CHAPTER 101

Multicriteria Optimization

PO LUNG YU
University of Kansas
National Chiao Tung University

CHIN I. CHIANG
GWO HSHIUNG TZENG
National Chiao Tung University

1. INTRODUCTION

All animals have multiple goals to fulfill in their lives. They want to survive, to perpetuate their own species (including sex), to dominate, and so on. If you kick a dog, the dog will fight back or run away. Humans are no exception. We want to have a good life, which may mean more wealth, power, respect, and time for ourselves, together with good health and a successful next generation, and so on. We want food to taste, smell, and look good and be nutritious. Ever since Adam, human beings have been continuously confronted with multiple-criteria decision problems.

Although there have been abundant records of decision analyses based on multiple criteria in human history, putting the analyses in a formal mathematical setting is fairly new. Though still young, the literature of such mathematical treatments has exploded during the last three decades. The interested reader is referred to Stadler (1979, 1981), Steuer et al (1996) for surveys and historic notes.

Handbook of Industrial Engineering: Technology and Operations Management, Third Edition.
Edited by Gavriel Salvendy

In this chapter, we will report on six basic concepts of multicriteria optimization. Just as the three primary colors (red, yellow, and blue) can produce an infinite number of pictures and the seven basic notes of the musical scale (do, re, mi, etc.) can produce an infinite number of songs, the six basic concepts we will describe should allow the reader to generate an infinite number of models to solve the complex multiple-criteria decision problems. The six basic concepts are:

1. *Preferences as binary relations,* in which mathematical functional relations are generalized to represent revealed preferences and some basic solution concepts of MCDM (multiple-criteria decision making) will be introduced (Section 2).
2. *Value functions,* in which preferences are represented by powerful numerical ordering and multicriteria optimization problems are reduce to single-criterion optimization (Section 3).
3. *Satisficing, goal programming, and compromise solutions,* in which we model the preferences in terms of human goal-seeking behavior and introduce goal programming (which is well known). Again in this framework multicriteria problems are reduced to single-criterion ones (Section 4).
4. *Domination structures,* in which the preferences are represented in terms of multidimensional comparisons (in contrast to 2 and 3, which are of single-dimensional comparison) and solution concepts and methods to obtain the solutions are introduced (Section 5).
5. *MC²-simplex method for linear cases,* in which powerful computation method for linear cases are introduced (Section 6).
6. *Fuzzy multicriteria optimization,* is sketched in Section 7.

In Section 8, we offer further comments on multicriteria optimization.

2. PREFERENCES

Given any pair of decision outcomes y^1 and y^2, one and only one of the following can occur:

1. We are convinced that y^1 is better than or preferred to y^2, denoted by $y^1 \succ y^2$.
2. We are convinced that y^1 is worse than or less preferred to y^2, denoted by $y^1 \prec y^2$.
3. We are convinced that y^1 is equivalent to or equally preferred to y^2, denoted by $y^1 \sim y^2$; or
4. We have no sufficient evidence to say either 1, 2, or 3, denoted by $y^1 ? y^2$. Thus, the preference relation between y^1 and y^2 is indefinite or not yet clarified.

Note that each of the above statements involves a comparison or relation between a pair of outcomes. Let Y denote the set of all possible outcomes. Any revealed preference information, accumulated or not, can be represented by a subset of the Cartesian product $Y \times Y$, or by a so-called binary relation. We have the following definition.

Definition 2.1. A binary relation R on Y is a subset of the Cartesian product $Y \times Y$. Binary relation R is

1. *Reflexive* if $(y, y) \in R$ for all $y \in y$
2. *Symmetric* if $(y^1, y^2) \in R$ implies that $(y^2, y^1) \in R$ for all $y^1, y^2 \in y$
3. *Transitive* if $(y^1, y^2) \in R$ and $(y^2, y^3) \in R$ imply that $(y^1, y^3) \in R$, for all $y^1, y^2, y^3 \in Y$
4. *C*omplete if $(y^1, y^2) \in R$ or $(y^2, y^1) \in R$ for all $y^1, y^2 \in y$ and $y^1 \neq y^2$
5. An *equivalence* if R is reflexive, symmetric, and transitive

Definition 2.2.

1. A preference based on $\succ$ (respectively $\prec$, $\sim$, or ?) is a binary relation on Y, denoted by $\{\succ\}$ (respectively $\{\prec\}$, $\{\sim\}$, or $\{?\}$), so that whenever $(y^1, y^2) \in \{\succ\}$ (respectively $\{\prec\}$, $\{\sim\}$, or $\{?\}$) $y^1 \succ y^2$ (respectively $y^1 \prec y^2$, $y^1 \sim y^2$, or $y^1 ? y^2$).
2. For convenience, we also define $\{\succ \sim\} = \{\succ\} \cup \{\sim\}; \{\succ ?\} = \{\succ\} \cup \{?\}$, $\{\prec \sim ?\} = \{\prec \sim\} \cup \{?\}$, etc.

By a preference (or revealed preference) structure we mean the collection of all the above preferences. Because such structures are uniquely determined by $\{\succ\}$, $\{\sim\}$ and $\{?\}$, a preference structure will be denoted by $\mathcal{P}(\{\succ\}, \{\sim\}, \{?\})$, or simply by $\mathcal{P}$.

Remark 2.1. Note that the larger $\{?\}$ is, the less the revealed preference is clarified, which will usually cloud the decision process. Sometimes it may take discipline and/or new information for the decision maker to clarify his or her preference in order to reach the final solution. The concept of *pseudo-orders* (Roy and Vincke 1987) may help in resolving some of the problems.

Example 2.1 (*Pareto preference*). Let greater values be more preferred for each component y_i and assume that no other information on the preference is available or established. Then Pareto preference is defined by $y^1 > y^2$ if and only if $y_i^1 \geq y_i^2$ for all i and $y^1 \neq y^2$. Therefore,

$$\begin{aligned} \{>\} &= \{(y^1, y^2) | \; y_i^1 \geq y_i^2 \text{ for all } i \text{ and } y^1 \neq y^2\} \\ \{\sim\} &= \{(y, y) | y \in Y\} \\ \{?\} &= \{(y^1, y^2) | \text{there exist } i \text{ and } j \text{ such that } y_i^1 > y_i^2, \, y_j^1 < y_j^2\} \end{aligned}$$

Example 2.2 (*value function*). A value function v is a real-valued function which is defined on Y, the set of decision outcomes, such that $v(y^1) > v(y^2)$ if and only if y^1 is preferred to y^2 and $v(y^1) = v(y^2)$ if and only if y^1 is indifferent to y^2. It is easy to see that

$$\begin{aligned} \{>\} &= \{(y^1, y^2) \in Y \times Y | v(y^1) > v(y^2)\} \\ \{\sim\} &= \{(y^1, y^2) \in Y \times Y | v(y^1) = v(y^2)\} \\ \{?\} &= \emptyset \end{aligned}$$

Example 2.3 (*a lexicographic ordering*). Let $y = (y_1, y_2, \ldots, y_q)$ be indexed so that the kth component is overwhelmingly more important than the $(k + 1)$th component for $k = 1, 2, \ldots, q - 1$. A lexicographic ordering preference is defined as follows: the outcome $y^1 = (y_1^1, y_2^1, \ldots, y_q^1)$ is preferred to $y^2 = (y_1^2, y_2^2, \ldots, y_q^2)$ if and only if $y_1^1 > y_1^2$, or there is some k so that $y_k^1 > y_k^2$ and $y_j^1 = y_j^2$ for $j = 1, 2, \ldots, k - 1$. We see that

$$\begin{aligned} \{>\} &= \{(y^1, y^2) \in Y \times Y | y^1 \text{ is lexicographically preferred to } y^2\} \\ \{\sim\} &= \{(y, y) \in Y \times Y\} \\ \{?\} &= \emptyset \end{aligned}$$

Definition 2.3. Given a preference and a point $y^0 \in Y$, we define the *better, worse, equivalent,* and *indefinite* sets with respect to (*w.r.t.*) y^0 as:

1. $\{y^0 <\} = \{y \in Y | y^0 < y\}$ (the better set *w.r.t* y^0)
2. $\{y^0 >\} = \{y \in Y | y^0 > y\}$ (the worse set *w.r.t* y^0)
3. $\{y^0 \sim\} = \{y \in Y | y^0 \sim y\}$ (the equivalent set *w.r.t* y^0)
4. $\{y^0 ?\} = \{y \in Y | y^0 ? y\}$ (the indefinite set *w.r.t* y^0)
5. $\{y^0 > \sim\} = \{y^0 >\} \cup \{y^0 \sim\}$
6. $\{y^0 > ?\} = \{y^0 >\} \cup \{y^0 ?\}$
7. $\{y^0 > \sim ?\} = \{y^0 > \sim\} \cup \{y^0 ?\}$

Now we can define various solution concepts as follows.

Definition 2.4. Given a preference structure $\mathcal{P}$ $(\{>\}, \{\sim\}, \{?\})$ defined on the outcome space Y, we define:

1. $y^0 \in Y$ is an $N[<]$-solution (point) if and only if $\{y^0 <\} \cap Y = \emptyset$; the collection of all such solutions is denoted by $N[<]$.
2. $y^0 \in Y$ is an $N[<\sim]$-solution if and only if $\{y^0 < \sim\} \cap Y = \{y^0\}$; the collection of all such solutions is denoted by $N[<\sim]$.
3. $y^0 \in Y$ is an $N[<?]$-solution if and only if $\{y^0 ? <\} \cap Y = \emptyset$; the collection of all such solutions is denoted by $N[?]$.
4. $y^0 \in Y$ is an $N[<\sim ?]$-solution if and only if $\{y^0 < \sim ?\} \cap Y = \{y^0\}$; the collection of all such solutions is denoted by $N[<\sim ?]$.

The following can be easily established (Chien 1987; Chien et al. 1990):
Theorem 2.1.

1. $N[<\sim?] \subseteq N[<?] \subseteq N[<]$
2. $N[<\sim?] \subseteq N[<\sim] \subseteq N[<]$
3. $N[<\sim?]$ contains at most one point

Remark 2.2. In application, we may first try to locate $N[<]$, then $N[<\,?]$ or $N[<\,\sim]$, and finally $N[<\,\sim?]$, which, if nonempty, will contain only one clear superior solution, and no alternative optimals would occur.

3. VALUE FUNCTIONS

3.1. Revealed Preferences and Value Functions

Numerical systems have a prevailing impact on our culture and thinking. It is natural and important for us to ask, Is it possible to express our preference over the outcomes in terms of numbers so that the larger the number the stronger the preference, and if it is, how do we do it?

Suppose that our preference structure $\mathcal{P}$ can be represented by a value function $v{:}Y \to R$ so that $v(y^1) > v(y^2)$ if and only if y^1 is preferred to y^2 and $v(y^1) = v(y^2)$ if and only if y^1 is indifferent to y^2 and that Y is convex and v is continuous on Y. We immediately can obtain the following properties for $\mathcal{P}$ (see Example 2.2).

1. $\{?\} = \varnothing$
2. $\{>\}$ and $\{>\,\sim\}$ are transitive.
3. Let $\mathcal{G}$ be the collection of all isovalued curves (or surfaces) of v in Y. Note that different isovalued curves never intersect. Since Y is convex and v is continuous, $V = v[Y]$ is an interval which contains dense countable rational numbers. Their corresponding isovalued curves are thus countable and dense in $\mathcal{G}$.
4. $\{y\,<\}$ and $\{y\,>\}$ are open sets in Y for every $y \in Y$, if Y is open.

We summarize some important results on value functions in the following remark.

Remark 3.1. (a) Property 1 of the above is essential. If $\{?\} \neq \varnothing$, then value function representation cannot offer meaningful numerical ordering; (b) Properties 1 and 2 together mean that $\mathcal{P}$ is a weak order in literature; (c) Properties 1–3 conversely ensure the existence of value function representation; (d) Properties 1–2 and 4 conversely ensure the existence of *continuous* value function representation. The reader is referred to Fishburn (1970), Debreu (1954, 1960), Keeney and Raiffa (1976), Yu (1985), Gorman (1968), Yu and Takeda (1987), and others for further detailed discussion on the existence condition of value functions and their forms.

3.2. Methods of Constructing Value Functions

Assuming that the existence conditions are satisfied, we are interested in methods of constructing value functions to approximate the revealed preferences. Let us first roughly classify the existing methods before we illustrate some popular methods.

One large class of techniques for constructing value functions is a direct application of calculus. These methods are usually based on the construction of approximate indifference curves. This class includes methods using trade-off ratios, tangent planes, gradients, and line integrals. Some of these methods are discussed in Yu (1985).

A second class has specifically been developed for the case of additive value functions. One of the well-known methods in this class is the midvalue method (Keeney and Raiffa 1976; Yu 1985), which is based on the pairwise information of $\{\sim\}$ and $\{>\}$.

A third class takes into account the fact that usually the revealed preference contains conflicting information, making it a virtually impossible task to construct a consistent value function. The conflicting nature of the information may be due to an unclear perception on the part of the DM of his true preference structure or to imperfect and/or incorrectly used interaction techniques. The objective of these methods is to find a value function and/or ideal point that minimizes the inconsistencies. Some of the techniques, such as regression analysis, are based on statistical theory, others on mathematical programming models, such as least distance and minimal inconsistency methods (using some appropriate l_p-norm). Included are methods based on weight ratios, pairwise preference information, or the distance from a (perhaps unknown) idea/target point (see Yu 1985, Section 6.3.4 and the citations therein). Another group of methods in this class is that of eigenweight vectors (see, e.g., Saaty 1980; Cogger and Yu 1985; Yu 1985, Section 6.3.3.). Yet another group uses holistic assessment utilizing the orthogonal design of statistical experiment (see Yu 1985, Section 6.3.3.2 and citations therein).

Each method has its strengths and weaknesses. The selection of the best/correct method is truly an art and poses a challenge for the analyst and decision maker. Assume that there are q criteria under consideration and that more is better for each y_k ($k = 1, 2, \ldots, q$) with y_k taking values in the interval $[a_k, b_k]$. Thus, the outcome space is the rectangle $[a_1, b_1] \times [a_2, b_2] \times \ldots \times [a_q, b_q]$. Assume that the value function is additive as follows:

$$v(y) = w_1 v_1(y_1) + w_2 v_2(y_2) + \ldots + w_q v_q(y_q)$$

where $v_k(y_k)$, $k = 1, 2, \ldots, q$ are the individual components and $(w_1, w_2, \ldots, w_q)$ is the weight distribution. Without loss of generality, we can assume that $v_k(a_k) = 0$ and $v_k(b_k) = 1$. To assess individual $v_k(y_k)$, there are several methods. For details, see Yu (1985, chap. 5).

Assume that we have successfully generated the individual function v_k ($k = 1, 2, \ldots, q$). We want to determine the weights $w_1, w_2, \ldots, w_q$ so that we can obtain the overall value function

$$v(y) = w_1 v_1(y_1) + w_2 v_2(y_2) + \ldots + w_q v_q(y_q)$$

by aggregating the individual components. There are many ways to assess the weights (see, e.g., Hwang and Masu 1979; Hwang and Yoon 1981; Yu 1973). Being limited by space, we shall discuss only two popular methods for different situations.

Method 1. Recall that $v_k(a_k) = v_k(y_k^0) = 0$ and $v_k(b_k) = v_k(y_k^1) = 1$. Set $y^0 = (y_1^0, \ldots, y_q^0)$ and $y^1 = (y_1^1, \ldots, y_q^1)$. Denote

$$(y_k^1, y_{\hat{k}}^0) = (y_1^0, \ldots, y_{k-1}^0, y_k^1, y_{k+1}^0, \ldots, y_q^0)$$

then

$$v((y_k^1, y_{\hat{k}}^0)) = w_k v_k(y_k^1) = w_k$$

This simply means that we need only to know the value of the value function at point $(y_k^1, y_{\hat{k}}^0)$ to know w_k. If this is difficult to accomplish, we can alternatively identify $q - 1$ pairs of indifferent points, then, using the additive form of the value function, we can obtain $q - 1$ equations for the q unknowns $(w_1, w_2, \ldots, w_q)$. After normalizing, we obtain a unique solution.

Example 3.1. Assume that there are three criteria under consideration and that $Y_1 = Y_2 = Y_3 = [0, 2]$. Suppose that we have obtained the individual functions as

$$\begin{aligned} v_1(y_1) &= (5/6)y_1 - (1/6)\, y_1^2 \\ v_2(y_2) &= y_2 - (1/4)\, y_2^2 \\ v_3(y_3) &= (3/4)\, y_3 - (1/8)\, y_3^2 \\ (1, 2, 0) &\sim (1, 0, 2) \\ (1, 0, 1) &\sim (0, 2, 0) \end{aligned}$$

Using the additive form of the value function, we obtain

$$\begin{aligned} v((1, 2, 0)) &= w_1 v_1(1) + w_2 v_2(2) + w_3 v_3(0) = (2/3)w_1 + w_2 \\ v((1, 0, 2) &= w_1 v_1(1) + w_2 v_2(0) + w_3 v_3(2) = (2/3)w_1 + w_3 \end{aligned}$$

Therefore, $(1, 2, 0) \sim (1, 0, 2)$ implies that $w_2 = w_3$. Similarly, from $(1, 0, 1) \sim (0, 2, 0)$, we have $(2/3)w_1 + (5/8)w_3 = w_2$. Combined with $w_1 + w_2 + w_3 = 1$, we obtain the weights

$$w_1 = \frac{9}{41},\ w_2 = \frac{16}{41},\ w_3 = \frac{16}{41}$$

Therefore, the overall value function is given by

$$v(y) = \left(\frac{15}{82}\right) y_1 - \left(\frac{3}{82}\right) y_1^2 + \left(\frac{16}{41}\right) y_2 - \left(\frac{4}{41}\right) y_2^2 + \left(\frac{12}{41}\right) y_3 - \left(\frac{2}{41}\right) y_3^2$$

Method 2. Saaty's Eigenweight Vector Method (Saaty 1980). Suppose that the value function is of the form

$$v(y) = w_1y_1 + w_2y_2 + \ldots + w_qy_q$$

Thus, the value function is determined if the weights are known. Without losing generality, we can assume that $w_k > 0$ ($k = 1, 2, \ldots, q$).

If weights are given, we can define the weight ratio by

$$w_{ij} = \frac{w_i}{w_j}$$

Then the weight ratio matrix $W = [w_{ij}]_{q\times q}$ is *consistent* in the sense that for any i, j, and k,

$$w_{ij} = w_{ji}^{-1}$$
$$w_{ik} = w_{ij}w_{jk}$$

On the other hand, given a consistent matrix, we can find the weight vector that generates it.

Theorem 3.1. Let $w_{q\times q}$ be any consistent matrix. Then

1. The maximum eigenvalue for W is q and all the other eigenvalues are 0.
2. The eigenvector corresponding to the maximum eigenvalue, which is unique after normalizing, is the weight vector $w = (w_1, w_2, \ldots, w_q)$, which generates W by $w_{ij} = w_i/w_j$.

Therefore, to find the weights $w_1, w_2, \ldots, w_q$, we need only to find the weight ratios w_{ij} ($i, j = 1, 2, \ldots, q$). Saaty proposed the following procedure:

Step 1: For $i < j$, estimate or elicit the weight ratio w_{ij}, by a_{ij}. Let $a_{ii} = 1$ for all i and $a_{ij} = a_{ij}^{-1}$ for $i > j$. Denote $A = [a_{ij}]_{q\times q}$.

Step 2: Since A is found as an approximation for W, when the consistency conditions are "almost" satisfied for A, one would expect that the normalized eigenvector corresponding to the maximum eigenvalue of A, denoted by $\lambda_{\max}$, will be close to w. Thus, w can be approximated by the normalized eigenvector corresponding to $\lambda_{\max}$.

Here the problem is, however, that the estimated weight ratios are generally not consistent, especially when criteria are not few, since human perception and judgment are subject to change when the information input or psychological state of the decision maker change. The using of the above procedure is partially justified by the following theorem.

Theorem 3.2 (Saaty 1980).

1. Let $\overline{w} = (\overline{w_1}, \overline{w_2}, \ldots, \overline{w_q})$ be the normalized eigenvector corresponding to $\lambda_{\max}$ of A. Then $\overline{w_i} > 0$ for all i.
2. Given i and k, if $a_{ij} \geq a_{kj}$ for all j, then $\overline{w_i} \geq \overline{w_k}$.

Example 3.2 (Saaty 1980). Three criteria are under consideration. Assume that the following estimations of weight ratios are obtained:

$$a_{12} = 9,\ a_{13} = 7,\ a_{23} = 1/5$$

We construct the following matrix:

$$A = \begin{bmatrix} 1 & 9 & 7 \\ 1/9 & 1 & 1/5 \\ 1/7 & 5 & 1 \end{bmatrix}$$

To find $\lambda_{\max}$, we solve det$[A - \lambda\ \mathrm{I}] = 0$, or $(1 - \lambda)^3 - (1 - \lambda) + 9/35 + 35/9 = 0$. The maximum solution is $\lambda_{\max} \approx 3.21$. Then we find the corresponding normalized eigenweight vector $\overline{w}$ with

$$\overline{w_1} = 0.77,\ \overline{w_2} = 0.05,\ \overline{w_3} = 0.17$$

The value function is therefore given by

$$v(y) = 0.77y_1 + 0.05y_2 + 0.17\ y_3$$

Remark 3.2. In Saaty's method, one needs to solve a nonlinear equation to find the maximum eigenvalue. Cogger and Yu (1985) proposed another eigenweight vector method that can be applied to find the eigenweight vector easily by using the deductive formula. The weights determined by the latter method are different from those determined by the former. It is important to note that there is no need to have all a_{ij}, $i < j$, in order to apply the above method. For such exploration, the reader is referred to Takeda and Yu (1995).

Many other methods have been developed that we cannot discuss here due to limited space. The interested reader is referred to Yu (1985), Yu and Takeda (1987, 1995), Hwang and Masud (1979), Hwang and Yoon (1981), and Yu (1973) and quotes therein. It should also be stressed that all these methods are approximation methods. When inconsistencies occur, we may wish to represent the decision maker's preference in ways other than value functions. This will be discussed shortly.

4. SATISFICING AND COMPROMISE SOLUTIONS

Each human being has a set of ideal goals to achieve and maintain (see Yu 1990, 1995, and 1985, chap. 9 and citations therein). When a perceived state (or value) has unfavorably deviated from its targeted or ideal state (or value), a charge (tension or pressure) will be produced to prompt an action to reduce or eliminate the deviation. The behavior of taking actions, including adjustment of the ideal values to move the perceived states to the targeted ideal states is called *goal-seeking behavior*.

This goal-seeking behavior has an important and pervasive impact on human decision making. In this section, we shall focus on two concepts that model human goal-seeking behavior. These are satisficing goal programming and compromise models.

4.1. Satisficing Models

Example 4.1. Consider the problem of selecting an athlete for a basketball team. Assume that quickness, f_1, and accuracy of shooting, f_2, of the player are the main concerns. Let both f_1 and f_2 be indexed from 0 to 10, where the higher indices, the better the player. Assume that a player, x, will be selected with satisfaction if $f_1(x) \geq 9$ or $f_2(x) \geq 9$, or $f_1(x) + f_2(x) \geq 15$. Let $f = f(x) = (f_1(x), f_2(x))$ be the score vector of x and $G_1(f) = f_1 - 9$, $G_2(f) = f_2 - 9$, and $G_3(f) = f_1 + f_2 - 15$. If an athlete's score vector is in at least one of the sets , $S_i = \{f | G_i(f) \geq 0\}$, $i = 1, 2, 3$, then he or she is to be selected with satisfaction. Note that these sets are defined in terms of the scores (outcomes) instead of the candidates (decision alternatives). Here S_i, $i = 1, 2, 3$, so defined, are called satisficing sets.

Note that the specifying of lower bounds 9, 9, and 15 in the above example is an act of *goal setting*. Goal setting for satisficing models is defined as the procedure of identifying a satisficing set S such that whenever the decision outcome is an element of S, the decision maker will be happy and satisfied and is assumed to have reached the optimal solution. When S contains only a single point, the point is called the goal or target point.

Let our MCDM problem be specified by the criteria $f = (f_1, f_2, \ldots, f_q)$ and the feasible set is $X = \{x \in R^n | g(k) \leq 0\}$, where $g = (g_1, g_2, \ldots, g_m)$. Note that the outcome of a decision x is specified by $y(x) = f(x)$.

To specify S, one can start with each individual f_i and find its satisficing or acceptable intervals. One can then consider two or more criteria simultaneously for their corresponding trade-off. In a general form, in the final satisficing set S can be defined as the union of

$$S_k = \{f | G_k(f) \geq 0\},\ k = 1, 2, \ldots, r$$

where G_k is a vector function that reflects the trade-off over f. Note that when the union contains a single set (i.e., $r = 1$), S is defined by a system of inequalities.

Once S, the satisficing set, has been determined, finding a satisficing solution x^0, such that $f(x^0) \in S$, is a mathematical programming problem.

Since $S = \cup \{S_k | k = 1, 2, \ldots, r\}$, $Y \cap S \neq \varnothing$ if and only if there is an S_k such that $Y \cap S_k \neq \varnothing$. Because we need only one point of $Y \cap S$ when it is nonempty, we can verify individually if $Y \cap S_k \neq \varnothing$, $k = 1, 2, \ldots, r$, and find one point from the nonempty intersections.

To verify whether or not $Y \cap S_k \neq \varnothing$ is empty and to find a point in $Y \cap S_k$, if it is nonempty, we can use the following mathematical program. First rewrite each satisficing set as

$$S_k = \{f(x) | G_{kj}(f(x)) \geq 0, j = 1, 2, \ldots, J_k\}$$

where G_{kj}, $j = 1, 2, \ldots, J_k$, are components of G_k, and J_k is the number of components of G_k.

Program 4.1. $V_k = \min d_1 + d_2 + \ldots + d_{J_k}$

$$\text{s.t. } G_{kj}(f(x)) + d_j \geq 0, j = 1, 2, \ldots, J_k$$

$$x \in X, d_j \geq 0, j = 1, 2, \ldots, J_k$$

It is readily verified that $Y \cap S_k \neq \varnothing$ if and only if $V_k = 0$. We thus have:

Theorem 4.1. Let $S = \cup [S_k | k = 1, 2, \ldots, r\}$. Then a satisficing solution exists if and only if there is at least one $k \in \{1, 2, \ldots, r\}$ such that Program 4.1 yields $V_k = 0$.

Observe that when one satisficing solution is found, the decision problem is solved, at least temporarily. Otherwise the decision maker can activate either positive problem solving or negative problem avoidance to restructure the problem. The former will enable careful restudy and restructuring of the problem so as to find a solution, perhaps a new one, that lies in the satisficing set. The latter will try to reduce the aspiration levels or play down the importance of making a good decision, thus, lowering the satisficing set to have a nonempty intersection with Y. While the psychological attitude in problem solving and in problem avoidance may be different (see Yu 1990, 1995, 1985, chap. 9 for further discussion), the consequences are the same. That is, eventually the newly structured problem enables the decision maker to have $Y \cap S \neq \varnothing$. The following provides some helpful information for decision maker for restructing S and Y.

1. Depending on the formation of S, if Program 4.1 is used to identify a satisficing solution, one can produce x^k, $k = 1, 2, \ldots, r$, which solves Program 4.1 with S_k as the satisficing set. One then can compute $f(x^k)$, $G_k(f(x^k))$, the distance between $f(x^k)$ and S_k, and the trade-off among the $G_{kj}(f(x))$, $j = 1, 2, \ldots, J_k$, at x^k. All of these can be helpful for the decision maker to reframe S_k.
2. Suppose that it has been revealed or established that more is better for each criterion and that the revealed preference contains at least the Pareto preference. One can then generate (a) all or some representatives of the $N[<]$ points, the corresponding $f(x)$, and optimal weights $\Lambda(x)$ for the $N[<]$ points (see Sections 5 and 6 for further discussion); (b) the value $f_k^* = \max \{f_k(x) | x \in X\} (k = 1, 2, \ldots, q)$, which gives the best value of $f(x)$ over X. This information can help the decision maker to restructure his or her S. Relaxing of the target goal with reference to the ideal point $(f_1^*, f_2^*, \ldots, f_g^*)$ sequentially and interactively with the decision maker, until a satisficing solution is obtained, is certainly an art. It is especially useful when X and f are fairly fixed and not subject to change.
3. If possible (for instance, in bicriteria cases), displaying Y and S, even if only partially, can help the decision maker conceptualize where Y and S are, to make it easier to restructure the problem.

One must keep in mind that with X, $f(x)$, and S, one can generate as much information as one wishes, just as one can generate as many statistics from sample values as one wants. However, relevant information (e.g., useful statistics) that can positively affect the problem solving may not be too much. Irrelevant information can become a burden for decision analysis and decision making.

Example 4.2. Consider a sample production problem. Let the resource constraints in the amount produced, x_1 and x_2, be given as follows:

$$g_1(x) = 3x_1 + x_2 - 12 \leq 0$$
$$g_2(x) = 2x_1 + x_2 - 9 \leq 0$$
$$g_3(x) = x_1 + 2x_2 - 12 \leq 0$$
$$x_1 \geq 0, x_2 \geq 0$$

The following two objective functions are related to gross output and net profit respectively:

$$y_1 = f_1(x) = x_1 + x_2$$
$$y_2 = f_2(x) = 10x_1 - x_1^2 + 4x_2 - x_2^2$$

Assume that the decision maker specifies his or her satisficing set as

$$S = \{(y_1, y_2) | y_1 \geq 10, y_2 \geq 30\}$$

Then, using Program 4.1, we solve the problem of

$$
\begin{aligned}
V = {} & \min d_1 + d_2 \\
\text{s.t.} \quad & x_1 + x_2 + d_1 \geq 10 \\
& 10x_1 - x_1^2 + 4x_2 - x_2^2 + d_2 \geq 30 \\
& 3x_1 + x_2 \leq 12 \\
& 2x_1 + x_2 \leq 9 \\
& x_1 + 2x_2 \leq 12 \\
& x_1 \geq 0,\ x_2 \geq 0
\end{aligned}
$$

It turns out that $V > 0$, thus $S \cap Y = \emptyset$. To find a satisficing solution, we must restructure X, f, and/or S. If X and f are fairly fixed, we may offer the information that $f_1^* = 7$ and $f_2^* = 26.5$, which simply means that any outcome y with $y_1 > 7$ or $y_2 > 26.5$ is unobtainable. Thus, the goals $y_1 > 10$ and $y_2 > 30$ must come down to allow a satisficing solution. The restructuring of S will continue until a satisficing solution is found.

4.2. Compromise and Goal Programming Solutions

Assume that each criterion $f_i (i = 1, 2, \ldots, q)$ is characterized by "more is better." Let $y^* = (y_1^*, y_2^*, \ldots, y_q^*)$ where $y_i^* = \sup\{f_i(x) | x \in X\}$ with X the feasible set. The point y^* is called *ideal* (or *utopia*) *point* because it is usually unattainable even though y_i^* may individually be attainable.

Example 4.3. Consider the following maximization problem with two criteria:

$$
\begin{aligned}
\max \quad & y_1 = f_1(x) = 6x_1 + 4x_2 \\
\max \quad & y_2 = f_2(x) = x_1 \\
\text{s.t.} \quad & g_1(x) = x_1 + x_2 \leq 100 \\
& g_2(x) = 2x_1 + x_2 \leq 150 \\
& x_1, x_2 \geq 0
\end{aligned}
$$

To find y_1^*, we solve

$$
\begin{aligned}
& \max y_1 = f_1(x) = 6x_1 + 4x_2 \\
& \text{s.t. } x_1 + x_2 \leq 100 \\
& \quad 2x_1 + x_2 \leq 150 \\
& \quad x_1, x_2 \geq 0
\end{aligned}
$$

The solution is $x_1^* = x_2^* = 50$. Thus, $y_1^* = f_1(x^*) = 500$. Similarly, we find $y_2^* = 75$. Therefore, the ideal point is $y^* = (500, 75)$. Note that y^* is not attainable.

In group decision problems, if each criterion represents a player's payoff, then y^*, if attainable, would make each player happy because it would simultaneously maximize each player's payoff. Even if one is a dictator, he cannot do better than y^* for himself. As y^* is usually not attainable, a compromise is needed if no other alternative is available to dissolve the group conflict. This offers a natural explanation of why the solution to be introduced is called a *compromise solution* (Yu 1973). The reader can extend this explanation easily to multiple-criteria problems.

Now given $y \in Y$, the outcome space, the *regret* of using y instead of obtaining the *ideal* point y^* may be approximated by the distance between y and y^*. Thus, we define the (group) regret of using y by

$$r(y) = \|y - y^*\|$$

where $\|y - y^*\|$ is the distance between y and y^* according to some specific norm. Typically, the l_p-norm will be used in our discussion because it is easy to understand, unless otherwise specified. To make this more specific, define for $p \geq 1$,

$$r(y; p) = \|y - y^*\|_p = \left[\sum |y_i - y_i^*|^p \right]^{1/p}$$

and

$$r(y; \infty) = \max \{|y_i - y_i^*|, i = 1, 2, \ldots, q\}$$

Then $r(y; p)$ is a measurement of regret from y to y^* according to the l_p-norm.

Definition 4.1. The compromise solution with respect to the l_p-norm is $y^p \in Y$, which minimizes $r(y; p)$ over Y, or is $x^p \in X$, which minimizes $r(f(x); p)$ over X. When the ideal point y^* is replaced by a specific goal or target point, the resulting compromise solution is called the goal programming solution with respect to the goal point.

Remark 4.1. In group decision problems, $r(y; p)$ may be interpreted as group regret and the compromise solution y^p, is the one which minimizes the group regret in order to maintain a cooperative group spirit. As the parameter p varies, the solution y^p can change.

Note that $r(y; p)$ treats each $|y_i^* - y_i|$ as having the same importance in forming the group regret and the multiple criteria problems. If the criteria have different degrees of importance, then a weight vector may be assigned to signal the different degrees of importance. The regret function $r(y; p)$ can be modified in a natural way (Yu 1985). Observe that using weights in the regret function has the effect of changing the scale of each criterion. Conversely, the scale can be adjusted so that the regret function is reduced to that of equal weight. Thus, in studying the properties of compromise solutions, without loss of generality, one may focus on the equal weight case. We shall assume the equal weight case from now on, unless specified otherwise.

Remark 4.2. Observe that the compromise solution is not scale *independent*. Scale independence, an important criterion in group decision problems, can prevent players from artificially changing the scale to obtain a better arbitration for themselves. Note that when the scale of $f_i(x)$ or y_i is changed and the weight of importance of each criterion is changed, so is the compromise solution.

Remark 4.3. Compromise solutions with proper assumptions enjoy a number of properties, such as feasibility, least group regret, no dictatorship, Pareto optimality, uniqueness, symmetry, independence of irrelevant alternatives, continuity, monotonicity, and boundedness. In terms of parameter p (associated with p-norms), we may say that when $p = 1$, the sum of the group utility is mort emphasized, and when $p = \infty$, the individual regret of the group is mort emphasized. For the details and further exploration, see Yu (1985, 1973) and Freimer and Yu (1976).

4.3. Computing Compromise Solutions and Goal Programming Solutions

Now we turn to the computation of compromise solutions. We first consider the case when the target point is the ideal point, then the general case.

4.3.1. The Ideal Point as the Target Point

To find a compromise solution when the ideal point is the target point, we have to solve $q + 1$ mathematical programming problems. The first q problems are to find the utopia or ideal point where $y^* = (y_1^*, y_2^*, \ldots, y_q^*)$, where $y_i^* = \sup\{f_i(x)|x \in X\}$. The last one is to find the compromise solution y^p. Suppose that X contains only countable points. We will have $q + 1$ integer programming problems. Otherwise, if X is a region, we will have $q + 1$ nonlinear programming problems.

If X and $f_i(x)$, $i = 1, 2, \ldots, q$ have some special structure, then more efficient computational techniques are available. For instance, if X is a convex set and each $f_i(x)$ is concave, then we first have q concave programming and then a convex programming (because $r(f(x); p)$ is convex under the assumptions). If X is a polyhedron defined by a system of linear inequalities and each $f_i(x)$ is linear, then the ideal points y^* can be found by q simple linear programming problems. Furthermore, the compromise solutions of y^1 and y^∞ can be found by a linear programming problem (the other compromise solutions, y^p, $1 < p < \infty$, can be found by convex programming) (Yu 1985).

Example 4.4. Reconsider Example 4.3. The ideal point is (500, 75). Now we show how to find compromise solutions y^1, y^2, and y^∞.

1. To find the y^1 compromise solution, we solve

$$\begin{aligned} \min \quad & \{[500 - (6x_1 + 4x_2)] + [75 - x_1]\} \\ \text{s.t.} \quad & x_1 + x_2 \leq 100 \\ & 2x_1 + x_2 \leq 150 \\ & x_1, x_2 \geq 0 \end{aligned}$$

We obtain $x_1^* = 50$, $x_2^* = 50$ and the compromise solution is then $y^1 = (500, 50)$.

2. Similar to (l), we obtain compromise solution $y^2 = (490, 55)$.

3. For the y^∞ compromise solution, we solve

$$\begin{aligned} \min\max \quad & \{500 - (6x_1 + 4x_2), 75 - x_1\} \\ \text{s.t.} \quad & x_1 + x_2 \leq 100 \\ & 2x_1 + x_2 \leq 150 \\ & x_1, x_2 \geq 0 \end{aligned}$$

This problem can be simplified as

$$\begin{aligned} \min \quad & v \\ \text{s.t.} \quad & v \geq 500 - (6x_1 + 4x_2) \\ & v \geq 75 - x_1 \\ & x_1 + x_2 \leq 100 \\ & 2x_1 + x_2 \leq 150 \\ & x_1, x_2 \geq 0 \end{aligned}$$

Note that this is a linear program. We can obtain the solution easily as $x_1^* = 58.33$, $x_2^* = 33.33$, and $y^\infty = (483.33, 58.33)$.

4.3.2. General Target Points and Goal Programming

The computation for the compromise solution with general target point can be slightly more complicated. This is due to the possibility that $y_i^* \geq y_i$ for all $y \in Y$ and $i = 1, 2, \ldots, q$ no longer hold. In this case, we need some transformation of variables.

Let d_i^+ and d_i^- be respectively the positive and negative pairs of $y_i - y_i^*$, i.e., $d_i^+ = y_i - y_i^*$ if $y_i > y_i^*$ and 0 if otherwise, and $d_i^- = y_i^* - y_i$ if $y_i \leq y_i^*$ and 0 if otherwise. Now given a target point y^*, the compromise solution with l_p-norm can be found by solving the following (Yu 1985):

Program 4.2.

$$\begin{aligned} \min \quad & \sum (d_i^+ + d_i^-)^p \\ \text{s.t.} \quad & y_i^* - f_i(x) = d_i^- - d_i^+, \; i = 1, 2, \ldots, q \\ & d_i^+, d_i^- \geq 0, \; i = 1, 2, \ldots, q \\ & g(x) \leq 0 \end{aligned}$$

Remark 4.4.

1. Suppose that all $f_i(x)$, $i = 1, 2, \ldots, q$, and $g(x)$ are linear. Then Program 4.2 for l_l-compromise solutions reduces to a linear program typically known as (linear) *goal programming*. By adding weights to or imposing lexicographical ordering on the criteria, one can generalize the concept discussed here to a variety of goal programming formats. Goal programming and compromise solutions are closely related. The major advantages of goal programming and l_l-compromise solutions are that they are easily understood and can be easily computed by linear programs.

2. When all $f_i(x)$, $i = 1, 2, \ldots, q$, and $g(x)$ are linear, Program 4.2 for l_2-compromise solutions becomes a quadratic program. The program can be solved without major difficulty.

3. When all $f_i(x)$, $i = 1, 2, \ldots, q$, and $g(x)$ are linear, Program 4.2 for l_∞-compromise solutions becomes a linear program.

4.3.3. Interactive Methods of Compromise and Goal Programming Models

For nontrivial decision problems, an optimal decision usually cannot be reached by applying the compromise solution only once. The following interactive method, according to Figure 1, may be helpful.

Box (*1*): Identifying X, f, and y^* is an art that requires careful observation and conversation with the decision maker. The target point y^*, weight vector w, and p for l_p-norm may be more difficult to specify precisely. However, one can use the ideal point as the first step of approximation for y^*, select some representative weights for w, and let $p = 1, 2$, or ∞ to begin.

Box (*2*): One can locate compromise solutions according to Program 4.2 for the specified values of y^*, w, and p.

Box (*3*): All relevant information obtained in Box (1) and (2) can be presented to the decision maker. These include X, f, Y, ideal points y_i^*, $i = 1, 2, \ldots, q$, the optimal vector value that maximizes the ith criterion, and the compromise solutions y^1, y^2, and y^* with their weights and norms.

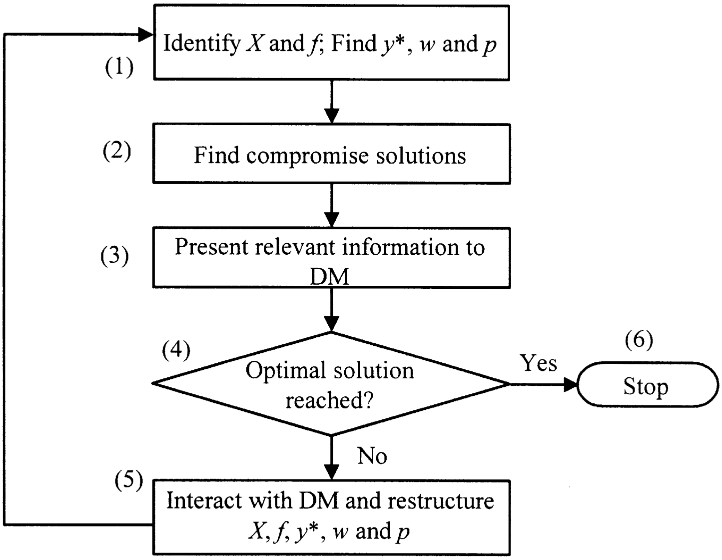

Figure 1 An Interactive Method for Compromise Solutions.

Boxes (4) and (6): If an optimal solution is reached in Box (3) and the decision maker is satisfied, the process is terminated at Box (6); otherwise, we go to Box (5) to obtain more information.

Box (5): Through conversation, we may obtain new information on X, f, y^*, weight vector w, and norm parameter p. Note that each of these five elements may change with time. To help the decision maker locate or change the target point, the "one-at-a-time method may be used. That is, only one value of f_i, or y_i, $i = 1, 2, \ldots, q$, is to be changed and the rest remain unchanged. One may ask the decision maker, "Since no satisfactory solution is obtained with the current target point y^*, would it be possible to decrease (or increase) the value of y_i^*? And, if so, by how much?" This kind of suggestive question may help the decision maker to think carefully of possible changes in the target points. The other method is called the pairwise trade-off method. We first select two criteria, say f_i and f_j, and then ask: "To maintain the same degree of satisfaction for the target point everything else being equal, how many units of f_j must be increased in order to compensate a one-unit decrease in f_i?" Again, this kind of suggestive question can help the decision maker to think carefully of the possible changes in the target points. Finally, we recall that the target point may be regarded as the satisficing set containing a single satisficing solution. Thus, by locating the satisficing set, we may locate the target point.

Example 4.5. Let us use Example 4.3 to illustrate the interactive method.

Step 1: Identify X, f, y^*, w, and p. The decision space X is given by

$$\begin{aligned} g_1(x) &= x_1 + x_2 \leq 100 \\ g_2(x) &= 2x_1 + x_2 \leq 150 \\ x_1, x_2 &\geq 0 \end{aligned}$$

and the objective functions are

$$\begin{aligned} y_1 &= f_1(x) = 6x_1 + 4x_2 \\ y_2 &= f_2(x) = x_1 \end{aligned}$$

Through conversation with the decision maker, assume that the ideal point will be used as the approximation for the target point. The ideal point (y_1^*, y_2^*) is (500, 75). We also assume that the two criteria are of the same weights. We shall use $p = 1, 2$, and ∞ for l_p-norm.

Step 2: Find compromise solutions. The computation of compromise solutions for $p = 1, 2$, and ∞ was illustrated in Example 4.4. The compromise solutions are $y^1 = (500, 50)$, $y^2 = (490, 55)$, and $y^\infty = (483.33, 58.33)$.

Step 3: Present relevant information to DM. All relevant information obtained in steps 1 and 2 will be presented to the decision maker. This includes X, f, Y, ideal points and the compromise solutions y^1, y^2, and y^∞ with their weights (w) and norms (p).

Step 4: Optimal solution reached? Suppose the decision maker is not satisfied with the compromise solutions presented. Assume that he or she is not satisfied with the target point (y^*) and the weight vector (w). The decision maker prefers to have the target point a little bit closer to the set Y and also feels that more weight should be given to the first criterion (f_1) rather than having equal weights.

Step 5: Through conservation with the decision maker, new relevant information on the target point (y^*) and the weight vector (w) will be obtained. Assume that the information on X, f, and p remains the same and that the new y^* is (450, 75) and the new w is (2/3, 1/3). With this new information compromise solutions y^1, y^2, and y^∞ can be recalculated and the new solutions will be presented to the decision maker. If the decision maker is satisfied with the new solutions, the procedure will be terminated; otherwise, more information from the decision maker on X, f, y^*, w, and p will be needed. We go to Box 5 and the process continues.

4.4. Further Comments

Utilizing human goal-seeking behavior, we have described the main concepts of satisficing and compromise solutions. In the real world, although specifying the satisficing set or the target point is not trivial, we can always start with the ideal point and then interact with the decision maker to gradually reach the final decision. For more details see Yu (1985, chap. 4).

There is a rich literature related to satisficing and compromise models. For instance, for goal programming see Charnes (1975), Ignizio (1976), Lee (1972) and those quoted therein; for different kinds of norms or penalty functions see Hwang and Yoon (1981), Gearhart (1979), Pascoletti and Serafini (1984), and the citations therein; for restructuring ideal points see Zeleny (1975).

Finally, observe that methods of this and the previous section are called methods of one-dimensional comparison, which tries to convert the preferences into a single-criterion numerical ordering. Other methods, such as Elimination et Choice Translating Reality (ELECTRE) (see Roy 1971 for further details) also belong to this class.

5. DOMINATION STRUCTURES

As mentioned in the introduction, preferences may be represented by one-dimensional comparison, which we discussed in the previous two sections, or in terms of multidimensional comparison, which we will discuss in this and the following section. Note that in one-dimensional comparison, we implicitly or explicitly assume that $\{?\} = \emptyset$ and that no ambiguity exists in preference. Once the value function or proper regret function is determined, MCDM becomes a one-dimensional comparison or a mathematical programming problem. In this section we shall tackle the problems with $\{?\} \neq \emptyset$.

Recall Definition 2.3 and $\{y^0\prec\}$, $\{y^0\succ\}$, $\{y^0 \sim\}$, and $\{y^0?\}$ are respectively the sets of points in Y that are better (preferred), worse (less preferred), equivalent and indefinite to y^0. For simplicity, we rewrite:

$$\{y^0\prec\} = y^0 + P(y^0)$$
$$\{y^0\succ\} = y^0 + D(y^0)$$
$$\{y^0\sim\} = y^0 + I(y^0)$$
$$\{y^0?\} = y^0 + U(y^0)$$

where $P(y^0)$, $D(y^0)$, $I(y^0)$, and $U(y^0)$ are respectively the sets of preferred, dominated, indifferent (equivalent), and unclarified (indefinite) factors.

Example 5.1

1. In Pareto preference (Example 2.l), for each y, $P(y) = \Lambda^{\geq} = \{d \in R^q | d_i \geq 0$ for all i and $d \neq 0\}$, $D(y) = \Lambda^{\leq} = \{d \in R^q | d_i \leq 0$ for all i and $d \neq 0\} = -P(y)$, $I(y) = \{0\}$, $U(y) = R^q \backslash (\Lambda^{\geq} \cup \Lambda^{\leq} \cup \{0\})$.
2. If the preference is represented by a concave differentiable value function $v(y)$, then $D(y)$ contains $\{d | \nabla v(y) \cdot d < 0\}$ which is a "half" space no matter what the dimensionality of Y is. Except linear $v(y)$, $D(y)$ is a function of y and in general $D(y) \neq -P(y)$. Note that in general $D(y)$ and $P(y)$ are not convex cones. Nevertheless, if we use the *tangent* cones of $\{y\succ\}$ and

$\{y\prec\}$ at y to represent the local sets of the preferred and dominated factors, denoted by $LD(y)$ and $LP(y)$, respectively, then $LD(y)$ and $LP(y)$ can be convex cones and $LD(y) = -LP(y)$. For a detail of such treatment, see Yu (1985, chap. 7).

In order to simplify our presentation, we shall assume the following throughout this section.

Assumption 5.1. For each $y \in Y$, $D(y)$ and $P(y)$ are convex cones. Furthermore, $D(y) = -P(y)$.

Remark 5.1. Pareto preference (Example 5.l number 1) satisfies Assumption 5.l, so does the preference represented by a linear value function. As indicated in Example 5.l, number 2, one can use $LD(y)$ and $LP(y)$ to replace $D(y)$ and $P(y)$ respectively when the assumption does not hold. If we do so, our results stated in this section are still valid, but only in a local sense (local optimal vs. global optimal). For the details of such treatment and conditions for the local results to be valid as the global results refer to Yu (1985, chap. 7).

Definition 5.1. A point $y^0 \in Y$ is a *nondominated point* or N-points if $y^0N[\prec]$, where $N[\prec]$ is as defined in Definition 2.4.

In Section 5.1, we discuss some basic properties of N-points when $D(y)$ is constant. In Section 5.2, we explore the case when $D(y)$ varies with y.

5.1. Constant Cone Domination Structures and Solutions

For simplicity, when $D(y) = \Lambda$ for all y, we shall call Λ the dominated cone and denote the set of all nondominated points or (N-points) by $N(Y, \Lambda)$. In this subsection we shall discuss two sets of conditions that characterize N-points. To facilitate our discussion, let us introduce the following concepts:

Definition 5.2

1. Y is Λ-convex if $Y + \Lambda$ is a convex set.
2. The polar cone of Λ is defined by $\Lambda^* = \{\lambda|\lambda\cdot d \leq 0 \text{ for all } d \in \Lambda\}$; interior is denoted by int Λ^*.
3. Given $\lambda \in \Lambda^*$, $\lambda \neq 0$, the set of all maximum points in Y with respect to linear functional $\lambda \cdot y$ is denoted by $Y^0(\lambda)$.

For further discussion on cone convexity, see Yu (1974, 1985).

Theorem 5.1

1. $\cup \{Y^0(\lambda)|\lambda \in \text{int } \Lambda^*, \lambda \neq 0\} \subseteq N(Y, \Lambda)$
2. If Y is Λ-convex and Λ is pointed (that is, $\Lambda \cap (-\Lambda) = \{0\}$), then

$$N(Y, \Lambda) \subseteq \cup \{Y^0(\lambda)|\lambda \in \Lambda^*, \lambda \neq 0\}$$

Note that the theorem states that conditions for N-points to be found by maximizing a linear function over Y; 1 is a sufficient condition and 2 a necessary condition. If one is interested in the entire set of N-points, then the theorem serves as an approximation for the set; 1 is the inner approximation and 2 the outer approximation for $N(Y, \Lambda)$. The results are derived in Yu (1974, 1985); their further refinement can be found in Hartley (1978).

A cone that is a closed polyhedron is called a *polyhedral cone*. If Λ is a polyhedral cone, so is Λ^*. In this case, there are a finite number of vectors $\{H^1, H^2, \ldots, H^p\}$ so that Λ^* is the cone generated by $\{H^1, H^2, \ldots, H^p\}$. That is,

$$\Lambda^* = \{a_1H^1 + a_2H^2 + \ldots + a_pH^p|a_i \in R^1, a_1 \geq 0, i = 1, 2, \ldots, p\}$$

$\{H^1, H^2, \ldots, H^p\}$ will be called a *generator* of Λ^*. From now on, unless otherwise specified, whenever Λ is a polyhedral cone, we shall assume that $\{H^1, H^2, \ldots, H^p\}$ is a generator for Λ^*.

Let $r(j)$ be the vector in R_k^{p-1} representing $\{r_k \in R^1|k \in \{1, 2, \ldots, p\}\backslash\{j\}\}$. Define $Y(r(j)) = \{y \in Y|H^k \cdot y \geq r_k, k \in \{1, 2, \ldots, p\}\backslash\{j\}\}$. Note that r_k may be regard as an aspiration level or satisfying level for $H^k \cdot y$.

Theorem 5.2 (Yu 1974). Let $\Lambda' = \Lambda \cup \{0\}$ be a polyhedral cone. Then $y^0 \in N(Y, \Lambda)$ if and only if for any arbitrary $j \in \{1, 2, \ldots, p\}$, there is $r(j)$ such that y^0 uniquely maximizes $H^j \cdot y$ over $Y(r(j))$.

This theorem essentially says that in search of $N(Y, \Lambda)$ by mathematical programming, constraints and objective functions are interchangeable.

5.2. Variable Cone Domination Structures and Methods for Seeking Good Solutions

In this subsection we describe a convergent method and a heuristic method to locate $N(Y, D(\cdot))$. To begin, let us assume that each $D(y)$ contains $\Lambda^{\leqq}$. Define $\Lambda^0 = \cap \{D(y)|y \in Y\}$ and set $Y^0 = Y$. Note that $\Lambda^{\leqq} \subseteq \Lambda^0$. Recursively, let us define, for $n = 0, 1, 2, \ldots$

$$\Lambda^n = \cap \{D(y)|y \in Y^n\} \text{ and } Y^{n+1} = N(Y^n, \Lambda^n)$$

Then

$$Y^{n+1} \subseteq Y^n \text{ and } \Lambda^n \subseteq \Lambda^{n+1}$$

It follows that the sequences converge. Denote by lim Y^n and lim Λ^n the limits. It can be shown (Yu 1985) that for each n, $N(Y, D(\cdot)) \subseteq Y^n$ and $N(Y, D(\cdot)) \subseteq \lim Y^n$. This observation allows us to locate the N-points according to Flowchart 1 (method l) of Figure 2.

Flowchart 1 is self-explanatory. Although the method guarantees a convergent set containing the N-points, it may not be effective and efficient. The following heuristic method according to Flowchart 2 (method 2) can be more efficient in reaching the final decision at the risk of missing some N-points. Let us explain Flowchart 2.

Box (*0*): We first explore the relationship among the criteria and locate some plausible aspiration levels and trade-offs among the criteria. Then we use Theorem 5.1 or 5.2 to locate a set of initial "good" alternatives corresponding to different aspiration levels and trade-offs, using mathematical programming if necessary. One can also start with the points that respectively maximize the individual criteria. This initial set is denoted by Z^0.

Box (*1*): We should first estimate $D(y)$ for some representative points in Z^n and find their intersection for Λ^n. The more representative points considered, the smaller is Λ^n and the less chance there is to miss N-points; but it may take longer to find the final solutions. In estimate $D(y)$, we can use the definition to locate it directly, or we can use the bounds of trade-off ratios to locate it indirectly. For further discussion, see Yu (1985).

Box (*2*): Here we can use Theorem 5.1 or 5.2 to locate the entire set of Z^{n+1} or to locate a representative set for Z^{n+1}, depending on how much we want to avoid missing some N-points.

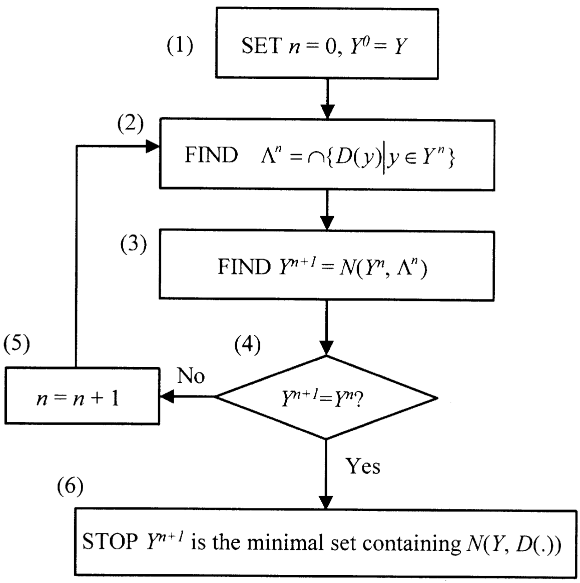

Figure 2 Flowchart 1: The First Method for Nondominated Solutions.

Boxes (3) and (4): Box (3) is a comparison to see if the process has reached a steady state. If not, the process will continue to Box (4), which is to eliminate those dominated points in Z^{n+1}. Here $\hat{D}(Z^{n+1})$ denotes estimated dominated points in Z^{n+1}, and W^{n+1} will be the remaining "good" alternatives after the elimination.

Boxes (5) and (6): Box (5) is a comparison to see if the elimination in Box (4) is effective if $W^{n+1} = Z^{n+1}$, then Z^{n+1} can be a set of N-points and the process can reach its steady state. Box (6) is to replace Z^{n+1} by W^{n+1} for the next iteration.

Boxes (7)–(10): Z^{n+1} is stable if no element in Z^{n+1} is dominated by any other element in Z^{n+1} and every element outside of Z^{n+1} is dominated by some element in $Z^{n+1} = N(Z^{n+1}, D(\cdot))$. If this condition is satisfied, Z^{n+1} can probably be the set of all N-points. In Box (9), nondominance is verified either by Theorem 5.1 or 5.2. If Z^{n+1} is the set of all N-points, the process is stopped at Box (10) with Z^{n+1} as the set for the final decision. Otherwise, the process should be repeated again from Box (0). If Z^{n+1} is not stable, then Z^{n+1} contains some dominated points or there are N-points not contained in Z^{n+1}. We shall accordingly either eliminate the dominated points from Z^{n+1} or add new N-points into Z^{n+1}. This adjustment on Z^{n+1} is performed at Box (8).

Box (11): This step is obvious for changing the step variable in the process.

For examples of applying the two methods described in this subsection, the interested reader is referred to Yu (1985).

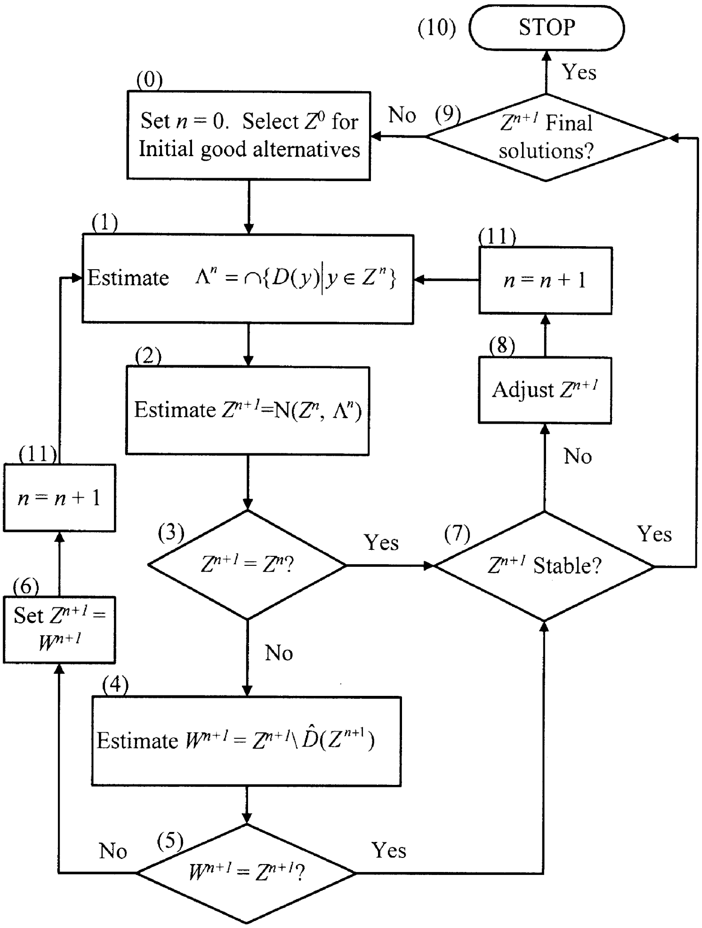

Figure 3 Flowchart 2: The Second Method for Nondominated Solutions.

6. MC²-SIMPLEX METHOD FOR LINEAR CASES

Linear function is the best function in analysis and application because it is easy to understand and compute. Because of these traits of linear functions, linear programming is popular and powerful. In this section we shall extend the linear programming with single criterion to that with multiple criteria and multilevel constraints of the resource. Instead of the simplex method with single criterion, we shall discuss the multicriteria (MC) simplex method and the multicriteria multiconstraint-level (MC²) simplex method to help resolve the difficulty of decision problems.

Algebraically, the MC²-simplex method can be represented by

$$\begin{aligned} \max \quad & Cx \\ \text{s.t.} \quad & Ax \leq D,\ x \geq 0 \end{aligned} \tag{1}$$

where $C = C_{q\times n}$, $A = A_{m\times n}$, and $D = D_{m\times k}$ are matrices. Note that if $k = 1$, then D is a vector and (1) becomes the MC simplex problem; if $q = 1$, then C is a vector and (1) becomes the ordinary simplex problem.

The MC²-simplex formulation can arise in many ways. Recall that Theorem 5.2 implies that to find N-points, the constraints and the objective functions in mathematical programming formulation are interchangeable. One may put some criteria in constraints with multiple levels, and thus the result will be a MC²-simplex form. Another case is in the design of optimal systems, in which one wishes to design a system that is optimal over all contingencies rather than find an optimal point within a given system (see Seiford and Yu 1979; Yu 1985, chap. 8 for further discussion). Alternatively, one may view the constraint level as occurring according to a random rule or influenced by some uncertain factor but contained within a set. In multiple-person decision making, the resource levels may reflect the views of different coalitions of the players.

6.1. Nondominated Solutions

The following result (Yu and Zeleny 1974) connects the relationship between N-points and the MC-simplex method.

Theorem 6.1. When D is a column vector, x^0 is an N-point in X with respect to $D(y) = \Lambda^{\leqq}$ (i.e., $Cx^0 \in N(Y, \Lambda^{\leqq})$) if and only if there exists some $\lambda > 0$ such that x^0 solves

$$\begin{aligned} \max \quad & \lambda Cx \\ \text{s.t.} \quad & Ax \leq D_{m\times 1},\ x \geq 0 \end{aligned} \tag{2}$$

Thus, by varying over λ over $\Lambda^{>} = \{\lambda | \lambda > 0\}$, we can locate all possible nondominated extreme points (N_{ex}-points) of X. Indeed, there are only a finite number of N_{ex}-points in X, and they are connected. (That is, by simplex pivoting, one can always arrive at any N_{ex}-point from other N_{ex}-point without leaving the bases of N_{ex}-points.) Locating all N_{ex}-points is in fact computationally feasible. Once all N_{ex}-points are located, they can be used to locate the entire N-points. (See Yu 1985, Yu and Zeleny 1974 for further discussion.)

To find N_{ex}-points, one can use the MC-simplex method, which appends multiple-criteria rows to the simplex tableaus. Because the MC-simplex method is a special case of the MC²-simplex method, we shall only describe the MC² simplex method in the next subsection.

Now suppose $\Lambda' = \Lambda \cup \{0\}$ is a general pointed polyhedral cone with $(H^1, H^2, \ldots, H^p)$ as a generator for Λ^*. One can define $C' = HC$, where H is the matrix with H^j as its jth row. Then $Cx^0 \in N(Y, \Lambda)$ if and only if there is $\lambda_{1\times p} > 0$ so that x^0 is a solution of (2) with C' replacing C. (See Yu 1985; Yu and Zeleny 1974 for further discussion.) This result suggests that in locating N_{ex}-points for a general polyhedral dominated cone Λ one can first convert the objective coefficient C into C', and then it becomes a problem of locating N_{ex}-points with dominated cone $\Lambda^{\leqq}$. Thus, unless specified, we shall assume $D(y) = \Lambda^{\leqq}$ for all y throughout the remainder of this subsection.

6.2. Potential Solutions and MC²-Simplex Method

Returning to (1), we generalize the concept of nondominated solutions into that of a potential solution by defining: A basis J is a potential basis (without confusion we also call J a potential solution) for the MC² problem (1) if there are $\lambda > 0$ and $\gamma > 0$ such that J is an optimal basis for

$$\begin{aligned} \max \quad & \lambda Cx \\ \text{s.t.} \quad & Ax \leq D\gamma,\ x \geq 0 \end{aligned} \tag{3}$$

Note that if D is a vector, $\gamma \in R^1$. By normalization, we can set $\gamma = 1$ and (3) reduces to (2). Then Theorem 6.1 ensures that a potential solution with D being a vector is indeed an N_{ex}-point. Thus, the concept of potential solutions is a generalization of that of nondominated solutions.

Note that the simplex tableau of (3) can be written as:

$$\begin{array}{cc|c} A & I & D\gamma \\ \hline -\lambda C & 0 & 0 \end{array} \qquad \begin{array}{c} (4) \\ (5) \end{array}$$

Let B be the basis matrix associated with J. Since each set of basic vectors J is uniquely associated with a column index set, we shall, without confusion, let J be this set of indices and J' the set of non-basic columns. The simplest tableau associated with J is

$$\begin{array}{cc|c} B^{-1}A & B^{-1} & B^{-1}D\gamma \\ \hline \lambda C_B B^{-1}A - \lambda C & \lambda C_B B^{-1} & \lambda C_B B^{-1} D\gamma \end{array} \qquad \begin{array}{c} (6) \\ (7) \end{array}$$

where (6) $= B^{-1} \cdot$ (4) (i.e., premultiply (4) by B^{-1} on both sides of the equation), (7) $= \lambda C_B \cdot$ (6) + (5), and C_B is the submatrix of criteria columns associated with the basis vectors.

Dropping γ and λ, we obtain the MC^2 tableau associated with basis J:

$$\begin{array}{cc|c} B^{-1}A & B^{-1} & B^{-1}D \\ \hline C_B B^{-1}A - C & C_B B^{-1} & C_B B^{-1}D \end{array} \qquad \begin{array}{c} (8) \\ (9) \end{array}$$

which we write as

$$\begin{array}{c|c} Y & W \\ \hline Z & V \end{array}$$

where $Y = [B^{-1}A, B^{-1}]$, $W = B^{-1}D$, $Z = [C_B B^{-1}A - C, C_B B^{-1}]$ and $V = C_B B^{-1}D$.

Let $W(J)$ and $Z(J)$ be the submatrices of the tableau associated with basis J Define:

$$\Gamma(J) = \{\gamma > 0 | W(J)\gamma \geq 0\}, \ \Lambda(J) = \{\lambda > 0 | \lambda Z(J) \geq 0\}$$

Note that, because of (6) and (8), the basis is feasible for all $\gamma \in \Gamma(J)$; and because of (7) and (9), the basis is also optimal for all $\lambda \in \Lambda(J)$. Immediately, we have (Yu 1985; Seiford and Yu 1979):

Theorem 6.2.

1. J is a potential solution if and only if $\Lambda(J) \times \Gamma(J)\varnothing$.
2. Given basis J, $\Gamma(J) \neq \varnothing$ if and only if $w_{\max} = 0$ for

$$\begin{array}{ll} \max & w = e_1 + e_2 + \ldots + e_q \\ \text{s.t.} & Z(J)x + e = 0 \\ & x \geq 0,\ e \geq 0, \text{ where } e \in R^q \end{array}$$

3. Given basis J, $\Gamma(J) \neq \varnothing$ if and only if $w_{\max} = 0$ for

$$\begin{array}{ll} \max & w = d_1 + d_2 + \ldots + d_k \\ \text{s.t.} & yW(J)x + d = 0 \\ & y \geq 0,\ x \geq 0, \text{ where } d \in R^k \end{array}$$

There are practical ways to verify whether a basis is a potential solution. It can be shown (Yu 1985; Seiford and Yu 1979) that the set of all potential bases is connected (that is, using pivoting, one can arrive at any potential solution without leaving the potential bases). This result makes locating all potential solutions feasible.

Given a potential basis, its corresponding sets of weights $\Lambda(J)$ and $\Gamma(J)$ on criteria and constraint levels can be specified. Even if we do not know the precise weights λ and γ, as long as $\lambda \in \Lambda(J)$ and $\gamma \in \Gamma(J))$, we know that J is the optimal basis and the decision process terminates at J for the solution. Since there are only a finite number of potential bases, finding them and identifying their corresponding weight sets can greatly simplify the difficulty in finding the final solution.

Suppose that we have $N = \{1, \ldots, n\}$ opportunities or products, from which we want to choose a subset to undertake or produce as to maximize profit. We can formulate the problem as in (3). We want to maximize the objective $\lambda\ Cx$ with constraint and $Ax \leq D\lambda$ and $x \geq 0$. Here λ and γ are uncertain. In this way we have a system design problem with multiple-criteria and multiple-resource availability levels. We can use MC^2-simplex method to identify a set of potentially good systems as candidates for the optimal system. Here, each potentially good system is a subset of a given opportunity set, each of which optimizes the objective when the parameters of contribution coefficients and that of resource availability levels fall in certain region. For further discussion along this line and applications, see Lee et al. (1990), Shi (1998a, b).

7. FUZZY MULTICRITERIA OPTIMIZATION

In addition to the above methods, we could imbed fuzzy set theory into multiple-criteria linear programming. Bellman and Zadeh (1970) propose the concept of fuzzy set, and Zimmermann (1978) first propose this idea. In fuzzy set theory, there is a membership function $\mu(x)$ indicating each element x the degree of membership for x to belong to a set. Fuzzy multiple-objective linear programming formulates the objectives and the constraints as fuzzy sets, characterized by their individual linear membership functions. The decision set is defined as the intersection of all fuzzy sets and the set defined by relevant hard constraints. A crisp (nonfuzzy) solution is generated by selecting the solution that has the highest degree of membership in the decision set. For further discussions, the reader is referred to Zimmermann (1978), Werners (1987), Martinson (1993), and Lee and Li (1993).

The fuzzy multiple objectives linear programming (f-MOLP) usually has the following format:

$$
\begin{aligned}
&\max\ \tilde{z}_k = \sum_{j=1}^{n} \tilde{c}_{kj} x_k,\ k = 1, 2, \ldots, q_1 \\
&\min\ \tilde{w}_k = \sum_{j=1}^{n} \tilde{c}_{kj} x_j,\ k = q_1 + 1, \ldots, q \\
&\text{s.t.} \sum_{j=1}^{n} \tilde{a}_{ij} x_j \leq \tilde{b}_i,\ i = 1, 2, \ldots, m_1 \\
&\qquad \sum_{j=1}^{n} \tilde{a}_{ij} x_j \geq \tilde{b}_i,\ i = m_1 + 1, \ldots, m_2 \\
&\qquad \sum_{j=1}^{n} \tilde{a}_{ij} x_j = \tilde{b}_i,\ i = m_2 + 1, \ldots, m \\
&\qquad x_j \geq 0,\ j = 1, 2, \ldots, n
\end{aligned}
\tag{10}
$$

where $\tilde{c}_{kj}$ is the jth coefficient of kth objective, $\tilde{a}_{ij}$ is jth coefficient of ith constraint, and $\tilde{b}_i$ is the right-hand side (RHS) of ith constraint. Note that $\tilde{c}_{kj}$, $\tilde{a}_{ij}$, and $\tilde{b}_i$ are fuzzy numbers. The f-MOLP problem (10) can be solved by transferring it into crisp MOLP (c-MOLP), shown as (11).

$$
\begin{aligned}
&\max\ (z_k)_a = \sum_{j=1}^{n} (c_{kj})_a^U x_j,\ k = 1, 2, \ldots, q_1 \\
&\min\ (w_k)_a = \sum_{j=1}^{n} (c_{kj})_a^L x_j,\ k = q_1 + 1, \ldots, q \\
&\text{s.t.} \sum_{j=1}^{n} (a_{ij})_a^L x_j \leq (b_i)_a^U,\ i = 1, 2, \ldots, m_1, m_2 + 1, \ldots, m \\
&\qquad \sum_{j=1}^{n} (a_{ij})_a^U x_j \geq (b_i)_a^L,\ i = m_1 + 1, \ldots, m \\
&\qquad x_j \geq 0,\ j = 1, 2, \ldots, n
\end{aligned}
\tag{11}
$$

where $(c_{kj})_a^U$ and $(c_{kj})_a^L$, $(a_{ij})_a^U$ and $(a_{ij})_a^L$ and $(b_i)_a^U$ and $(b_i)_a^L$ are upper and lower bound of fuzzy number $\tilde{c}_{kj}$, $\tilde{a}_{ij}$ and $\tilde{b}_i$, respectively, by taking α-level cut. Problem (11) can be solved by a fuzzy algorithm interactively. For details, see Zimmermann (1978) and Lee and Li (1993). For applications and extensions along this line, see Sakawa et al. (1994, 1995), Shibano et al. 1996), Shih et al. (1996), Ida and Gen (1997), and Shih and Lee (1999) and those quoted therein.

8. EXTENSIONS AND CONCLUDING REMARKS

We have briefly sketched six important topics of MCDM problems. Many more topics, such as interactive methods including adapted gradient search method, surrogate-worth trade-off methods (Haimes and Hall 1974), the Zionts–Wallenius method (Zionts and Wallenius 1976), the paired comparison simplex method (Malakooti and Ravindran (1986), the bireference procedure (Michalowski and Szapiro 1992), the paired comparison method (Shin and Allen 1994), preference over uncertain outcomes, and second-order games can be found in Yu (1985, chap. 10 and the citations therein). For multiple-criteria dynamic optimization problems, refer to Li and Haimes (1989), Yu and Seiford (1981), Yu and Leitmann (1974) and those quoted therein. The reader who is interested in decision aid and MCDM in abstract spaces is referred to Roy (1977) and Dauer and Stadler (1986) respectively, and the citations therein. For other mathematical analysis on multiple criteria optimization, see Metev and Yordanova-Markova (1997), Ehrgott and Klamroth (1997), and Gal and Hanne (1999).

There are a number of computer programs to solve multiple-criteria decision making (MCDM) problems. Because many MCDM problems are reformulated and solved by converting them into single-criterion optimization problems (as discussed in the previous sections), many computer programs for mathematical programming are adopted in MCDM computer software. Since the problem domains of MCDM are rich and complex, so are the computer software programs. Space limitations make it difficult to list and discuss them here. However, interested readers can refer to www.cba.uga.edu/mcdm.html, the website of the International Society on Multiple Criteria Decision Making, for MCDM software listing.

With creative thinking and practicing, the basic concepts of MCDM described above can generate an infinite number of concepts and be applied to an infinite number of multicriteria problems. Rigid habitual ways of thinking usually are the stumbling blocks to creativity. In solving nontrivial problems, we need not only the mastery of the mathematical tools but also a good understanding of human behavior and the habitual domains of the decision makers (see Yu 1990, 1995 for a detailed discussion). At the least, we should not become so overenthusiastic about one particular concept or method that, so to speak, we cut our feet to fit the already-made shoes.

REFERENCES

Bellman, R. E., and Zadeh, L. A. (1970), ‘‘Decision-Making in a Fuzzy Environment,’’ *Management Science*, Vol. 17, pp. 140–164.

Charnes, A., and Cooper, W. W. (1975), ‘‘Goal Programming and Constrained Regression—A Comment,’’ *Omega,* Vol. 3, pp. 403–409.

Chien, I. S. (1987), ‘‘Restructuring and Computer Support in Multiple Criteria Decision Making,’’ Ph.D. dissertation, University of Kansas, Lawrence, KS.

Chien, I. S., Yu, P. L., and Zhand, D. (1990), ‘‘Indefinite Preference Structures and Decision Analysis,’’ *Journal of Optimization Theory and Application*, Vol. 64, pp. 71–85.

Cogger, K. O., and Yu, P. L. (1985), ‘‘Eigen Weight Vectors and Least Distance Approximation for Revealed Preference in Pairwise Weight Ratios,’’ *Journal of Optimization Theory and Application,* Vol. 46, pp. 483–491.

Dauer, J. P., and Stadler, W. (1986), ‘‘A Survey of Vector Optimization in Infinite-Dimensional Space, Part 2,’’ *Journal of Optimization Theory and Application,* Vol. 51, pp. 205–241.

Debreu, G. (1954), ‘‘Representation of a Preference Ordering by a Numerical Function,’’ in *Decision Processes*, R. M. Thrall, C. H. Coombs, and R. L. Davis, Eds., John Wiley & Sons, New York.

Debreu, G. (1960), ‘‘Topological Methods in Cardinal Utility Theory,’’ in *Mathematical Methods in Social Sciences*, K. J. Arrow, S. Karlin, and P. Suppes, Eds., Stanford University Press, Stanford, CA.

Ehrgott, M., and Klamorth, K. (1997), ‘‘Connectedness of Efficient Solutions in Multiple Criteria Combinatorial Optimization,’’ *European Journal of Operational Research,* Vol. 97, pp. 159–166.

Freimer, M., and Yu, P. L. (1976), ‘‘Some New Results on Compromise Solutions for Group Decision Problem,’’ *Management Science*, Vol. 22, pp. 688–693.

Fishburn, P. C. (1970), *Utility Theory for Decision Making*, John Wiley & Sons, New York.

Gal, T., and Hanne T. (1999), ‘‘Consequences of Dropping Nonessential Objectives for the Application of MCDM Methods,’’ *European Journal of Operational Research,* Vol. 119, pp. 373–378.

Gearhart, W. B. (1979), ‘‘Compromise Solutions and Estimation of the Noninferior Set,’’ *Journal of Optimization Theory and Application,* Vol. 28, pp. 29–47.

Gorman, W. M. (1968), ‘‘The Structure of Utility Functions,’’ *Review of Economic Studies,* Vol. 35, pp. 367–390.

Haimes, Y. Y., and Hall, W. A. (1974), ‘‘Multiobjectives in Water Resources System Analysis: The Surrogate Worth Trade off Method,’’ *Water Resources Research*, Vol. 10, pp. 615–623.

Hartley, R. (1978), ‘‘On Cone-Efficiency, Cone-Convexity and Cone-Compactness,’’ *SIAM Journal of Applied Mathematics*, Vol. 34, pp. 211–222.

Hwang, C. L., and Masud, A. S. M. (1979), *Multiple Objective Decision Making—Methods and Applications: A State-of-the-Art Survey*, Springer, New York.

Hwang, C. L., and Yoon, K. (1981), *Multiple Attribute Decision Making—Methods and Applications: A State-of-the-Art Survey,* Springer, New York.

Ida, K. and Gen, M. (1997), "Improvement of Two-Phase Approach for Solving Fuzzy Multiple Objective Linear Programming," *Journal of Japan Society for Fuzzy Theory and System,* Vol. 19, pp. 115–121 (in Japanese).

Ignizio, J. P. (1976), *Goal Programming and Extensions,* D.C. Heath, Lexington, MA.

Keeney, R. L., and Raiffa, H. (1976), *Decisions with Multiple Objectives: Preferences and Value Tradeoffs,* John Wiley & Sons, New York.

Lee, E. S., and Li, R. J. (1993), "Fuzzy Multiple Objective Programming and Compromise Programming with Pareto Optimum," *Fuzzy Sets and Systems,* Vol. 53, pp. 275–288.

Lee, S. M. (1972), *Goal Programming for Decision Analysis*, Auerback, Philadelphia.

Lee, Y. R., Shi, Y., and Yu, P. L. (1990), "Linear Optimal Designs and Optimal Contingency Plans," *Management Science*, Vol. 36, pp. 1106–1119.

Li, D., and Haimes, Y. Y. (1989), "Multiobjective Dynamic Programming: The State of the Art," *Control Theory and Advanced Technology,* Special Issue on Multiobjective Discrete Dynamic System Control Theory and Advanced Technology, Y. Y. Haimes and D. Li, Eds., Vol. 5, pp. 471–483.

Malakooti, B. and Ravindran, A. (1986), "Experiments with an Interactive Paired Comparison Simplex Method for MOLP Problems," *Annals of Operations Research,* Vol. 5 pp. 575–597.

Martinson, F. K. (1993), "Fuzzy vs. Minmax Weighted Multiobjective Linear Programming Illustrative Comparisons," *Decision Sciences,* Vol. 24, pp. 809–824.

Metev, B. S., and Yordanova-Markova, I. T. (1997), "Multi-objective Optimization over Convex Disjunctive Feasible Sets Using Reference Points," *European Journal of Operational Research,* Vol. 98, pp. 124–137.

Michalowski W. and Szapiro T. (1992), "A Bi-reference Procedure for Interactive Multiple Criteria Programming," *Operations Research*, Vol. 40, pp. 247–258.

Pascoletti, A., and Serafini, P. (1984), "Scalarizing Vector Optimization Problems," *Journal of Optimization Theory and Application,* Vol. 42, pp. 499–524.

Roy, B. (1971), "Problems and Methods with Multiple Objective Functions," *Math Programming,* Vol. 1, pp. 239–266.

Roy, B. (1977), "A Conceptual Framework for a Prescriptive Theory of Decision-Aid," in *Multiple Criteria Decision Making,* M. K. Starr and M. Zeleny, Eds., TIMS Studies in Management Sciences, Vol. 6, North-Holland, Amsterdam.

Roy, B. and Vincke, P. H. (1987), "Pseudo-Orders: Definition, Properties, and Numerical Representation," *Mathematical Social Sciences,* Vol. 14, pp. 263–274.

Saaty, T. L. (1980), *The Analytic Hierarchy Process*, McGraw-Hill, New York.

Sakawa, M., Inuiguchi, M., Sundad, H., and Sawada, K. (1994), "Fuzzy Multiobjective Combinatorial Optimization through Revised Genetic Algorithms," *Journal of Japan Society for Fuzzy Theory and System,* Vol. 6, pp. 177–186 (in Japanese).

Sakawa, M., Kato, K., Sundad, H., and Enda, Y. (1995), "An Interactive Fuzzy Satificing Method for Multiobjective 0-1 Programming Problems through Revised Genetic Algorithms," *Journal of Japan Society for Fuzzy Theory and System,* Vol. 17, pp. 361–370 (in Japanese).

Seiford, L., and Yu, P. L. (1979), "Potential Solutions of Linear Systems: The Multi-Criteria Multiple Constrain Levels Program," *Journal of Mathematical Analysis and Applications,* Vol. 69, pp. 283–303.

Shi, Y. (1998a), "Finding Dual Flexible Contingency Plans for Optimal Generalized Linear Systems," *International Transactions in Operational Research,* Vol. 5, pp. 303–315.

Shi, Y. (1998b), "Optimal System Design with MC^2 Linear Programming: A Dual Contingency Plan Approach," *European Journal of Operational Research,* Vol. 107, pp. 692–709.

Shibano, T., Sakawa M., and Obata H. (1996), "Interactive Decision Making for Multiobjective 0–1 Programming Problems with Fuzzy Parameters through Genetic Algorithms," *Journal of Japan Society for Fuzzy Theory and System,* Vol. 18, pp. 1144–1153 (in Japanese).

Shih, H. S., and Lee, E. S. (1999), "Fuzzy Multi-Level Minimum Cost Flow Problems," *Fuzzy Sets and Systems*, Vol. 107, pp. 159–176.

Shih, H. S., Lai, Y. J., and Lee, E. S. (1996), "Fuzzy Approach for Multiple-Level Programming Problems," *Computers and Operation Researches,* Vol. 23, pp. 73–91.

Shin, W. S., and Allen, D. B. (1994), "An Interactive Paired Comparison Method for Bicriterion Integer Mathematical Programming," *Naval Research Logistics,* Vol. 41, pp. 423–434.

Stadler, W. (1979), "A Survey of Multicritena Optimization or the Vector Maximization Problem, Part I: 1776–1960," *Journal of Optimization Theory and Application,* Vol. 29, pp. 1–52.

Stadler, W. (1981), "A Comprehensive Bibliography on Multicriteria Decision Making and Related Areas," Working Paper, University of California, Berkeley, CA.

Steuer, R. E., Gardiner, I. R., and Gray, J. (1996), "A Bibliographic Survey of the Activities and International Nature of Multiple Criteria Decision Making," *Journal of Multi-Criteria Decision Analysis,* Vol. 5, No. 3, pp. 195–217.

Takeda, E., and Yu, P. L. (1995), "Assessing Priority Weights from Subsets of Pairwise Comparisons in Multiple Criteria Optimization Problems," *European Journal of Operational Research*, Vol. 86.

Werners, B. (1987), "Interactive Multiple Objective Programming Subject to Flexible Constraints," *European Journal of Operational Research,* Vol. 31, pp. 342–349.

Yu, P. L. (1973), "A Class of Solutions for Group Decision Problems," *Management Science,* Vol. 19, pp. 936–946.

Yu, P. L. (1974), "Cone Convexity, Cone Extreme Points and nondominated Solutions in Decision Problems with Multiobjectives," *Journal of Optimization Theory and Applications*, Vol. 14, pp. 319–377.

Yu, P. L. (1985), *Multiple Criteria Decision Making Concepts, Techniques and Extensions*, Plenum Press, New York.

Yu, P. L. (1990), *Forming Winning Strategies—An Integrated Theory of Habitual Domains,* Springer, Berlin.

Yu, P. L. (1995), *Habitual Domains: Freeing Yourself from the Limits on Your Life,* Highwater Editions, Shawnee Mission, KS.

Yu, P. L., and Leitmann, G. (1974), "Nondominated Decisions and Cone Convexity in Dynamic Multicriteria Decision Problems," *Journal of Optimization Theory and Applications,* Vol. 14, pp. 573–584.

Yu, P. L., and Seiford, L. (1981), "Multistage Decision Problems with Multicriteria," in *Multicriteria Analysis: Practical Methods*, P. NijKamp and J. Spronk, Eds, Gower Press, London.

Yu, P. L., and Takeda, E. (1987), "A Verification Theorem of Preference Separability for Additive Value Functions," *Journal of Mathematical Analysis and Applications,* Vol. 126, pp. 382–396.

Yu, P. L. and Zeleny, M. (1974), "The Set of All Nondominated Solutions in the Linear Case and a Multicriteria Simplex Method," *Journal of Mathematical Analysis and Applications,* Vol. 49, pp. 430–468.

Zeleny, M. (1975), "The Theory of the Displaced Ideal," in M. Zeleny, Ed., Springer, New York, pp. 151–205.

Zimmermann, H. J. (1978), "Fuzzy Programming and Linear Programming with Several Objective Functions," *Fuzzy Sets and Systems,* Vol. 1, pp. 45–55.

Zionts, S. and Wallenius, J. (1976), "An Interactive Programming Method for Solving the Multiple Criteria Problem," *Management Science*, Vol. 22, pp. 652–663.

ADDITIONAL READING

Changkong, V., and Haimes, Y. Y., *Multiobjective Decision Making Theory and Methodology,* North-Holland, New York, 1983.

Dyer, J. S., "Interactive Goal Programming," *Management Science,* Vol. 19, 1970, pp. 62–70.

Fandel, G., Grauer, M., Kurzhanski, A., and Wierzbicki, A. P., Eds., *Large-Scale Modelling and Interactive Decision Makings, Proceedings, Eisenbach, GRD, 1985,* Lecture Notes in Economics and Mathematic No. 273, Springer, New York, 1986.

Gal, T., *Postoptimal Analysis Mathematic Programming and Related Topics,* McGraw-Hill, New York, 1979.

Haimes, Y. Y., and Changkong, V., Eds., *Decision Making with Multiple Objectives. Proceedings, Cleveland, Ohio, 1984,* Lecture Notes in Economics and Mathematical Systems No. 242, Springer, New York, 1985.

Hansen, P., Ed., *Essays and Surveys on Multiple Criteria Decision Making, Proceedings, Mons, 1982,* Lecture Notes in Economics and Mathematical Systems No. 209, Springer, New York, 1983.

Hazen, G. B., and Morin, T. L., "Optimality Conditions for Non-conical Multiple-Objective Programming," *Journal of Optimization Theory and Application,* Vol. 40, 1983, pp. 25–59.

Morse, J. N., Ed., *Organization: Multiple Agents with Multiple Criteria. Proceedings, University of Delaware, Newark, 1980,* Lecture Notes in Economics and Mathematical Systems No. 190, Springer, New York, 1981.

Pareto, V., 1906; Piccola *Manual of Political Economy*, A. S. Schwier, Trans., MacMillan, New York, 1971.

Sawaragi, Y., Nakayama, H., and Tamno, T. (1985), *Theory of Multiobjective Optimization,* Academic Press, Orlando, FL.

Steuer, R. E., *Multiple Criteria Optimization,* John Wiley & Sons, New York, 1985.

Tanino, T., and Sawaragi, Y., ''Stability of Nondominated Solutions in Multicriteria Decision Making,'' *Journal of Optimization Theory and Application,* Vol. 30, 1980, pp. 229–253.

Wang, P. Z., *Fuzzy Sets and the Fall-Shadow of Random Sets,* Beijing Normal University Press, Beijing, 1985 (in Chinese).

White, D. J., *Optimality and Efficiency,* John Wiley & Sons, New York.

Zeleny, M. (1982), *Multiple Criteria Decision Making*, McGraw-Hill, New York.