Algoritma seçilmiş bir ana amaç fonksiyonunu diğer amaç fonksiyonlarının değerlerini tatminkar (satisficing) düzeylerde tutma kısıtlarını sağlayarak en iyiler. Bu kısıtlar arasından seçilen birinin sağ tarafı parametrize edilerek ve en iyilenen amaç fonksiyonunun değişen değerleri bu parametre üzerinde çizilerek, bir değiş-tokuş eğrisi oluşturulur. KV'den beklenen, bu değiş-tokuş eğrisi üzerinde bir nokta, yani bir uzlaşık çözüm belirlemesidir. Bu seçim, paremetrize edilen amaç fonksiyonu için bir tatminkar değer belirler. Bundan sonra diğer bir kısıtın sağ tarafı parametrize edilir ve benzer işlemler tekrarlanır vb. Bu süreç, KV'yi tatmin eden bir çözüme ulaşıncaya veya belirli diğer bir kriteri sağlayıncaya kadar sürer. Sisteme KV'nin yargısını kullanmayı kolaylastırıcı marjinal ikame eğrisi, üçlü karşılaştırma eğrileri, yakınsama eğrileri, ve algoritmaya başlama ve bitirme için yol gösterici ögeler konmuştur. Algoritma, yöntemin işlerliğini göstermek amacıyla, iki örnek problem üzerinde uygulanmıştır. Algoritma ve yöntemin bütününün geçerliği kanıtlanmıştır.

Anahtar Sözcükler: Çok Amaçlı Karar Verme, Karar Destek Sistemleri, Görsel Etkileşimli Yöntem.

Acknowledgment

I would like to express my gratitudes to my thesis advisor Prof. Halim Doğrusöz for his constant help and guidance during this study. He has been a source of motivation and energy to me, and I benefited from every insight he provided.

I would like to express my thanks to Dr. Serpil Sayın and to Dr. Selim Aktürk for their valuable suggestions.

I am particularly indebted to Mustafa for his love, understanding and support.

Contents

List of Figures

Chapter 1

Introduction

Multiple Criteria Decision Making (MCDM) or Decision Making with Multiple Objectives, had arised from the need to solve applied decision problems. It has its roots in a variety of mathematical disciplines and has developed its own field only recently. MCDM started to be regarded as an important field of study when it was realized that for a real decision problem, there are more objectives than one, which are in conflict. Therefore, a simple optimization, as in the case of single objective, is not applicable. Hence, there were several solutions that have the property that no improvement in any one objective was possible without sacrificing on one or more of the other objectives. Therefore, *compromises* between objectives had to be made, necessitating the application of *judgement* by an involved Decision Maker (DM).

All of the early work on this subject was done in connection with welfare and utility theory, initiated by political economists such as Edgeworth and Pareto. However, this concept of conflicting objectives can be said to have been founded by Adam Smith in '*The Wealth of Nations*' in 1776, [22]. According to Arrow [22], however, '*the full recognition of the general equilibrium concept can unmistakenly be attributed to Leon Walras*'. Later, the theory of psychological games and the notion of strategy by Borel in 1921, and by Von Neumann in 1927, respectively, made their contributions, and in 1951 with Koopmans, the *efficient point set* entered in the context of production theory.

Then, Cantor and Hausdorff laid the mathematical foundations in the late 19th Century, but, another fifty years were needed for the subject to become a mathematical discipline until the introduction of the concept of vector maximum problem by Kuhn and Tucker.

The subject was introduced even much later in the engineering literature where it would find the widest area of application. This introduction was due to a short note by Zadeh in 1963, and its use in this field and in the sciences started in the 1970s with a lag of ten years or so.

From then on, practical methodologies suitable for application to the real world decision problems started to be designed by various researchers. Among these are Keeney and Raiffa who developed the theory and methods for multiattribute utility assessment, and Zionts and Wallenius who proposed a practical man-machine interactive programming method, [26]. Apart from these pioneers, there are many others that we are indebted to for bringing the research up to this point.

Although the mathematical framework, which started to develop before the 1970s, constituted the basis of the development of the theory of MCDM, in the 1980s emphasis shifted form multiple objective optimization to providing multiple criteria decision support to the Decision Makers (DMs). This shift implies that more and more research is focusing on capturing the DM's actual decision/choice behavior instead of solving well-structured problems under hypothetical and unrealistic assumptions concerning the DM's preference structure and behavior.

Although there are still many mathematically challenging and important problems left that include multiple objective integer, nonlinear and stochastic optimization problems which require mathematically oriented research [8], the emphasis of Decision Support Systems (DSSs) was brought about by the ever increasing need for applicable procedures for solving real life problems. Especially, when no assumptions can be made a priory on the preference structure of the DM, the implementation of a DSS which would find the most preferred solution to the DM is crucial. This means that, any tool should be available for the DM to apply her/his judgement over the problem effectively. However, this should not be understood that the DM is left alone with a bunch of procedures and expected to dig out a solution. Instead, a systemic view of the problem at hand should be provided to the DM, and the analysis should be conducted in such a way that starting from less complicated and proceeding by supplying intrinsic details of the solutions, the DM will be able to "learn" her/his problem, and if necessary, or believes so, would be free to change the structure of the problem. This is what we call 'a design (or planning) approach'. Sobol [21] stresses that '*...multiple criteria decision methods are much less tools of optimization than they are tools of learning and communication...*'

Decision Support Systems can perform the best when they are incorporated in the context of *Interactive MCDM* approaches where the preferences of the DM are continuously checked and updated. Especially, when this checking is done in such a way that the DM will be informed about the current solution, the real benefit out of the interaction process can be obtained. Also, with the rapid development in PC and computer technology, problems that were once very cumbersome to solve, have started to be handled in seconds, and any type and amount of analysis can be performed very fast. Therefore, any method that is claimed to be applicable, should benefit from this technology and in the most user friendly manner.

One such attempt to design a computer aided procedure, is the *Pareto Race* proposed by Korhonen and Wallenius [11], [9], [13]. Pareto Race represents a dynamic, visual and interactive procedure for multi objective linear programming. It implements reference points (goals) of the DM, and allows her/him to search the efficient frontier of the problem by controlling the speed and direction of the motion. On a display, the DM sees the objective function values in numeric form and as bar graphs whose lengths are dynamically changing as s/he moves about on the efficient frontier. The DM is expected to single out a final decision by observing the values that the objectives can take on the efficient frontier. Although this method has the advantage of working with efficient solutions, it may still be difficult for the DM to compare the achievement of every objective function all at once. It is well known that, from a behavioral perspective, vector valued comparisons are not among the easiest comparison styles. Especially, in Pareto Race, this type of comparisons are continuously made one after another, thus, necessitating that the DM keeps in mind every previous vector of objective functions.

In this thesis, recognizing the need of interaction with DM, we propose a computer aided application of the interactive method introduced by Dogrusoz [2], and we improve, elaborate and refine the method with various additional features. Hereafter, this method will be called 'our method' for ease of referencing.

Our method, makes use of the well-established *ϵ-constraint* concept to generate solutions. We successively optimize an objective chosen from the set of objective functions by converting the remaining objectives to constraining objectives by setting satisficing values as least acceptable values, and by parametrically changing the right hand side of one of them. Then, we draw the tradeoff curve between the optimized and the parametrized objective, assuring constraining objectives (satisficing values guaranteed). This tradeoff curve which is an approximation of the projection of the efficient surface is presented to the DM, and·the DM is asked to choose a compromise solution between these two objectives. Given that the other objectives are guaranteed to perform at least as much as their satisficing values, the choice on the curve is independent from these objectives and considers only the tradeoff information between the two. This chosen point is used as the new aspiration level of the parametrized objective, and another tradeoff curve will be generated by parametrizing another objective function's right hand side, and will be presented to the DM again, and so on. This interactive procedure is continued until the DM is satisfied or some other termination criterion is met.

Owing to the enhanced visualization implemented in our method, we name it VISTA (*Visual Interactive Sequential Tradeoffs Algorithm*). The 'sequential' is due to the fact that the algorithm generates tradeoff curves sequentially. VISTA has various properties that would facilitate the solution of a decision making problem. First, it is easy to understand both from the part of the Analyst and the Decision Maker. Second, the interaction style demands the least possible cognitive effort from the DM to use her/his judgement effectively, since the whole range of possible realizations of the two objectives are shown to the DM, thus reducing the decision process to choosing a point on a two dimensional curve. Third, it has the flexibility to change any satisficing value when desired. It does not use any surrogate functions, such as achievement scalarizing function, which force the same aspiration values throughout the process. Fourth, it does not presume any restrictions set on the objective functions, constraints or the unknown preference function. Thus, it bears the freedom for the DM to be indecisive or inconsistent during any stage of the algorithm. Fifth, under some regularity assumptions, the algorithm converges to the optimum solution with respect to an assumed form of utility function.

As a final remark, VISTA seems to be very appropriate if a practical decision making procedure is wished to be applied. Together with the features proposed in this thesis, we believe that VISTA is a candidate to be classified in the MCDM literature as a significant practical method.

In the second chapter the terminology developed for and used by field of MCDM together with the related literature are reviewed. In the third chapter, our method will be presented as a Decision Support System and its mathematical foundation is discussed. The fourth chapter presents two example problems that illustrate the algorithm, and the final two chapters validate VISTA with respect to existing methods and conclude the thesis.

Chapter 2

Concepts and A Review of MCDM Literature

A Multiple Criteria Decision Making (MCDM) problem is a decision problem which has more than one objective functions. Although it is expected to find a solution which 'optimizes' all of the objectives simultaneously, such a solution is not feasible due to the conflicts among the objectives inherent in the problem. This conflict may occur due to the scarcity of the resources or due to the counteractions arising from system behavior. Therefore, special tools are needed in order to find the *best compromise solution*, and these tools are supplied by the MCDM methodology.

In this chapter, we shall review the definitions of the concepts which have a bearing on the solution methodologies for MCDM problems that have been developed so far. In the next section, we will review terms that are used throughout this thesis, then we will briefly describe the Rational Ideal Model which bears an idealization of the Decision Maker as well as solutions for a MCDM problem. In the following section, an approach proposed as an alternative to this ideal setting is presented. We will conclude that the MCDM methodology challenges the Rational Ideal Model and , or in a sense, complements it. Finally, we will present a literature review on interactive continuous MCDM methodologies along which our method can be classified.

2.1 Definitions

Concepts related to MCDM and associated terminology, as appear in literature, is full of confusion and ambiguities. As a good example, for the very subject studied in this thesis many terms are used:
Multiple Criteria Decision Making (MCDM), [8] [23], [1]
Multi Objective Decision Making, [13]
Multi Objective Programming (MOP), [18]
Multi Attribute Utility Theory (MAUT), [7]
Decision Making With Multi Dimensional Value [2]
without being appropriately defined.

Other related concepts are not exceptions, as it is clearly stressed by Keeney and Raiffa [7] by saying '*...there are no universal definitions of the terms objective, goal, attribute, measure of effectiveness, standard, and so on...*'

Therefore, we feel a need to provide some clarification about our understanding of these concepts as they are used in this study. In the following, we will provide definitions of these concepts as clearly as we understand them.

- **Decision Making**

  Decision Making is the process of making a choice from a set of alternative courses of action, which is called Decision Space, while aiming at achieving a number of objectives. If these multiple objectives are transformed into a single optimizing objective, then the decision making process is converted to a simple optimization process (the concept of optimizing objective will be clarified later). There is a crucial difference between single objective decision making and MCDM. Once the single objective optimization problem has been formulated and relevant data have been collected, the solution process does not involve the decision maker since the solution is embedded in the formulation, and becomes a decision maker independent algorithm. However, in MCDM the DM is the only one who can provide the preference information which is required to determine the best compromise solution. Decision making is no longer an independent computational process, but is now a process of search, evaluation, communication and learning where the decision maker's values and preferences gradually become explicit. Zeleny commented on this with '*Letting the man back in*'.

- **Attributes**

An *attribute* of a system is a measure to characterize a property of the system. Here, the concept of measurement is taken in most general sense (i.e. nominal, ordinal, interval or ratio scales). Attributes characterize different properties of alternatives, such as age, height, price, surface area, quality, location, etc. In other way of saying, an attribute is an indicator of a property and its specific value is the property. For example, weight is an attribute for a person and weight of that person being equal to 75 kg. is her/his property.

At a more technical level, let a designate a feasible alternative, such that $a \in A$, where A is the set of all feasible alternatives (Act Space). To each $a \in A$ we associate n indices of value: $X_1(a), ..., X_n(a)$. We can think of the n evaluators $X_1, ..., X_n$ as mapping each a into a point in an n-dimensional consequence space. Here, the evaluators determine values of the attributes. Often, an attribute and its evaluator are denoted with the same symbol X, [7]. This simply means that the value of its evaluator determines the value of an attribute. Fig. 2.1 illustrates the concept further.

- **Criteria**

In the Webster's Dictionary, *criterion* is defined by '*a standard rule or test by which something can be judged.*' Criteria are functions which determine how desirable the attributes of an alternative are with respect to those of another alternative. Criteria, therefore, are rules to distinguish which values of performance measures are preferred to which other values. For example, the height of a basketball player may be an attribute to determine the effectiveness of this player, and this can be

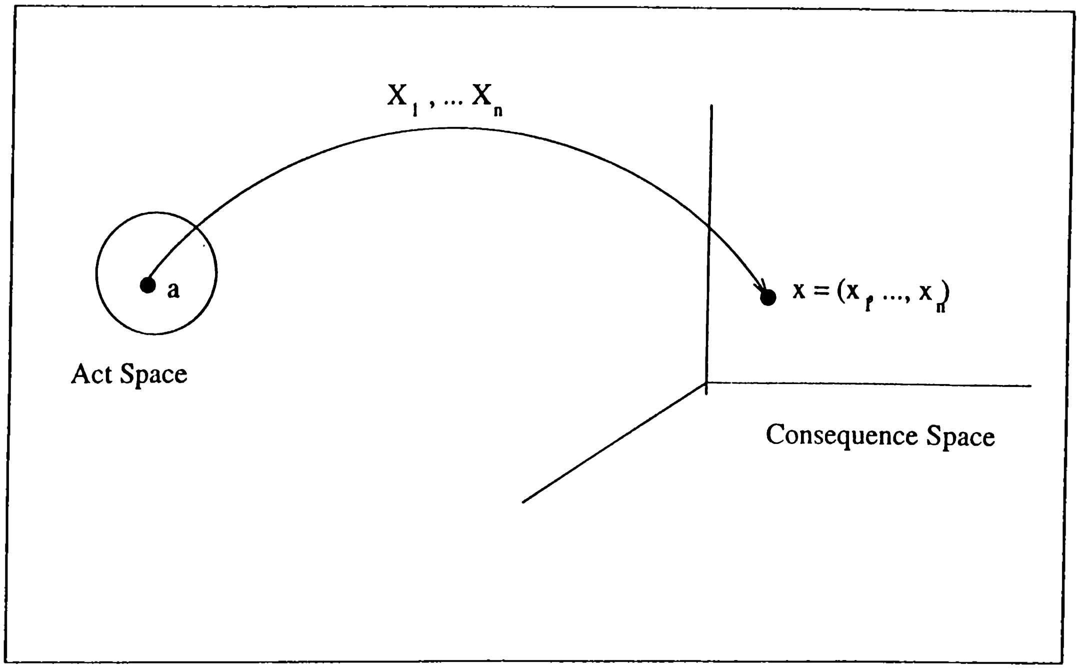

Figure 2.1: Mapping of an alternative to the attribute space

expressed as a rule with 'taller players are preferred to shorter ones'. Here, this rule is the criterion.

- **Objectives**

According to Keeney and Raiffa [7] '*An objective generally indicates the direction in which we should strive to do better.*' More formally, an *objective* is specified by means of two components:

1. An objective variable, and
2. A rule of choice on that objective variable.

Therefore, an objective is a function of objective variables, and this function is determined by the rules of choice which are called criteria. Depending on the rule of choice, an objective may be of one of the following types:

a. Optimizing Objective: Here, the values of objective variables are completely preference ordered (e.g. maximizing profit, minimizing cost),

b. Constraining Objective: This indicates that a subset of values are preferred to its complement (e.g. earning at least \$ 100,000 per year),

c. Goal: Here one value is chosen to be attained (e.g. becoming an academician).

- **Value**

Value is an order preserving function which maps elements of the objective space (outcomes of decisions) to the Real line. And by the use of the values assigned to the objectives, alternative courses of action can be compared with each other.

A value system is an internalized preference ordering strategy of an individual. This system is utilized to make preferences on the alternatives to choose from and evaluate them accordingly. It has been evolved starting at the birth of the individual and shaped through the continuous interaction with the society he lives in. In their book, Bogetoft and Pruzan [1] indicate that '*The choice of an alternative corresponds to a culmination of a learning process where values, objectives, criteria, alternatives and preferences continually interact and redefine each other and lead -explicitly or implicitly- to a compromise which dissolves the intrapersonal conflict.* It is therefore unique to the individual in question, and except that the social control over an issue is extremely strong, no two individuals' values are expected to be the same.

Bogetoft and Pruzan [1] comment that '*...no matter how we consider our objectives, each of them can be derived from certain more fundamental values...In other words, when we use the word 'value' we are referring to a more fundamental concept than 'objective'. One way to distinguish between these concepts is to consider values to be ends and objectives to be means to achieving the more fundamental values. When a question as to 'why do you want to achieve that objective' cannot be answered, by referring to a new objective, but simply by the reply 'because', we are dealing with a value. And when such a question can be answered by referring to some other, more fundamental objective, we are dealing with an objective and not a value...*'

Decision theorists have suggested the use of a value (or utility) function to express the preference behavior of the decision makers by which the choice over a multidimensional solution space would be reduced to the optimization of a single objective function. However, derivation of such a function is a highly demanding task, and, may not be accurate. On this issue, Ozernoi [17] comments that '*Decision theory assumes that a decision maker uses some rational, uncontradictory preference system (or structure). Each decision maker has such a system, although he is not always aware of it. Assessing the DM's preference system is the most difficult step in formulating decision rules in multi criterion problems. A DM's preference judgments are linked closely with the relative importance of criteria, the estimates generated through the scales he uses, and so on...Another difficulty in assessing the preference system is that a decision maker may in practice be inconsistent. It is often difficult to discover inconsistency in the preference system, both for the decision maker and for the analyst who takes part in the decision process.*'

- **Surrogate Value**

  The surrogate value (called proxy attribute or proxy objective in some other contexts) is utilized to select an objective to be used in place of one or more other objectives which are more expressive to the decision maker's underlying values but which are more difficult to measure and express. For example, most business people would react with the objective of high profits when asked to increase the well-being of an enterprise. However, profit is an ill-defined concept which says very little about the qualities characterizing an enterprise's performance. It is one means to escape from difficult measurements. Therefore, profit is used as a surrogate measure for value measure that should normally have many other components.

- **Ideal and Nadir Points**

  The ideal point is a vector of an optimizing objective space whose elements are best values of the component objective functions, given other objectives are ignored.

  The nadir point is a vector of an optimizing objective space whose elements are the best values of the negative of the component objective functions given other objectives are ignored..

The above definitions should make it clear that the term used to indicate decision making by considering tradeoffs between alternatives should *not* be Multiple Criteria Decision Making (MCDM) since criteria are merely rules to specify objective functions. Therefore, the term criteria does not correspond to the objective functions of the problem at hand. The DM may have only one criterion (for example, the criterion of maximizing), but various objectives to evaluate the alternatives whether they are preferable in the sense that their achievement will be in the direction indicated by this criterion. Therefore, the term Multiple *Objective* Decision Making looks better. Moreover, if we would like to approach the problem from a meta-level, we could incorporate the inevitable component, *judgment*, implied by the use of *value* in the name of the subject. It follows that, the term *Decision Making With Multidimensional Value* [2] could be appropriate. However, in order to be consistent with the literature, we will use the term MCDM .

2.1.1 Compromises and Tradeoffs

Given that the ideal point of an MCDM problem is not feasible because of the conflicts among objectives, in order to choose a feasible alternative, the Decision Maker has to deal with conflicts, by taking into account the sacrifices s/he has to make in some objectives in favor of some other objectives. In choosing such an alternative, the DM has to implement her/his value system to determine how much more important an objective is than another one. S/he therefore '*...has to make compromises between objectives to solve this problem, and, in a sense, has to choose the most preferred conflict...*' [1].

Using tradeoffs, on the other hand, is a way to illustrate the meaning of compromising. When the DM makes a choice which corresponds to the implicit choice of a vector of objective values as being the most preferred, we can interpret this choice by saying that the DM implicitly trades off all the losses in some criteria for the gains in the others, meaning that the achievement of best value of her/his value measure. For example, given that we have two vectors of three criteria (stands for objectives), and the objectives are maximizing, if the DM prefers one over the other, then s/he is said to have traded off between these two alternatives. Let the first vector be $a_1 = (3, 5, 2)$, and the second be

$a_2 = (5, 4, 2)$. If the DM prefers a_2 over a_1, this means that s/he traded off 1 unit loss in the second objective with two units gain in the first objective. Therefore, at that point, the tradeoff ratio between the first two objectives is $-1/2$.

2.1.2 The MCDM Problem

The MCDM problem that we are going to consider is defined on a continuous decision and objective space which are both Euclidean. In the most general form this problem is,

$$\begin{array}{c} \max \quad f_1(\mathbf{x}) \\ \max \quad f_2(\mathbf{x}) \\ \vdots \\ \max \quad f_n(\mathbf{x}) \\ \text{subject to} \\ \mathbf{x} \in X(\text{Feasible Subset of decision space}) \end{array} \tag{2.1}$$

where $\mathbf{x}$ is a p-dimensional vector of real numbers, and $f_1(\mathbf{x}), f_2(\mathbf{x}), ..., f_n(\mathbf{x})$ are so called *objective functions* each of which represent an optimizing objective. Here, $f_i(\mathbf{x})$ is a value measure or surrogate value measure, i.e., an order preserving real function.

2.1.3 Efficient Solutions

Since the ideal point of an MCDM problem is not feasible, we need to find a solution that reflects the preference of the DM the best. However, the feasible region over which such a solution is sought contains infinitely many points, thus necessitates the introduction of a concept which will reduce this region to a smaller set, possibly of one element. This is exactly the goal of MCDM, but difficult to attain. The idea of efficient set is conceived as an intermediary step to make the final step a bit easier. The advantage of this intermediary step is its being independent of the DM.

Let $\mathbf{X}$ be the set of alternatives or feasible solutions, and let $\mathbf{Y}$ be the objective space such that

$$\mathbf{Y} = \{\mathbf{y} | \mathbf{y} = \{y_1, ..., y_n\}\} = \{(f_1(\mathbf{x}), ..., f_n(\mathbf{x})), \mathbf{x} \in \mathbf{X}\} \tag{2.2}$$

Definition 1. (*Domination*) A vector $\mathbf{y} \in \mathbf{Y}$ is said to be dominated by a vector $\mathbf{y}' \in \mathbf{Y}$ if and only if $\forall i \;\; y_i \leq y_i'$ and $\exists j$ such that $y_j < y_j'$, $\quad i, j = 1, ..., n$.

Definition 2. (*Efficiency*) A vector $\mathbf{y}' \in \mathbf{Y}$ is said to be efficient if and only if it is dominated by no other feasible vector $\mathbf{y} \in \mathbf{Y}$. An alternative $\mathbf{x} \in \mathbf{X}$ is called efficient if $(f_1(\mathbf{x}), ..., f_n(\mathbf{x}))$ is efficient.

Fig. 2.2 illustrates these definitions. Here, objective functions are $f_1(\mathbf{x}) = x_1$ and $f_2(\mathbf{x}) = x_2$, and they are to be maximized given the feasible region in the figure. For this example, the efficient points are along an edge, which is called the *efficient edge.* Had there been more than two objective functions, they would build a surface called *efficient surface.*

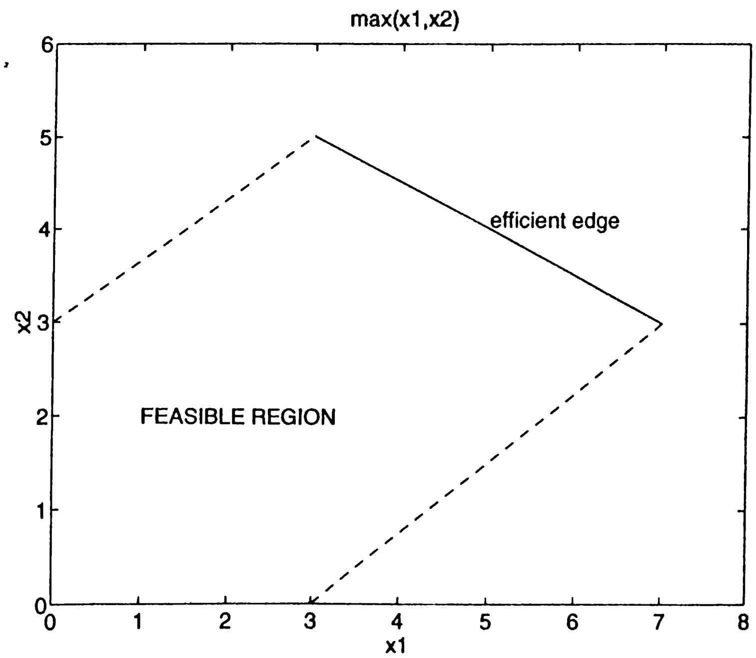

Figure 2.2: Illustration of efficient set

2.2 Conceptual Models to Represent Value Systems

In this section, we will present two different conceptualizations of the decision making strategies used by various MCDM procedures. The first one, the Rational Ideal Model idealizes the DM as well as the solution space, i.e., the DM's preferences are consistent and not changing over time, and the set of alternative solutions is predetermined. The second conceptualization is Procedural Rationality, in which the DM is perceived as a human being, and a behavior style, namely satisficing, is defined. We will further explain and compare these two strategies in the following.

2.2.1 The Rational Ideal Model

The Rational Ideal Model is conceptualized as an approach to solve problems involving multiple objectives. Given the Decision Maker's *true preferences* and the *set of all alternative solutions*, a solution which is optimizing to the MCDM problem can be determined. In this situation, the term *rational* describes the behavior of the DM which is unchanging and consistent over time, and the term *ideal* describes the set of alternative solutions which is predetermined and stable too. This conceptualization makes use of *substantive rationality* which is defined to be a preference made in the specific abstract world of the Rational Ideal Model where preference functions are employed to measure the attractiveness of alternatives.

The preference (or utility, or value) function of a DM however, is not easy to derive, and requires a lot of time and effort, and even then, is still subject to error. It also limits the control of the DM over a specific problem, since an 'optimal' solution can be found without consulting the DM once her/his preference function is at hand. Furthermore, it makes group decision making impossible since everyone in the group will stick to her/his optimal solution since they have different value functions, thus, eliminating the chance of discussion.

Bogetoft and Pruzan [1] observe that:

(1) Real multiple objective problems do not simply exist as objective realities, but are subjective products of our cognition,

(2) They, therefore, do not present themselves with clearcut descriptions of wishes and possibilities, and

(3) Humans have limited capacities for information production, processing and analysis.

On subjectivity Tversky [24] made the following remark:'*...Our research has shown that subjective judgments generally do not obey the basic normative principles that sometimes lead to reasonable answers and sometimes to severe and systematic errors. Moreover, our research shows that the axioms of rational choice are often violated consistently by sophisticated as well as naive respondents, and that the violations are often large and highly persistent...one's original erroneous response does not lose its appeal even after one has learned the correct answer.*'

Earlier, similar observations led psychologists and economists to suggest alternative models of rationality. H.A. Simon [19] used the term *bounded rationality* about theories that incorporate an individual's limited information processing capacities (another term for this is *substantive rationality*) . More specifically, the above observations led to models of human decision making in terms of *satisficing* rather than *optimizing* behavior. We will explain these as well as other terms such as *procedural rationality* [20], which is used to supplement the *substantive rationality* of the Rational Ideal model in the following subsection.

2.2.2 Satisficing Models and Procedural Rationality

In *satisficing models*, the distinction between possibilities and wishes is not very apparent. '*...People are depicted as thinking in terms of aspiration levels which function as a sort of mediator between the ideal and the realizable outcomes, and which may be dynamically modified throughout the decision making process. The search for appropriate action can be terminated whenever the aspiration level is even nearly satisfied...*', [1].

Bogetoft et al. argue that the concept of procedural rationality seeks to focus attention on the effectiveness of different decision making procedures rather than the effectiveness of a given decision as emphasized by substantive rationality. Therefore, procedural rationality is process-oriented while substantive rationality of the Rational Ideal model is outcome-oriented.

In practice, however, planners try to integrate these two approaches in a best way for their specific problems. They intuitively decide how to balance the effort required to generate and analyze alternatives with the possible improvements in the resulting solutions. They thus integrate optimizing and satisficing behavior as well as substantive and procedural rationality. Therefore, the rational ideal model can be used as a means of justification of the applied approximating model. We will attempt to justify our method using the concepts borrowed from the Rational Ideal Model also, in Chapter 4, section 4.4 Convergence where we discuss the convergence of our procedure.

2.3 Evolution of MCDM Methodology and Techniques

The field of Multiple Criteria Decision Making as a scientific discipline, is perceived to be one of the most active, international, and interdisciplinary fields of research in management science and operations research [10]. Although its roots are founded by Adam Smith, and Pareto, it started to evolve with goal programming introduced by Charnes and Cooper in 1960's, and developed with Keeney and Raiffa's theory of Multiattribute Utility. MCDM has been a popular research area for more than two decades and over the years several approaches and underlying theory have been developed in several countries by various researchers.

In the 1970's, research focused on the theoretical foundations of multiple objective mathematical programming and on the development of procedures and algorithms for solving multiple objective mathematical programming problems. Mathematical programs, especially linear and discrete problems dominated the field, and tools of mathematical programming theory were used. The algorithms were programmed for mainframe computers and were used mainly for illustrative purposes. The systems were often of a prototypical nature, lacked user-friendly interfaces and not well documented.

In the 1980's however, emphasis shifted away from multiple objective optimization towards providing multiple criteria decision support to DM's and practitioners. Korhonen believes that more and more research is focusing on capturing the DM's actual decision/choice behavior, instead of solving well-structured problems under hypothetical and unrealistic assumptions concerning the DM's preference structure and behavior [8]. According to him, this emphasis on decision support has brought to light many important issues, such as:
-the importance of developing appealing communication facilities to the DM (e.g., interfaces based on the use of spreadsheets, colors, graphical representations, windows, and on-line help capabilities, providing a simple grammar of the communication language);
-the realization that problem solving should not be seen in isolation; the organizational context is important;
-the fact that the entire process of decision making from problem identification to solution implementation should be supported. For instance, it is not realistic to assume that a DM is able to formulate a problem precisely prior to the solution process and then solve it. It is essential that s/he can approach a problem on a more evolutionary basis, in which several steps of redefining and solving follow each other.

In similar lines Dogrusoz stressed the importance of DM's judgement in decision making as follows, [2]: '*With or without the help of science, judgement cannot be taken out of decision making. All scientific investigations and analyses in general, and cost-effectiveness in particular, are only to provide information so that judgment can be applied by decision maker more effectively (more efficiently, more rapidly, with less effort and higher chance of optimality etc.) in making the choice. Such information may also boost the confidence in the choice made which may help to expedite implementation, but analysis and the analyst cannot replace the DM, unless both functions is charged to the same* *individual.*'

Bogetoft argues that: '*Multiple Criteria Decision Making (MCDM) is both an approach and a body of techniques designed to help people make choices which are in accord with their values in cases characterized by multiple, noncommensurate and conflicting criteria. It is considered by many to be a sub-discipline within Operations Research. From this vantage point, it has probably been the single most expanding branch of OR in the past decade. Interestingly enough, it represents at the same time a renewal and revitalization of OR by recalling its original character as an approach to problem solving based on systems thinking, multidisciplinarity and a scientific approach. In particular, the emphasis is again seen as one of helping a decision maker to structure his problems and to make good choices. The optimization of given, well-structured problems using more and more computationally efficient algorithms is* **not** *in focus, particularly in the case of more significant or strategic decisions.*

... In particular, research on the relationship between contextual factors and the methods applied is needed...A typical article on MCDM in a scientific journal only devotes a few lines to the contextual characteristics that are supposed to motivate its technical developments. This lack of explicit contextual consideration, motivation and precision has serious implications. No sound foundation has been developed which permits synthesizing, comparing and evaluating different MCDM procedures.

...In addition, the literature on MCDM seldom pays attention to such vital matters as the choice of criteria, the identification of alternative actions, and the symbiotic relationship between these two activities. Ignoring these behavioral and cognitive aspects leads once again to a fixation on algorithmic procedures and their characteristics based upon the presupposition that the means and ends have already been operationally identified.', [1]

The conclusion that can be arrived at is that the behavioral and pragmatic realism of decision tools has been and still is increasing. One such tool is called Multiple Criteria Decision Support Systems (MCDSSs) that allow the users to analyze multiple criteria and to incorporate their preferences over these criteria into the analysis. These analyses are of "what to do - to achieve" type whereas traditional mathematical programming framework suggests the use of "what - if" type analyses. Therefore, consequences are manipulated and depending on the levels of criteria (or objectives) that the DMs are expected to use their judgments.

2.4 Solution Methodologies

Solution methods developed for solving MCDM problems are categorized in the following manner depending on different assumptions made with respect to the preference function:

(1) Complete information of the preference function (utility function) is available from the DM;

(2) No information of the preference function is available from the DM;

(3) Partial information can be obtained progressively from the DM.

In the first approach, called prior articulation of preferences, the DM's preference function is assessed or the DM's aspirations are determined before attempting to solve the MCDM problem at hand. The problem is either reduced to a single objective optimization problem whose objective is to maximize utility over the system constraints, or transformed into a series of scalar optimization problems such as goal programming. However, it is well agreed upon that determination of the explicit form of the preference function, as it is the case in the utility theory, may require a prohibitive amount of time and effort.

The second approach, called prior articulation of alternatives, the DM is presented with efficient solutions only and is expected to select the one which is the most preferred. Here, the effort is made to find all efficient solutions from the part of the analyst, and to find the most preferred one from the part of the DM. These methods have been criticized for their computational burden on the DM in selecting a solution from an infinite number of alternatives.

The third approach, called interactive method, does not require a priori preference information, instead it elicits the DM's preference structure through DM-Analyst, [2], or Human-Machine interactions. At every iteration of the method, the DM provides preference information about the current solution either implicitly or explicitly. Since the DM is involved in the entire solution process, this approach has found better acceptance in practice. In the next section, we will briefly review interactive methods along their main lines since the method that we are going to propose and improve in this thesis can be categorized under this heading.

Emphasizing the necessity of DM's judgement, we could rephrase this classification of MCDM procedures, by focusing on where during the process of decision making, the judgement of the DM is considered:

a. Procedures that use DM's judgement *before algorithm starts*,

b. Procedures that use DM's judgment *after algorithm ends*,

c. Procedures that use DM's judgment *interactively*.

2.5 Interactive Approaches

Here, we would like to make a little more elaborate review of interactive approaches, since ours fall into this category. Interactive approaches rely on the progressive articulation of preferences by the DM. These methods can be characterized by the following three steps:

(1) Finding an interim solution (feasible, preferably efficient);

(2) Interacting with the DM to obtain her/his reaction and response to the solution;

(3) Repeating steps (1) and (2) until satisfaction or some other termination criteria is met.

According Shin and Ravindran, [18], when interactive algorithms are applied to real-world problems, the most critical factor is the functional restrictions placed on the objective functions, constraints and the unknown preference function. Another important factor is preference assessment styles which is also called interaction styles. The cognitive burden on the DM during the solution process depends heavily on interaction styles. Typical interaction styles in ascending order of cognitive burden are listed:

(a) Binary pairwise comparison: the DM must compare a pair of two-dimensional vectors at each interaction.

(b) Pairwise comparison: the DM must compare a pair of p-dimensional vectors and specify a preference.

(c) Vector comparison: the DM must compare a set of p-dimensional vectors and specify the best, the worst of the order of preference (this can be done by a series of pairwise comparisons).

(d) Precise local tradeoff ratio: the DM must specify precise values of local tradeoff ratios at a given point.

(e) Interval tradeoff ratio: the DM must specify an interval for each local tradeoff ratio.

(f) Comparative tradeoff ratio: the DM must specify his preference for a given tradeoff ratio.

(g) Index specification and value tradeoff: the DM must list the indices of objectives to be improved or sacrificed, and specify the amount.

(h) Aspiration levels (or reference point): the DM must specify or adjust the values of the objectives which indicate her/his optimistic wish concerning the outcomes of the objectives.

It is believed that vector comparisons are easier to respond than value tradeoff ratios. On the other hand, methods that use vector comparisons may require more interactions. The DM also may prefer a certain interaction style, and therefore the selection of an interaction style is case dependent.

According to Shin and Ravindran, interactive methods can be classified in the following scheme. The references are given in [18].

- FEASIBLE REGION REDUCTION METHODS:

  These methods try to reduce the feasible region of the problem (eliminating the unpreferred subset of objective space) by introducing extra constraints derived from the answers obtained from the DM. Three steps in each iteration of the method are followed. In the calculation phase, an efficient solution which is in the minimax sense nearest to the ideal solution is obtained. This solution is presented to the DM in the decision phase, and her/his responses are used in the feasible region reduction phase. However, this method is criticized to be an ad hoc approach since no preference function concept is utilized. Some approaches in this category are STEP method (or STEM), and GPSTEM (Goal Programming STEP Method).

- FEASIBLE DIRECTION METHODS:
  Feasible direction methods guide the DM in finding the most preferred solution by making a search along a direction where the preference of the DM appears to increase. Then the step size with which to proceed on this direction is determined and the obtained solution is presented to the DM. The DM provides information by specifying values of local tradeoffs among criteria. The GDF and GRG methods are examples of this approach.

- CRITERION WEIGHT SPACE METHODS:
  Criterion weight space methods are very popular and rely on the most easily understood form of MCDM methodology. Namely, they reduce the multiple objective optimization problem into a single objective one by taking the weighted sum of the objective functions. These weights are expected to reflect the DM's preferences, and doing either a parametric or interval search on the weight space, optimal set of weights are seeked. Zionts and Wallenius [26] proposed a method assuming a pseudo concave preference function.

- TRADEOFF CUTTING PLANE METHODS:
  These methods try to reduce the objective space (or criterion space) by cutting planes, and therefore do not require a line search. However, in order to implement these methods successfully, the DM has to supply exact local tradeoff ratios.

- LAGRANGE MULTIPLIER METHODS:
  These methods make use of the Lagrange multipliers obtained from the solution of optimizing one objective function subject to the other objective functions treated as

constraints, and whose bounds are varied. The DM is asked to assess the indifference bounds to define a surrogate worth function. However, the work may be cumbersome for both the analyst and the DM.

- VISUAL INTERACTIVE METHODS:
  In order to relax the assumption that the preference function of the DM remains unchanged during the decision making procedure, or the assumption that such a function does exist, Korhonen and Laakso [9] suggested a graphic-aided interactive approach which is later called VIG and Pareto Race. Pareto Race uses reference directions concept which reveals the tradeoffs between objective functions when striving to achieve the ideal solution. The DM can control the efficient frontier during the interactive process. Visual interactive methods are expected to grow rapidly since PC technology allows the use of computer graphics, windows, and user friendly tools quite efficiently. We can classify our method, which will be developed and presented in the next chapter, within this category.

- BRANCH AND BOUND METHODS:
  These methods, as their name implies, use the concept of branching the objective space and fathoming unpromising branches after a branch's ideal solution is obtained and found to be dominated. This method terminates with and efficient solution, and does not depend on the preference function regardless of whether it exists or not. However, no real-world applications are reported.

- OTHER IMPORTANT METHODS:
  Some other methods are Relaxation Methods, Sequential Methods, Scalarizing Function Methods, Fuzzy Satisficing Methods. These all use interactions with the DM to find a compromise solution after many iterations. The references are given in Shin and Ravindran [18].

It should be noted that, however, most methods are applicable only for long term planning purposes instead of frequent decision making situations. For scheduling jobs in a factory, for example, interactive approaches may be very time consuming since they require constant involvement of the DM. However, for investment planning decisions this interaction with the DM is made once for all, therefore is crucial. We assume that the method we propose in the following is to be applied to long term planning decision situations in the context of interactive methods.

Chapter 3

A Visual Interactive Decision Support System

In this chapter, the core of this study, the Visual Interactive Sequential Tradeoffs Algorithm (VISTA) as a decision support system for MCDM, will be developed and discussed. First, the importance of the principle of visualization of information, which constitutes the foundation of the method is discussed and justified. Then, the basic idea, on which the method evolves, is presented. To make this idea operational, a working algorithm is developed by constructing missing elements and refinements, and the detailed working methodology is presented. This is first presented intuitively as a step by step algorithm, then the logical foundation is discussed rigorously.

3.1 Importance of Visualization and Interaction Style in Information Support

The method that is developed and presented here is based on the philosophy that in solving MCDM problems, use of the DM's judgement is inevitable, and, the analyst's interaction with her/him is a crucial component of an effective solution approach.

From a behavioral perspective, a human being can learn and judge in a best way by visualization of the material under consideration. Instead of long tabulated lists, mere descriptions, or definitions; visual aids such as simulations, graphics, charts, photographs, analog devices, etc., perform better in helping someone to become acquainted with something. Also in MCDM, visualization is considered to be of prime importance and especially with the growing PC technology, it has become easier to implement. Although suggested much earlier [2], the direction of MCDM research began to grow towards supplying as much visual information as possible only in the 1980's [11], [16].

The interaction style, also, plays an important role if an interactive algorithm is expected to be relatively easy and efficient in finding the most preferred solution. It should provide the decision maker with maximum information, and require minimum effort from her/him to come up with an answer. However, in general, the easier the interaction style, the longer the analysis takes. On the contrary, the less the number of interactions, the more complex the answers required from the DM are. The interaction style, therefore, is itself a tradeoff problem in nature. However, the fact that binary pairwise comparisons require less cognitive effort by the DM than other interaction styles [18] (see previous chapter for this) will help us in developing our method.

3.2 The Basic Idea

According to many descriptive studies [15], the transfer of a criterion to a constraint, and the search for a satisfactory level is a typical human operation. This idea of satisficing has been proposed by H.A. Simon [19] and used mainly in economic contexts, called rationing mechanism in resource (or budget) directive planning methods. However, to the best of our knowledge, although there has been some studies on purely mathematical grounds (for example [25]), there hasn't been any attempt to formalize this concept into a decision making process until mid-1970s. One such approach where the basic idea of the method developed here has first been suggested in Dogrusoz [2]. His suggestions remained unnoticed in literature, except an attempt of its application to an aircraft design [4], until

1980 when Nakayama et al. (apparently unaware of Dogrusoz' paper) used the same idea of interaction with DM with visual displays of information to her/him in the decision making process, and came up with somewhat a similar method and applied to an example problem [16]. Both approaches make use of binary comparisons of two objective measures by presenting the DM all the points on the efficient frontier obtained by considering these two objectives at a time, and keeping other objective measures at satisfactory values to the DM. The method presented by Nakayama, however, can be visualized for three objectives only, whereas the procedure proposed by the former handles general multiple objective problems (with any number of objectives) by visual interactions. Although in principle, Dogrusoz' method is not restricted to continuous decision space, for all practical purposes, we take the decision space continuous, thus, this efficient frontier is in fact a continuous surface which represents the exact tradeoffs between a pair of two-dimensional value vectors. This curve will be hereafter called the tradeoff curve.

3.2.1 A Bi-objective Problem

To illustrate the idea on simplistic terms, let us now consider a bi-objective optimization problem. The efficient frontier of this problem can be obtained by optimizing one objective (main objective) and treating the other objective as a constraint (parametrized objective) and varying its right hand side (satisficing or aspiration level) parametrically. The resulting pairs of right hand side and optimum value are plotted and a tradeoff curve is obtained. As an example, let the problem at hand be,

$$\begin{gathered} \max \quad x_1 + 2x_2 \\ \max \quad 2x_1 + x_2 \\ \text{subject to} \\ 3x_1 + 2x_2 \leq 8 \\ x_1 + x_2 \leq 3 \\ x_1, x_2 \geq 0 \end{gathered} \tag{3.1}$$

Optimizing the first objective which becomes the main objective, and converting the second to a constraining objective which becomes the parametrized objective having

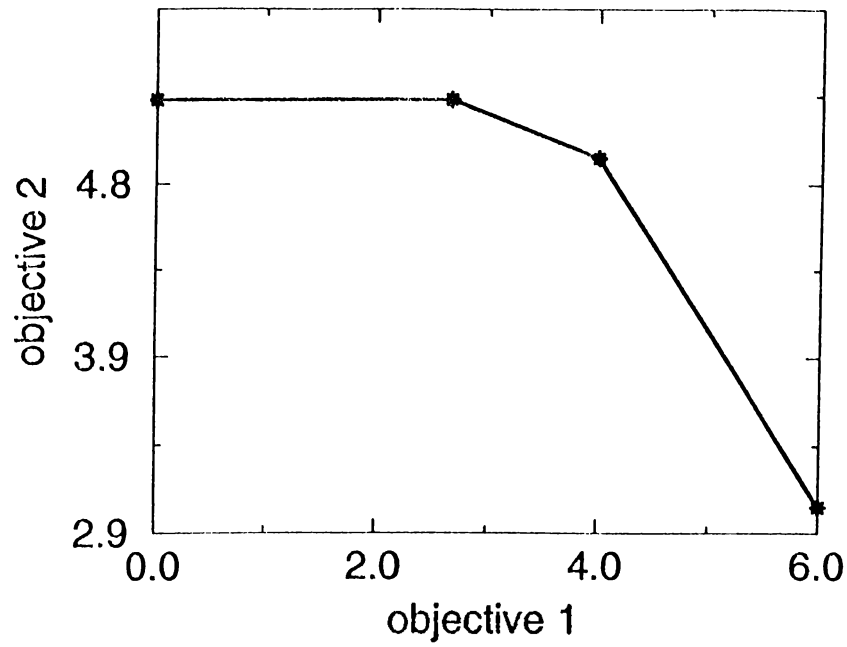

Figure 3.1: Tradeoff curve of objective 1 vs. objective 2

aspiration value p, we have,

$$\begin{array}{c} \max \quad x_1 + 2x_2 \\ \text{subject to} \\ 2x_1 + x_2 \geq p \\ 3x_1 + 2x_2 \leq 8 \\ x_1 + x_2 \leq 3 \\ x_1, x_2 \geq 0 \end{array} \tag{3.2}$$

where p is the parameter which varies from a lower bound up to the upper bound of the region in which this problem remains feasible.

Solving (3.2), we obtain the tradeoff curve in Fig. 3.1. This curve is very informative since it involves all efficient combinations of the two objective functions. The DM can easily see the conflicting nature of the objectives, tradeoff ratios in different intervals, and ideal and nadir solution values. S/he can easily locate a most preferred solution. Note that, maximizing the second objective function and parametrizing the first generates the

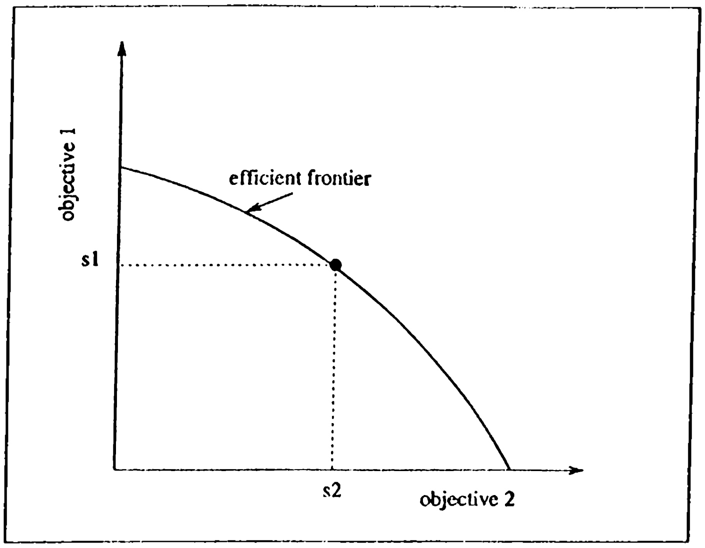

Figure 3.2: Tradeoff curve of objective 1 vs. objective 2

same tradeoff curve, which is intuitively obvious.

Here, visualization is considered to be of prime importance. Being able to actually "see" every point on the efficient frontier, and needing to tradeoff only two-dimensional vectors would facilitate the DM's and the analyst's job (boosting the effectiveness of the use of the DM's judgement). The analyst will only present the DM the tradeoff curve, and, depending on the answer, start a new iteration if necessary in the case of more than two objectives. In the example above, no further iteration is needed since there are two objective functions only.

Remark 3.1 *The tradeoff curve obtained by choosing the first objective as the main objective and the second as the parametrized one would be the same had we reversed the order of the objectives (i.e., second objective as the main objective, and first objective as the parametrized one). This is due to the fact that we are interested in optimizing one of the functions by keeping the other at a satisficing value, therefore, as Fig. 3.2 illustrates, the order in which the two objectives are treated does not, in fact matter. In Fig. 3.2, objective 1 is the main objective, and objective 2 is the parametrized objective, and the efficient frontier is as shown. Assume that the satisficing value of the second objective is* s_2. *Thus, the maximum of objective 1, for this aspiration level, occurs at the* *point indicated on the efficient frontier, and gives s_1 for objective 1. Had we reversed the objectives, then, for a satisficing value of s_1 for objective 1, would yield the same point on the efficient frontier since objective 2 is now maximized, and its value would be s_2.*

3.2.2 The Method For Multiple Objectives

We now review Dogrusoz's method in general, i.e. we will look at MCDM problems involving more than two objectives. Without loss of generality, as a convention, we assume that all objectives are maximizing, and in conflict with each other. A typical MCDM problem can be expressed as:

$$\begin{array}{c} \max \quad \mathbf{f} = (f_1(\mathbf{x}), f_2(\mathbf{x}), ..., f_n(\mathbf{x})) \\ \text{subject to} \\ \mathbf{x} \in X(\text{decision space}) \end{array} \tag{3.3}$$

where $\mathbf{x}$ is an m-dimensional vector of real numbers.

Since we assumed that objectives are in conflict, a vector $\mathbf{x}^*$ which maximizes all objectives simultaneously cannot be found. Instead, there exists a set of efficient solutions from which the DM has to choose the most preferred one. The method aims at isolating this solution in a simple and yet efficient manner.

Proceeding as in the bi-criteria case, we separate two objective functions $f_i(\mathbf{x})$ and $f_j(\mathbf{x})$, and preserve them as optimizing objectives for which a tradeoff curve will be obtained. Remaining objective functions are converted to constraining objectives by specifying a satisficing level for each, i.e. $f_k(\mathbf{x})$ is appended to system constraints as $f_k(\mathbf{x}) \geq \epsilon_k$, where ϵ_k is the satisficing value of the k'th objective function. For the time being, assume that satisficing values are given, and the problem at hand is feasible. One of the separated objectives will be optimized (main objective), and the other will be treated as another system constraint but its right hand side, p, will be changed parametrically (parametrized objective). This problem has been called the 'kth objective constraint problem' for a static p, however, as in our case p is dynamically changing, this parametric optimization problem will be called $P_i(\epsilon, j)$, where i refers to the index of the main objective and j to that of the parametrized objective. We will thus have the following problem if we choose to separate the first two objectives:

$$P_1(\epsilon, 2): \quad \begin{array}{c} \max \quad f_1(\mathrm{x}) \\ \text{subject to} \\ f_2(\mathrm{x}) \geq p \\ f_3(\mathrm{x}) \geq \epsilon_3 \\ f_4(\mathrm{x}) \geq \epsilon_4 \\ \vdots \\ f_n(\mathrm{x}) \geq \epsilon_n \\ \mathrm{x} \in \mathrm{X} \end{array} \tag{3.4}$$

Solving (3.4) by parametrically changing p (i.e., solving successive kth objective constraint problems by varying p) we draw a tradeoff curve between $f_1(\mathrm{x})$ and $f_2(\mathrm{x})$, and the DM is expected to choose the best compromise solution of these two objectives. Let this solution be $f_1(\mathrm{x}) = \epsilon_1$, and $f_2(\mathrm{x}) = \epsilon_2$. Then, the satisficing value of $f_2(\mathrm{x})$ will be set to ϵ_2. Note that, the DM is sure that her/his satisficing values on the other objectives are achieved.

A new iteration will have to be done, this time by choosing another objective to be parametrized, and by adding $f_2(\mathrm{x}) \geq \epsilon_2$ to the system constraints. Let us proceed sequentially using indices and choose $f_3(\mathrm{x})$ to be parametrized, we have:

$$P_1(\epsilon, 3): \quad \begin{array}{c} \max \quad f_1(\mathrm{x}) \\ \text{subject to} \\ f_2(\mathrm{x}) \geq \epsilon_2 \\ f_3(\mathrm{x}) \geq p \\ f_4(\mathrm{x}) \geq \epsilon_4 \\ \vdots \\ f_n(\mathrm{x}) \geq \epsilon_n \\ \mathrm{x} \in \mathrm{X} \end{array} \tag{3.5}$$

The resulting tradeoff curve presented to the DM will allow her/him to make a choice that sets a new satisficing value to $f_3(\mathrm{x})$. S/he might choose to sacrifice from the main objective in order to improve the parametrized one depending on her/his judgement.

Next iteration is the same except index is increased by one and $f_4(x)$ is chosen to be parametrized. The corresponding problem is $P_1(\epsilon, 4)$ This time, $f_4(x)$ is set to a satisficing value and the tradeoff curve is obtained.

The algorithm proceeds like this until all objectives are compared with the first objective. The DM, then, can choose to repeat these steps by reoptimizing $f_1(x)$ or any other objective function instead of $f_1(x)$. Therefore, the main objective can be any of the objective functions. The stopping condition, therefore, is the choice of the DM. We expect that by applying the algorithm, the DM will eventually "learn" her/his problem, and take control over it. These issues will be discussed in the following section.

3.3 VISTA-The Algorithm

This crude idea establishes the basis of VISTA, but stops short for some important details to make it practically operational. In the following, we will remedy these shortcomings to make the idea operationally viable and demonstrate that it is so. First, we will summarize the steps of VISTA, then we will present the features that are developed to facilitate the implementation of our method.

3.3.1 VISTA-Visual Interactive Sequential Tradeoffs Algorithm

The steps followed can be resumed in the following algorithm which will thereafter be called VISTA (Visual Interactive Sequential Tradeoffs Algorithm):

step0 DM ranks the objectives in decreasing order of importance. Objectives are indexed with i, $i = 1, .., n$.

step1 Set $k = 2$. DM specifies aspiration levels for objectives $i \geq k + 1$, these are added to the original problem as constraints.

step2 While ($k < n + 1$) do {

step2.1 kth objective ($f_k(\mathbf{x})$) is set to its lower bound.

step2.2 $P_1(\epsilon, k)$ is solved. Resulting tradeoff curve is presented to the DM, and DM selects a compromise solution on this curve. Aspiration level of $f_k(\mathbf{x})$ is set equal to the selected point.

step2.3 Set $k = k + 1$ }

The above is one full iteration of the algorithm. If the DM is not satisfied with the solution obtained, then s/he makes a second iteration by reindexing the objectives, and using already obtained solution as aspiration levels of objectives in step1. These issues will be discussed in the following.

3.3.2 Starting The Algorithm

Here, we undertake the question 'How to start the procedure?' Initially, we assumed that the method starts with given aspiration levels. But now, there are practical questions related to the determination of the values of these parameters.

First, the problem should be feasible. We want to start the procedure with a feasible and preferably efficient solution vector $\mathbf{x}$ keeping in mind that at the very beginning, the decision maker does not know every aspect of the problem. The DM, therefore, should be assisted in finding an initial feasible solution that will be improved in each iteration of the algorithm. Second, starting satisficing values should be meaningful to the DM. This means that, for the sake of feasibility, the core of the problem should not be sacrificed. Third, as the initial values are in a sense forcing the DM to proceed toward some direction, these values should assure for convergence to a final decision efficiently. This is rather a difficult objective to achieve, because determination of the final solution depends on the DM's learning of her/his problem. These initial aspiration levels should not mistakenly lead her/him to undesired consequences. One way to avoid this problem, is to repeat the procedure from the scratch with different initial aspiration levels. Fourth, the choice of the initial aspiration values should assure the DM to learn and understand the problem situation, and become aware of her/his value system. Since each interaction with the DM, i.e. every choice on the tradeoff curve, is an indication of the value measure inherent in the DM, even if the DM has been unaware of this system, s/he discovers how much an increase in an objective is desirable when compared to a decrease in another one. If initial aspiration values are balanced (i.e., not some of them are close to their nadir and some others to their ideal values, but, all are in the middle of the way.), then conflicts will become apparent, and the impact of small changes will be easily detected.

Here, we propose two different methods in finding initial aspiration levels via interaction with the DM.

Method1. First method is to solve the maximization and minimization problems for each objective function with respect to system constraints. That is, we solve:

$$\begin{array}{ccc} \max \quad f_i(\mathbf{x}) & & \min \quad f_i(\mathbf{x}) \\ \text{subject to} & \text{and} & \text{subject to} \\ \mathbf{x} \in \mathbf{X}, i = 1,..,p & & \mathbf{x} \in \mathbf{X}, i = 1,..,p \end{array} \tag{3.6}$$

This will result in maximum and minimum possible values of each objective. The maximum value is called the "ideal" point (the upper bound) and the minimum is called the "nadir" point (the lower bound) -provided that it exists, of course.

Next, these points are presented to the DM, and the DM is expected to specify aspiration levels within these bounds. The kth objective constraint problem for the first objective (*main objective*) is solved by keeping the second objective (*parametrized objective*) function's right hand side at its nadir value. If the problem turns out to be feasible then this right hand side will be increased parametrically as large as possible to obtain the first tradeoff curve. If, however, the problem is not feasible, then the decision maker is asked to reevaluate the aspiration levels. This process is repeated until a feasible solution to start iterating the method is found.

Remark 3.2 *If the substitution of all nadir points for aspiration levels results in an infeasible solution, then the problem itself is infeasible.*

Method2. The second method that we propose in order to specify initial aspiration levels to the problem, is easier than the first one. It involves solving a weighted sum of the objective functions for arbitrary positive weights.

$$\begin{array}{c} \max\ \sum_i w_i.f_i(\mathbf{x}) \\ \text{subject to} \\ \mathbf{x} \in \mathbf{X},\ \ i = 1,..,p \\ \text{where} \\ \sum_i w_i = 1, w_i \geq 0. \end{array} \tag{3.7}$$

As indicated above, the choice of w_is is arbitrary, and supplying equal weights to all of them is possible. Once a solution vector x is found, right hand sides of all objective functions except f_1 and f_2 are computed simply by substituting this solution vector x in $f_i(\mathbf{x})$, $i = 3, ..., n$, and, $P_1(\epsilon, 2)$ is solved by parametrically varying the right hand side of f_2 starting from its lower bound.

Remark 3.3 *These starting points may affect the judgement of the decision maker in the process of converging to a compromise solution. Therefore, in order to justify the validity of a solution after a complete application of the algorithm, we may want to check the consistency of the Decision Maker by using a different set of initial aspiration levels. If the solutions obtained in both cases are close, we can then conclude that the DM is consistent in using his/her judgement. However, if the two solutions differ at a large extent (e.g. more than a prespecified α percentage for at least one objective function), a reiteration of the procedure with initially set aspiration levels is recommended.*

3.3.3 Terminating The Algorithm

Up to now, we described in detail how the method works and how solutions are generated for the DM to use his/her judgement. However, we also need to determine when the solution procedure should terminate with an efficient solution which is satisfactory to the DM. For this, we propose the following two methods:

Method1. When the DM thinks that a solution is the best, find a nearest efficient point to this solution if necessary (see subsection 3.4.1). Interact with DM. If s/he likes the solution then stop. Otherwise continue by trying to improve the chosen objective or objectives.

Method2. When during a full iteration (trade offs of f_1 vs. f_i, $i = 2, .., p$) the same epsilon vector is obtained, and if the DM is satisfied with the solution, then stop, the solution is efficient (see section 3.4.1, Theorem 3.3), else (the DM is not satisfied with the solution), perturb right hand side values by a small amount and continue iterating.

In order to give the DM an idea about whether a final solution is to be obtained soon or not, a procedure is suggested in subsection 3.3.6. This procedure keeps track of the values of the first objective with respect to the other objective functions when the latter were treated as constraints for each iteration separately. The DM therefore would know how much the process approached to a possibly unchanging compromise solution. S/he would then be able to choose to iterate further or not.

3.3.4 Computer Display Of Tradeoff Information To Aid DM

One merit of the advance in computer technology is that a large amount of information is accessible in a short period of time. Complicated programs and calculations can be handled easily, and due to developed interfaces, users need not be experts in programming.

In this subsection, we will present computer aided features that are developed to facilitate the implementation of our method especially in displaying tradeoff information. We will introduce a visual aid that is expected to eliminate possible confusion due to scaling of the plots. Then, we will present another visual aid, namely Convergence Plot, which will assist the DM during iterations. For problems involving many objective functions, a hybrid approach that eliminates some objectives from our consideration will be discussed. Then, we will present another extension to the method which involves simultaneous comparisons of three objective functions. We will discuss a scheme for a software program that is expected to help the DM by incorporating all the above suggestions. We will conclude the chapter by presenting step by step summary of these features applied to a conceptual decision making problem.

TRADEOFF CURVES

The method described above, can be best implemented with the use of a computer. Especially, for generating the tradeoff curves, a standard optimization package and a subroutine is generally enough. The DM specifies the lower bound from which the parameter of the second optimizing objective will start to be increased (or s/he could simply set it to its nadir value), and the corresponding tradeoff curve will be plotted by the computer program. This process can be applied repeatedly, and in seconds, therefore allowing the DM change her/his mind anywhere during the process.

A DECISION AID: MARGINAL RATE OF SUBSTITUTION CURVE

The power of the above method lies in the visualization of the information of tradeoff potential between objectives with associated alternative courses of action. We expect the DM to use her/his judgement more easily and effectively once the tradeoffs are presented on a two dimensional curve. We do not, however, content with this information only, since graphical displays may have scaling problems and may incorporate slight differences that are not easy to perceive by simple observation of the curves. We therefore try to support the DM in this decision making process by introducing another decision aid information, namely the marginal rate of substitution (MRS) curve.

The MRS curve is obtained by plotting the derivative of the tradeoff curve at every point, since the rate of substitution between f_1 and f_2 is by definition df_1/df_2 at a given f_2 value. Therefore, plotting this MRS curve, we provide the DM with extra information to show how much to sacrifice in one objective per unit gain in the other, or vice versa. Hence, the DM will have an overall view of gains and sacrifices, i.e. tradeoffs between two objectives. This information will be provided visually on the slope curve along with the tradeoff curve and will be presented to the DM in the easiest way possible by plotting it

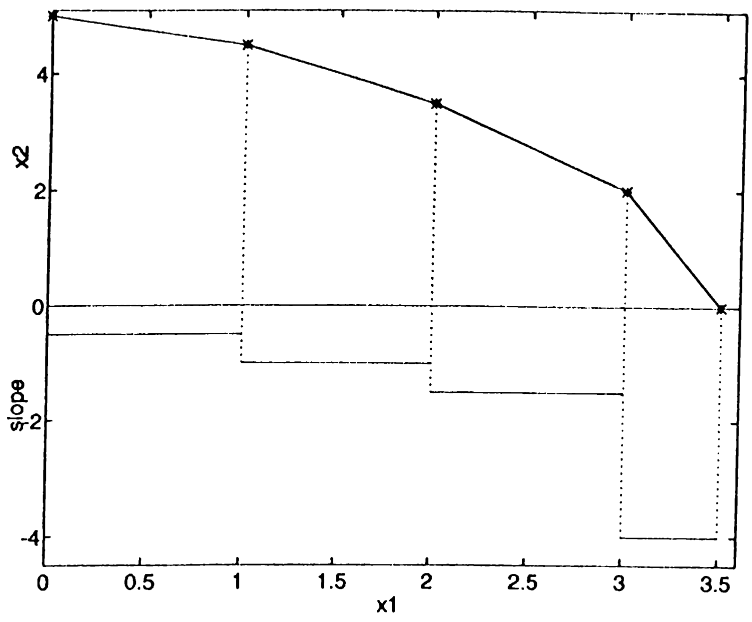

Figure 3.3: Tradeoff Curve plotted with slope information

right under the tradeoff curve display. Choosing the below part of the ordinate axis as the derivative axis, we plot the derivative of the tradeoff curve (i.e., MRS curve) under the initial tradeoff curve as in Fig. 3.3. This scheme will be very helpful to the DM since s/he will observe how the MRS between the two objective functions change, and be able to determine the point where her/his subjective measure of value is highest with respect to these two conflicting objectives.

TRADEOFF WITH A THIRD OBJECTIVE

So far, we have claimed that for a human being the easiest way of making tradeoff decisions is when the tradeoff information is given in the form of binary comparisons. Also, from a behavioral perspective, when other objectives are "guaranteed" to perform at least some value, making comparisons involving only a pair of objectives becomes trivially easy. Particularly, the visualization of the tradeoff information plays an essential role in this.

Since we assume that the relative importance of objectives do not differ from one another at a large extent, our method works best for problems having small to moderate number of objective functions. Therefore, we expect that the DM, after various iterations, has learned the aspects of the problem and internalized it. This means that the DM has started to ask more about the problem and can thus handle cognitively any extra information that we supply. Therefore, by making use of visualization as a decision aid, we extend our method to allow the DM to make comparisons involving *three objective functions simultaneously.* We, therefore, generate triary comparisons which need to be graphed on a three dimensional figure. However, three dimensional here means involving more than two objective functions, the plots will still be two dimensional in form.

We are going to propose two strategies to choose from, but there is no restriction on using them altogether.

Method1. The tradeoff curves for f_1 vs. f_i, $i = 2, \ldots, p$, are plotted on the same graph by changing the value of another objective function's right hand side. That is, the following procedure is used:

Procedure MULTIPLE1

step 0. Set p', its upper bound and step size δ

repeat

step 1. Solve

$$\begin{array}{c} \max \quad f_1(\mathbf{x}) \\ \text{subject to} \\ f_2(\mathbf{x}) \geq p \\ f_3(\mathbf{x}) \geq p' \\ f_i(\mathbf{x}) \geq \epsilon_i \\ i \neq 1,2,3 \\ \mathbf{x} \in \mathbf{X} \end{array} \tag{3.8}$$

by parametrically changing p as usual,

step 2. Set p'=$p' + \delta$.

until p' reaches its upper bound.

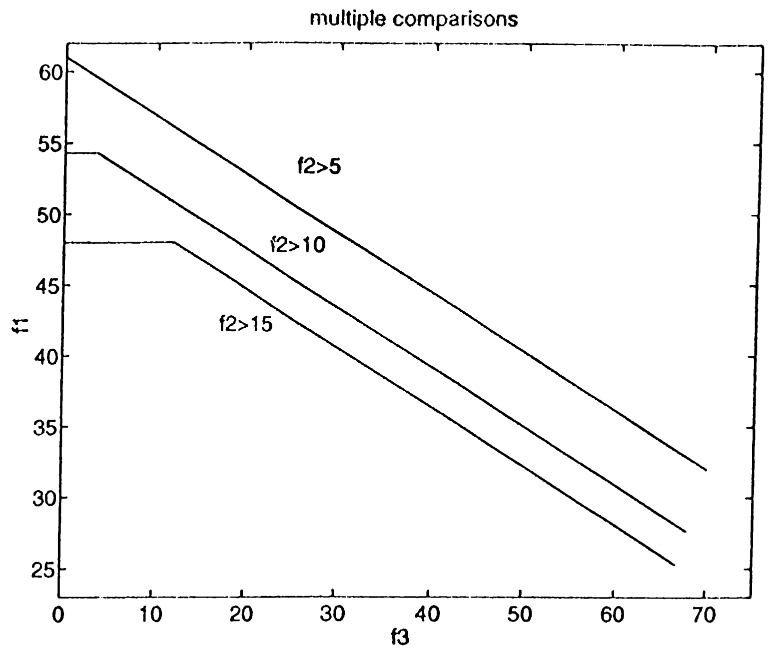

Figure 3.4: Tradeoff Curves for multiple comparison, method1.

This will give multiple tradeoff curves to choose from depending on the achievement on these three objectives simultaneously. Here, since the levels of the remaining objective functions will be kept constant, one good strategy is to determine a minimum valued curve (where p' is smallest) which will give the best performance of the first two objectives. Fig. 3.4 illustrates this procedure. This method should be used for a small range of p' and preferably not very small δ to allow the DM to differentiate between the tradeoff curves.

Method2. The second method is developed again for an 'experienced' DM, i.e., the DM has to have performed various iterations of the initial algorithm and is about to come up with a decision, and wishes to further investigate a region where her/his most preferred solution is expected to be found by considering a third objective function. This time, the tradeoff curve between two objective functions f_i and f_j is plotted as usual, and for each point $(c1, c2)$ on this curve, the solution of the following auxiliary optimization problem is plotted also:

$$
\begin{aligned}
& \max \quad f_m(\mathbf{x}) \\
& \text{subject to} \\
& f_i(\mathbf{x}) \geq c_1 \\
& f_j(\mathbf{x}) \geq c_2 \\
& f_k(\mathbf{x}) \geq s_k \qquad \text{for } k = 1, \ldots, p, k \neq i, j, m \\
& \mathbf{x} \in \mathbf{X}
\end{aligned}
\tag{3.9}
$$

where f_m is the third objective function chosen to be investigated, and s_k's are current aspiration levels of the remaining objectives.

Having plotted this curve, the DM will know how much s/he will have on the mth objective given that s/he locates a compromise solution on the tradeoff curve between f_i and f_j. Fig. 3.5 illustrates the usage of this method.

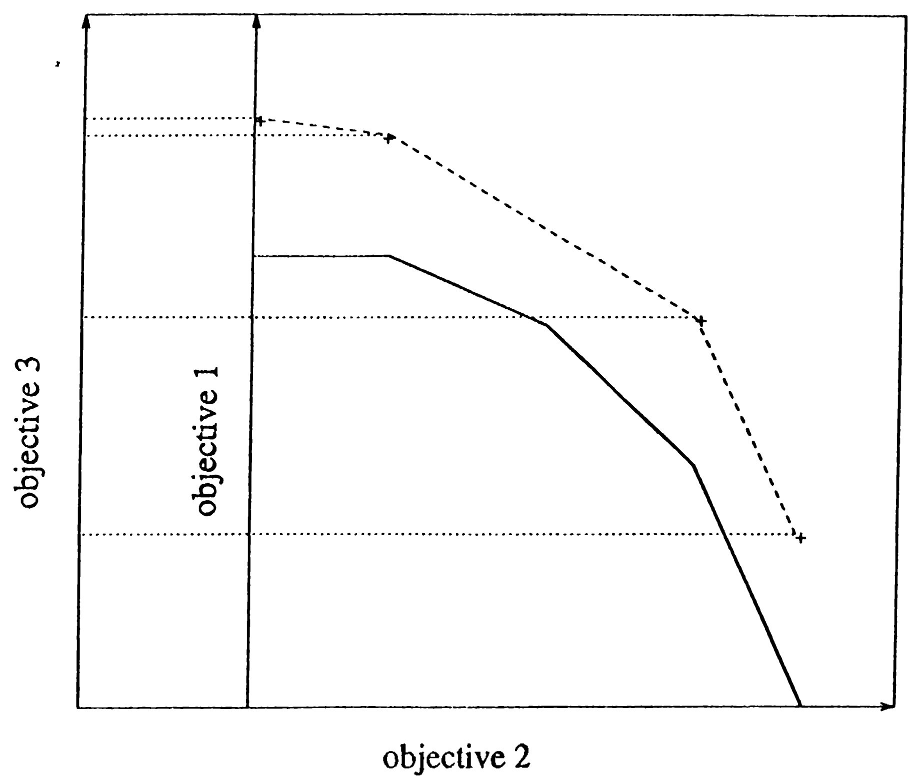

Figure 3.5: Tradeoff Curves for multiple comparison, method2.

TRACKING MAIN OBJECTIVE FUNCTION VALUE FOR CONVERGENCE

In this section, we are concerned with providing a scheme of tracking the progress made toward arriving at a final solution. For this, we suggest the use of two different methods separately:

Single Plot The variation of the maximized value of the main objective function is visualized by a plot. This plot displays the maximum value of the main objective function obtained by each choice on the tradeoff curve as the procedure proceeds. The DM can thus follow the progress, and determines how to finalize the procedure. We call this plot the *Convergence Plot*, and every point on this plot represents the performance of the main objective when choice is made by the DM after $P(\epsilon_i, j)$ is solved repeatedly with every $j = 2, ..., p$. The vertical axis of the Convergence Plot

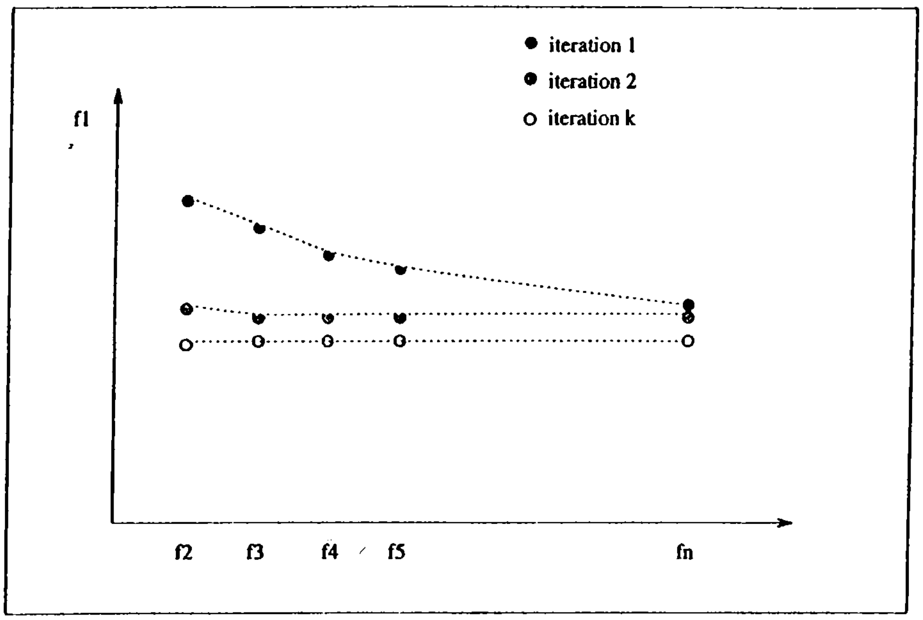

Figure 3.6: Convergence Plot

in Fig. 3.6 is the value of the main objective function, and on the horizontal axis is the index of the objective function that is currently parametrized. Each point on the plot determines the value of the objective function on the vertical axis, obtained from the interaction process with the DM. Having kept initial aspiration levels not too high, we expect that at each interaction, the DM will be willing to increase the values of these objectives by compromising (with some sacrifice) from the main objective. Therefore, the level of the main objective is subject to decrease after each tradeoff choice, and is expected to level off towards a final choice. Note that, the main objective function is not necessarily decreasing at every step of the algorithm. There are cases such that the marginal rate of substitution between the main and the parametrized objectives is very high, and the DM is not willing to sacrifice a huge amount from the main objective for a relatively little gain from the parametrized one. For example, in Fig. 3.7, At P, for improving f_2 some δ units, the DM should

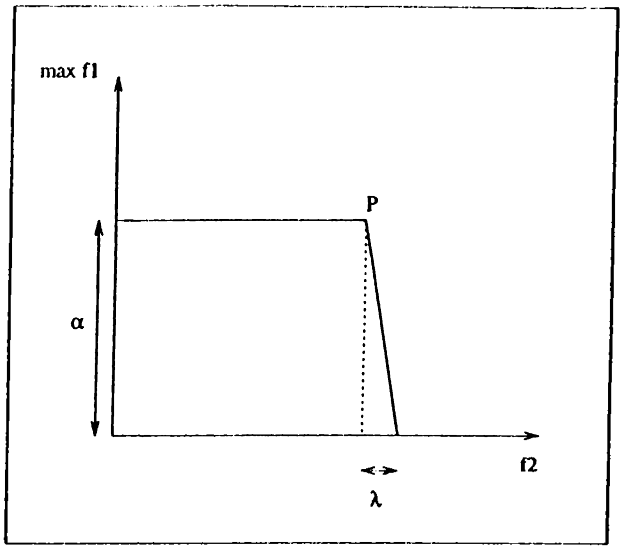

Figure 3.7: example for not changing f1

sacrifice $\delta(\alpha/\lambda)$ units from f_1. However, this ratio α/λ may be so high that the DM may not want to decrease f_1. Therefore, in this step f_1 remains the same. This plot is considered as a test of consistency of the DM's choices. If this convergence scheme is not observed, then we suspect that the DM's actual decision/choice behavior is not consistent partly due to initial satisficing values. Therefore, in order to alleviate this problem, we may need to reiterate the procedure with different and carefully determined aspiration levels. If this situation persists, then another method should be suggested to the DM. The Hybrid Approach proposed in the next section is helpful in such a case.

Multi Plot Another approach for making the Convergence Plot more illustrative is to keep track of the values of the other objective functions also. For the DM, it is extremely informative to observe how every objective function has changed throughout the algorithm. Provided that the number of objectives is small this

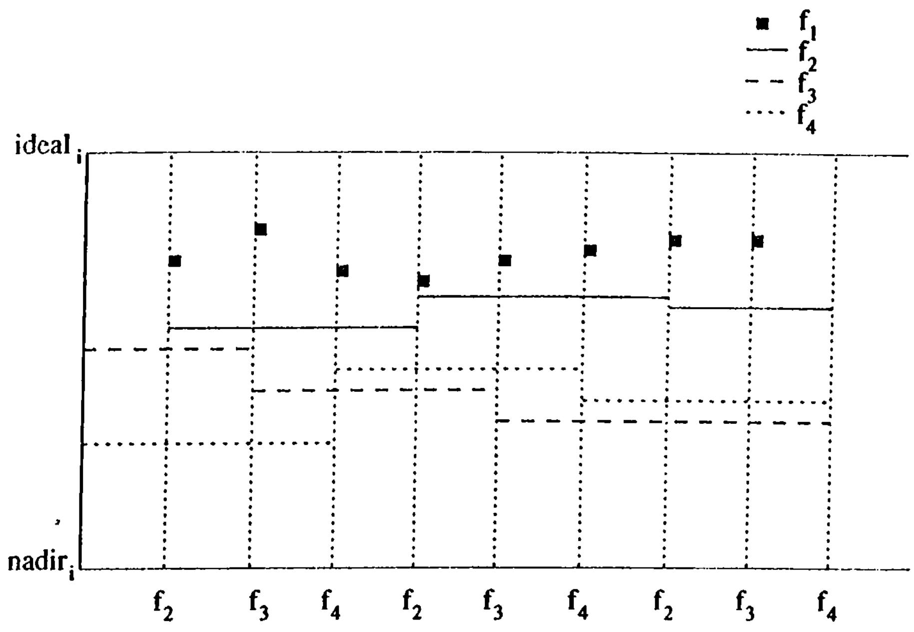

Figure 3.8: Multi Plot

information is available in Multi Plot in Fig. 3.8. An objective function's value on this plot is indicated after a choice on the tradeoff curve has been made. This time, on the horizontal axis are the indices of the parametrized objectives, and on the vertical axis is a scale to show how each objective function performs relative to *its* ideal and nadir values. Note that, on the horizontal axis, the indices of the parametrized objectives are repeated, meaning that a new iteration has been performed with the same main objective (here, with f_1). By using color or different line styles, this plot is made understandable, and analyzable for consistency.

Remark 3.4 *In Fig. 3.8, it is assumed that after each interaction with the DM (i.e., after each choice on the tradeoff curve), only the values of the main and parametrized objective functions are updated. However, it should be noted that, due to possible dependency to the* *same decision variables, values of some constraining objectives may have changed although their satisficing values remain unchanged. However, it is preferred not to update them, because otherwise the DM will lose control of the problem since these values would change dynamically. This issue will be discussed in section 3.4.*

As it appears in the above discussion, these tools are highly intuitive since they are intended to be visual aids to facilitate decision making. Therefore, their validity is subject to verification but again intuitively or experimentally, since they address to the cognitive capacity of the DM.

A Hybrid Approach

When the problem at hand consists of numerous objective functions, it may be very difficult for the DM to iterate the algorithm through every objective function. Since our primary goal is to facilitate the DM's decision making process, we propose the use of a hybrid approach. The hybrid approach consists of treating some objectives as constraints and the rest with a weighted function. In MCDM, this approach was suggested by Yu [25], studied by Henig [5].

Objective functions that are considered to be more 'important' than the others (here, the word important is used cautiously to mean those objectives that refer to planning purpose rather than those that refer to hard constraints such as product mix constraints, or those that could not be eliminated due to other reasons) are handled as constraining objectives, and the rest are given not necessarily equal weights, and their weighted sum is incorporated as the single objective to the problem. For the sake of notation, let us divide the index set into two, and let J_1 denote the index set of the constraining objectives, and J_2 denote that of the remaining objectives.

$$(PH): \qquad \begin{array}{c} \max \quad \sum_{i\in J_2} w_i . f_i(\mathbf{x}) \\ \text{subject to} \\ f_i(\mathbf{x}) \geq \epsilon_i, \;\; i \in J_1, \\ \mathbf{x} \in \mathbf{X}, \;\; w_i \geq 0. \end{array} \tag{3.10}$$

We now propose a procedure for solving this single objective optimization problem. The steps of the procedure HYBRID are as follows:

procedure HYBRID;

*step0.*Determine sets J_1 and J_2. Set satisficing values for objectives in J_1 (i.e., the vector ϵ), and set weights for objectives in J_2 (for convention give them equal positive weights). Choose two objective functions f_j and f_k, $j \neq k$, and $j, k \in J_1$ to start with.

*step1.*Solve the problem (PH) , and with the obtained solution x, evaluate every objective function in both J_1 and J_2. If the DM is satisfied with this solution y, then stop, else goto step2.

*step2.*Solve the standard ϵ-constraint problem using all objective functions (in both J_1 and J_2) with f_j in the objective function, by parametrically changing the right hand side of f_k as usual, and obtain the tradeoff curve between f_j and f_k to present to the DM.

*step3.*Present this curve to the DM, interact with her/him, and upon the choice of a point on the curve turn to step1 with another pair of objective functions, and updated right hand side vector.

The determination of the sets J_1 and J_2 is important, since the form of the objective functions chosen to belong to each of them provides the efficiency of the obtained solution. Henig [5] proves that when the objective functions in J_2 are concave, and those in J_1 are concave or quasiconcave, then the solution to (PH), obtained with positive weights in the objective function is efficient.

3.3.5 Classification of The Features

The operational principles and additional features of VISTA have been presented in previous sections. In this section, a clearcut distinction of which to be handled by the analyst and which by the Decision Maker will be made. Furthermore, the features themselves will be categorized according to specific purposes they serve. These clarifications will be made by describing the steps of VISTA verbally as applied on a conceptual decision making problem.

CLASSIFICATION OF DECISION AIDS

The features that have been proposed to accompany the basic idea of tradeoff curve generation are classified into two groups according to their purpose of use.

1. *Facilitating Aids:* The features that fall in this category are START, TERMINATE and HYBRID procedures. These are used particularly for helping the DM in applying VISTA, the algorithm itself. They are relatively more technical than others, and their impact is high on the final solution.

2. *Information Aids:* SLOPE, CONVERGENCE PLOTS and MULTIPLE COMPARISONS are information aids that help the DM in applying her/his *judgement*. Their use supplies extra information apart from tradeoff curves that VISTA generates. As VISTA aims at facilitating the decision making process, these aids are crucial in the sense that they represent the correct tradeoff information, the history of the process, and, multidimensional comparisons, respectively. For an efficient implementation of VISTA, these aids are indispensable.

A DIVISION OF LABOR

VISTA incorporates two different types of work load. First, there is the decision making process which requires determination of satisficing values, choices on the tradeoff curves, termination, etc... These are perceived as a cognitive burden on the Decision Maker. Second, there is the computational process which, although complicated, is straightforward, and, requires calculations, optimizations, and drawing graphs. The Analyst is responsible of this process. In brief, all work except decision making is handled by the Analyst. This very division of labor suggests that a computer software can replace the analyst, and the DM could work interactively with a computer on her/his problem. In subsection 3.3.6, we will describe a scheme for the application of this idea.

SUMMARIZED DESCRIPTION OF THE ALGORITHM

VISTA, together with the decision aids is summarized in the following.

- *Starting the Algorithm:* If there are too many (more than 7) objective functions then go to Hybrid Approach, otherwise continue. Pick, if possible, a most important objective function, otherwise select one arbitrarily.
  Find ideal and nadir points of each objective separately, and by trial and error, specify satisficing values for each of them, *or* give weights (not necessarily equal) to the objective functions, and set all but two objective functions' right hand sides to the solution obtained.
  If this solution is satisfactory, then stop, otherwise go to Tradeoff Curve Generation.
- *Hybrid Approach:* Determine the objectives that are not wanted to be dealt with interactively, assign weights to them (equal weights will do), solve problem (PH), obtain the value of every objective function, go to Tradeoff Curve Generation, and turn back again.
- *Tradeoff Curve Generation:* Obtain a tradeoff curve between two objectives as described in the algorithm accompanied with the slope information curve (Marginal Rate of Substitution curve) (see Fig. 3.3), and let the DM make a choice on this curve.
- *Convergence Plot:* Continue iterating the algorithm, and at the same time keep a record of the objective function value pairs and present this plot to the DM (see Fig. 3.6 , and Fig. 3.8).
- *Third Objective:* If after many iterations, the DM is unsatisfied (indecisive), and asks for more, implement the feature of comparing three objective functions simultaneously, either by plotting the family of tradeoff curves by changing the right hand side of a third objective parametrically(*Method1*) (see Fig. 3.4), or, by optimizing a third objective given a chosen point on the tradeoff curve for as many points as it is wished (*Method2*) (see Fig. 3.5).

- *Termination:* When the DM is satisfied with a solution, and if it is efficient, then stop, if it is not efficient, then make the inequalities of the constraining objectives binding and present the solution to the DM. If s/he likes the solution, then stop (and check weak efficiency), otherwise continue iterating.
  If the solution does not change upon a full iteration, then the solution is efficient, and if the DM is satisfied with this solution, then stop, otherwise perturb the right hand side vector by a small amount, and continue iterating.

In order to make these steps more illustrative, the algorithm and decision aids are also summarized in Fig. 3.9. The distinction between facilitating and information aids are made clearer by using dashed lines to represent the information flow from the latter towards the former.

3.3.6 Use of Computer Graphics

This method supported with the above suggestions can be implemented as a software program that will be run in a windows environment. The analyst, therefore, will be replaced by a computer, and the DM will follow the steps of the algorithm by clicking on the mouse and typing in numbers. All the features of the method will be listed in the IMPLEMENT and OPTIONS menus, and the DM should choose START from the IMPLEMENT menu and determine an initial feasible solution, thus setting satisficing values. And proceeding from simplest towards more complicated analysis s/he will learn the MCDM problem at hand gradually and will be able to handle more and more information as the process is continued.

TRADEOFF in IMPLEMENT menu will draw the tradeoff curve of the current problem in a separate window. The DM can click on any point to see its coordinates. From the OPTIONS menu, when the SLOPE option is selected, the slope information is activated and this decision aid is used. Once a choice on the tradeoff curve is made it is selected with the mouse.

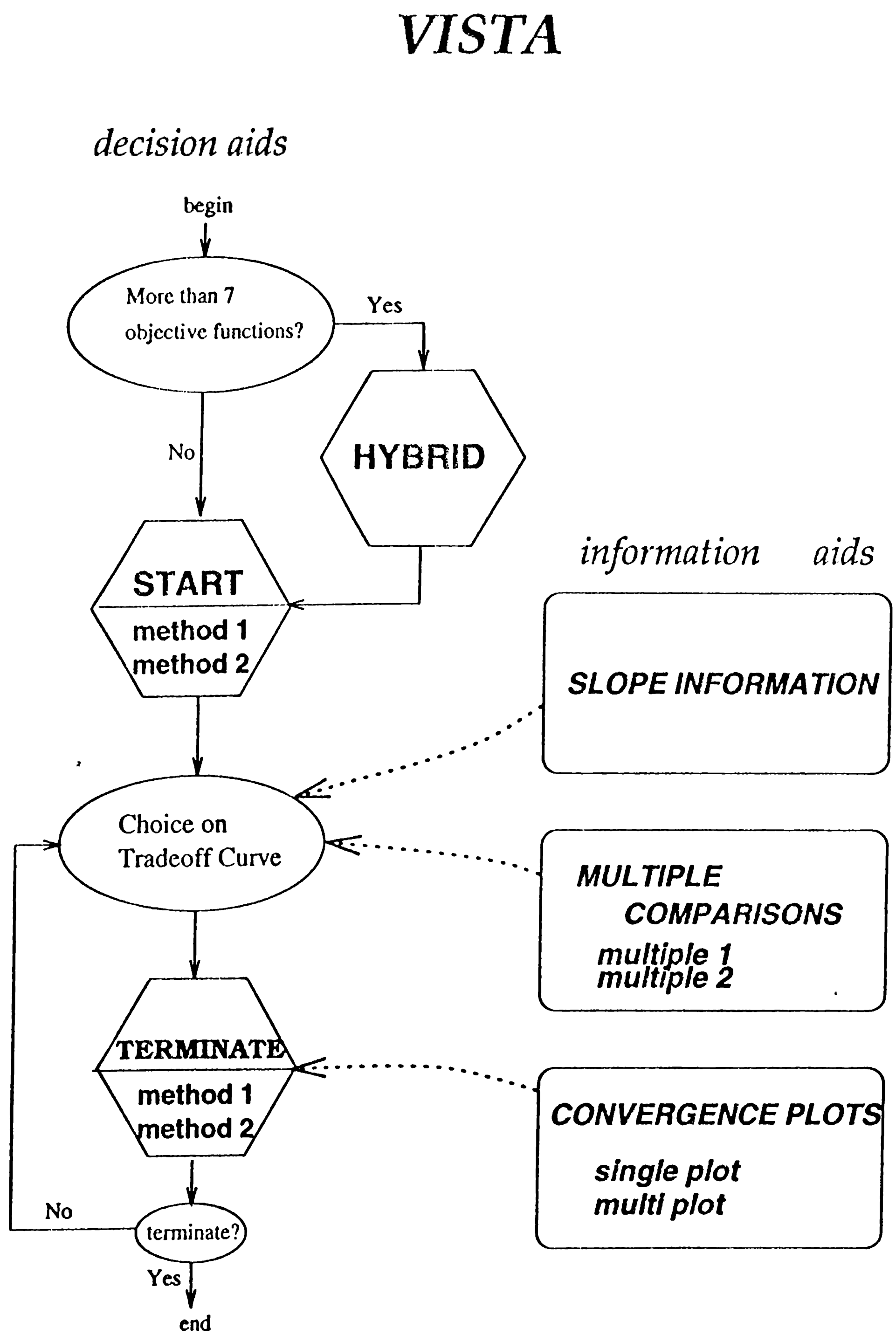

Figure 3.9: Summarized Description of VISTA

After a full iteration of the algorithm, or at any time during the iteration, ADJUST

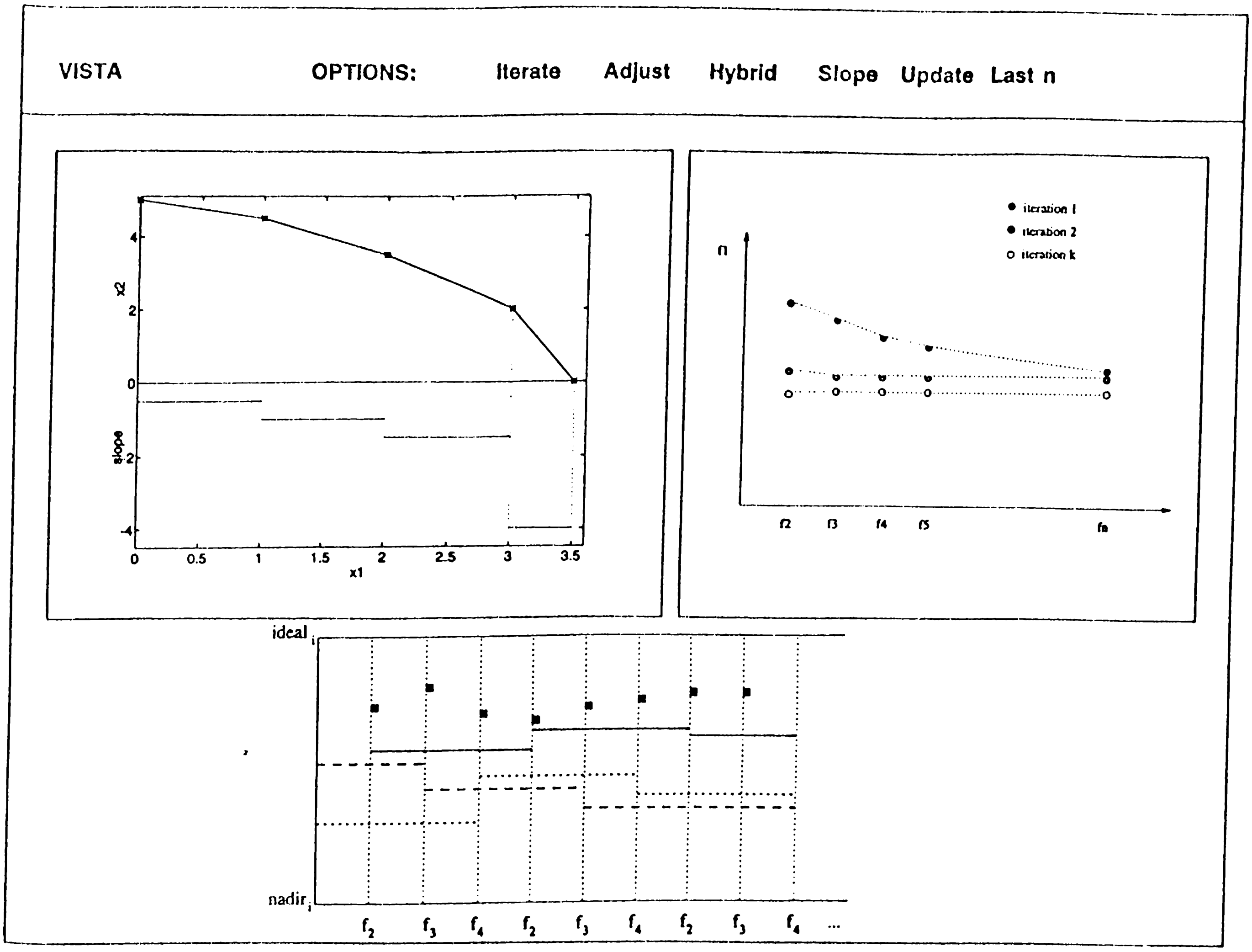

Figure 3.10: Proposed Scheme for VISTA Software

option can be chosen from the OPTIONS menu in order to find the right hand sides that are binding. This option evaluates every objective function with the current solution, and the DM may perform the next iteration with either another or the same objective function as the primary objective starting with these adjusted satisficing values. During this process, the levels of every objective function value will be displayed on a chart which also indicates the ideal and nadir values of each objective function together with the initial aspiration level. The DM will therefore have the chance to see the performance of all objective functions at the same time.

In another window, the graph called CONVERGENCE will keep track of the value of the primary objective function set at each kth constraint problem. (This plot is explained in subsection 3.4.2 in detail.) The continued examination of this plot will warn the DM whether a final solution is about to be reached or not, i.e., whether convergence or divergence is present. The DM is free to choose among SINGLE PLOT or MULTI PLOT to be displayed in the CONVERGENCE window. Another feature that will be implemented in this software is the possibility to make comparisons involving three objective functions simultaneously. Only after the DM has learned the problem and feels confident that s/he can handle three objective functions at the same time that the 3-DIM option from the OPTIONS menu should be chosen. Then, either one of Method1 and Method2 or both of them for three dimensional comparisons will be chosen.

The HYBRID option in the OPTIONS menu may be chosen if there are many objective functions at hand, and some seem to be not very important initially and is wanted to be taken care of in a simple yet efficient manner. Therefore, in HYBRID, the DM will select those objectives and assign weights depending on the judgement of the DM.

Fig. 3.10 shows the proposed scheme of the software which is expected to be helpful to the DM in finding her/his most preferred solution. This system is quite easy to use provided that a warm-up period is allowed for the DM to become intimate with the problem at hand.

3.4 Mathematical Foundation Of The Algorithm

3.4.1 Efficiency

Our method sequentially solves kth objective constraint (ϵ-constraint) problems suggested in [3]. The kth objective constraint problem is formulated (as in [3]) by taking the kth objective function f_k as the optimizing objective and letting all the other objectives f_j ($j \neq k$) be constraining objectives (inequality constraints). That is, given the initial

MCDM problem

$$(P): \qquad \max \quad \mathbf{f} = (f_1(\mathrm{x}), f_2(\mathrm{x}), ..., f_n(\mathrm{x})) \quad \text{subject to} \quad \mathrm{x} \in \mathrm{X}(\text{decision space}). \tag{3.11}$$

The kth objective constraint problem is defined as a scalar optimization problem

$$(P_k(\epsilon)): \qquad \max \quad f_k(\mathrm{x}) \quad \text{subject to} \quad f_i(\mathrm{x}) \geq \epsilon_i, \quad i = 1, ..., n, i \neq k, \quad \mathrm{x} \in \mathrm{X} \tag{3.12}$$

where $\epsilon = (\epsilon_1, ..., \epsilon_n) \in R^n$.

In economic contexts, $P_k(\epsilon)$ represents rationing mechanism, i.e. resource (or budget) directive planning method.

During one trade-off curve generation of our problem, we solve $P_k(\epsilon)$ by parametrically varying the right hand side, ϵ_i, of a predetermined constraint $f_i(\mathrm{x}) \geq \epsilon_i$. Thus, we solve subsequent kth objective constraint problems. Therefore, all properties and theorems related to this type of problems should apply to our case also. In this thesis, we have adopted a new notation to refer to this problem. We have used $P_k(\epsilon, j)$ to mean that the kth objective constraint problem is solved by parametrizing the jth objective function's right hand side.

The following theorems provide the relationship between the Pareto optimal solutions of the initial MCDM problem (P) and the solutions of the kth objective constraint problems $(P_k(\epsilon))$.

Theorem 3.1 *A point $\hat{\mathrm{x}}$ is a Pareto optimal solution of (P) if and only if $\hat{\mathrm{x}}$ solves $(P_k(\hat{\epsilon}))$ for every $k = 1, ..., p$, where $\hat{\epsilon} = f(\hat{\mathrm{x}})$.*

Proof. If $\hat{\mathrm{x}}$ is not a Pareto optimal solution of (P), there exists $\mathrm{x} \in \mathrm{X}$ such that $f_i(\mathrm{x}) \geq f_i(\hat{\mathrm{x}})$, $i = 1, ..., n$, with the strict inequality holding for at least one k. This implies that

$\hat{x}$ does not solve $(P_k(\hat{\epsilon}))$. Conversely, if $\hat{x}$ does not solve $(P_k(\hat{\epsilon}))$ for some k, then there exists $x \in X$ such that $f_k(x) > f_k(\hat{x})$ and $f_i(x) \geq f_i(\hat{x})$ $(i \neq k)$, implying that $\hat{x}$ is not a Pareto optimal solution of (P).

Theorem 3.2 *If $\hat{x}$ is a unique solution of $(P_k(\hat{\epsilon}))$ for some $k \in (1, ..., p)$ with $\hat{\epsilon} = f(\hat{x})$, then $\hat{x}$ is a Pareto optimal solution of (P).*

Proof. Since $\hat{x}$ uniquely maximizes $(P_k(\hat{\epsilon}))$, for all x satisfying $f_i(x) \geq f_i(\hat{x})$, $(i \neq k)$, $f_k(x) < f_k(\hat{x})$. Hence, $\hat{x}$ is a Pareto optimal solution of (P).

The above theorems are due to the equivalence theorem of Haimes, Lasdon and Wismer [3] which is a combination of these two theorems. A more general theorem is given as follows [1]:

Theorem 3.3 *$\hat{x}$ is a Pareto optimal solution of (P) if and only if there exists an ϵ such that $\hat{x}$ is a unique optimal solution of $(P_k(\epsilon))$.*

The utility of results in these theorems is that they allow efficient solutions to be generated by solving $P_k(\epsilon)$ for some ϵ and k. By systematically varying ϵ_i in $f_i(x) \geq \epsilon_i$, a subset of efficient solutions can be obtained. Computationally, any method based on Theorem 3.1 will not be too practical since it requires $P_k(\epsilon)$ to be solved for all k to generate each efficient solution.

On the other hand, Theorem 3.2 requires only one $P_k(\epsilon)$ to be solved. However, the uniqueness of the solution is sometimes difficult to check. If the objective functions are strictly concave, the uniqueness of the solution is guaranteed without further checking. If the objective functions are differentiable and known to be quasi concave, then second-order sufficiency conditions should be checked with the solution found.

Remark 3.5 *When objective functions are linear, uniqueness condition is needed to be checked. Unfortunately, this requires considerable effort and should be made at each iteration, and would make the decision procedure very complicated and cumbersome, so the* *algorithm is iterated without checking the uniqueness condition. Therefore, on a tradeoff curve, there may be Weak Pareto solutions that need to be improved (for Weak Pareto solutions, there may exist another solution that improves some criteria while other criteria are left unchanged, and hence these solutions seem to be inadequate as a decision making solution). However, after numerous iterations of the algorithm, we may want to see the closest efficient solution to the solution at hand. Then, with some extra effort, the current solution can be projected on the efficient surface. This is accomplished by the following procedure involving an LP:*

Procedure EFFICIENT

Solve:

$$\begin{array}{c} \max \quad \sum_i w_i f_i(\mathrm{x}) \\ \text{subject to} \\ f_i(\mathrm{x}) \geq f_i(\hat{\mathrm{x}}) \\ i = 1, ..., n, \mathrm{x} \in \mathbf{X} \end{array} \tag{3.13}$$

where $\hat{\mathrm{x}}$ *is the current solution. The solution of this problem will give an efficient solution depending on the* w *vector. This vector could be determined by interacting with the DM, in the form of an answer to "which objective would you like to improve most?". And depending on the answer, the weights are assigned so that weight on this objective would be the highest, and those of the remaining would be equal and they all sum up to one.*

Remark 3.6 *When objective functions are linear, as seen in Remark 3.5, solutions obtained may not be efficient, thus should be dealt with the procedure EFFICIENT. This procedure of finding an efficient solution, however, should be saved to latter iterations of the algorithm, so, in the intermediate iterations, we are contented with solutions of* $P_k(\epsilon, j)$ *, and do not question whether the solution is unique, thus efficient, or not. Since the algorithm works with tradeoff curves only, we know that the whole feasible region under these curves are already excluded from consideration throughout the process. Therefore, although these tradeoff curves may contain dominated solutions, it is obvious that VISTA generates solutions which are as close to the efficient frontier as possible. Therefore, the procedure EFFICIENT is not needed to be called frequently, instead, the final solution of* *each iteration should be elaborated with it.*

Remark 3.7 *Even when the objective functions are strictly concave, although a solution* $\hat{\mathbf{x}}$ *is efficient, objective functions which are treated as constraining objectives often need to be evaluated with this solution vector. Since,* $P_k(\epsilon, j)$ *assumes the constraining objectives be* $f_i(\mathbf{x}) \geq \epsilon_i$*, at a solution* $\hat{\mathbf{x}}$*,* f_i *may perform better than* ϵ_i*. This occurs due to lack of conflict between this objective and the main objective in this region. If, however, the DM wants to know the actual performance of the objective functions at* $\hat{\mathbf{x}}$*, the following procedure is used:*

Procedure ADJUST

Let the solution at hand be $\hat{\mathbf{x}}$*. Evaluate all objective functions with the given solution*

$$f(\hat{\mathbf{x}}) = (f_1(\hat{\mathbf{x}}), ..., f_n(\hat{\mathbf{x}})). \tag{3.14}$$

and assign these computed values to the epsilon vector.

$$\begin{aligned} \hat{\epsilon} &= f(\hat{\mathbf{x}}) \\ \hat{\epsilon}_i &= f_i(\hat{\mathbf{x}}), \forall i. \end{aligned} \tag{3.15}$$

The inequalities are thus binding. If this new right hand side vector is different from the one with which the solution above is obtained, then the epsilon vector is updated with the newly computed values. And the algorithm is continued with this new vector of aspiration levels (keeping the inequality sign for each constraint).

Remark 3.8 *The above arguments should make it clear that the solutions obtained in the objective space* $\mathbf{X}$ *are all efficient, however, the* ϵ *vector of the current right hand sides is not necessarily efficient, and needs to be ADJUST'ed. ADJUST procedure results in efficient solutions in the objective space* $\mathbf{Y}$*. It requires little effort to perform, and it is useful to show the DM the solution status.*

Corollary 3.1 *If the* ϵ*-vector does not change during a full iteration of the algorithm, then the solution of* $P_i(\epsilon, j)$ *is efficient.*

Proof. At each tradeoff curve generation, f_i vs. f_j in $P_i(\epsilon, j)$ the DM actually handles two maximization problems simultaneously, namely $P_i(\epsilon)$ and $P_j(\epsilon)$. The tradeoff curve obtained by solving $P_i(\epsilon, j)$ is the same (contains the same pairs of values) as the one that would be obtained had we chosen to solve $P_j(\epsilon, i)$. Therefore, at each trade-off choice, $P_i(\epsilon, j)$ is solved as well as $P_j(\epsilon, i)$, and this is true for all $i = 1, ..., p$. If the ϵ vector remains the same for $P_i(\epsilon, j)$ solved for every i, j pair, then this ϵ vector solves all $P_k(\epsilon)$ with equality holding for every objective treated as constraint. This is the case of Theorem 1, therefore the solution is efficient.

3.4.2 Convergence

VISTA is a sequential algorithm which aims at arriving at a final solution that the DM prefers over all other feasible solutions. The theory that proves the convergence of VISTA to such a final solution will be provided in the following.

VISTA is a straightforward (not user-driven) algorithm that requires as little effort as possible from the DM via sequential interactions. Thus, once the DM has made her/his choice on a tradeoff curve, s/he is led to another tradeoff curve to make another choice, and so on. Therefore, it is important that there is 'convergence' to a final solution even though the DM is not striving for this explicitly. The algorithm has to have properties inherent in itself that outputs a convergence scheme in order to single out a final solution, thus, the groundwork to show that this scheme is present in VISTA, will be laid. Assuming that a compromising choice on a tradeoff curve is utility maximizing, the following claim needs to be defined first and then proven.

Claim. The solutions generated by VISTA converge to a global optimum.

- *solutions generated:* The solutions generated by VISTA, are determined by the DM's choice on a tradeoff curve. Thus, when $P_1(\epsilon, 2)$ is solved, each point on the corresponding tradeoff curve is given by a function $f(y_1, y_2) = f(y_1, y_2, y_3^s, ..., y_n^s)$, where $y_i^s \in f$ is the satisficing value of the ith objective function at the current

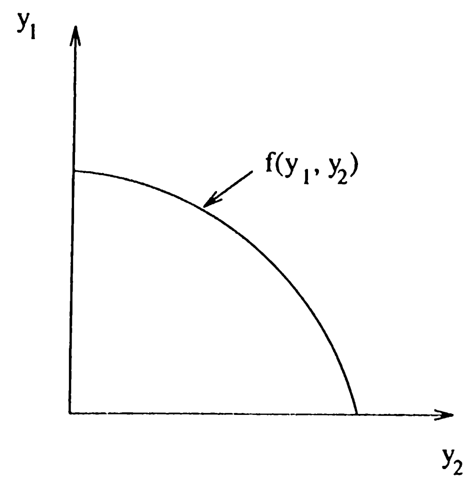

Figure 3.11: Tradeoff curve between f_1 and f_2

step and is constant. Assuming all objectives are in conflict with each other, at every point on the tradeoff curve the constraining objectives are achieved at their satisficing values (inequalities will be binding).

- *global optimum:* Since an MCDM problem involves the application of judgement by a DM, it is not possible to talk about an 'optimum solution' of a problem with multiple objectives. Although for an interactive MCDM method, the preference structure of the DM is assumed not to be known in advance and is not attempted to be assessed explicitly, for the sake of completeness of the theory, the form of this preference structure will be assumed, and will be used in proofs in place of a DM. Therefore, the use of the term global optimum of an MCDM problem with respect to the assumed Preference (or Utility or Value) function is allowed.

Let the tradeoff curve generated by $P_1(\epsilon, 2)$ be as in Fig. 3.11. Let the preference function of the DM be represented by $U = U(y_1, ..., y_n)$, where U is concave and monotonically increasing, meaning that higher values of objective functions are preferred over lower values. And, let $U_1 = U(y_1, y_2, y_3^s, ..., y_n^s)$ be the utility curve corresponding to the tradeoff curve in Fig. 3.11. This utility curve is shown in Fig. 3.12. A choice on the tradeoff curve is made so as to maximize U_1. The point P in Fig. 3.12 maximizes utility, thus the DM, whose preference function is as given, would identify P, the projection of P' on the

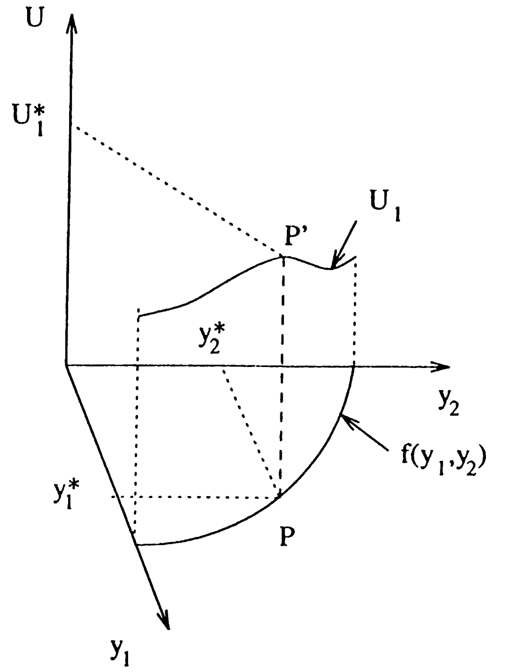

Figure 3.12: Tradeoff and Utility curves between f_1 and f_2

tradeoff curve, as the compromise solution. The value of the utility at P is given by $U_1^* = U(y_1^*, y_2^*, y_3^s, ..., y_n^s)$. The next tradeoff curve is generated by solving $P_1(\epsilon, 3)$, and by setting the satisficing value of y_2 equal to y_2^*. Therefore, the utility curve corresponding to the new tradeoff curve is $U_2 = U(y_1, y_2^*, y_3, y_4^s, ..., y_n^s)$, as shown in Fig. 3.13, similar to Fig. 3.12.

Conjecture 3.1 *Let the decision space X be convex, and let $f_i(\mathrm{x})$ be concave objective functions. Assuming that the utility function is concave, the sequence of choices made on the efficient curves by VISTA converges to the global optimum of the utility function defined.*

Proof. In order to prove convergence to the global optimum, it is sufficient to show that at every tradeoff choice, value of the utility function is increased. Then, by concavity of the utility function, the result follows.

Let $P_1(\epsilon, 2)$ be the current problem to be solved. Let the DM's choice on the tradeoff curve $f(y_1, y_2) = f(y_1, y_2, y_3^s, ..., y_n^s)$, be the point shown in Fig. 3.14. This chosen point

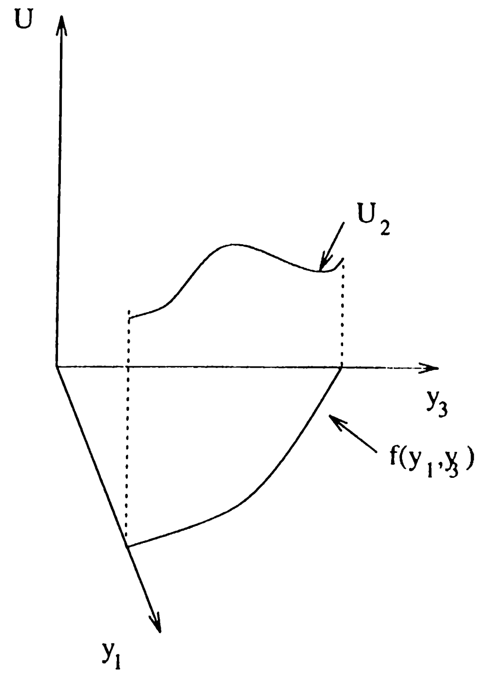

Figure 3.13: Tradeoff and Utility curves between f_1 and f_3

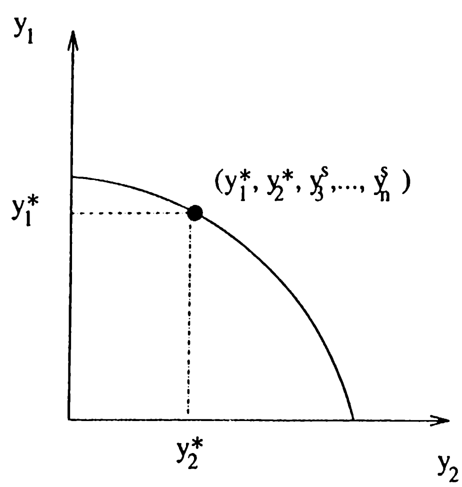

Figure 3.14: Tradeoff curve between f_1 and f_2

maximizes $U_1 = U(y_1, y_2, y_3^s, ..., y_n^s)$ at $(y_1^*, y_2^*, y_3^s, ..., y_n^s)$, this point is indicated in Fig. 3.14, where the value of the utility is $U_1^* = U(y_1^*, y_2^*, y_3^s, ..., y_n^s)$.

The next step is to solve $P_1(\epsilon, 3)$, and plot the tradeoff curve y_1 vs. y_3 keeping y_2 at y_2^*, thus plot $f(y_1, y_3) = f(y_1, y_2^*, y_3, ..., y_n^s)$ as in Fig. 3.15. Then, $U_2 = U(y_1, y_2^*, y_3, y_4^s, ..., y_n^s)$

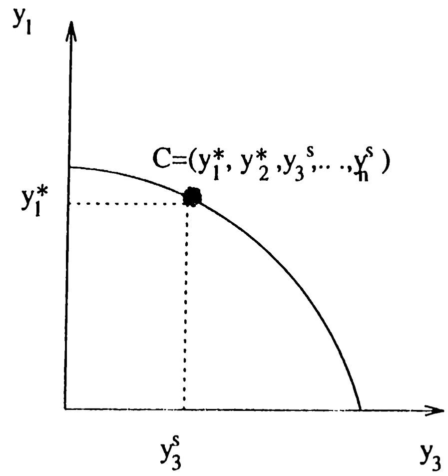

Figure 3.15: Tradeoff curve between f_1 and f_3

is to be maximized. On this tradeoff curve, however, note that the point $C = (y_1^*, y_2^*, y_3^s, ..., y_n^s)$ results in a utility value equal to U_1^*, since the point C has the same coordinates as the maximum solution of the previous iteration. Therefore, at this iteration, the maximizing point has to perform at least as much as C on U_2. Thus, $U_2^* \geq U_1^*$. Thus, at every choice on a new tradeoff curve, the utility will either increase or stay the same. If $U_2^* > U_1^*$, then the algorithm will be continued with another tradeoff curve obtained by solving $P_1(\epsilon, 4)$. If, however, $U_2^* = U_1^*$, then this implies that on this tradeoff curve the same set of points $C = (y_1^*, y_2^*, y_3^s, ..., y_n^s)$ has been chosen (by concavity of the utility function). Note that, this also implies that,

$$\frac{\partial U}{\partial y_1}\bigg|_{C=(y_1^*, y_2^*, y_3^s, ..., y_n^s)} = 0, \text{ and}$$
$$\frac{\partial U}{\partial y_2}\bigg|_{C=(y_1^*, y_2^*, y_3^s, ..., y_n^s)} = 0.$$

The algorithm is continued with $P_1(\epsilon, 4)$.

When at some stage during the algorithm, the same utility function value is being obtained through one full iteration for every tradeoff curve, i.e. $U_1^{i*} = U_2^{i*} = ... = U_n^{i*}$, where $U_1^i = U(y_1, y_2, y_3^i, ..., y_n^i)$, and U_i^* is the maximum of U_i, and occurs at a point $P' = (y_1^{i*}, y_2^{i*}, y_3^i, ..., y_n^i)$, then the following will hold:

$$\frac{\partial U}{\partial y_1}\bigg|_{P'=(y_1^{i*}, y_2^{i*}, y_3^i, ..., y_n^i)} = 0,$$
$$\frac{\partial U}{\partial y_2}\bigg|_{P'=(y_1^{i*}, y_2^{i*}, y_3^i, ..., y_n^i)} = 0,$$

$$\left.\frac{\partial U}{\partial y_3}\right|_{P'=(y_1^{i*},y_2^{i*},y_3^i,\ldots,y_n^i)} = 0,$$

$$\vdots$$

$$\left.\frac{\partial U}{\partial y_n}\right|_{P'=(y_1^{i*},y_2^{i*},y_3^i,\ldots,y_n^i)} = 0.$$

The above is the criterion of global optimality for a concave function, therefore, the point P' gives the highest utility value to the DM, and the algorithm terminates at that point.

Chapter 4

Examples to Illustrate VISTA

In this chapter two different example problems are studied to further illustrate the usage of VISTA. The examples are chosen so as to show how the distinction between an MCDM problem involving to solve successive LPs, and one involving NLPs is made clearer. First, a nonlinear MCDM problem is presented, and since NLPs require no extra effort to obtain an efficient solution, this problem is relatively less complicated. Then, a linear MCDM problem, which illustrates the procedure of obtaining an efficient solution, is studied.

4.1 A Nonlinear MCDM Problem

In this section, we will illustrate our method for a decision problem, involving four objectives. The problem is hypothetical, and it is so conceived that it is easily handled and yet demonstrates all features of the algorithm.

ROAD CONSTRUCTION:

Let us assume a hypothetical problem of constructing a road which is extremely important for the future of a rapidly growing city whose main source of income is tourism. The city in concern is located at the outskirts of a forest surrounding a lake which is a major tourist attraction. The only access to this city, however, is a highway located at the opposite side

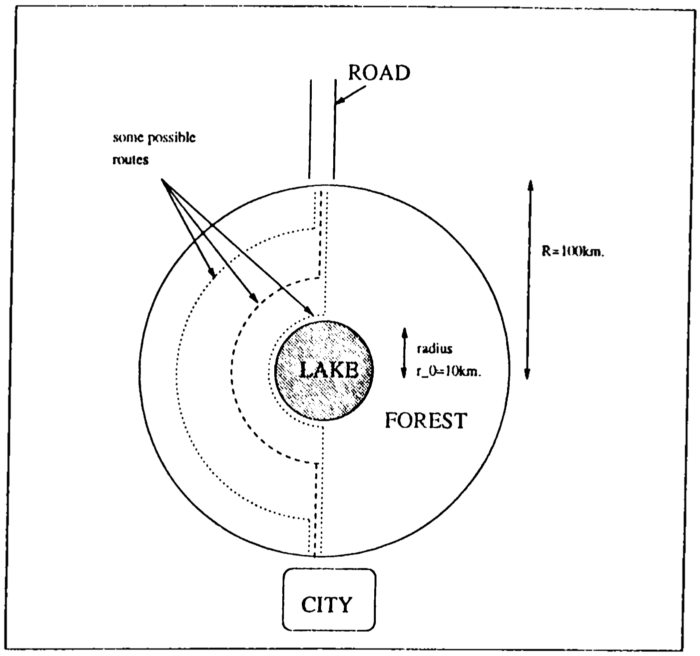

Figure 4.1: Situation in the Example Problem

of the lake, 200 km. away from the city. Therefore, the new road will have to pass close to the lake and inescapably through the forest. The Fig. 4.1 illustrates the situation.

4.1.1 Model Construction

We now construct an MCDM model to be used in a decision process by assuming that a feasible alternative will be similar to those shown in the Fig. 4.1 as 'some possible routes'.

Main objectives of the City Council are determined after various discussions on the subject, and are listed under two headings:

- **COST**: The Council would like to choose the least costly alternative,
- **TOURISM**: The Council would like to maximize income from tourism.

These two are classified as high level objectives. Although the first one is clear enough to be formulated with distance and construction cost data, the second objective should be further broken down into sub-objectives (in fact, surrogate objectives) to become operational. The Council decides that, income from tourism can be maximized by:

- **Environmental Protection**: The Council would like to preserve touristic attractions of the city forever, and,
- **Quality of the Road**: The Council believes that tourists would like to travel on wide and comfortable roads.

The first of the sub-objectives is further broken down into two:

- **Pollution**: The Council would like to minimize air and water pollution around the lake that will be caused by exhausts gases of the cars passing by,
- **Trees Cut**: The Council would like to minimize the number of trees cut from the forest during the construction of the road, to preserve the quality of the forest.

The second sub-objective, the quality of the road, is taken to be proportional to the width of the road, and the Council would like to maximize it. Summing up, we conceive to represent these objectives all, as follows:

1. Minimize Cost of Construction, f_1
2. Minimize Number of Trees Cut, f_2
3. Minimize Air and Water Pollution Around The Lake, f_3
4. Maximize the Width of The Road, f_4

Now, let us consider these objectives one by one, and construct their functional forms in terms of decision variables and parameters:

- **Cost, to be minimized**

  The cost of the road to be constructed is proportional to its area. It is:

$$f_1(r, \delta) = c_1\delta(2R + (\pi - 2)r). \tag{4.1}$$

r: the radius of the curvature of the road to the mid-point of the lake,
c_1: the cost of constructing $1km^2$ of road, and we set c_1=1,000 USD/(km^2), and
δ: the width (in meters) of the road.

Note that, r and δ are decision variables for f_1 since their values determine the level of that objective. The plots of f_1 vs r for various values of δ are given in Fig. 4.2.

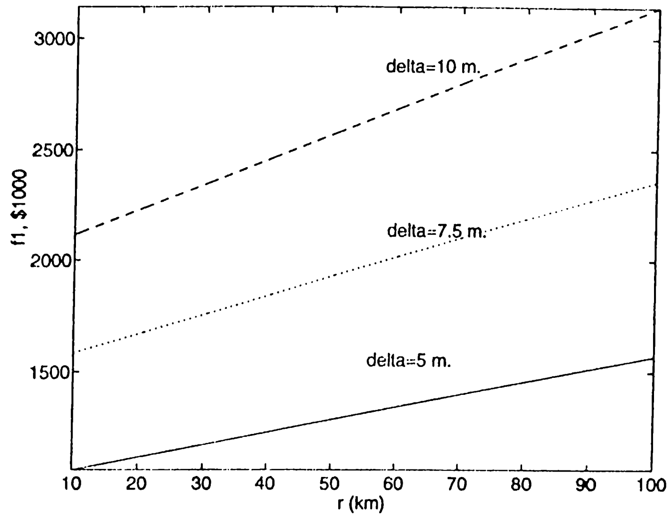

Figure 4.2: f1 vs r

- **Number of Trees Cut, to be minimized**

  The number of trees to be cut is proportional to the area of the road, and, to the density of the forest. The forest around the lake becomes denser towards to lake, and sparser towards outside. Assuming that the density $a = 0.25$ tree per m^2 near the lake and decreases exponentially, by a factor of $\exp(-(r - r_0)/r_n))$, where r_n is the normalizing constant and equals 30, we have:

$$f_2(r, \delta) = -2\delta a r_n \exp(-(R - r_0)/r_n) + [2\delta a r_n + \pi a \delta r] \exp(-(r - r_0)/r_n). \quad (4.2)$$

  We can see that the number of trees cut is a decreasing function of r (distance from the lake) in Fig. 4.3, therefore the objective of minimizing this number is in conflict with the objective of minimizing the cost.

- **Air and Water Pollution, to be minimized**

  Pollution of the Lake due to exhaust gases of the cars is considered on a scale of 0-1 such that highest value incurs when the road is closest to the lake, and decreases as the distance increases. Hence, the objective is conceived as to minimize

$$f_3(r) = r_0/r = 10/r. \quad (4.3)$$

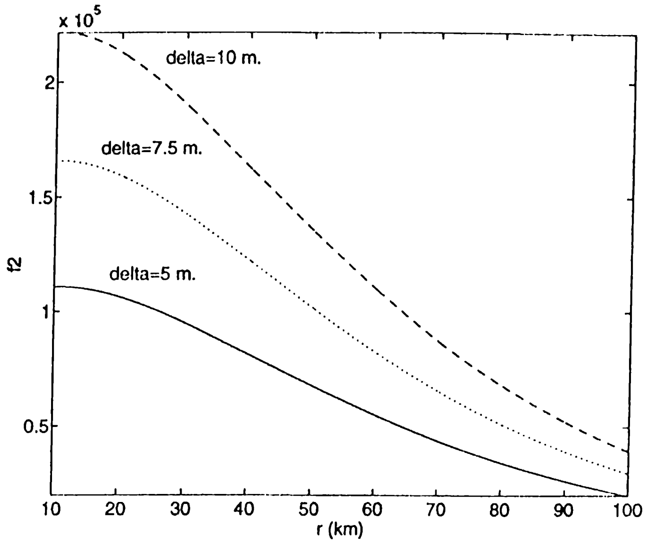

Figure 4.3: f2 vs r

This objective is clearly in conflict with the first objective.

- **Width of the Road, to be maximized**

  This objective is very simple and it is the maximization of the width of the road whose feasible values range within 5 and 10 meters. It is in conflict with the first and the second objectives, since an increase in the width of the road will increase construction costs as well as number of trees to be cut.

$$f_4(\delta) = \delta. \tag{4.4}$$

More formally, the model is constructed with a total of four objective variables $f_1, ..., f_4$, each being a function of decision variables r and δ, and evaluated by a rule of choice (criteria) of either minimization or maximization. Therefore, there are four objective functions and they are all optimizing.

4.1.2 Implementation of VISTA

VISTA has started by determining the ideal and nadir values for every objective function.
f_1: nadir= 3,141,600 USD., ideal= 1,057,100 USD.

f_2: nadir= 221,180 trees, ideal= 19,551 trees

f_3: nadir= 1, ideal= 0.1, and f_4: nadir= 5 m., ideal= 10 m.

The Committee is asked to rank the objectives in decreasing order of importance, and they find the above ranking appropriate. The Committee determines initial satisficing values for f_3 and f_4: ϵ_3=2/3, and $\epsilon_4=7m$. We then solve $P_1(\epsilon, 2)$, that is:

$$\begin{gathered} \min \quad f_1(r,\delta) \\ \text{subject to} \\ f_2(\mathrm{x}) \leq p \\ f_3(\mathrm{x}) \leq 2/3 \\ f_4(\mathrm{x}) \geq 7 \\ 10km \leq r \leq 100km \\ 5m \leq \delta \leq 10m \end{gathered} \tag{4.5}$$

The solution of the above nonlinear problem by parametrically changing p, generates the tradeoff curve between f_1 and f_2 as in Fig. 4.4.

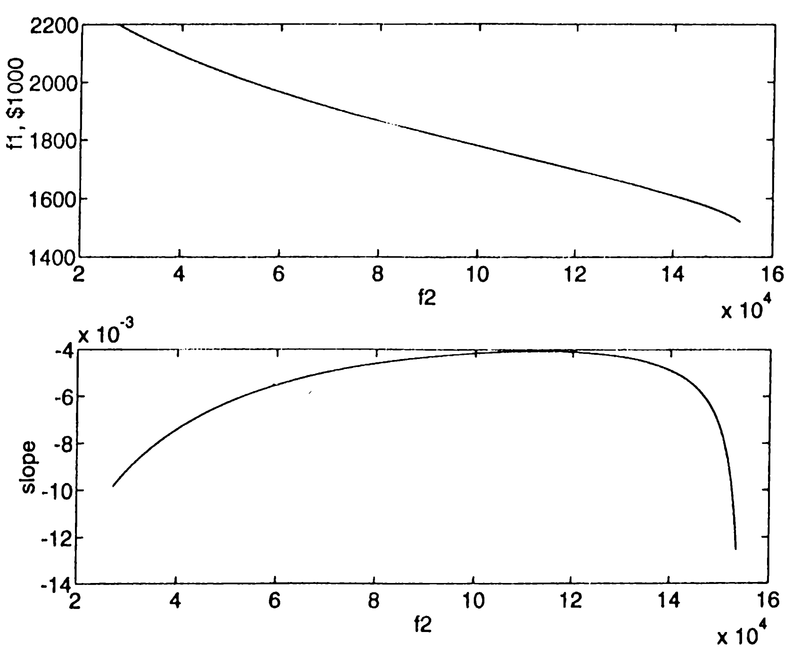

Figure 4.4: Tradeoff curve between f1 and f2

The DMs have to locate a point on this curve by trading off values of the cost and number of trees to be cut. They observe the slope (marginal rate of substitution) information supplied right below the tradeoff curve, and conclude that the number of trees to be cut should be no more than 100,000, and the corresponding cost of construction is affordable. We move to $P_1(\epsilon, 3)$:

$$\begin{gathered} \min \quad f_1(r, \delta) \\ \text{subject to} \\ f_3(\mathbf{x}) \leq p \\ f_2(\mathbf{x}) \leq 100000 \\ f_4(\mathbf{x}) \geq 7 \\ 10km \leq r \leq 100km \\ 5m \leq \delta \leq 10m \end{gathered} \tag{4.6}$$

which produces the following tradeoff curve in Fig. 4.5.

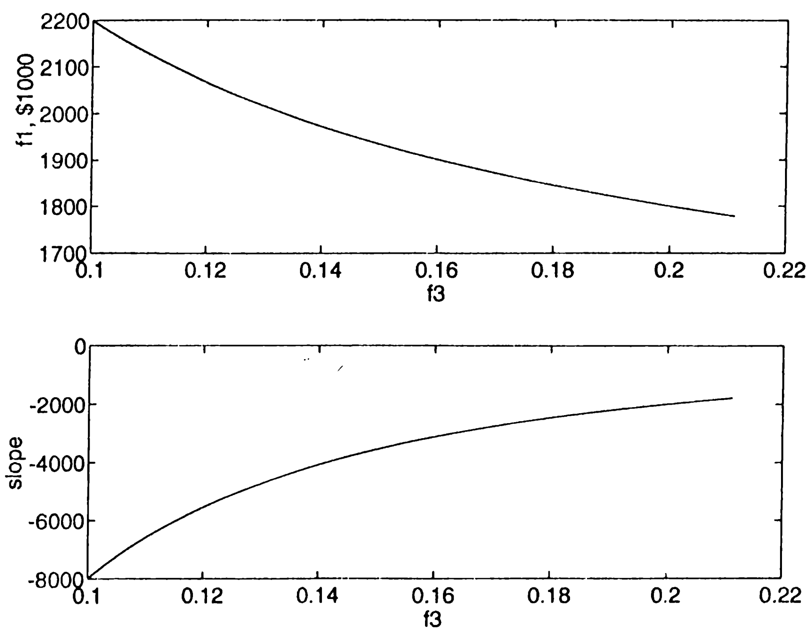

Figure 4.5: Tradeoff curve between f1 and f3

DMs observe that the pollution index on this curve is far less than their acceptable level, they are contented with even higher values of pollution, therefore they do not want to make a choice on this curve and proceed to the next step. Here, note that, the earlier choice on the number of trees to be cut forced the pollution index to become even smaller. So, the DMs were indecisive on whether to decrease the pollution index that much or not. Therefore, they preferred not to change their initial aspiration level of 2/3 at the moment. The next problem is $P_1(\epsilon, 4)$:

$$\begin{aligned} \min \quad & f_1(r,\delta) \\ & \text{subject to} \\ & f_4(\mathrm{x}) \le p \\ & f_2(\mathrm{x}) \le 100000 \\ & f_3(\mathrm{x}) \ge 2/3 \\ & 10km \le r \le 100km \\ & 5m \le \delta \le 10m \end{aligned} \tag{4.7}$$

The parametric optimization of this problem produces the tradeoff curve in Fig. 4.6. The

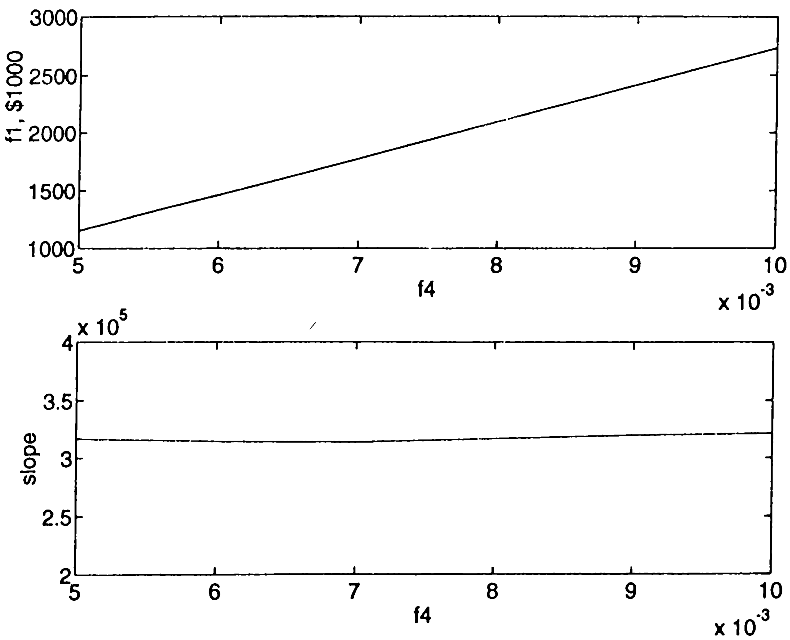

Figure 4.6: Tradeoff curve between f1 and f4

DMs of the Committee feel more concerned about the dramatic increase in the cost for just 1m. wider road. So they feel that a reduction of their initial aspiration level is well justified for a reduction in the cost of constructing the road. They set the right hand side of f_4 to 6 meters. Next, $P_1(\epsilon, 2)$ begins with

$$\begin{aligned} \min \quad & f_1(r, \delta) \\ & \text{subject to} \\ & f_2(\mathbf{x}) \leq p \\ & f_3(\mathbf{x}) \leq 2/3 \\ & f_4(\mathbf{x}) \geq 6 \\ & 10km \leq r \leq 100km \\ & 5m \leq \delta \leq 10m \end{aligned} \tag{4.8}$$

and produces the tradeoff curve in Fig. 4.7. DMs start to pay more attention to the slope

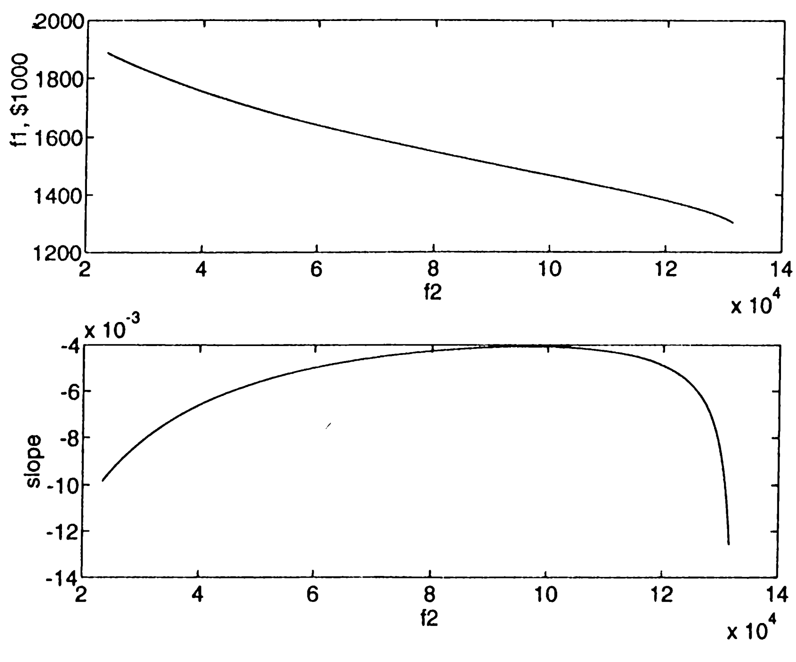

Figure 4.7: Tradeoff curve between f1 and f2, second iteration

information and observe that due to a flattening of the slope curve they could save almost 20,000 trees by a slight increase in the construction cost which is already reduced due to the decrease of the width. They ask the final parameters corresponding to this choice, and they are told that: f_1=1,548,000 USD, f_2=80,000 trees, f_3=0.196, f_4=6 meters. DMs feel that this solution comforts them best, and the procedure stops with the decision: r=50.83 km., δ=6 meters.

4.2 A Linear MCDM Problem

Some parts of this example problem has been quoted from [23], and involves finding a solution to the MCDM problem of a chocolate manufacturer.

CHOCOLATE PRODUCTION:

The company, a certain Chocolate Manufacturers, Inc. (Chocoman), is a manufacturer of various types of chocolate bars, candy and waffer. It has both production and marketing capability to produce and sell all or a mixture of the following products:

- Milk Chocolate Bars, 250g weight, (MB).
- Milk Chocolate Bars, 100g weight, (MS).
- Crunchy Chocolate Bars, 250g weight, (CB).
- Crunchy Chocolate Bars, 100g weight, (CS).
- Chocolate with Nuts, 250g weight, (NB).
- Chocolate with Nuts, 100g weight, (NS).
- Chocolate Candy, packed in 300g weight each, (CD).
- Chocolate Waffer, packed in 12 pcs at 10g per piece, (WF).

The materials to be used for production of these products are Cocoa, Milk, Nuts, Confectionary Sugar, Flour, Aluminum foil for packaging, Plastic sheets for packaging. Usage of these raw materials vary for each product. Also, the factory has facilities for production, namely, Cooking, Mixing, Forming, Grinding, Waffer Making, Cutting, Packaging 1, and Packaging 2 Units, and, of course Labor. Material and facility availabilities and costs are given, therefore a linear programming model is to be constructed given that there is limited demand for each end product.

4.2.1 Model Construction

As for the objectives of the problem, the company managers aspire for the following:

- Maximize Revenue
  Revenue is equal to price of each product multiplied by units produced. Therefore,
  Maximize $Z_1 = 375MB + 150MS + 400CB + 160CS + 420NB + 175NS + 400CD + 150WF$

- Maximize Profit
  Profit is equivalent to gross contribution in these models. The variable prices can be calculated and profit would be selling price less variable cost. Therefore,
  Maximize $Z_2 = 179.95MB + 82.90MS + 153.08CB + 72.15CS + 129.95NB + 69.90NS + 208.50CD + 83WF$

- Maximize Market Share of Chocolate Bar Products
  Maximizing market share is equivalent to maximizing the tonnage of chocolate bars produced. For every 1000 unit of 250 gram chocolate bar, the weight would be 0.25 tons. Therefore,
  Maximize $Z_3 = 0.25MB + 0.10MS + 0.25CB + 0.10CS + 0.25NB + 0.10NS$

- Maximize the Units of Products Produced
  The advertising department would like to maximize the exposure of the company's brand name. In order to achieve this, the consumers had to be constantly reminded of the product through the packaging. In order to maximize on the repeated exposure of the brand, the units of products sold must be maximized. This, therefore, would be,
  Maximize $Z_4 = MB + MS + CB + CS + NB + NS + CD + WF$

- Maximize Plant Machinery Utilization
  To maximize plant utilization, the machines should be loaded to the maximum tonnage for maximum number of hours. Since coefficients of usage are known, the total machine capacity utilization can be calculated by adding usage of each machine

per product. For the cutting and packaging machines, since there are no restrictions on weight, the loading can be assumed unity, and simply added to the others. The objective function would be,

Maximize $Z_5 = 1.65MB + 0.9MS + 1.975CB + 1.03CS + 1.75NB + 0.94NS + 4.2CD + 1.006WF$

- Minimize Budget

  In order to buy raw materials needed and pay for the labor, a budget should be allocated beforehand. Therefore, the minimization of the budget is a company objective as well. The corresponding objective is,

  Minimize $Z_6 = 195.05MB + 67.1MS + 246.93CB + 87.85CS + 290.05NB + 105.1NS + 191.5CD + 67WF$

The overall problem, thus, is:

$$
\begin{aligned}
\max\ Z_1 &= 375MB + 150.0MS + 400CB + 160CS + 420NB + 175NS \\
&\quad + 400CD + 150WF \\
\max\ Z_2 &= 179.95MB + 82.90MS + 153.08CB + 72.15CS + 129.95NB \\
&\quad + 69.9NS + 208.50CD + 83WF \\
\max\ Z_3 &= 0.25 \quad B + 0.10MS + 0.25CB + 0.10CS + 0.25NB + 0.10NS \\
\max\ Z_4 &= MB + MS + CB + CS + NB + NS + CD + WF \\
\max\ Z_5 &= 1.65MB + 0.9MS + 1.975CB + 1.03CS + 1.75NB + 0.94NS \\
&\quad + 4.2CD + 1.006WF \\
\max\ Z_6 &= 195.05MB + 67.1MS + 246.93CB + 87.85CS + 290.05NB \\
&\quad + 105.1NS + 191.5CD + 67WF
\end{aligned}
\tag{4.9}
$$

subject to

$$
\begin{aligned}
&\text{FacilityConstraints}\\
&\text{Cook}: \ 0.5MB + 0.2MS + 0.425CB + 0.17CS + 0.35NB + 0.14NS\\
&\qquad + 0.6CD + 0.096WF \le 1000\\
&\text{Mix}: \ + 0.15CB + 0.06CS + 0.25NB + 0.1NS \le 200\\
&\text{Form}: \ 0.75MB + 0.3MS + 0.75CB + 0.3CS + 0.75NB + 0.3NS\\
&\qquad + 0.9CD + 0.36WF \le 1500\\
&\text{Grind}: \ 0.25CB + 0.1CS \le 200\\
&\text{Wfrmkg}: \ 0.3WF \le 100\\
&\text{Cut}: \ 0.1MB + 0.1MS + 0.1CB + 0.1CS + 0.1NB + 0.1NS + 0.2CD \le 400\\
&\text{Pkg1}: \ 0.25MB + 0.25CB + 0.25NB + 0.1WF \le 400\\
&\text{Pkg2}: \ 0.05MB + 0.3MS + 0.05CB + 0.3CS + 0.05NB + 0.3NS\\
&\qquad + 2.5CD + 0.15WF \le 1000\\
&\text{SizeMixConstraints}:\\
&MB - 0.6MS \le 0\\
&CB - 0.6CS \le 0\\
&NB - 0.6NS \le 0\\
&\text{ProductMixConstraint}:\\
&-56.25MB - 22.5MS - 60CB - 24CS - 63NB - 26.25NS + 400CD + 150WF \le 0\\
&\text{DemandConstraints}:\\
&MB \le 500MS \le 800\\
&CB \le 400CS \le 600\\
&NB \le 300NS \le 500\\
&CD \le 200WF \le 400\\
&\text{All variables are } \ge 0
\end{aligned}
\tag{4.10}
$$

Note that, in this formulation of the LP, the inclusion of the Budget Objective is due to De Novo Programming applied in [23]. This will allow more flexibility and full utilization of the budget allocated due to conflict among the objectives.

4.2.2 Implementation of VISTA

The manager of Chocoman (the DM) ranks the objective of maximization of profits first, but fails to give a proper ranking for the other objectives. Therefore, VISTA will consider $P_2(\epsilon, k)$ problems for $k = 1, 3, 4, 5, 6$.

With the help of the analyst of the consulting firm, the company manager starts iterating the algorithm with the following initial satisficing values that he specified by using his judgement,
$Z_3 \geq 400$, $Z_4 \geq 3200$, $Z_5 \geq 4250$, $Z_6 \leq 500000$.

The first iteration of the algorithm solves $P_2(\epsilon, 1)$. This corresponds to solving the problem for Z_2 given that all other objective functions are held at their satisficing values. VISTA solves $P_2(\epsilon, 1)$ by parametrically changing the RHS of Z_1 from zero until the problem becomes infeasible. The tradeoff curve in Fig. 4.8 is thus obtained and presented to the DM.

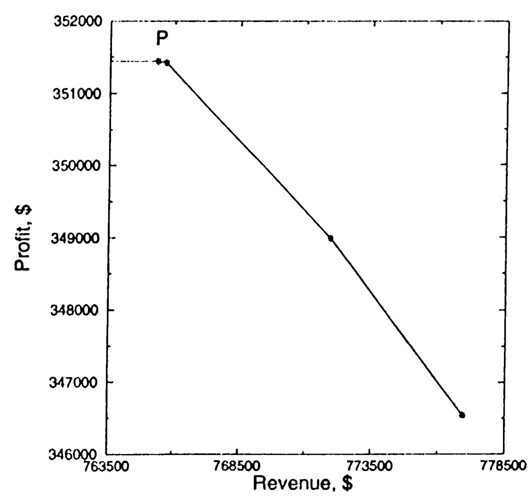

Figure 4.8: Tradeoff curve between Profit and Revenue

This figure clearly shows that, up to the point P=(765344, 351442), the objectives of maximizing revenue and maximizing profit are not in conflict. This indicates that given a certain level of revenue less than 765344, the maximum profit obtainable is constant at 351442. However, after P, the DM observes that, in order to increase revenues further, he has to sacrifice from profits that could be gained! That information actually is invaluably important, and, the DM, upon questioning the analyst learns that profits are lost due to increase in utilization of the machinery that are used to produce high priced products (which increase revenues). The DM, willing not to sacrifice from profits selects the point P on the tradeoff curve in Fig. 4.8.

Next, VISTA solves $P_2(\epsilon, 3)$, maximization of profits versus maximization of market share of chocolate bars. All other objectives are set at their satisficing values. The tradeoff curve is in Fig. 4.9, and again indicates a lack of conflict up to the point (422.5, 351442). The DM, still not wishing to sacrifice from his most preferred objective chooses this point which results in the highest profit level. Note that, at this point the market share is higher than its initial satisficing value.

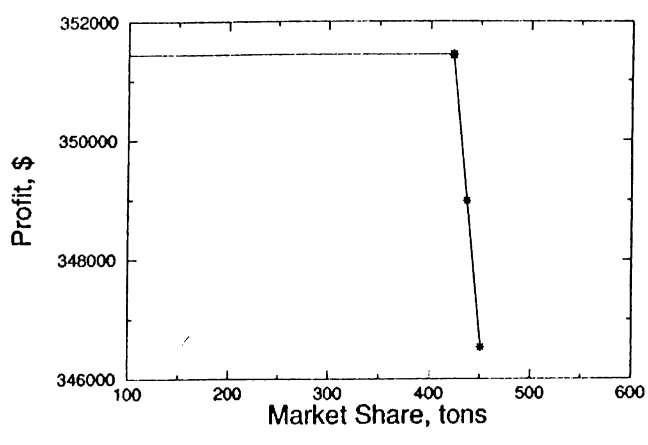

Figure 4.9: Tradeoff curve between Profit and Market Share

$P_2(\epsilon, 4)$ is solved similarly, producing the curve in Fig. 4.10. Similarly, the DM chooses (3288, 351442).

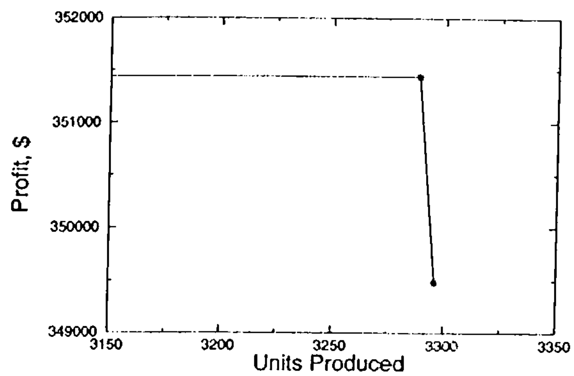

Figure 4.10: Tradeoff curve between Profit and Units Produced

$P_2(\epsilon, 5)$ corresponds to the tradeoff between maximizing profits and maximizing plant utilization, Fig. 4.11. This gives a rather flat curve, and the DM sticks with (4328, 351442).

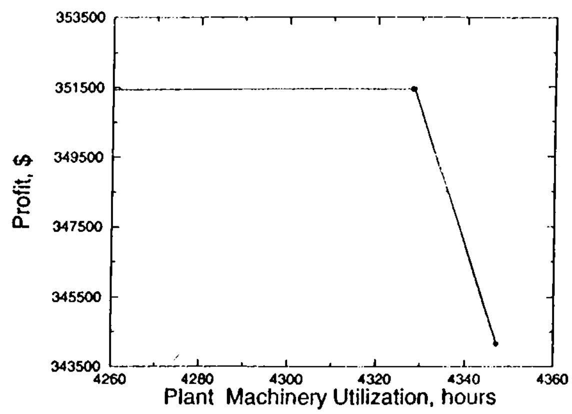

Figure 4.11: Tradeoff curve between Profit and Plant Utilization

The last step of this iteration is to solve $P_2(\epsilon, 6)$, maximization of profits versus minimization of budget. However, note that, if the DM is to choose the same profit value at this iteration too, then VISTA will satisfy one of the TERMINATION criteria, with an efficient solution, (see corollary 3.1). The corresponding tradeoff curve is given in Fig. 4.12 which is quite different from previous ones since it is increasing in both axes. However, knowing that these objectives strive to perform in reverse directions, this curve can be explained easily. The two tradeoff points are (413903, 351442.16), and, (413852, 351425). In order to increase profit by 17 USD, the budget allocated should be increased by almost 51 USD. The DM, thus, chooses not to spend that money for such a little gain.

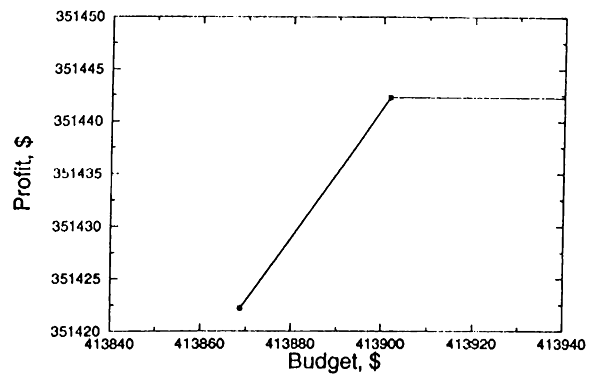

Figure 4.12: Tradeoff curve between Profit and Budget

At this point, VISTA performs TERMINATION procedure, and depending on whether the DM is satisfied with the solution at hand or not, the procedure stops. In this case, the DM is satisfied with the solution that gives, $Z_1 = 765344$, $Z_2 = 351442$, $Z_3 = 422.5$, $Z_4 = 3288$, $Z_5 = 4328$, $Z_6 = 413903$. This solution gives the production volume of each product in tons: MB = 480.0, MS = 800.0, CB = 360.0, CS = 600.0, NB = 90.5, NS = 500.0, CD = 124.6, WF = 333.2

It can be noted that the DM of this company believes that the objective of maximization of profits is extremely more important than the remaining ones. This choice scheme indicates that had we deleted all other objective functions, and optimized the second objective function alone subject to the problem constraints, we would obtain the same solution.

Chapter 5

Validation of VISTA

In this chapter we validate and compare VISTA with other interactive methods using a validation scheme, and in terms of interaction style, efficiency, and providing information. We will identify the major advantages of VISTA which make it perform better than other methods.

5.1 Interaction Style

In VISTA, a tradeoff curve between two of the objective functions is presented to the DM. The DM, in turn, is expected to choose one point which represents a compromise between these two objectives which can be interpreted as an assessment of a utility (preference) function by the DM on the tradeoff curve and maximize it.

Choice on this two dimensional curve is equivalent to comparing a set of two dimensional vectors. Knowing that making binary comparisons is the easiest interaction style with the DM, we are almost sure that making a choice on the tradeoff curve is not cumbersome. Also, the well-known drawback of binary comparisons, i.e., the fact that numerous interactions with the DM are needed, is overcome by visualizing the entire feasible region. Therefore, when all possible pairs of objective functions, together with the slope information, are presented to the DM, the task of locating the most preferred solution is reduced to a single interaction. For example, in Pareto Race [11], which is a visual interactive method, this tradeoff information between objectives is presented by supplying the value of every objective function simultaneously on the same plot. This implies that the DM has to compare multi dimensional vectors, moreover, since these are presented on the same plot, that visualization can be found difficult to follow.

Usually, in interactive methods, the DM is considered to be a special reliable device providing information about her/his unknown utility function, [15]. In reality, the DM should be regarded as an individual who has a limited capacity for handling and processing information. The negligence of psychological facts have led many interactive procedures to various drawbacks in real life applications. An analysis and discussion of the psychological aspects of MCDM can be found in Kahneman, et al [6]. These briefly indicate that, many information processing problems such as comparing multiple criteria, ordering, assigning criteria weights, etc., are too complex for a DM. In solving such problems, people usually make errors, display contradictions and employ auxiliary heuristics to simplify the problems. This behavior may be unnoticed due to the complexity of the methods themselves. Larichev [15] indicates that, "this behavior is considerably dependent on the deviation of the requirements on a DM within the framework of interactive procedure from the limits of his capabilities. This is not the problem of 'insignificant' errors, instead, the real problem is in the fact that beyond the capacity limit, people may stop using a considerable part of information and may give inconsistent answers".

5.1.1 A Scheme for Validation: Larichev's Study

Oleg I. Larichev's [15] systematic study aims at determining the cognitive burden on the part of the DM in an interactive process. He, therefore, defined elementary data processing operations made by a DM, and classified them according to their complexity:

(a) *Complex (C)*, if psychological research indicates that in performing these operations the DM is often inconsistent and/or makes use of simplifying strategies. Examples of such elementary operations are assignment of criteria weights, comparison of two vectors of criteria and identification of the better one, identification of variables that must be increased or decreased,

(b) *Complex, except for small problems (CS)*, if psychological research shows that the DM successfully performs these operations on small problems (2-3 criteria, 2-3 alternatives),

(c) *Admissible (A)*, if the research indicates that the DM can manage these elementary operations reliably, i.e. with a small number of inconsistencies, and using complex strategies. Examples are criteria ordering by importance, comparison of two criteria values variation, identification of criteria whose values must be improved, lowered or remain at least equal to the attained satisfactory level,

(d) *Uncertain (U, UC, UA)*, if an insufficient number of studies on these operations have been conducted but it is possible to judge about them by analogy. UA stands for uncertain admissible, and UC for uncertain complex. Example elementary operations are use of gradient methods (UC), and assignment of satisfactory value for a criterion (UA).

In [14], it was concluded that interactive procedures using as search for a pairwise tradeoff between criteria are more correct in terms of information elicitation from DMs.

We will evaluate our method's properties by using the elementary operations that VISTA implements with respect to the following criteria cited in Larichev's work,

1. **Admissibility.** If a method uses admissible (A) or supposedly admissible (UA) operations to elicit information from the DM, then it is considerably superior to an interactive method employing complex (C) or supposedly complex (UC) operations,

2. **Stability to Random Error.** Although in any data processing operation, DMs may commit random errors, a stable interactive method should not exclude the from consideration a large part of the solution domain containing the DM's best solution.

If an interactive method does so, then it is classified as being highly sensitive to random errors. However, if in a method, a random error by a DM just leads to a larger number of iterations, then such methods are hardly sensitive.

Methods which pass these two tests are referred to as correct.

According to the above scheme, our method uses two elementary operations to interact with the DM. One such operation is the requirement that the DM compares two criteria values which are varying against each other. The systematic studies cited in this paper have shown that with the number of criteria no more than eight, a DM can perform this operation quite reliably with a small number of contradictions. Given that the criteria are continuous, the answer from the DM are not sensitive to insignificant variations. Therefore, this elementary operation is evaluated to be A (Admissible), thus easy. The second elementary operation is the assignment of a satisfactory value for a criterion (constraining objective). According to many descriptive studies, the transfer of a criterion to a constraint, and the search for a satisfactory level is a typical human operation applied to different problems. Therefore, the DM should not have much problems in setting aspiration levels for the objective measures. This operation can be performed with a small number of contradictions and is found to be rather stable under the assumption that the considered criteria are important to the DM. As these results depend, however, on analogy with the known facts, it is classified as UA (Uncertain Admissible). Therefore, VISTA has the property of Admissibility.

With respect to the second criterion to evaluate interactive methods, however, our method lacks a formal experimental study to see on what extent it is sensitive to random errors made by the DM. However, one may conclude that since no scalarizing functions that implement some prespecified ideal values are used, the DM is free to choose any point in the solution space, thus the best point is not excluded from her/his reach. On the other hand, due to the MCDM problem structure, there may be cases where a random error excludes some regions from consideration.

We could therefore classify our method as a *correct* but with a reservation on stability. We thus conclude that assuming stability, *VISTA* is a structured interactive method.

5.2 Ease of Use

In this context, we evaluate the ease of use of the procedure in solving a decision making problem, and compare it with the Pareto Race.

VISTA is a computer aided decision support system which fully utilizes the merits of the computer technology. It is fast since solution times of the problems can be short, and especially for LP problems parametric optimization is quite easy to perform with any standard optimization package. Past iterations of the algorithm are stored in memory, and can be displayed when desired. Any amount of information can be made available with the use of *windows* and user friendly interfaces. The analyst is replaced by a computer program which provides an OPTIONS menu to choose from during the iterations of the algorithm. This menu incorporates various visual aids such as Convergence Plots, Single or Multiple Comparison plots, to facilitate the decision making process.

In Pareto Race, however, a systematized procedure for DM is not presented, and is left to the learning process of the DM, [12]. Pareto Race uses an achievement scalarazing function whose weights are computed by the range values of the objective functions specified by the DM. A solution, thus, is projected on the efficient frontier by using this function, and presented to the DM for evaluation. The DM is free to specify which objectives are to be treated as constraining objectives and which to be traded off. Thus, the DM may have to compare multidimensional vectors of objective function values according to her/his choices on the objectives. This comparison is significantly more complex than pairwise comparisons which is implemented in VISTA. Therefore, VISTA is superior to Pareto Race in terms of interaction style, and straightforwardness.

5.3 Information

VISTA, in essence, implements the simple yet very fruitful idea of tradeoff curve generation by varying the RHS of one objective function. This very specialty of VISTA is in fact its most powerful tool. The DM decides on whether to improve one objective at the expense of another only after inspecting this curve. This curve, which represents all feasible alternatives at that stage, together with the MRS, is very informative. Particularly, the leveling off of the tradeoff curve at specific regions, indicate that there is absolutely no conflict between the two objective functions under consideration. This piece of information gives an insight on the nature of the problem at hand, and is nonexistent in numerical algorithms such as [13], [26], [9]. The visualization of conflict and no conflict guarantees that the DM does not choose a point that is clearly inefficient, furthermore, teaches her/him that some objectives that do not conflict conceptually, indeed do on a numerical level. For example see the second example in Chapter 4.

In the same context, Nakayama [16] has developed a procedure called Interactive Relaxation Method (IRM), which sequentially optimizes each objective function given that other objectives remain at some satisficing values. They obtain the preferred solution by cyclically changing the pair of objectives on which tradeoff information is obtained. In the case where the number of objective functions is three or less, the efficient frontier can be visualized, however, when it is more than three, bisectioning methods are used to interact with the DM. Their method, in essence, implements the same principles as VISTA, however, IRM's main concern is working with efficient solutions only, and they fail to visualize problems with many objective functions. Therefore, although VISTA gives primary importance to visualization of the *approximation* of the efficient set (which becomes *exact* under the conditions given in the previous section), IRM prefers to supply obtained values to the DM, and according to her/his reaction updates them. That is, the DM cannot see all the possible efficient combinations of two objectives, instead s/he judges whether at some point her/his tradeoff value for these objectives is equal, larger or less than the marginal rate of substitution at that point. If a range where this tradeoff ratio switches from less than to larger than, then a mid-point is evaluated and presented to the DM. By doing so, the point where tradeoff ratio and the marginal rate of substitution are equal is seeked.

In this sense, IRM uses more interactions in each step than VISTA, since, once VISTA presents the DM the whole efficient curve between two objectives, the DM can select a point immediately. This selection on the tradeoff curve corresponds to choice of the point where the tradeoff ratio and the marginal rate of substitution between two objectives are equal. Therefore, in VISTA, burden is on the analyst's part whereas in IRM, burden is on the DM's part. Considering the improvement of the computer technology, we are pretty sure that the burden of the analyst would be handled by a computer package which would generate the efficient frontier in seconds, whereas asking the DM her/his preferences repeatedly would disturb her/him much, and might lead to erroneous answers. Therefore, information provided with one tradeoff curve by VISTA, replaces numerous interactions of the IRM procedure.

Chapter 6

Conclusion

Throughout this thesis, we have developed a method for solving Multiple Criteria Decision Making Problems by interacting with the Decision Maker (DM). We have presented the algorithm and additional features to facilitate the decision making process by decreasing the cognitive burden of the DM. Our method, which we called VISTA, accomplishes this by implementing an easy interaction style: Choice on a two dimensional tradeoff curve. This tradeoff curve is obtained by optimizing one objective function (main objective) and treating the remaining objectives as constraints whose right hand sides are set to their satisficing values, and parametrizing the RHS of one of these constraints (parametrized objective). The corresponding tradeoff curve would plot the values of main objective vs. parametrized objective. And under special circumstances, this curve is the projection of the efficient surface on this two dimensional space. Without any special assumptions, however, the tradeoff curve is a good approximation of the efficient surface, and at any time, by solving a secondary optimization problem, an efficient solution can be found. This tradeoff curve is presented to the DM, and the DM is asked to choose a compromise solution on this curve, and given that this solution determines the RHS of the parametrized constraint, next step is to parametrize the RHS of another constraining function. Special features, such as how to start and terminate the algorithm, how to supply the DM more information on the status of the problem have also been proposed in previous chapters.

We have, then, validated our model by an example problem and considering universal criteria established on decision making process. We have concluded that VISTA is a Structured Interactive Method which is not highly sensitive to random errors made by the DM. We have also supplied the theory that VISTA converges to the global optimum if the preference function assumes a special structure.

There are, however, open ended questions regarding the application of VISTA. A thorough experimental study is needed to be done with assumed preference functions replacing DMs. However, this type of study will lack a theoretical background on "satisficing behavior", and its effects on the preference function. Since the question asked to the DM is: *"given that every other objective function is guaranteed to perform at least some value, what is your choice considering the tradeoffs between two objectives?"*, the choice indicated on the tradeoff curve gives a tradeoff ratio between the two objectives without considering the achieved values of the other objective functions at that point. Note that, no assumption of preferential independence of the preference function has been done, therefore, these tradeoff ratios, in fact, cannot be used in deriving such a preference function, nor a preference function can be used in simulating a DM. Had we have the theory of satisficing behavior, we could test VISTA, and furthermore, use it to determine the coefficients of the preference function of the DM. Therefore, until such a theory has been developed, the only option of testing it is with real and concerned DMs on real problems. Random errors, as well as convergence to a solution have to be tested in the same manner. The effect of starting point and the ranking of the objectives on the obtained solution deserves more attention, and such a study with real DMs should not exclude these.

Although VISTA has shortcomings related to insufficient experimentation, it has the following advantages that cannot be neglected, it:

- Facilitates the decision making process by decreasing the cognitive burden on the DM,
- Implements elementary operations that are classified to be *easy* such as pairwise

comparisons, satisficing value assignment, ranking of the objectives

- Visualizes the solution space and exact tradeoffs among objectives
- Provides other aids such as slope information, convergence plot, hybrid approach, multiple comparisons
- Is flexible meaning that the DM is free to change main and parametrized objectives any time during the process
- Enables the DM to learn her/his problem, i.e. provides a learning instrument

We believe that, VISTA can find a large application area in the body of MCDM Methodologies since its emphasis is on simplifying the use of judgment, thus, decision making itself.

Bibliography

[1] P. Bogettoft and P. Pruzan. *Planning with Multiple Criteria.* North-Holland, Amsterdam, 1991.

[2] H Dogrusoz. Decision maker-analyst interactive decision making for military equipment with balanced cost and effectiveness. In *Proceedings of the 17th NATO Defence Research Group Seminar on The Balance between cost and effectiveness of military equipment*, 1975.

[3] Y. Y. Haimes, L. Lasdon, and D. A. Wismer. On a Bicriterion formulation of the problems of integreted system identification and system optimization. *IEEE Trans., Sys., Man, Cyber.*, 1:296–297, July 1971.

[4] H. Hain. Untersuchungen zum Verfahren der Rekursiven Systemoptimierung von Dogrusoz am Beispiel Eines Flugzeugwaffensystems. Technical report, Hochshule der Bundeswehr Munchen, 1978.

[5] M. Henig. A generalized method of approximating the set of efficient points with respect to a convex cone. In J. Morse, editor, *Organizations: Multiple Agents with Multiple Criteria.* Springer-Verlag, 1980.

[6] D. Kahneman, P. Slovic, and A. Tversky. *Judgment Under Uncertainty: Heuristics and Biases.* Cambridge University, 1982.

[7] R. L. Keeney and H. Raiffa. *Decisions with Multiple Objectives: Preferences and Value Tradeoffs.* Wiley, New York, 1976.

[8] P. Korhonen. Multiple Criteria Decision Support: The State Of Research and Future Directions. *Computers Ops. Res.*, 19:549–551, 1992.

[9] P. Korhonen and J. Laakso. A Visual Interactive Method for Solving the Multiple Criteria Problem. *European Journal of Operational Research*, 24:277–287, 1986.

[10] P. Korhonen, H. Moskowitz, and J. Wallenius. Multiple Criteria Decision Support: A Review. *European Journal of Operational Research*, 63:361–375, 1992.

[11] P. Korhonen and J. Wallenius. A Pareto Race. *Naval Research Logistics*, 35:615–623, 1988.

[12] P. Korhonen and J. Wallenius. A Multiple Objective Linear Programming Decision Support System. *Decision Support Systems*, 6:243–251, 1990.

[13] P. Korhonen, J. Wallenius, and S. Zionts. A Computer Graphics-based Decision Support System for Multiple Objective Linear Programming. *European Journal of Operational Research*, 60:280–286, 1992.

[14] O. I. Larichev. Multicriteria problems with objective models. In *Proceedings of the International Conference on Multiobjective Problems of Mathematical Programming*, 1988.

[15] O. I. Larichev and A. D. Nikiforov. Analytical survey of procedures for solving multicriteria mathematical programming problems (mmpp). In *Proceedings of the Seventh International Conference on Multiple Criteria Decision Making: Toward Interactive and Intelligent Decision Support Systems*, 1986.

[16] H. Nakayama, T. Tanino, and Y. Sawaragi. An Interactive Optimazition Method in Multicriteria Decisionmaking. *IEEE Trans. Syst., Man, and Cybern.*, SMC-10:163–169, March 1980.

[17] V. M. Ozernoi and M. G. Gaft. Multicriterion decision problems. In D. E. Bell, R. L. Keeney, and H. Raiffa, editors, *Conflicting Objectives in Decisions*. Wiley, 1978.

[18] W. S. Shin and A. Ravindran. Interactive Multiple Objective Optimization: Survey I-Continuous Case. *Computers Ops. Res.*, 18:97–114, 1991.

[19] H. A. Simon. *Theories of Bounded Rationality*. C. B. McGuire and R. Radmer(ed.), 1972.

[20] H. A. Simon. *From Substantive to Procedural Rationality*. J. S. Latsis(ed.), 1976.

[21] I. M. Sobol. Problems with multiple objectives and ill-posed problems. In *Proceedings of the International Conference on Multiobjective Problems of Mathematical Programming*, 1988.

[22] W. Stadler. Initiators of multicriteria optimization. In *Proceedings of the International Conference on Vector Optimization: Recent Advances and Historical Development of Vector Optimization*, 1986.

[23] M. T. Tabucanon. *Multiple Criteria Decision Making in Industry*. Elsevier, Amsterdam, 1988.

[24] A. Tversky. On the elicitation of preferences: Descriptive and prescriptive considerations. In D. E. Bell, R. L. Keeney, and H. Raiffa, editors, *Conflicting Objectives in Decisions.* Wiley, 1978.

[25] P. L. Yu. Cone Convexity, Cone Extreme Points, and Nondominated Solutions in Decision Problems with Multi-Objectives. *J. Optimization Theory Appl.*, 14:319–377, 1974.

[26] S. Zionts and J. Wallenius. An Interactive programming method for solving the multiple criteria problem. *Management Science*, 22:652–663, February 1976.