13
Multi-objective Optimization: Theory and Methods

13.1 Introduction

By now we are well versed with the concepts of linear and nonlinear programming problems involving optimization of a single objective function. We now take a step further and enter in the area where the programming problems require optimization of more than one objective function. To get comfortable with this idea, we begin our discussion by considering a very simple decision making problem of buying a car. Let us pose a question to ourself. What features we will look for in a car of our choice? Cost, comfort, space, mileage, safety, engine power, height from the ground level, power steering, color, and may be some additional technical features, like, power windows, quality of AC, and so on. From this list it is obvious that a decision of buying a particular car is not based on a single criterion, of say cost alone, but many more criteria are equally vital in the final decision. Problems of this kind can not be modeled through a single objective optimization problems. One needs to look beyond that and talk about *multiobjective programming* (MOP). As the name is self explanatory, a multiobjective programming problem (MOPP) deals with optimization of more than one objective criterion. MOPPs are frequently encountered in problems of product design, management decisions, resource planning, to name a few. This branch of optimization is also known by other names, like, vector optimization, multicriteria optimization, multiattribute optimization and so forth.

In this chapter, we shall be concentrating on MOPPs unlike the previous chapters where the focus was on a single objective optimization problems. We shall be describing and characterizing various solution concepts associated with the class of MOPPs. In the later part of the chapter, we shall be discussing a technique, known as *goal programming technique*, to solve a particular class of linear multiobjective programming problems. Our aim is to get familiar with the otherwise vast and complex topic of MOP. We therefore shall be avoiding the technical complexities and restrict ourselves to understand the fundamental ideas involved in the context.

13.2 Conflicting Objectives and Tradeoff

Going back to the problem of buying a car, it is evident that some of the several listed criteria are conflicting in nature, like, more comfort can generally be achieved by increasing the cost of a car, more advance technical features will also enhance the cost of a car, luxury car provides more comfort but generally gives less mileage. In practice too, MOPP involves several conflicting and non-commensurate objective functions that have to be optimized simultaneously over a feasible region.

For instance, consider two objectives represented by functions, say, $f_1(x) = x^2$ and $f_2(x) = (x-1)^2$. We wish to minimize them simultaneously over the set $[0,1]$.

We may sketch the graphs of $f_1(x)$ and $f_2(x)$, and observe that while f_1 is increasing in $[0,1]$, f_2 is decreasing in $[0,1]$, thereby resulting in a situation where it is not possible to find an $x \in [0,1]$ that can minimize both the objectives.

If we can find a feasible vector $x^\star$ that optimizes all the objective criteria simultaneously then we have certainly achieved an ideal solution of the problem. But quite often, improvement in one criterion results in a loss in another criterion leading to the unlikely existence of an ideal solution. It can be seen as some kind of tradeoff between various objectives. Visualizing and resolving the tradeoffs is one of the key aspect of MOP. For this reason, one has to look for the 'best' compromise solution. Now, 'best' can be defined differently in different situations. In economics, 'best' is referred to the decisions taken by the buyers and sellers or the governments which simultaneously optimize several criteria. One of the most frequently quoted example thereof is Taxation. An optimal collected tax is one which maximizes the revenue for common goods while maintaining sufficient incentives for individuals to earn reasonably good income from their work.

Not surprisingly, the first person to describe such a tradeoff was an Economist F. Y. Edgeworth . He defines an optimum within the context of two consumers criteria, P and Q, as "It is required to find a point (x, y) such that in whatever direction we take an infinitely small step, P and Q do not increase together but that, while one increases, the other decreases". Vilfredo Pareto, chair of political economy at the University of Lausanne, Switzerland, wrote in 1906, "The optimum allocation of the resources of a society is not attained so long as it is possible to make at least one individual better off in his own estimation while keeping others as well off as before in their own estimations".

The above statements beautifully capture the balance of the socio-economic structure of any society. The very same principle is applicable to MOPPs. One needs to create a balance between various conflicting objective functions to arrive at the 'best' or 'optimal' solution in some appropriate sense. Of course, we need to know how to designate a particular feasible point to be a 'best' compromise solution. We shall be examining this issue from mathematical viewpoint in the sections to follow.

13.3 Various Solution Concepts

The following form of MOPP is studied in this chapter

$$\underset{x \in S}{Min}\ f(x) = (f_1(x), \dots, f_p(x)). \tag{13.1}$$

Here we assume that S is a nonempty subset of $\mathbf{R}^n$ representing the feasible set of problem (13.1), and $f : S \to \mathbf{R}^p$ is a given vector function comprising of p objective criteria to be minimized.

For $p = 1$, problem (13.1) reduces to a scalar nonlinear programming problem which has been a subject of study in the previous chapters. So, for the multiobjective case, we take $p \geq 2$. Moreover, it is not necessary that all the p objective criteria are to be minimized, some criteria may involve maximization process. For instance, in a car buying problem discussed earlier, we would like to maximize comfort, maximize mileage and minimize cost of a car. Actually in context of modeling the problem it does not matter whether we investigate minimization or maximization problem. One can convert all the maximization criteria into the minimization form by using the identity, $Max f_i(x) = -Min(-f_i(x))$. With this understanding, we study MOPP in the minimization form in problem (13.1). Observe that we have yet to define what we mean by minimization of a vector objective function, $f(x) \in \mathbf{R}^p$ with $x \in S$, in (13.1).

The basic problem to realize here is that unlike the real space $\mathbf{R}$, the space $\mathbf{R}^p$, $p \geq 2$, is not an ordered space unless we define an appropriate partial order. It simply means that given any two distinct real numbers x and y it is always possible to determine the greater among them. But the same is not true if x and y are vectors in $\mathbf{R}^p$. For instance, it is not possible to compare the vectors, $(2,1)^T$ and $(1,2)^T$, in general.

Now, for $x \in S$, we get the resultant objective vector $f(x)$ in $\mathbf{R}^p$. Let $f(S) = \{y \in \mathbf{R}^p : \exists x \in S \text{ such that } y = f(x)\}$ denotes the image set of S under f. To define an optimal solution in the sense of minimization we need to compare vectors in the image set $f(S)$, and for this, we need to identify a partial order relation in $\mathbf{R}^p$.

To appreciate this aspect of MOPP, we consider the problem

$$\underset{x \in [0,1]}{Min}\ (x^2,\ (x-1)^2)$$

Let $y_1 = x^2$ and $y_2 = (x-1)^2$. For $y = (y_1, y_2) \in f(S)$, there exists $x \in S$ such that $f(x) = y$, i.e. $y_1 = x^2$, $y_2 = (x-1)^2$. As $x \in S = [0,1]$, $y_1 \geq 0$, $y_2 \geq 0$. Moreover, $x = \sqrt{y_1}$ and $(x-1) = -\sqrt{y_2}$ yields the set, $f(S) = \{(y_1, y_2) : y_1 \geq 0, y_2 \geq 0, \sqrt{y_1} + \sqrt{y_2} = 1\}$.

Now if we choose any two vectors in the set $f(S)$, say $(1,0)$ and $(0,1)$, or $(\frac{1}{4}, \frac{1}{4})$ and $(\frac{9}{16}, \frac{1}{16})$, we can not decide as which one is greater of the two unless some priorities are attached with the two objective functions.

Thus the major thrust in defining the solution concepts for MOPP (13.1) lies in

In this chapter we assume that $\mathbf{R}^p$ is partially ordered by a binary relation induced by $\mathbf{R}^p_+$, the nonnegative orthant of $\mathbf{R}^p$. By this we mean the following. For $x, y \in \mathbf{R}^p$,

$$
\begin{aligned}
x \leqq_{\mathbf{R}^p_+} y \quad &\Leftrightarrow \quad y - x \in \mathbf{R}^p_+; \\
x \leq_{\mathbf{R}^p_+} y \quad &\Leftrightarrow \quad y - x \in \mathbf{R}^p_+ \setminus \{0\}; \\
x <_{\mathbf{R}^p_+} y \quad &\Leftrightarrow \quad y - x \in int\mathbf{R}^p_+.
\end{aligned}
$$

Thus, $\begin{pmatrix} 1 \\ 2 \end{pmatrix} <_{\mathbf{R}^2_+} \begin{pmatrix} 2 \\ 3 \end{pmatrix}$ while $\begin{pmatrix} 1 \\ 2 \end{pmatrix} \leq_{\mathbf{R}^2_+} \begin{pmatrix} 1 \\ 3 \end{pmatrix}$.

It is important to note the difference between $x \leqq_{\mathbf{R}^p_+} y$ and $x \leq_{\mathbf{R}^p_+} y$. While the first one means $x_i \leqq y_i$ $(i = 1, \ldots, p)$, the latter one imply $x_i \leqq y_i$ $(i = 1, \ldots, p, i \neq j)$ and $x_j < y_j$, for some j. From now onwards we suppress the subscript $\mathbf{R}^p_+$ in the above relations. Moreover, for $x, y \in \mathbf{R}$, we continue to use $x \leq y$ to denote x is less than equal to y. The partial order $\leq$ is to be understood in the right sense in the given context.

We are now ready to introduce two solution concepts for MOPP (13.1).

Definition 13.3.1 (Weak Efficient Solution). *$x^\star \in S$ is called a weak efficient solution of problem (13.1) if there does not exist $x \in S$ such that $f(x) < f(x^\star)$. In other words, it says that whenever $x \in S$, $f(x) - f(x^\star) \notin -(int\mathbf{R}^p_+)$. In set notation form the same can be explained as $(f(S) - f(x^\star)) \bigcap (-int\mathbf{R}^p_+) = \emptyset$.*

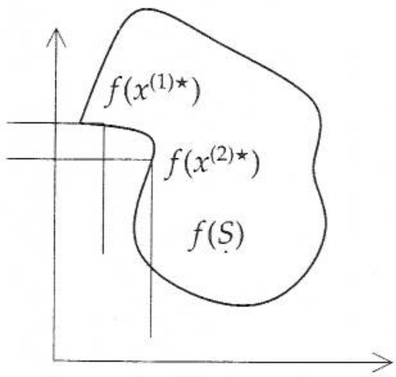


Fig. 13.1.

Remark 13.3.1 *In two or three dimensional spaces, we can graphically illustrate the concept of weak efficiency. Let $p = 2$, i.e. $f(S) \subseteq \mathbf{R}^2$. The above definition suggests that on shifting the origin to a point $f(x^\star)$, if we find that $int\mathbf{R}^2_-(= -int\mathbf{R}^2_+)$ does not intersect the set $f(S)$, then $x^\star$ is a weak efficient solution of (13.1), see Fig 13.1. $x^{(1)\star}$ is a weak efficient solution of some minimization problem represented by $f(S)$ whereas $x^{(2)\star}$ is not*

We present below few examples in support of above definition. The examples also illustrate that the weak efficient solution of MOPP is not necessarily unique.

Example 13.3.1 *Let* $S = \{x = (x_1, x_2) : x_1+x_2 \geq 2, (x_1-1)^2+(x_2-1)^2 \leq 4, x_1, x_2$ *are integers* $\}$ *and* $f : S \to \mathbf{R}^2$ *be defined as* $f(x_1, x_2) = (x_1+x_2^2,\ x_1^2+x_2)$. *Consider the problem* $\underset{x \in S}{Min}\ f(x)$, *and identify the set of weak efficient solutions.*

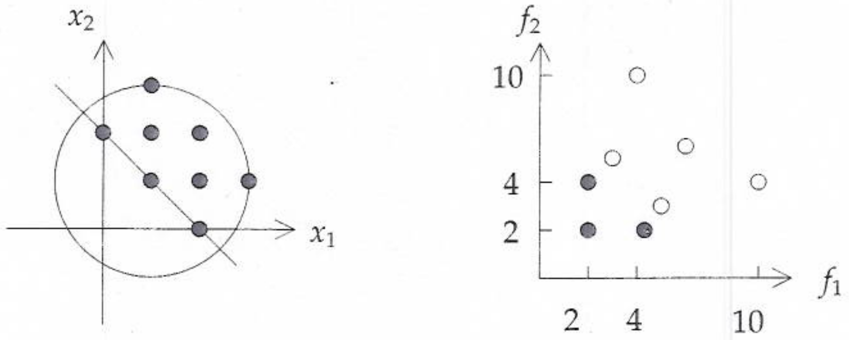


Fig. 13.2.

Solution Observe that $S = \{(0,2), (1,1), (1,2), (1,3), (2,0), (2,1), (2,2), (3,1)\}$, consequently, $f(S) = \{(4,2), (2,2), (5,3), (10,4), (2,4), (3,5), (6,6), (4,10)\}$.

The sets S, $f(S)$ are shown in Fig 13.2. The set of weak efficient solutions of the problem is given by $\{(0,2), (1,1), (2,0)\}$.

Example 13.3.2 *Let* $S = [0,1] \times [0,1]$ *and* $f : S \to \mathbf{R}^2$ *be the identity map given by* $f(x_1, x_2) = (x_1, x_2)$. *Consider the problem*

$$\underset{x \in S}{Min}\ f(x), \tag{13.2}$$

and obtain the set of weak efficient solutions.

Solution Obviously $f(S) = S$. $x^\star = (0,1) \in S$ is a weak efficient solution of (13.2), as there is no $x \in S$ with $x_1 < 0$ and $x_2 < 1$, i.e. $f(x) < f(x^\star)$. Similarly for $x^\star = (1,0) \in S$, there is no $x \in S$ with $x_1 < 1$ and $x_2 < 0$, in other words, there exists no $x \in S$ with $f(x) < f(x^\star)$. Consequently, $(1,0)$ is also a weak efficient solution of (13.2). In fact it can easily be seen that all the points in the set $\{(x_1, 0) : 0 \leq x_1 \leq 1\} \bigcup \{(0, x_2) : 0 \leq x_2 \leq 1\}$ are weak efficient solutions of (13.2).

Example 13.3.3 *Let* $S = \mathbf{R}^2$ *and* $f : S \to \mathbf{R}^2$ *be defined as* $f(x_1, x_2) = (x_1^2, x_2^2)$. *Obtain the set of weak efficient solutions of* $\underset{x \in S}{Min}\ f(x)$.

Solution It is clear that $f(S) = \mathbf{R}^2_+$. Thus the set of weak efficient solutions of the

Example 13.3.4 *Let* $S = \{x = (x_1, x_2) : x_1 \geq 1, x_2 \geq 1, x_1 + x_2 \geq 4, \max\{2x_1, 3x_2\} \geq 6\}$ *and* $f : S \to \mathbf{R}^2$ *be* $f(x_1, x_2) = (x_1, x_2)$. *Obtain the set of weak efficient solution of* $\underset{x \in S}{Min}\ f(x)$.

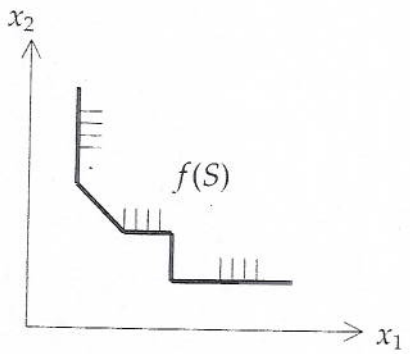


Fig. 13.3.

Solution The set $f(S)$ is depicted in Fig 13.3. The set of weak efficient solutions is marked by a bold curve.

We now define another solution concept called efficient solution of MOPP (13.1). The idea of efficiency is based upon the conviction that no criterion can be improved without worsening at least one other criterion.

Definition 13.3.2 (Efficient Solution). $x^\star \in S$ *is called an efficient solution of problem (13.1) if there does not exist* $x \in S$ *such that* $f(x) \leq f(x^\star)$. *In other words, it says that whenever* $x \in S$, $f(x) - f(x^\star) \notin -(\mathbf{R}^p_+ \setminus \{0\})$. *In set notation form the same can be explained as* $(f(S) - f(x^\star)) \bigcap (-\mathbf{R}^p_+ \setminus \{0\}) = \emptyset$.

This solution is also known by the names *Pareto solution* or *non-inferior solution.*

Definition 13.3.3 (Efficient Frontier). *The set of efficient solutions of MOPP (13.1) is called efficient frontier.*

Remark 13.3.2 *It follows from Definition 13.3.1 and Definition 13.3.2 that every efficient solution is a weak efficient solution of problem (13.1). But the converse need not hold. Some of the following examples justify this fact.*

Example 13.3.5 *Let* $S = \{x = (0, x_2) : x_2 \leq 0\}$ *and* $f : S \to \mathbf{R}^2$ *be* $f(x_1, x_2) = (x_1, x_2)$. *Consider* $\underset{x \in S}{Min}\ f(x)$, *and identify the set of efficient solutions.*

Solution Here $f(S) = S$. The set of efficient solutions is an empty set because for any $(x_1^\star, x_2^\star) \in S$, we can always find another $(x_1, x_2) \in S$ with $x_1 = x_1^\star = 0$ and $x_2 < x_2^\star$. Note that the set of weak efficient solutions is the entire set S.

Going back to Example 13.3.1, we find that the problem has only one efficient solution, namely, $(1, 1)$. Also, recall Example 13.3.2. It is clear that $(0, 0)$ is the only efficient solution of the MOPP considered therein while the set of weak efficient solutions is an uncountable subset of S.

Example 13.3.6 *Let* $S = \{x = (x_1, x_2) : x_1 \in [0, 1], \sqrt{x_1} \leq x_2 \leq \sqrt[3]{x_1}\}$. *Identify the sets of weak efficient and efficient solutions of* $\underset{x \in S}{Min}\ f(x_1, x_2) = (x_1, x_2)$.

Solution The set $f(S) = S$ is depicted in the first figure in Fig 13.4. Here, both the set

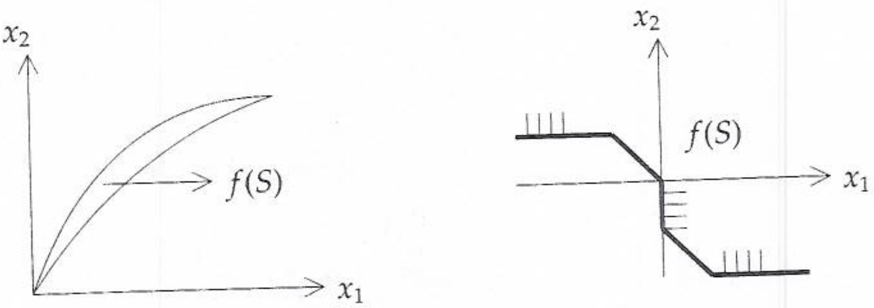


Fig. 13.4.

of weak efficient solutions and the set of efficient solutions are equal to the singleton set $\{(0, 0)\}$.

However, we need to realize that a MOPP can have many efficient solutions. For instance, in Example 13.3.4, the efficient frontier is $\{(1 - \lambda)(1, 3) + \lambda(2, 2) : \lambda \in [0, 1]\} \bigcup \{(3, 1)\}$. The following examples also confirm the statement.

Example 13.3.7 *Let* $S = \{(x_1, x_2) : x_1 \geq 1\} \bigcup \{(x_1, x_2) : x_1 \leq 0,\ x_2 \geq 1\} \bigcup \{(x_1, x_2) : x_1 \leq 0,\ x_2 \leq 1,\ x_1 + x_2 \geq 0\} \bigcup \{(x_1, x_2) : 0 \leq x_1 \leq 1,\ x_1 + x_2 \geq -1\}$.
Consider $\underset{(x_1, x_2) \in S}{Min}\ f(x_1, x_2) = (x_1, x_2)$, *and sketch its efficient frontier.*

Solution The graphical illustration of the set $f(S) = S$ is presented in the second figure in Fig 13.4. The efficient frontier is described by

$$\{(1 - \lambda)(-1, 1) : \lambda \in (0, 1)\} \cup \{(1 - \lambda)(0, -1) + \lambda(1, -2) : \lambda \in [0, 1]\}.$$

Note that the efficient frontier is not a connected set. Moreover, if instead we have considered the MOPP, $Min_{(x_1, x_2) \in S}\ f(x_1, x_2) = (-x_2, x_1)$, then the set of efficient solutions is an empty set.

Example 13.3.8 *Consider the multiobjective programming problem*

$$\begin{array}{ll} Min & f(x_1, x_2) = (-x_1,\ x_1 + x_2^2) \\ subject\ to & \\ & \begin{array}{ll} x_1^2 - x_2 & \le 0 \\ x_1 + 2x_2 & \le 3, \end{array} \end{array}$$

and sketch its efficient frontier.

Solution The feasible set S and its image set $f(S)$ are shown in Fig 13.5.

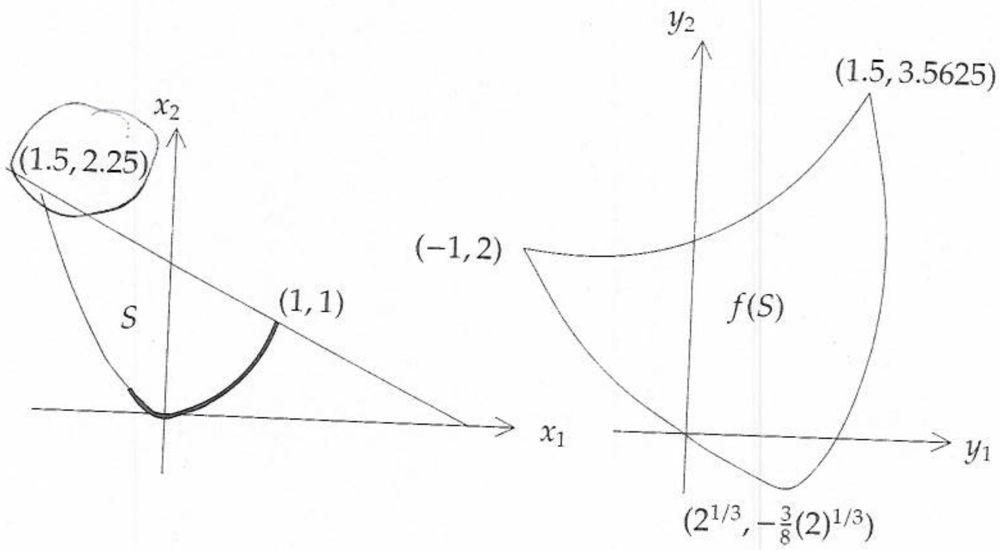


Fig. 13.5.

The efficient frontier is given by $\{(x_1, x_2) : x_1 \in \left[-\frac{1}{2}\sqrt[3]{2},\ 1\right],\ x_2 = x_1^2\}$.

Through the above examples it is clear that the efficient solution of MOPP (13.1) is one for which if one criterion improves than at least one other criterion deteriorate in the sense of minimization. In other words, if value of one criterion decreases at some feasible point than the value of at least one other criterion increases at the same feasible point. However, it is possible that improvement in one criterion is marginal as compared to the deterioration in the other criterion leading to anomalous efficient solutions. To exclude such efficient solutions, Arthur M. Geoffrion [67] introduced a sharper notion of solution called properly efficient solution. The idea behind the proper efficiency is to eliminate unbounded tradeoffs between various criteria.

Definition 13.3.4 (Properly Efficient Solution). $x^\star \in S$ *is called a properly efficient solution of problem (13.1) if* $x^\star$ *is an efficient solution of problem (13.1), and there* *exists a real number $M > 0$ such that for every index i ($i = 1, \ldots, p$) and every $x \in S$ with $f_i(x) < f_i(x^\star)$, there exists at least one index j ($j = 1, \ldots, p$), such that $f_j(x^\star) < f_j(x)$ and*

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

A properly efficient solution is sometimes refer to as *Geoffrion properly efficient solution* or *properly Edgeworth-Pareto optimal solution*. We shall restrict ourself to call it a properly efficient solution.

Remark 13.3.3 *(i) It follows from the definition that every properly efficient solution is an efficient solution of problem (13.1).*
(ii) An efficient solution which is not a properly efficient solution is called an improperly efficient solution.

Definition 13.3.5 (Improperly Efficient Solution). *An efficient solution $x^\star$ is called an improperly efficient solution of problem (13.1) if for every real number $M > 0$, there exist an index i ($i = 1, \ldots, p$) and some $x \in S$ with $f_i(x) < f_i(x^\star)$ such that for every index j ($j = 1, \ldots, p$) with $f_j(x^\star) < f_j(x)$, we have,*

$$\frac{f_i(x^\star) - f_i(x)}{f_j(x) - f_j(x^\star)} > M.$$

The commonsense reasoning says that improperly efficient solutions are not desired as improvement in one criterion comes only with a large sacrifice in the other criterion.

Example 13.3.9 *Consider the linear multiobjective programming problem*

$$\begin{aligned} &\textit{Min} && f(x_1, x_2) = (x_1 + 2x_2,\ 2x_1 - x_2) \\ &\textit{subject to} && \\ & && x_1 + x_2 \leq 1 \\ & && x_1, x_2 \geq 0. \end{aligned}$$

Identify the set of efficient solutions and show that $x^\star = (0, 1)$ is a properly efficient solution of the given MOPP.

Solution Let $y_1 = x_1 + 2x_2, y_2 = 2x_1 - x_2$. Then the feasible set S gets transformed to the set $f(S) = \{(y_1, y_2) : 3y_1 + y_2 \leq 5, y_1 + 2y_2 \geq 0, 2y_1 - y_2 \geq 0\}$, shown in Fig 13.6.

It is clear that the set of efficient solutions is given by $\{(0, x_2) : x_2 \in [0, 1]\}$. Consider an efficient solution $x^\star = (0, 1)$. Then $f(x^\star) = (2, -1)$, and for any $x \in S, x \neq x^\star$, $y_1 < 2$, $y_2 > -1$. Take $i = 1$ and $j = 2$ in the definition of properly efficient solution. It can easily be verified that

$$\frac{f_1(x^\star) - f_1(x)}{f_2(x) - f_2(x^\star)} = \frac{2 - y_1}{y_2 + 1} \leq 2.$$

Thus, (0

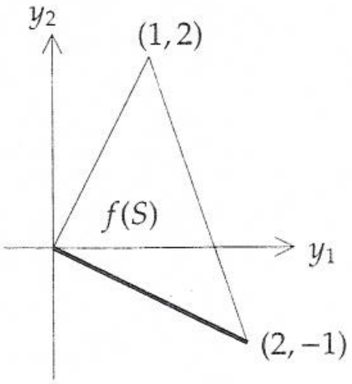


Fig. 13.6.

Example 13.3.10 *Let* $S = \{x = (x_1, x_2) : x_1^2 + x_2^2 \leq 1\}$ *and* $f : S \to \mathbf{R}^2$ *be the identity map,* $f(x_1, x_2) = (x_1, x_2)$. *Consider the MOPP:* $\underset{x \in S}{\text{Min}}\ f(x_1, x_2)$. *Identify the set of efficient solutions. Are* $(0,-1)$ *and* $(-1,0)$ *properly efficient solutions? Justify your answer.*

Solution The set of efficient solutions is $\left\{(x_1, x_2) : x_1 \in [-1, 0],\ x_2 = -\sqrt{1 - x_1^2}\ \right\}$. Consider an efficient solution $x^\star = (0, -1)$. For any natural number n, the point $x_n = (-\frac{1}{n}\sqrt{2n-1}, -1 + \frac{1}{n}) \in S$, and we have,

$$f_1(x_n) = -\frac{1}{n}\sqrt{2n-1} < f_1(x^\star), \quad f_2(x_n) = -1 + \frac{1}{n} > f_2(x^\star).$$

Consequently,

$$\frac{f_1(x^\star) - f_1(x)}{f_2(x) - f_2(x^\star)} = \sqrt{2n-1} \to \infty \text{ as } n \to \infty.$$

Thus $(0,-1)$ is an improperly efficient solution of MOPP. In fact we can show that $(-1,0)$ is also an improperly efficient solution of MOPP while all other efficient solutions are properly efficient solutions of MOPP.

Remark 13.3.4 *There are several variants of properly efficient solutions in literature with more general partial ordering and in more general vector spaces. Geoffrion's properly efficient solution is based on a natural partial ordering in* $\mathbf{R}^p$. *Its simplicity makes it easy to visualize and understand. For this reason we have included it in the text.*

The next section governs the characterization of properly efficient solution which in turn suggests a computational procedure for finding the properly efficient solutions of MOPP (13.1).

13.4 Weighted Sum Approach

For ready reference recall the MOPP

$$\underset{x \in S}{Min}\ f(x) = (f_1(x), \dots, f_p(x)), \tag{13.3}$$

S is the feasible set and $f : S \to \mathbf{R}^p$, $p \geq 2$, is the objective function.

For arbitrary $\lambda_i > 0$ $(i = 1, \dots, p)$, $\sum_{i=1}^{p} \lambda_i = 1$, associate a scalar problem with MOPP (13.3) as

$$\underset{x \in S}{Min} \sum_{i=1}^{p} \lambda_i f_i(x) \tag{13.4}$$

The scalar λ_i $(i = 1, \dots, p)$ can be interpreted as some positive weight or priority assigned to the i-th objective criterion by the decision maker. The sum of all the weights is usually taken to be one to ensure that the criteria are appropriately scaled and relatively placed. We present a hypothetical situation for clarity. Suppose $p = 3$ and $\lambda_1 = \frac{1}{2}$, $\lambda_2 = \frac{1}{3}$, $\lambda_3 = \frac{1}{6}$. It means that the first objective criterion is 3 times important as compared to the third objective criterion and 1.5 times important in comparison with the second objective criterion, while the second objective criterion is twice important in comparison to the third objective criterion. Equivalently, the objectives priorities are set as 3:2:1 by the decision maker.

From now onwards we take,

$$\Lambda = \{\lambda = (\lambda_1, \dots, \lambda_p)^T \in \mathbf{R}^p \, : \, \lambda_i > 0 \ (i = 1, \dots, p), \ \sum_{i=1}^{p} \lambda_i = 1\}.$$

The following two theorems relate the properly efficient solutions of MOPP (13.3) with the optimal solutions of the scalar nonlinear programming problem (13.4).

Theorem 13.4.1 *Let $x^\star \in S$ be an optimal solution of problem (13.4), for fixed $\lambda \in \Lambda$. Then $x^\star$ is a properly efficient solution of problem (13.3).*

Proof. The proof is achieved in two parts. First we prove that $x^\star$ is an efficient solution of problem (13.3), and later, proper efficiency of $x^\star$ for problem (13.3) is worked out. Both parts are proved by contradiction.

Suppose $x^\star$ is not an efficient solution of problem (13.3). Then there exists $x \in S$ and an index i, such that

$$f_j(x) \leq f_j(x^\star) \ \ (j = 1, \dots, p, \ j \neq i), \ \text{ and } \ f_i(x) < f_i(x^\star). \tag{13.5}$$

Since $\lambda_i > 0$ $(i = 1, \dots, p)$, it follows from (13.5) that

$$\sum_{i=1}^{p} \lambda_i f_i(x) < \sum_{i=1}^{p} \lambda_i f_i(x^\star),$$

a contradiction to the optimality of $x^\star$ for problem (13.4).

Next, assume that $x^\star$ is an improperly efficient solution of problem (13.3). Choose $M = (p-1) \max_{i,j=1,\dots,p} \frac{\lambda_j}{\lambda_i}$. Then, for some index i ($i = 1, \dots, p$) and some $x \in S$ with $f_i(x) < f_i(x^\star)$, we obtain,

$$\frac{f_i(x^\star) - f_i(x)}{f_j(x) - f_j(x^\star)} > M \quad (j = 1, \dots, p) \text{ with } f_j(x) > f_j(x^\star).$$

This implies

$$f_i(x^\star) - f_i(x) > M(f_j(x) - f_j(x^\star)) \geq (p-1)\frac{\lambda_j}{\lambda_i}(f_j(x) - f_j(x^\star)) \quad (j = 1, \dots, p,\ j \neq i).$$

Multiplying throughout by $\dfrac{\lambda_i}{p-1}$ and summing over all j ($j = 1, \dots, p,\ j \neq i$), yields

$$\lambda_i(f_i(x^\star) - f_i(x)) > \sum_{j=1,\, j\neq i}^{p} \lambda_j(f_j(x) - f_j(x^\star)).$$

Consequently,

$$\sum_{j=1}^{p} \lambda_j f_j(x) < \sum_{j=1}^{p} \lambda_j f_j(x^\star),$$

again contradicting optimality of $x^\star$ for problem (13.4). The conclusion is now evident from the two contradictions.

□

Theorem 13.4.2 *Let S be a convex set and each f_i ($i = 1, \dots, p$) be a convex function on S. If $x^\star \in S$ is a properly efficient solution of problem (13.3) then there exists $\lambda \in \Lambda$ such that $x^\star$ is an optimal solution of problem (13.4) with this λ.*

We skip the proof here as it requires the knowledge of the 'separation theorem of convex sets'. However, we recommend the readers to the research article of Geoffrion [67] for the detailed proof.

Example 13.4.1 *Consider the two criteria linear MOPP*

$$\begin{aligned} &\mathit{Min} && f(x_1, x_2) = (-2x_1 + x_2,\ x_1 - 2x_2) \\ &\textit{subject to} && \\ & && -x_1 + x_2 \leq 2 \\ & && 2x_1 - x_2 \leq 10 \\ & && x_1 + x_2 \leq 8 \end{aligned} \tag{13.6}$$

Construct the weighted sum LPP and analyze the same in the light of Theorem 13.4.1 and Theorem 13.4.2.

Solution Let $y_1 = -2x_1 + x_2,\ y_2 = x_1 - 2x_2$. The image set $f(S)$ is given by $\{(y_1, y_2) : y_1 - y_2 \leq 6,\ y_1 \geq -10,\ y_1 + y_2 \geq -8,\ 2y_1 + y_2 \leq 0,\ y_1 + 2y_2 \leq 0\}$.
Fig 13.7 shows the feasible set S and the set $f(S)$.

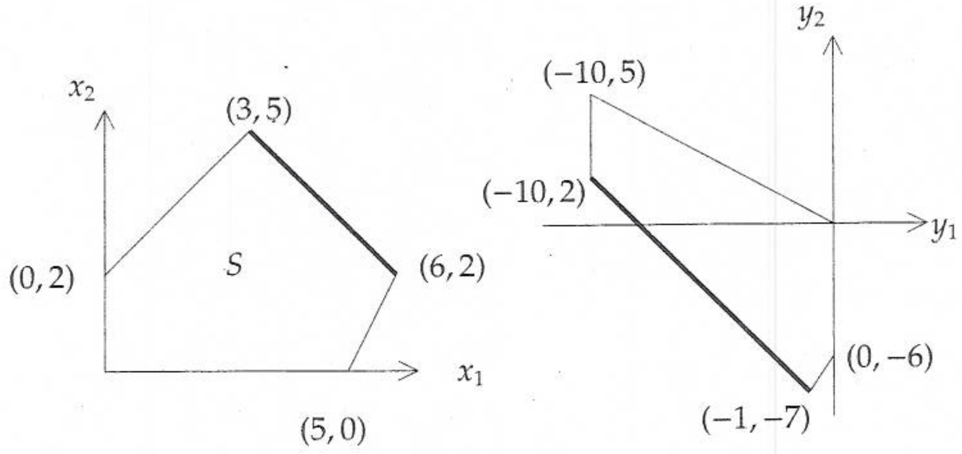


Fig. 13.7.

The efficient frontier is marked by bold line in the first figure in Fig 13.7, and is described by $\{\mu(3,5) + (1-\mu)(6,2) : \mu \in [0,1]\}$.

We formulate the weighted sum LPP with weights $(\lambda, 1-\lambda)$, $\lambda \in (0,1)$

$$\begin{array}{ll} \text{Min} & (1-3\lambda)x_1 + (3\lambda - 2)x_2 \\ \text{subject to} & \\ & -x_1 + x_2 \leq 2 \\ & 2x_1 - x_2 \leq 10 \\ & x_1 + x_2 \leq 8 \\ & x_1, x_2 \geq 0. \end{array} \tag{13.7}$$

For $\lambda = \frac{1}{3}$, the optimal solution of (13.7) is (3,5), while for $\lambda = \frac{2}{3}$, the optimal solution is (6,2). For $\lambda = \frac{1}{2}$, the optimal solutions of the weighted linear program (13.7) generate the entire line segment connecting the points (3,5) and (6,2). Thus, on account of Theorem 13.4.1 and Theorem 13.4.2, all efficient solutions are properly efficient solutions of problem (13.7).

We next present an example of nonlinear convex MOPP to illustrate the weighted sum approach.

Example 13.4.2 *Consider the two criteria nonlinear MOPP*

$$\begin{array}{ll} \text{Min} & f(x_1, x_2) = (x_1^2 + x_2^2, (x_1 - 3)^2 + (x_2 - 3)^2) \\ \text{subject to} & \\ & (x_1 - 3)^2 + x_2^2 \le 9 \\ & 0 \le x_1 \le 3 \\ & 0 \le x_2 \le 2. \end{array} \tag{13.8}$$

Construct the weighted sum scalar problem, and analyze the same in the light of Theorem 13.4.1 and Theorem 13.4.2.

Solution The feasible set S and the corresponding image set $f(S)$ are shown in Fig 13.8. The efficient frontier of the problem is marked in bold in the first figure in Fig 13.8, and it is described by

$$\{(x_1, x_2) : x_1 = x_2, 0 \le x_1 \le 2\} \cup \{(x_1, 2) : 2 \le x_1 \le 3\}.$$

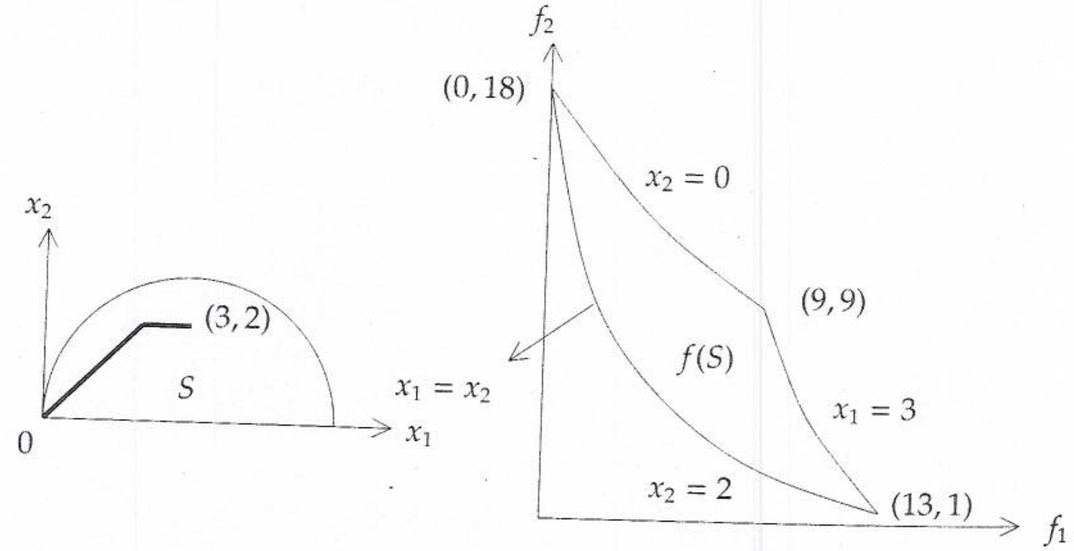


Fig. 13.8.

For weight vector $(\lambda, 1 - \lambda)$, $\lambda \in (0, 1)$, the weighted sum nonlinear convex programming problem is

$$\begin{array}{ll} \text{Min} & f(x_1, x_2) = \lambda(x_1^2 + x_2^2) + (1 - \lambda)((x_1 - 3)^2 + (x_2 - 3)^2) \\ \text{subject to} & \\ & (x_1 - 3)^2 + x_2^2 \le 9 \\ & 0 \le x_1 \le 3 \\ & 0 \le x_2 \le 2. \end{array} \tag{13.9}$$

Writing the KKT conditions for problem (13.9), we get the following system of ations

$$\begin{aligned}
2x_1 + 6\lambda + 2\mu x_1 - 6\mu + \nu - s &= 6\\
2x_2 + 6\lambda + 2\mu x_2 + \eta - t &= 6\\
\mu((x_1-3)^2 + x_2^2 - 9) &= 0\\
\nu(x_1 - 3) &= 0\\
\eta(x_2 - 2) &= 0\\
sx_1 &= 0\\
tx_2 &= 0\\
\mu,\ \nu,\ \eta,\ s,\ t &\geq 0.
\end{aligned}$$

After simple calculations it can be shown that when $\lambda \in [\frac{1}{3}, 1)$, all the efficient olutions $\{(x_1, x_2) : x_1 = x_2,\ 0 < x_1 \leq 2\}$ of (13.8) can be generated, while for $\lambda \in (0, \frac{1}{3})$, he resultant set, $\{(x_1, 2) : 2 < x_1 < 3\}$, of efficient solutions of (13.8) is obtained. ccording to Theorem 13.4.1 and Theorem 13.4.2, these efficient solutions constitute he set of properly efficient solutions of the given MOPP (13.8).

The following example of a nonlinear convex MOPP is included to show that the ondition of properly efficient solution in Theorem 13.4.2 can not be replaced by the fficient solution.

Example 13.4.3 *Consider the convex nonlinear MOPP in* $\mathbf{R}$

$$\begin{aligned}
&\mathit{Min} \quad f(x) = (-x^2,\ x^3)\\
&\mathit{subject\ to} \quad x \geq 0.
\end{aligned} \tag{13.10}$$

Show that the Theorem 13.4.2 fails to hold at $x^\star = 0$.

Solution Clearly $x^\star = 0$ is an efficient solution of (13.10). But $x^\star$ is an improperly efficient solution because for any $M > 0$ there exists a feasible $x > 0$ such that $f_1(x) < f_1(x^\star)$ and $f_2(x) > f_2(x^\star)$ but, $\dfrac{f_1(x^\star) - f_1(x)}{f_2(x) - f_2(x^\star)} > M$.

Now suppose we construct a weighted scalar problem of (13.10) as

$$\begin{aligned}
&\text{Min} \quad \lambda_1(-x^2) + \lambda_2 x^3\\
&\text{subject to} \quad x \geq 0,
\end{aligned} \tag{13.11}$$

with $\lambda_1 > 0,\ \lambda_2 > 0$. Then it can be shown that $x^\star = 0$ is not an optimal solution of (13.11), for any $\lambda_1, \lambda_2 > 0$. For if, $x^\star$ is its optimal solution then we must have, $\lambda_1(-x^2) + \lambda_2 x^3 \geq 0,\ \forall\, x \geq 0$ yielding $-\lambda_1 + \lambda_2 x \geq 0,\ \forall\, x > 0$. The latter is possible only if $\lambda_1 = 0$.

Remark 13.4.1 *(i) The approach described above to study MOPP (13.3) is known as a* *has been converted into a scalar nonlinear programming problem (13.4) with the help of weighted mean of the objective functions f_i. $\lambda \in \Lambda$ in problem (13.4) is appropriately refer to as a weight vector.*

(ii) In view of the Theorem 13.4.2, the weighted sum approach seems to be suitable for convex MOPP (13.3). However, for a general nonconvex MOPP, it is possible that not every properly efficient solution can be determined using this approach. Despite this, the approach works well for a large class of generalized convex MOPP with convex efficient frontier.

(iii) The above two theorems characterize properly efficient solutions of MOPP (13.3). Similar characterizations can be derived for weak efficient solutions and efficient solutions of MOPP (13.3). The main principle behind these characterizations is same. It relates various solutions of MOPP with optimal solutions of suitable scalar nonlinear programming problem. The difference emerges in the choice of the weight vector $\lambda \in \mathbf{R}^p$. We summarize the results in the following two theorems.

Theorem 13.4.3 *Let S be a convex set and each f_i $(i = 1, \ldots, p)$ be a convex function on S. Then $x^\star \in S$ is a weak efficient solution of problem (13.3) if and only if there exist $\lambda_i \geq 0$ $(i = 1, \ldots, p)$, $\sum_{i=1}^{p} \lambda_i = 1$, such that $x^\star$ is an optimal solution of scalar problem*

$$\underset{x \in S}{Min} \sum_{i=1}^{p} \lambda_i f_i(x). \tag{13.12}$$

Theorem 13.4.4 *Let $x^\star \in S$ be an optimal solution of problem (13.4), for fixed $\lambda \in \Lambda$. Then $x^\star$ is an efficient solution of problem (13.3). Conversely, let S be a convex set and each f_i $(i = 1, \ldots, p)$ be a convex function on S. If $x^\star \in S$ is an efficient solution of problem (13.3) then there exist $\lambda_i \geq 0$ $(i = 1, \ldots, p)$, $\sum_{i=1}^{p} \lambda_i = 1$, such that $x^\star$ is an optimal solution of problem (13.12).*

Recall example 13.4.3, $x^\star = 0$ is an efficient solution of problem (13.10), and if we take $\lambda_1 = 0$, $\lambda_2 = 1$, then $x^\star$ is an optimal solution of the weighted scalar problem (13.12).

The thrust of the weighted sum approach (sometimes also refer as a *scalarization* of a multiobjective programming problem) is to convert a MOPP into a single objective programming problem, thereby making it convenient to handle the problem. The approach is simple and most widely used to solve MOPP. Of course there are some obvious difficulties in choosing the right proportion of weights. The weights reflect how much importance the decision maker wishes to give to various objective criteria. But in many situations it is hard to estimate the precise numerical values of the weights. For example, in an aircraft design problem, if one objective is to decrease the manufacturing cost of the aircraft and the other is to increase the safety of the passenger, it is difficult to assign the numeric values to the weights simply because human life is precious and can not be quantified. Consider another simple scenario where the government has to provide more basic facilities to the weaker sections of the society at a low cost and at the same time wishes to decrease the subsidy on various goods. The conflict between the social responsibilities and economic development is difficult to be accurately measured by some weight vector. There is no mathematically designed mechanism to compute the weight vector, it completely rely on the knowledge of the decision maker, his preferences, and the set up of the problem. Furthermore, different weight vectors need not necessarily generate different properly efficient solutions of MOPP. For instance, in Example 13.4.1, $\lambda = \frac{1}{3}$ and $\lambda = \frac{2}{5}$ give the same optimal solution $(3, 5)$. This generally leads to wastage of search efforts in finding the properly efficient solutions of MOPP by weighted sum approach. The root cause behind this is that the mapping depicting the relationship between the weight vectors and the properly efficient solutions of MOPP is usually not known.

The practical limitations of the weighted sum approach motivate us to look for another approach which inherit the basic ideas of MOPP and at the same time computationally easy to implement. The search land us to 'goal programming'. We shall focus on this concept in the forthcoming sections.

13.5 Formulation of Goal Programming Problem

Rather than asking to provide the numerical value of weight for each objective criterion in MOPP (13.1), the decision maker is asked to rank the objectives according to their perceived importance. He is also asked to set up the aspired target values for each objective criterion. For instance, suppose $5x_1 + 2x_2 + 4x_3$ represents the profit function and $10x_1 + 3x_2 + 5x_3$ represents the maintenance cost of the finished goods inventory of some organization. An interactive discussion with the decision maker in an organization reveals that his first priority is to maximize profit and then to minimize the maintenance cost. Further, he also aspire to obtain a profit of Rs. 5000 and curtail the maintenance cost upto Rs. 1000. In this way, a dialogue with the decision maker reveals additional information of the aspired values for each objective criterion that can be used in formulation of a problem in a more constructive way.

The method of *goal programming* (GP) is based on this idea. It actually consists of formulating an optimization problem in such a manner that ensures that the objective criteria come close to the specified aspiration levels in order of priorities set up by the decision maker. It is worth to note that goal programming aims at satisfaction of the goals rather than exact achievement of the goals. Before proceeding further with our discussion we pause here to define related terminologies.

Definition 13.5.1 (Aspiration Level). *Aspiration level is the numerical value specified by the decision maker that reflects his desire or satisfactory level with regard to the objective function under consideration.*

Definition 13.5.2 (Goal). *An objective function along with its aspiration level is termed as goal.*

Definition 13.5.3 (Goal Deviation). *The difference between what we actually achieve and what we desire to achieve is called goal deviation. If the goal deviation is positive it reflects overachievement of a goal as the actual achieved value is more than the set up aspiration level. On the other hand negative value of the goal deviation indicates underachievement of the goal as the aspired level is more than what we could actually manage to achieve.*

For instance, suppose profit of an organization is described by a function $5x_1 + 2x_2 + 4x_3$. Now the decision maker wishes to attain a profit of at least Rs. 5000. So, the aspiration level of the decision maker is 5000 with regard to this objective function and the goal is described by an inequality $5x_1 + 2x_2 + 4x_3 \geq 5000$. If for some feasible vector $x^\star = (x_1^\star, x_2^\star, x_3^\star)$, $5x_1^\star + 2x_2^\star + 4x_3^\star > 5000$, then it shows that by taking the decision $x^\star$ the decision maker can achieve more profit than what he aspired. Whereas if for all feasible $x = (x_1, x_2, x_3)$, $5x_1 + 2x_2 + 4x_3 < 5000$, then no decision by the decision maker can help him to attain the aspired goal. This situation represents underachievement of the goal.

One obvious question arise here. How does the decision maker arrive at a figure of Rs. 5000? Arguably setting up too high value or too low value for the aspiration level is not advisable. First of all we assume that the decision maker is rationale and knowledgeable. Secondly, in a MOPP (13.1) with p objective criteria, one can solve p individual scalar programming problems

$$\underset{x \in S}{Min}\ f_i(x) \qquad (i = 1, \ldots, p). \tag{13.13}$$

Each of these problems, being single objective nonlinear constrained programming problem, can be solved by the techniques described in earlier chapters. Once the optimal values, say $f_i(x^{(i)\star})$ $(i = 1, \ldots, p)$, are known, they can be used as aspiration levels for the respective objective functions.

We now turn our attention to formulate a mathematical model of goal programming problem.

Suppose the i-th objective function is described by $f_i(x)$, $x \in \mathbf{R}^n$, and its aspiration level is specified by $v_i \in \mathbf{R}$. The possible form of the i-th goal is

$$\begin{aligned} \text{either} \quad & f_i(x) \leq v_i\,, \\ \text{or} \quad & f_i(x) \geq v_i\,, \\ \text{or} \quad & f_i(x) = v_i. \end{aligned}$$

By introducing two additional variables, $d_i^- \geq 0$, $d_i^+ \geq 0$, we can transform the above relations into the following equation

$$f_i(x) + d_i^- - d_i^+ = v_i. \tag{13.14}$$

important to note here that if the goal is of the form $f_i(x) \le v_i$, then d_i^+ is the ble variable in equation (13.14). Because if $d_i^- = 0$ and $d_i^+ > 0$ then equation implies $f_i(x) = v_i + d_i^+ > v_i$, thereby not satisfying the set up goal $f_i(x) \le v_i$. f $d_i^- > 0$ and $d_i^+ = 0$, then we get $f_i(x) = v_i - d_i^- < v_i$, consequently, achieving up goal. Thus, $f_i(x) \le v_i$ implies d_i^+ is an undesirable variable and we need to e it. Similarly, $f_i(x) \ge v_i$, gives d_i^- as an undesirable variable, and $f_i(x) = v_i$ yields as an undesirable expression. In all situations, we first identify the undesirable /expression in the goal and then attempt to minimize the same. The variable d_i^- negative deviational variable while d_i^+ is called the positive deviational variable ous reasons.

ll the MOPP

$$\underset{x \in S}{Min}\ f(x) = (f_1(x), \dots, f_p(x)). \tag{13.15}$$

ose the feasible set S is described by the m inequality constraints $g_i(x) \le 0$ $\dots, m$). Some of the constraints could be $\ge$ or $=$ types.

e the form of the goal function and the constraint function are alike, both are in m of mathematical inequalities or equations, goal programming treats the con- functions also as goals. Thereby the constraints too are converted into equations egative and positive deviational variables. In other words, constraint $g_i(x) \le 0$ is ed as $g_i(x) + d_i^- - d_i^+ = 0,\ d_i^-, d_i^+ \ge 0$.

olve an optimization problem, we must ensure feasibility of the problem. This is lished by assigning first priority to the goals representing the actual constraint ns. Once we attain this priority, we have a feasible solution of the problem in The actual constraints are therefore termed as *rigid goals or hard goals* whereas als representing the original objective functions are called *soft goals*.

marizing the above discussion we conclude the following procedure for the for- on of the goal programming problem (GPP).

ecify the aspiration level for each objective function.

up the goals and convert them into equations by using negative and positive viational variables.

eat the constraints present in the problem as goals and convert them into equations using negative and positive deviational variables.

sign first priority to the hard constraints and rank all other goals according to eir importance specified by the decision maker.

entify the appropriate undesirable deviational variables or expressions involving viational variables and make an attempt to minimize them in order of the specified iorities.

e task in point 5 is accomplished by constructing a suitable multiobjective pro- amming problem and by defining an appropriate ranking mechanism.

e following example illustrates the above procedure.

ion along with its aspiration
erence between what we actuall
ation. If the goal deviation is
al achieved value is more than
value of the goal deviation i
l is more than what we could
is described by a function $5x_1$
profit of at least Rs. 5000.
regard to this objective funct
$4x_3 \ge 5000$. If for some feasibl
it shows that by taking the
an what he aspired. Whereas i
en no decision by the decisio
ation represents underachieve
lecision maker arrive at a figu
w value for the aspiration lev
naker is rationale and knowle
ria, one can solve p individua
$\dots, p)$.
nonlinear constrained progra
in earlier chapters. Once the
can be used as aspiration le
ematical model of goal progra
by $f_i(x)$, $x \in \mathbf{R}^n$, and its asp
i-th goal is
$d_i^+ \ge 0$, we can transform the

Example 13.5.1 *Formulate the following three objective LPP as a goal programming problem.*

$$\begin{array}{ll} \textit{Min} & (2x_1 - x_2, 4x_1 - 5x_2, -x_1) \\ \textit{subject to} & \\ & \begin{array}{rl} 4x_1 + 5x_2 & \leq 20 \\ 3x_1 + 2x_2 & \leq 12 \\ x_1, x_2 & \geq 0. \end{array} \end{array}$$

Solution Suppose the decision maker give first priority to the second objective criterion, second priority to the third objective criterion and third priority to the first objective criterion. He wish to keep the first, second and third priorities objective values below $8, -2, 1$, respectively. So, the goals in order of their importance are given by

$$\begin{array}{rl} 4x_1 - 5x_2 & \leq 8 \\ x_1 & \geq 2 \\ 2x_1 - x_2 & \leq 1. \end{array}$$

Introducing the deviational variables both in the actual goals and the two constraints (hard goals), we get the following system of linear equations

$$\begin{array}{rl} 4x_1 + 5x_2 + d_1^- - d_1^+ & = 20 \\ 3x_1 + 2x_2 + d_2^- - d_2^+ & = 12 \\ 4x_1 - 5x_2 + d_3^- - d_3^+ & = 8 \\ x_1 + d_4^- - d_4^+ & = 2 \\ 2x_1 - x_2 + d_5^- - d_5^+ & = 1 \\ x_j,\ d_i^-,\ d_i^+ & \geq 0,\ (j = 1,2)\ (i = 1, \ldots, 5). \end{array} \tag{13.16}$$

The undesirable variables are, d_1^+, d_2^+, d_3^+, d_4^-, d_5^+, respectively.

Our next task is to construct an objective function depicting the priorities of the decision maker. The first rank is reserved for the hard constraints followed by the second objective function, third objective function and then the first objective function, respectively. Therefore, we make an attempt to minimize $(d_1^+ + d_2^+,\ d_3^+,\ d_4^-,\ d_5^+)$ in this order only. Thus the GPP becomes

$$\begin{array}{ll} \text{Min} & (d_1^+ + d_2^+,\ d_3^+,\ d_4^-,\ d_5^+) \\ \text{subject to} & (13.16). \end{array}$$

Note that we have yet to specify the meaning of 'minimization in order'.

Remark 13.5.1 *GPP is a MOPP in which the objective criteria in terms of the deviational variables are ranked according to their importance and the minimization process has to take care of the preassigned order. We simply can not bypass this order. Obviously this ranking is different from the concepts of weak efficiency, efficiency or proper efficiency, as in the latter definitions the order of the objective criterion is immaterial.*

owing three objective LPP as a go

$2x_1 - x_2,\ 4x_1 - 5x_2,\ -x_1)$

$$\begin{aligned} 4x_1 + 5x_2 &\le 20 \\ 3x_1 + 2x_2 &\le 12 \\ x_1, x_2 &\ge 0. \end{aligned}$$

ive first priority to the second objec
riterion and third priority to the
ond and third priorities objective
der of their importance are given

$$\begin{aligned} -5x_2 &\le 8 \\ x_1 &\ge 2 \\ -x_2 &\le 1. \end{aligned}$$

oth in the actual goals and the tw
of linear equations

$$\begin{aligned} &= 20 \\ &= 12 \\ &= 8 \\ &2 \\ &1 \\ &0, \ (j = 1,2)\ (i = 1,\ldots,5). \end{aligned}$$

$_3^+,\ d_4^-,\ d_5^+$, respectively.
tive function depicting the priori
or the hard constraints followed
tion and then the first objecti
t to minimize $(d_1^+ + d_2^+,\ d_3^+,\ d_4^-,$

$d_2^+,\ d_3^+,\ d_4^-,\ d_5^+)$
(13.16).

ing of 'minimization in order'.

he objective criteria in terms of
importance and the minimizatio
simply can not bypass this ord
s of weak efficiency, efficiency
of the objective criterion is imp

13.5.4 (Lexicographic Minimum Vector). *A vector $w^{(1)} \in \mathbf{R}_+^p$ is said to to another vector $w^{(2)} \in \mathbf{R}_+^p$ if $w_i^{(1)} = w_i^{(2)}$ $(i = 1,\ldots,k-1)$ and $w_k^{(1)} < w_k^{(2)}$, for $1,\ldots,p)$. For example, if $w^{(1)} = (0,12,3,17,25)$ and $w^{(2)} = (0,12,5,11,27)$. preferred to $w^{(2)}$. If, in some set $\mathfrak{B} \subseteq \mathbf{R}_+^p$, there is no vector preferred to $w^\star$ is called the lexicographic minimum vector in the set $\mathfrak{B}$.*

th to note here that in choosing the lexicographic minimum among the ndidate vectors, we search a vector with minimum first component. If this ique, we stop, else we continue to choose the minimum second component ering the minimum first component. Repeat this procedure till we get a or with first k minimum components, k can be equal to p also. The procedure t similar to searching an english word in a dictionary. The minimization in en in the sense of lexicographic minimum (Lexi-Min).

general model of GPP is described as follows

$$\begin{aligned} &\text{Lexi-Min} \quad (F_1(d^-,d^+),\ldots,F_K(d^-,d^+)) \\ &\text{subject to} \end{aligned}$$

$$\begin{aligned} f_i(x) + d_i^- - d_i^+ &= v_i \quad (i = 1,\ldots,p) \\ g_j(x) + d_j^- - d_j^+ &= 0 \quad (j = 1,\ldots,m) \\ d_i^-, d_j^-, d_i^+, d_j^+ &\ge 0 \quad (i = 1,\ldots,p)\ (j = 1,\ldots,m), \end{aligned} \tag{13.17}$$

K is the number of priorities specified by the decision maker. Remember the ity is reserved for the hard goals. Also, $F_r(d^-,d^+)$ $(r = 1,\ldots,K)$, are linear of the deviational variables.

he functions f_i and g_j are linear functions of the decision variable x then the 17) is called *linear goal programming problem* (LGPP). In the next section we he solution methodologies to solve LGPPs.

olution Methodologies for Linear Goal Programming Problems

fly present two traditionally known techniques to solve LGPPs. If the decision x involved in LGPP belongs to $\mathbf{R}^2$ then the problem can be solved by graph- nique. We illustrate this technique on Example 13.5.1. For convenience, we luce the example.

Example 13.6.1 *Solve* *Lexi-Min* $(d_1^+ + d_2^+,\ d_3^+,\ d_4^-,\ d_5^+)$
subject to

$$
\begin{aligned}
G_1 &: & 4x_1 + 5x_2 + d_1^- - d_1^+ &= 20\\
G_2 &: & 3x_1 + 2x_2 + d_2^- - d_2^+ &= 12\\
G_3 &: & 4x_1 - 5x_2 + d_3^- - d_3^+ &= 8\\
G_4 &: & x_1 + d_4^- - d_4^+ &= 2\\
G_5 &: & 2x_1 - x_2 + d_5^- - d_5^+ &= 1\\
& & x_j,\ d_i^-,\ d_i^+ \geq 0\ (j = 1, 2) & \quad (i = 1, \ldots, 5).
\end{aligned}
\tag{13.18}
$$

Solution Ignoring the deviational variables, the five goals are plotted as straight lines in Fig 13.9.

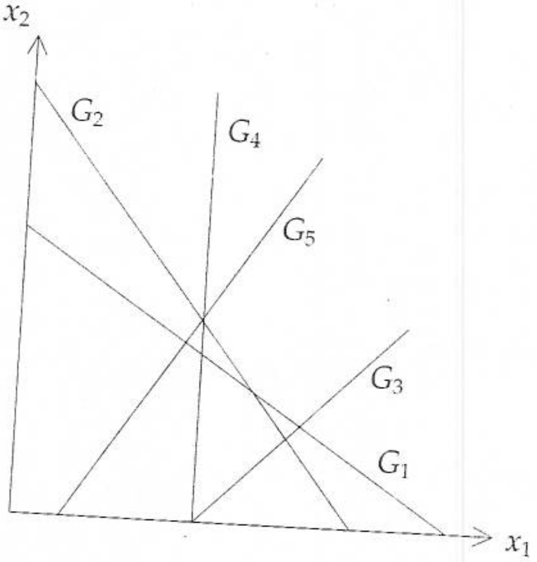


Fig. 13.9.

Concentrate on the first objective function, Min $(d_1^+ + d_2^+)$. The corresponding goals are G_1, G_2. We identify the region in the first figure in Fig 13.10 depicting the optimal solutions of the first priority objective criteria. Observe that in the shaded region $d_i^+ = 0$ $(i = 1, 2)$, thereby yielding zero optimal value of the objective function.

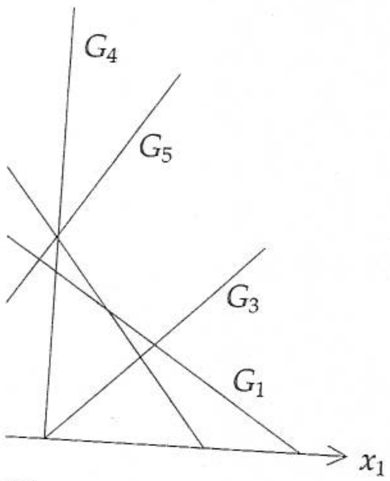
Next, move to minimize the second objective d_3^+, with corresponding goal G_3, over the optimal solution space of the first objective function (i.e. the shaded region in the first figure in Fig 13.10). Optimal value of this objective is zero and the new optimal solution space is shown in the second graph in Fig 13.10. This region will act as the feasible region for the third objective function of the GPP.

We repeat the above procedure with the third objective to get its optimal value zero in the new optimal solution space shown in Fig 13.11. This is the feasible region for the successive objective function.

$(d_1^+ + d_2^+,\ d_3^+,\ d_4^-,\ d_5^+)$

$$
\begin{aligned}
4x_1 + 5x_2 + d_1^- - d_1^+ &= 20 \\
3x_1 + 2x_2 + d_2^- - d_2^+ &= 12 \\
4x_1 - 5x_2 + d_3^- - d_3^+ &= 8 \\
x_1 + d_4^- - d_4^+ &= 2 \\
2x_1 - x_2 + d_5^- - d_5^+ &= 1 \\
x_j,\ d_i^-,\ d_i^+ \geq 0\ (j = 1, 2) &\quad (i = 1,
\end{aligned}
$$

les, the five goals are plotted as st



3.9.

n, Min $(d_1^+ + d_2^+)$. The correspond
; figure in Fig 13.10 depicting th
a. Observe that in the shaded re
e of the objective function.
ive d_3^+, with corresponding goal
ve function (i.e. the shaded regi
is objective is zero and the new
in Fig 13.10. This region will a
of the GPP.
rd objective to get its optimal v
13.11. This is the feasible regio

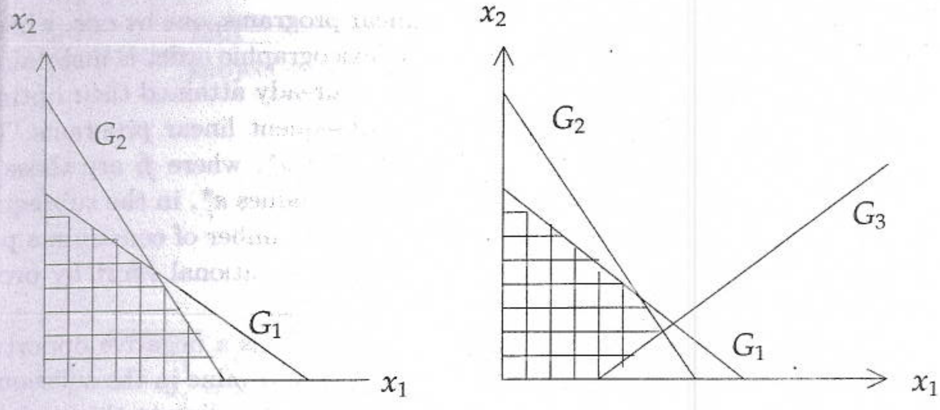


Fig. 13.10.

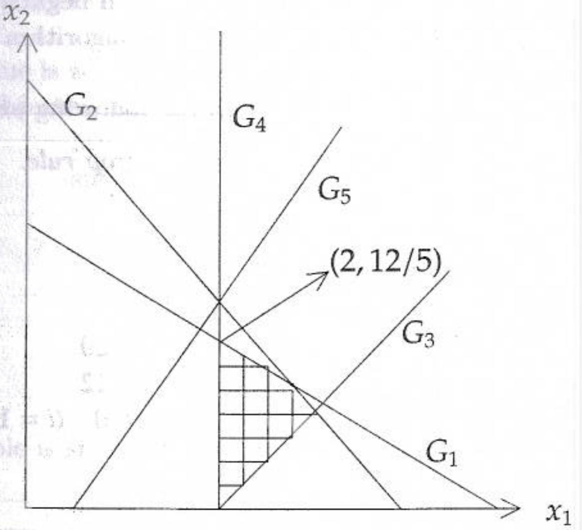


Fig. 13.11.

e final priority goal G_5 is outside the current feasible region, we move the objec-
ne G_5 parallel to itself in the upward direction to meet the feasible region. The
graphic optimal solution is $x^* = (2, 2.4)$ and the optimal achievement value of the
ive function of (13.17) is $(0, 0, 0, 0.6)$. The positive value 0.6 in the end indicates
ne aspired value 'below 1' for the objective criterion $2x_1 - x_2$ can not be achieved.
we can not bring down its value below 1.6. But since this objective was at least
o the decision maker, so, the compromise is reasonably acceptable.
higher dimension set up we can not rely on the graphical technique. So, we switch
o the *sequential goal programming technique* wherein the ideas of the graphical
que are preserved.

The scheme involves solving K number of linear programs, one by one, where K is the number of priorities, in such a manner that the lexicographic order is maintained and at the same time the objective functions which have already attained their optimal values should not deteriorate in their values in the subsequent linear programs. The latter is ensured by adding additional constraints, $f_i(x) = a_i^\star$, where f_i are those objective functions which have already achieved their optimal values $a_i^\star$, in the subsequent linear programs. This may cause substantial increase in the number of constraints particulary in a large size GPPs. Luckily we can reduce the computational effort by progressively dropping some variables according to the following rule.

Column Drop Rule. Any nonbasic variable that has a negative opportunity cost $z_j - c_j$ in the optimal table of a LPP can be assigned zero value in the subsequent linear programming problems and therefore the column corresponding to this variable can be dropped from the subsequent linear programming problems.

Implicitly the rule states that if a nonbasic variable with negative opportunity cost $z_j - c_j$ is introduced in the basis at the later stages of the algorithm it will degrade the solution in the Lexi-Min order.

The algorithm can easily be understood through the following example.

Example 13.6.2 *Solve the GPP (13.18) by the column drop rule.*

Solution To begin with, we solve the following LPP

$$\text{Min} \quad d_1^+ + d_2^+$$

subject to

$$\begin{aligned} 4x_1 + 5x_2 + d_1^- - d_1^+ &= 20 \\ 3x_1 + 2x_2 + d_2^- - d_2^+ &= 12 \\ x_i, d_i^-, d_i^+ &\geq 0 \quad (i = 1,2). \end{aligned}$$

The optimal table is given by

	x_1	x_2	d_1^+	d_2^+
$d_1^- = 20$	4	5	-1	0
$d_2^- = 12$	3	2	0	-1
$z_j - c_j$	0	0	-1	-1
			✓	✓

'✓' denotes that these non basic variables have negative $z_j - c_j$, hence according to the column drop rule, the marked columns can be dropped from the subsequent iterations. Observe that the optimal value of the LPP is zero.

The next level LPP and its optimal solution are respectively given by

Min $\quad d_3^+$

subject to

$$\begin{aligned} 4x_1 + 5x_2 + d_1^- &= 20 \\ 3x_1 + 2x_2 + d_2^- &= 12 \\ 4x_1 - 5x_2 + d_3^- - d_3^+ &= 8 \\ x_j, d_i^-, d_i^+ &\geq 0, \quad (j = 1,2)\ (i = 1,2,3). \end{aligned}$$

	x_1	x_2	d_3^+
$d_1^- = 20$	4	5	0
$d_2^- = 12$	3	2	0
$d_3^- = 8$	4	-5	-1
$z_j - c_j$	0	0	-1
			✓

The optimal value is zero. We then move on to construct the LPP corresponding to the third priority objective function.

Min $\quad d_4^-$

subject to

$$\begin{aligned} 4x_1 + 5x_2 + d_1^- &= 20 \\ 3x_1 + 2x_2 + d_2^- &= 12 \\ 4x_1 - 5x_2 + d_3^- &= 8 \\ x_1 + d_4^- - d_4^+ &= 2 \\ x_j, d_i^-, d_i^+ &\geq 0 \quad (j = 1,2)\ (i = 1,\ldots,4). \end{aligned}$$

The optimal table is shown below.

	x_2	d_4^-	d_4^+
$d_1^- = 12$	5	-4	4
$d_2^- = 6$	2	-3	3
$d_3^- = 0$	-5	-4	4
$x_1 = 2$	0	1	-1
$z_j - c_j$	0	-1	0
		✓	

The final priority criterion is represented by the following LPP, and its optimal solution table is subsequently generated.

Numerous research articles are devoted to study the KKT type optimality conditions and the duality results for nonlinear MOPP. We cite here only those contributions which had been novel in their approach, like, the works of P. Wolfe [167], Weir and Mond [163], Chankong and Haimes [33], Charnes and Cooper [34], Egudo [52], Jeyakumar and Mond [85], Singh [144], Hanson [75], to name a few. While citing the above references we have restricted ourself to differentiable settings. One can obviously go deep to find many interesting results for MOPP in nonsmooth versions too.

Evolutionary algorithms have been successfully used to solve MOPPs. The primary reason for their success is the ability of the evolutionary algorithms to generate several solutions in a single simulation run. A brief description of few such algorithms is given later in this book. The beginners in the field of multiobjective optimization with evolutionary algorithms can refer to a good text by Deb [46].

Extensive research in last many decades resulted in various techniques, like, multiattribute utility analysis, normal boundary intersection method, multiobjective heuristic algorithms, among others, for computing efficient frontiers of extremely complex engineering design problems and decision making problems. However, none of these techniques are perfect and selecting among them depends on the requirements of the problem. For more details on the solution methodologies, we suggest a website dedicated to the multiobjective programming: http://www.lania.mx/~ccoello/EMOO.

In this Chapter we have focussed on MOPP in finite dimensional real space. However, vector optimization problems set up in very abstract spaces have been investigated in detail in last many years. Many excellent texts are available to get the insight of the otherwise extremely involved subject. For instance, one can refer to, Borwein [25], Jahn [82], Luc [105], Luenberger [106], Steuer [148], Sawaragi et al. [140]. Furthermore, numerous research articles can be found dealing with nonsmooth nonlinear MOPPs.

13.8 Exercises

13.1 *Consider the linear MOPP*

$$\begin{array}{ll} \textit{Max} & f(x) = \{x_1 + 5x_2,\ x_1\} \\ \textit{subject to} & \\ & x_1 - 2x_2 \le 2 \\ & x_1 + 2x_2 \le 12 \\ & 2x_1 + x_2 \le 9 \\ & x_1, x_2 \ge 0. \end{array}$$

(i) Show graphically that the optimal solutions taken separately for two linear programming problems, each with a single objective function, do not coincide.

(ii) Determine graphically whether each of the given feasible solution is an efficient solution of the problem: $(2,0)$, $(4,7)$, $(3,3)$, $(2,5)$, $(2,2)$, $(0,6)$.
(iii) Identify the efficient frontier of this model in an objective values space.

13.2 *Do the above exercise for the linear MOPP*

$$\begin{array}{ll} \text{Min} & \{5x_1 - x_2,\ x_1 + 4x_2\} \\ \text{subject to} & \end{array}$$

$$\begin{aligned} -5x_1 + 2x_2 &\leq 10 \\ x_1 + x_2 &\geq 3 \\ x_1 + 2x_2 &\geq 4 \\ x_1, x_2 &\geq 0, \end{aligned}$$

and points $(4,0)$, $(2,1)$, $(3,3)$, $(1,2)$, $(5,0)$, $(0,0)$.

13.3 *Let* $S = \{(x_1, x_2, x_3) \in \mathbf{R}^3 : x_1^2 \leq x_2^3\}$. *Define a function* $f : S \to \mathbf{R}^2$ *by*

$$f(x_1, x_2, x_3) = (x_3^2 - x_1,\ x_2^2 - \sqrt[3]{x_3^7}\ ).$$

Is $(0,0,0)$ *an efficient solution of the MOPP:* $\underset{x \in S}{\text{Min}}\ f(x)$? *Give reasons for the answer.*

13.4 *Let* $S = [0,2]$. *Define* $f : S \to \mathbf{R}^2$ *by*

$$f(x) = \begin{cases} (x - 1, 1 - x), & x \in [0,1] \\ (0, 1 - x), & x \in [1,2]. \end{cases}$$

Find all the weak efficient solutions and efficient solutions of the two objective programming problem $\underset{x \in S}{\text{Min}}\ f(x)$.

13.5 *Using the weighted sum approach. determine the efficient frontier of the following linear multiobjective programming problems.*

(i)

$$\begin{array}{ll} \text{Max} & \{x_1 + 2x_2,\ -2x_1 - 4x_2\} \\ \text{subject to} & \end{array}$$

$$\begin{aligned} -x_1 + 2x_2 &\leq 4 \\ x_1, x_2 &\geq 0. \end{aligned}$$

(ii)

$$\begin{array}{ll} \text{Max} & \{-4x_1 + x_2,\ x_1 - x_2\} \\ \text{subject to} & \end{array}$$

$$\begin{aligned} -x_1 + 2x_2 &\leq 8 \\ -x_1 + 2x_2 &\geq 4 \\ x_1, x_2 &\geq 0. \end{aligned}$$

(iii) $\quad Max \quad \{4x_1 + 2x_2,\ 8x_1 + 10x_2\}$

subject to

$$\begin{aligned} x_1 + x_2 &\le 70 \\ x_1 + 2x_2 &\le 100 \\ x_1 &\le 60 \\ x_2 &\le 40 \\ x_1,\ x_2 &\ge 0. \end{aligned}$$

13.6 *Using the weighted sum approach, find the efficient frontier of the following multiobjective nonlinear programming problems.*

(i) $\quad \underset{x \in \mathbb{R}^2}{Min} \quad \{2x_1x_2,\ x_1^2 + x_2^2\};$

(ii) $\quad Min \quad \{x_1^2 + x_2^2,\ 3 - x_1 + x_2^2\}$

subject to $\quad -3 \le x_1,\ x_2 \le 3.$

13.7 *Consider the following MOPP*

$Min \quad \{-x_1 + x_2,\ x_1^2 + x_2^2\}$

subject to

$$\begin{aligned} x_1^2 + x_2^2 &\le 2 \\ -x_1 + x_2^2 &\ge 1 \\ x_1 - x_2 &\le 1 \\ x_1,\ -x_2 &\ge 0. \end{aligned}$$

Find the efficient frontier of the problem.

13.8 *Consider the following MOPP*

$Max \quad \{x_1,\ -x_1 - x_2^2\}$

subject to

$$\begin{aligned} x_1^2 - x_2 &\le 0 \\ x_1 + 2x_2 &\le 3. \end{aligned}$$

Determine all the efficient solutions of the problem corresponding to the weight vectors $(\frac{1}{2}, \frac{1}{2})$ *and* $(\frac{1}{3}, \frac{2}{3})$. *Is any of the generated efficient solution a properly efficient solution of the problem? Justify your answer.*

13.9 *Determine* $x = (x_1, x_2)$ *so as to*

(i) $\quad Lexi\text{ - }Min \quad \{(d_1^+ + d_2^+),\ (d_3^-),\ (d_4^+),\ (d_5^+)\}$

subject to

$$\begin{aligned} 4x_1 + 5x_2 + d_1^- - d_1^+ &= 80 \\ 4x_1 + 2x_2 + d_2^- - d_2^+ &= 48 \\ 80x_1 + 100x_2 + d_3^- - d_3^+ &= 800 \\ x_1 + d_4^- - d_4^+ &= 6 \\ x_1 + x_2 + d_5^- - d_5^+ &= 7 \\ x_i,\ d_i^-,\ d_i^+ &\ge 0 \end{aligned}$$

(ii) *Lexi - Min* $\{(d_1^-), (d_3^-), (d_2^-), (d_1^+ + d_2^+)\}$
subject to

$$\begin{aligned} 2x_1 + x_2 + d_1^- - d_1^+ &= 20 \\ x_1 + d_2^- - d_2^+ &= 12 \\ x_2 + d_3^- - d_3^+ &= 10 \\ x, d^-, d^+ &\geq 0. \end{aligned}$$

(iii) *Lexi - Min* $\{(d_1^- + d_1^+), (2d_2^+ + d_3^+)\}$
subject to

$$\begin{aligned} x_1 - 10x_2 + d_1^- - d_1^+ &= 50 \\ 3x_1 + 5x_2 + d_2^- - d_2^+ &= 20 \\ 8x_1 + 6x_2 + d_3^- - d_3^+ &= 100 \\ x, d^-, d^+ &\geq 0. \end{aligned}$$

(iv) *Lexi - Min* $\{(d_1^+, d_2^-, d_3^-)\}$
subject to

$$\begin{aligned} x_1 - x_2 + d_1^- - d_1^+ &= 10 \\ 2x_1 + x_2 + d_2^- - d_2^+ &= 26 \\ -x_1 + 2x_2 + d_3^- - d_3^+ &= 6 \\ x, d^-, d^+ &\geq 0. \end{aligned}$$

as the maximal transmission rate over all probabilities leading to nonlinear fractional programs. In nutshell we can say that fractional programming problems are frequently encountered in various fields thereby making their study important.

At this point one may critically raised the question as to why so much emphasis on studying fractional programming problems? Can not the class of fractional programming problems be treated just like other class of nonlinear optimization problems? These queries are very natural and need to be addressed before we proceed. One major reason for giving a special treatment to fractional programming problems is that even though, in general, these problems are non convex yet for certain specific types of nonlinear fractional programming problems the structure of the objective function allow us to design special algorithms for them.

In many cases, the numerator and the denominator of the objective function are found to be, respectively, a convex and a concave function. An extra nonnegativity condition, viz., $f(x) \geq 0,\ \forall x \in S$, when g is not a linear function on S, is generally imposed. This condition ensures that the convex-concave objective ratio $\dfrac{f(x)}{g(x)}$ is pseudoconvex and thus a quasiconvex function on S (recall, Theorem 12.3.1 and Theorem 12.3.2). In this case, a local min point of the fractional program (12.6) is also its global min point.

12.5 Linear Fractional Programming Problems

In this section, we shall be discussing two algorithms, namely, the *Charnes and Cooper algorithm* and the *simplex algorithm* to solve the following linear fractional programming problem

$$\begin{aligned} \text{Max} \quad & z = \frac{c^T x + \alpha}{d^T x + \beta} \\ \text{subject to} \quad & \\ & Ax = b \\ & x \geq 0. \end{aligned} \tag{12.7}$$

Here, $x, c, d \in \mathbf{R}^n$, $A \in \mathbf{R}^{m \times n}$, $b \in \mathbf{R}^m$, $\alpha, \beta \in \mathbf{R}$.

The feasible solution set $S = \{x \in \mathbf{R}^n : Ax = b,\ x \geq 0\}$ is assumed to be non empty and bounded. Furthermore, let $d^T x + \beta \neq 0,\ \forall x \in S$. As already discussed, for the validity of the ratio, we assume that the denominator function $d^T x + \beta$ keeps the same sign in S. If $\text{sign}(\underset{x \in S}{Min}\ (d^T x + \beta)) > 0$ then $d^T x + \beta > 0,\ \forall x \in S$, and if $\text{sign}(\underset{x \in S}{Max} d^T x + \beta) < 0$ then $d^T x + \beta < 0,\ \forall x \in S$. Without loss of generality, we take, $d^T x + \beta > 0,\ \forall x \in S$.

Charnes and Cooper Algorithm

In this algorithm, a variable transformation is used to convert problem (12.7) into a linear programming problem while preserving the geometry of the problem. For this, letting $d^T x + \beta = \frac{1}{w}$. The problem can be rewritten as

$$\begin{array}{ll} \text{Max} & z = (c^T x + \alpha) w \\ \text{subject to} & x \in S. \end{array}$$

Using a transformation $y = wx$, the above problem becomes

$$\begin{array}{ll} \text{Max} & z = c^T y + \alpha w \\ \text{subject to} & \end{array}$$

$$\begin{aligned} Ay - bw &= 0 \\ d^T y + \beta w &= 1 \\ y, w &\geq 0. \end{aligned} \tag{12.8}$$

It may be noted here that the transformed problem (12.8) is a linear programming problem with an additional variable and one extra constraint as compared to problem (12.7).

The feasible solution space of problem (12.8) is denoted by $S^1 = \{(y, w) : Ay - bw = 0,\ d^T y + \beta w = 1,\ y \geq 0,\ w \geq 0\}$. We next prove some results regarding the two feasible sets S and S^1.

Result 12.5.1 *If $(y, w) \in S^1$ then $w > 0$.*

Proof. Suppose $(y, 0) \in S^1$. Then, $Ay = 0$ and $d^T y = 1$. Therefore, $y \neq 0$. Let $x \in S$, and $\mu > 0$ be an arbitrary real number. Then

$$A(x + \mu y) = Ax + \mu Ay = b,$$

$$x + \mu y \geq 0, \text{ with atleast one positive component.}$$

Thus, $x + \mu y \in S$, $\forall\, \mu > 0$. This contradicts the boundedness assumption on S. Consequently, $w > 0$, $\forall\, (y, w) \in S^1$. □

Result 12.5.2 *There is one to one correspondence between the feasible solutions of (12.7) and (12.8) with equal objective values.*

Proof. Let $x \in S$. Set $w = \dfrac{1}{d^T x + \beta} > 0$ and $y = wx$. Then, $(y, w) \in S^1$. Conversely, if $(y, w) \in S^1$ then by the previous result $w > 0$, and hence, $x = \dfrac{y}{w} \in S$.

Moreover, the objective values of problems (12.7) and (12.8) are equal at the two feasible solutions as

$$\frac{c^T x + \alpha}{d^T x + \beta} = c^T y + \alpha w.$$ □

Result 12.5.3 *There is one to one correspondence between the extreme points of the sets S and S^1.*

Proof. Let $x \in S$ be an extreme point of S. The corresponding feasible point of (12.8) is $y = wx$, $w = \dfrac{1}{d^T x + \beta}$.

If $(y, w) \in S^1$ is not an extreme point of S^1 then there exist two distinct points $(y_1, w_1) \in S^1$ and $(y_2, w_2) \in S^1$ such that for some λ, $0 < \lambda < 1$.

$$\begin{aligned}
(y, w) &= \lambda(y_1, w_1) + (1 - \lambda)(y_2, w_2) \\
\Rightarrow x = \frac{y}{w} &= \frac{\lambda y_1 + (1 - \lambda) y_2}{w} \\
&= \bar{\lambda} x_1 + (1 - \bar{\lambda}) x_2,
\end{aligned}$$

where $0 < \bar{\lambda} = \dfrac{\lambda w_1}{w} < 1$ and $x_1 = \dfrac{y_1}{w_1} \in S$, $x_2 = \dfrac{y_2}{w_2} \in S$, $x_1 \neq x_2$.

This implies that x is not an extreme point of S, leading to a contradiction. Hence, (y, w) must be an extreme point of S^1. The converse of the proof, i.e. if $(y, w) \in S^1$ is an extreme point of S^1 then the corresponding feasible point $x = \dfrac{y}{w} \in S$ is an extreme point of S, can be derived on the same lines. □

Result 12.5.4 *There is one to one correspondence between the optimal solutions of problems (12.7) and (12.8).*

Proof. Problem (12.8) is a linear program, so its optimal solution is one of the extreme point of S^1. Now, observe that because of Theorem 12.2.5, the objective function of (12.7) is a quasiconcave function on the polytope S. It then follows from Theorem 12.2.4 that the optimal solution of (12.7) is also obtained at one of the extreme point of S, say $\hat{x}$. Further, invoking Results 12.5.3 and 12.5.2, there exists a corresponding extreme point $(\hat{y}, \hat{w})$ of S^1 with objective value of (12.7) at $\hat{x}$ equals the objective value of (12.8) at $(\hat{y}, \hat{w})$. Moreover, since $\hat{x}$ is an optimal solution of (12.7), thus, by virtue of Results 12.5.2 and 12.5.3, $(\hat{y}, \hat{w})$ is an optimal solution of problem (12.8). □

The above result lies at the root of the Charnes and Cooper algorithm. From this, one can easily see that solving (12.7) is equivalent to solving a linear programming problem (12.8).

We now present an example to illustrate the working of this algorithm.

Example 12.5.1 *Solve the following linear fractional programming problem by the Charnes and Cooper algorithm*

$$\begin{aligned} &Max \qquad z = \frac{x_1}{x_1 + x_2 + 1} \\ &subject\ to \\ &\qquad\qquad x_1 + x_2 \le 1 \\ &\qquad\qquad x_1 + 2x_2 \le 1 \\ &\qquad\qquad x_1,\ x_2 \ge 0. \end{aligned}$$

Solution Note that $x_1 + x_2 + 1 > 0$, for all (x_1, x_2) feasible for the problem. Applying the Charnes and Cooper transformation, $w = \dfrac{1}{x_1 + x_2 + 1}$ and $y_1 = wx_1,. y_2 = wx_2$, the problem becomes

$$\begin{aligned} &\text{Max} \qquad z = y_1 \\ &\text{subject to} \\ &\qquad\qquad y_1 + y_2 - w \le 0 \\ &\qquad\qquad y_1 + 2y_2 - w \le 0 \\ &\qquad\qquad y_1 + y_2 + w = 1 \\ &\qquad\qquad y_1,\ y_2,\ w \ge 0. \end{aligned}$$

Introduce the slack variables $s_1, s_2 \ge 0$, the problem reduces to

$$\begin{aligned} &\text{Max} \qquad z = y_1 \\ &\text{subject to} \\ &\qquad\qquad y_1 + y_2 - w + s_1 = 0 \\ &\qquad\qquad y_1 + 2y_2 - w + s_2 = 0 \\ &\qquad\qquad y_1 + y_2 + w = 1 \\ &\qquad\qquad y_1,\ y_2,\ w,\ s_1,\ s_2 \ge 0. \end{aligned}$$

Solving the above problem by the two phase method, the optimal table is given by

	$y^{(1)}$	$y^{(2)}$	$y^{(w)}$	$y^{(s_1)}$	$y^{(s_2)}$
$y_1 = 1/2$	1	1	0	1/2	0
$s_2 = 0$	0	1	0	-1	1
$w = 1/2$	0	0	1	-1/2	0
$z_B = 1/2$	0	1	0	1/2	0

Therefore, the optimal value is $\frac{1}{2}$ with the optimal solution $y^\star = (\frac{1}{2}, 0)$ and $w^\star = \frac{1}{2}$. Thus, the optimal solution of the original linear fractional program is $x_1^\star = \dfrac{y_1^\star}{w^\star} = 1,\ x_2^\star = \dfrac{y_2^\star}{w^\star} = 0$ and the optimal value is $\dfrac{1}{2}$.

Simplex Algorithm for Linear Fractional Programming

As already observed that the objective function of (12.7) is a quasiconcave function and the feasible set S is a polytope, so, the optimal solution of (12.7) is obtained at an extreme point of S. Taking motivation from this simple fact, we study the simplex algorithm for problem (12.7) without restoring to any variable transformation. The simplex algorithm for linear programming problem has already been explained in detail in the first few chapters of this book. We assume that the readers are familiar and well versed with the notations and interpretations of all the concepts used in linear programming problems. We carry forward the same notations while describing the simplex algorithm for (12.7). Moreover, many steps in the present algorithm follows on the similar lines, so, without going into the detailed reasoning we present only their outline.

We start with an assumption that a basic feasible solution (b.f.s.) with basis B is available. Therefore, $x_B = B^{-1}b$. Set

$$V_N = c_B^T x_B + \alpha, \quad V_D = d_B^T x_B + \beta.$$

The value of the objective function at x_B is $z = \dfrac{V_N}{V_D}$.

Let $a^{(j)}$ be a column in a matrix A that is not in B. Then, since B is a basis, $a^{(j)}$ can be expressed as a linear combination of the columns of B matrix, i.e.

$$a^{(j)} = By_j = \sum_{i=1}^{m} y_{ij} b^{(i)}.$$

Set $z_j^1 = c_B^T y_j$, $z_j^2 = d_B^T y_j$.

We first find the condition that ensures that the current b.f.s. can be improved to get another b.f.s. with improved objective value.

Suppose $\bar{B}$ is a new basis obtained from B by replacing a column $b^{(r)}$ of B by $a^{(j)}$. Therefore, the columns of $\bar{B}$ are given by $\bar{b}^{(r)} = a^{(j)}$, $\bar{b}^{(i)} = b^{(i)}$ $(i \neq r)$. The leaving variable x_{B_r} is chosen according to the minimum ratio test

$$\theta_j = Min\left\{\frac{x_{B_i}}{y_{ij}} \;:\; y_{ij} > 0\right\} = \frac{x_{B_r}}{y_{rj}} \qquad \text{(say)}.$$

The new basic variables $\bar{x}_{B_r}$ are given by

$$\begin{aligned} \bar{x}_{B_i} &= x_{B_i} - \theta_j y_{ij} \quad (i = 1, \ldots, m,\ i \neq r), \\ \bar{x}_{B_r} &= \theta_j. \end{aligned}$$

Let the new value of the objective function be $\bar{z} = \dfrac{\overline{V}_N^1}{\overline{V}_D^2}$, where

$$\overline{V}_N^1 = V_N - \theta_j(z_j^1 - c_j) \quad \text{and} \quad \overline{V}_D^2 = V_D - \theta_j(z_j^2 - d_j).$$

We pause here to again emphasize that the detailed working of all the above expressions is explained in Chapter 3.

The objective function value will strictly improve if $\bar{z} > z$, i.e.

$$\frac{\overline{V}_N^1}{\overline{V}_D^2} - \frac{V_N}{V_D} > 0,$$

which on simplification yields

$$\theta_j\{V_N(z_j^2 - d_j) - V_D(z_j^1 - c_j)\} > 0.$$

Letting $\Delta_j = V_N(z_j^2 - d_j) - V_D(z_j^1 - c_j)$, we obtain, $\theta_j\Delta_j > 0$.

For $\theta_j > 0$, the improvement in the objective value z is possible if $\Delta_j > 0$. Thus, the current b.f.s. is optimal for (12.7) if $\Delta_j \le 0$, for all those variables x_j which are currently nonbasic variables. Note that for basic variables, we already have, $\Delta_j = 0$. Thus, the optimality criteria (for maximization problem (12.7)) is that $\Delta_j \le 0$ $(j = 1, \ldots, n)$.

Suppose $\theta_j = 0$. It mean $x_{B_r} = 0$ and $\bar{z} = z$. This implies that there is no improvement in the objective function value. Here x_{B_r}, which is a basic variable with value zero, becomes a nonbasic variable and the new variable $\bar{x}_{B_r}$ becomes a basic variable with value zero, the values of all the other variables remain the same, leading to the situation of degenerate b.f.s. In this case, the basis changes but the corresponding extreme points remains the same. The case of degeneracy in b.f.s. has already been discussed in Chapter 3.

The above discussion is summarized in the following theorem.

Theorem 12.5.1 *If all $\Delta_j \le 0$ then the current b.f.s. x_B is an optimal solution of the linear fractional programming problem (12.7).*

Remark 12.5.1 *If some $\Delta_j > 0$ and for that $y_{ij} \le 0$, $\forall\, i$, then the linear fractional programming problem (12.7) has unbounded solution, which contradicts the boundedness of the feasible set S.*

Remark 12.5.2 *The objective function of (12.7), $\dfrac{c^T x + \alpha}{d^T x + \beta}$, is a pseudolinear function, hence, a local optimizer is a global optimizer. Consequently, the b.f.s. x_B with all $\Delta_j \le 0$ is the global optimal solution of problem (12.7).*

The mechanism of the simplex algorithm for linear fractional programming problem is now illustrated through an example.

Example 12.5.2 *Solve the following linear fractional programming problem by the simplex algorithm*

$$\begin{aligned} &\textit{Max} \quad z = \frac{x_1}{x_1 + x_2 + 1} \\ &\textit{subject to} \\ &\qquad x_1 + x_2 \le 1 \\ &\qquad x_1 + 2x_2 \le 1 \\ &\qquad x_1, x_2 \ge 0. \end{aligned}$$

Solution Introducing the slack variables $s_1, s_2 \ge 0$, the problem reduces to

$$\begin{aligned} &\text{Max} \quad z = \frac{x_1}{x_1 + x_2 + 1} \\ &\text{subject to} \\ &\qquad x_1 + x_2 + s_1 = 1 \\ &\qquad x_1 + 2x_2 + s_2 = 1 \\ &\qquad x_1, x_2, s_1, s_2 \ge 0. \end{aligned}$$

Solving the problem by the simplex method we get following tableaus

	$y^{(1)}$	$y^{(2)}$	$y^{(s_1)}$	$y^{(s_2)}$
$\leftarrow s_1 = 1$	$\boxed{1}$	1	1	0
$s_2 = 1$	1	2	0	1
$z_j^1 - c_j$	-1	0	0	0
$z_j^2 - d_j$	-1	-1	0	0
$z_B = 0, \ \Delta_j \rightarrow$	1	0	0	0
	$\uparrow$			

	$y^{(1)}$	$y^{(2)}$	$y^{(s_1)}$	$y^{(s_2)}$
$x_1 = 1$	1	1	1	0
$s_2 = 0$	0	1	-1	1
$z_j^1 - c_j$	0	1	1	0
$z_j^2 - d_j$	0	0	1	0
$z_B = 1/2, \ \Delta_j \rightarrow$	0	-2	-1	0

Since $\Delta_j \le 0, \ \forall j$, thus the optimality criteria is satisfied. The optimal solution of the given problem is $x_1^\star = 1, \ x_2^\star = 0$ and the optimal value is $\frac{1}{2}$.

12.6 Nonlinear Fractional Programming Problems

In the previous section, we have concentrated on studying the solution methodologies for linear fractional programming problems. In this section we move ahead to study a solution procedure for a subclass of the class of *nonlinear fractional programming problems*.

The class of nonlinear fractional programming problems is too large to be solved by a single algorithm. Thus, when we say a subclass we actually mean a particular class of nonlinear fractional programming problems in which the numerators are concave functions and the denominators are positive convex functions. Such nonlinear fractional programming problems are called concave-convex fractional programming problems. Note that for such problems the objective ratio is a pseudoconcave function.

The algorithm presented below is called the *Dinkelbach's algorithm*, named after Werner Dinkelbach of Germany who first described this algorithm in 1967. Recall that a general nonlinear fractional programming problem is given by

$$\begin{aligned} &\text{Max} \quad \frac{f(x)}{g(x)} \\ &\text{subject to} \\ &\qquad x \in S = \{x \in \mathbf{R}^n : h_i(x) \le 0 \ (i = 1, \dots, m)\}. \end{aligned} \tag{12.9}$$

We assume that S is a non empty compact convex set and $f, g : \mathbf{R}^n \to \mathbf{R}$ are continuous functions on S with $g(x) > 0$, $\forall\, x \in S$.

For instance the assumption that S is a convex set is true if each h_i $(i = 1, \dots, m)$, is a quasiconvex function on $\mathbf{R}^n$, as $S = \bigcap_{i=1}^{m}\{0 -$ level set of $h_i\}$. Also, S is a closed set if each h_i $(i = 1, \dots, m)$ is a continuous function on $\mathbf{R}^n$.

The algorithm for solving (12.9) shall make use of the following auxiliary problem with parameter $q \in \mathbf{R}$.

$$\begin{aligned} &\text{Max} \quad f(x) - qg(x) \\ &\text{subject to} \quad x \in S. \end{aligned} \tag{12.10}$$

Observe that if f is concave, g is convex and $q \ge 0$, then $f - qg$ is a concave function, whereas, $\dfrac{f}{g}$ is not a concave function. This makes (12.10) easier to solve than (12.9). Moreover, f and g are continuous functions on a compact convex set S, hence the two problems (12.9) and (12.10) possess optimal solutions in S.

Denote by $F(q)$ the optimum objective value of (12.10), i.e.

$$F(q) = Max\{f(x) - qg(x) : x \in S\}, \ q \in \mathbf{R}.$$

The function F has some nice properties which we would like to share with the readers.

Result 12.6.1 *F is a convex function on* $\mathbf{R}$.

Proof. Let $q_1, q_2 \in \mathbf{R}$, $\lambda \in [0, 1]$. Taking $q = \lambda q_1 + (1 - \lambda)q_2$. Let $x_q \in S$ be the max point of $F(q)$, i.e.

$$\begin{aligned}
F(q) &= Max\{f(x) - qg(x) : x \in S\} \\
&= f(x_q) - qg(x_q) \\
&= f(x_q) - (\lambda q_1 + (1-\lambda)q_2)g(x_q) \\
&= \lambda(f(x_q) - q_1 g(x_q)) + (1-\lambda)(f(x_q) - q_2 g(x_q)) \\
&\leq \lambda \underset{x\in S}{Max}(f(x) - q_1 g(x)) + (1-\lambda)\underset{x\in S}{Max}(f(x) - q_2 g(x)) \\
&= \lambda F(q_1) + (1-\lambda)F(q_2).
\end{aligned}$$

Thus,

$$F(\lambda q_1 + (1-\lambda)q_2) \leq \lambda F(q_1) + (1-\lambda)F(x_2),\ \forall \lambda \in [0,1],\ \forall q_1, q_2 \in \mathbf{R}.$$

This completes the proof. □

Result 12.6.2 *F is a continuous function on* $\mathbf{R}$.

Proof. Using Result 12.6.1 along with the fact that a convex function defined on an open convex set (here, it is $\mathbf{R}$) is continuous in its domain, we get the desired result. □

Result 12.6.3 *F is a strictly monotonic decreasing function on* $\mathbf{R}$.

Proof. Let $q_1, q_2 \in \mathbf{R}$ with $q_1 < q_2$. Suppose $x_2 \in S$ is the point where $F(q_2)$ attains its maximum value. Then,

$$\begin{aligned}
F(q_2) &= f(x_2) - q_2 g(x_2) \\
&< f(x_2) - q_1 g(x_2) \\
&\leq F(q_1),
\end{aligned}$$

where the strict inequality follows on account of $q_1 < q_2$ and $g(x_2) > 0$. Thus,

$$q_1 < q_2 \Rightarrow F(q_2) < F(q_1).$$

Thereby, implying that F is a monotonic decreasing function on $\mathbf{R}$. □

Result 12.6.4 *The nonlinear equation* $F(q) = 0$ *has the unique solution in* $\mathbf{R}$.

Proof follows by virtue of Result 12.6.2 and Result 12.6.3. □

Result 12.6.5 *Let* $x^\star \in S$ *and* $q^\star = \dfrac{f(x^\star)}{g(x^\star)}$. *Then* $F(q^\star) \geq 0$.

Proof. We have

$$\begin{aligned}
F(q^\star) &= Max\{f(x) - q^\star g(x) : x \in S\} \\
&\geq f(x^\star) - q^\star g(x^\star) = 0.
\end{aligned}$$

□

Theorem 12.6.1 $x^\star \in S$ *is an optimal solution of (12.9) if and only if* $x^\star$ *is an optimal solution of* $F(q^\star)$ *with optimal objective value* $F(q^\star) = 0$, *where,* $q^\star = \dfrac{f(x^\star)}{g(x^\star)}$.

Proof. Suppose $x^\star$ is an optimal solution of problem (12.9). Then,

$$q^\star = \frac{f(x^\star)}{g(x^\star)} \geq \frac{f(x)}{g(x)}, \quad \forall\, x \in S$$

implying

$$f(x) - q^\star g(x) \leq f(x^\star) - q^\star g(x^\star), \quad \forall\, x \in S.$$

Consequently, $x^\star$ is an optimal solution of the problem

$$F(q^\star) = Max\{f(x) - q^\star g(x) \,:\, x \in S\},$$

with $F(q^\star) = 0$.

The converse follows by tracing the steps backward. □

Similar theorem can be stated and proved for the minimization case as well.

Theorem 12.6.2 *$x^\star \in S$ is an optimal solution of $Min\left\{\frac{f(x)}{g(x)} \,:\, x \in S\right\}$ if and only if $x^\star$ is an optimal solution of $Min\{f(x) - q^\star g(x) \,:\, x \in S\}$ with optimal objective value zero, where, $q^\star = \frac{f(x^\star)}{g(x^\star)}$.*

Dinkelbach's Algorithm

As a consequence of the above results and theorems, it follows that there is a correspondence between the optimal solutions of the nonlinear fractional programming problem (12.9) and the nonlinear parametric programming problem (12.10). Taking clue from this, a mechanism is developed which solves the nonlinear parametric programming problem (12.10) that in turn provides an optimal solution of the original nonlinear fractional programming problem (12.9).

In view of Theorem 12.6.1, solving (12.9) is equivalent to finding the root of the equation $F(q) = 0$ which, on account of Result 12.6.4, is unique. An iterative scheme is proposed to achieve this aim.

We begin the algorithm with $q_0 = 0$ (or we can start with any other value of q with $F(q) \geq 0$).

$$F(q_0) = Max\{f(x) \,:\, x \in S\} \geq 0.$$

This problem is constrained convex program, hence, its optimal solution can be found by applying appropriate nonlinear convex optimization technique. Suppose x_0 is an optimal solution of this problem.

Two cases arise, either $F(q_0) = 0$ or $F(q_0) > 0$.

If $F(q_0) = 0$ then x_0 is an optimal solution of (12.9) and the process terminates.

12.8 *Let $S \subseteq \mathbf{R}^n$ be a nonempty convex set and $g, h : S \to \mathbf{R}$. Define a function $f : S \to \mathbf{R}$ by $f(x) = g(x)h(x)$. Show that f is quasiconvex if the following two conditions hold*

(i) g is convex on S and $g(x) \leq 0,\ \forall x \in S$
(ii) h is concave on S and $h(x) > 0,\ \forall x \in S$.

12.9 *Let $S \subseteq \mathbf{R}^n$ be a nonempty open convex set. Consider a function $f : S \to \mathbf{R}$ with $f(x) > 0,\ \forall x \in S$. Show that if $\ln(f)$ is a concave function on S then f is a pseudoconcave function on S.*

12.10 *Solve the following linear fractional programming problems by both the Charnes and Cooper method and the simplex method*

(i) *Max* $z = \dfrac{2x_1 + x_2}{-x_1 - 2x_2}$

subject to

$$\begin{aligned} x_1 + x_2 &\leq 6 \\ 2x_1 + x_2 &\geq 4 \\ x_1, x_2 &\geq 0. \end{aligned}$$

(ii) *Max* $z = \dfrac{2x_1 + 3x_2}{x_1 + x_2 + 7}$

subject to

$$\begin{aligned} 3x_1 + 5x_2 &\leq 15 \\ 4x_1 + 3x_2 &\leq 12 \\ x_1, x_2 &\geq 0. \end{aligned}$$

(iii) *Max* $z = \dfrac{x_2 - 5}{-x_1 - x_2 + 9}$

subject to

$$\begin{aligned} 2x_1 + 5x_2 &\geq 10 \\ 4x_1 + 3x_2 &\leq 20 \\ -x_1 + x_2 &\leq 2 \\ x_1, x_2 &\geq 0. \end{aligned}$$

12.11 *Using the variable transformation, associate a linear programming problem with the linear fractional programming problem (LFPP)*

Min $\dfrac{c^T x + c_o}{d^T x + d_o}$

subject to

$$\begin{aligned} Ax &\geq 0 \\ x &\geq 0. \end{aligned}$$

Let $(y^\star, w^\star)$ be an optimal solution of the linear programming problem. If $w^\star \neq 0$ then show that $x^\star = y^\star/w^\star$ is an optimal solution of the (LFPP) and the objective functions values of the (LFPP) and the corresponding linear program are equal.

12.12 *Suppose $w^\star = 0$ in the Charnes and Cooper transformation. Prove that the feasible set of (LFPP) is unbounded.*

12.13 *Solve the following fractional programming problems graphically*

$$\text{(i)} \qquad \underset{(x_1, x_2) \in S}{Min} \; \frac{6x_1 + 2x_2 + 6}{3x_1 + 5x_2 + 3}$$

$$\text{(ii)} \qquad \underset{(x_1, x_2) \in S}{Min} \; \frac{6x_1 + 2x_2 + 6}{3x_1 + 5x_2 + 3}$$

where $S = \{(x_1, x_2) : x_1 - x_2 \leq 2, \; 2x_1 - x_2 \leq 6, \; x_1, x_2 \geq 0\}$. *Which of (i) and (ii) has an alternate optimal solution? Justify your answer.*

12.14 *Solve the following fractional programming problems by the Dinkelbach's method*

$$\text{(i)} \quad Max \quad z = \frac{5x_1 + 3x_2}{5x_1 + 2x_2 + 1}$$

subject to

$$\begin{aligned} 3x_1 + 5x_2 &\leq 15 \\ 5x_1 + 2x_2 &\geq 10 \\ x_1, x_2 &\geq 0. \end{aligned}$$

$$\text{(ii)} \quad Max \quad z = \frac{2x_1 + 2x_2 + 1}{x_1^2 + x_2^2 + 3}$$

subject to

$$\begin{aligned} x_1 + x_2 &\leq 3 \\ x_1, x_2 &\geq 0. \end{aligned}$$

$$\text{(iii)} \quad Max \quad z = \frac{x_1 + 2x_2}{3x_1^2 + x_2^2 + 1}$$

subject to

$$\begin{aligned} x_1 + x_2 &\leq 2 \\ x_1 - x_2 &\leq 1 \\ x_1, x_2 &\geq 0. \end{aligned}$$