APPLIED SYSTEMS ANALYSIS

Engineering Planning and Technology Management

Richard de Neufville
Massachusetts Institute
of Technology

McGraw-Hill Publishing Company
New York St. Louis San Francisco Auckland Bogotá Caracas Hamburg
Lisbon London Madrid Mexico Milan Montreal New Delhi Oklahoma City
Paris San Juan São Paulo Singapore Sydney Tokyo Toronto

To calculate the reliability of the circuit, we note:

(*a*) P(Circuit functions) $= \prod_i$ P (part *i* functions)

(*b*) P(Part i functions) $= 1 -$ P (part *i* fails)
$= 1 -$ P (all components in *i* fail)
$= 1 - [$ P(one component in *i* fails)$]^n$
$= 1 - [1 - P_i]^n$

Assume a budget of \$16 and use dynamic programming to design the most reliable circuit.

CHAPTER 8

MULTIOBJECTIVE OPTIMIZATION

8.1 THE PROBLEM

Most systems either produce several different outputs or serve a variety of purposes at once. A factory almost always makes distinct products: an automobile factory may make cars and trucks; a refinery can turn crude petroleum into a range of distillates. Service facilities likewise typically accomplish several goals: schools, for instance, both teach children and provide custodial care so parents can work outside the home; a dam on a river may both control floods and provide some recreational benefits associated with its reservoir.

The preceding chapters have all dealt with optimization over only one objective, benefit or cost, in contrast to the reality of the multiple outputs for any system. This focus is based on two reasons. The first is that it is often realistic to assume that multiple outputs can all be translated into a single overriding objective. The managers of a factory producing cars and trucks may, for example, really be concerned with profits rather than types of vehicles; they may thus be quite prepared to express each different output in terms of its profitability. The second reason for the focus on single objective optimization is that it is virtually impossible to achieve an optimum over many objectives at once.

The essential difficulty with multiobjective optimization is that the meaning of the optimum is not defined so long as we deal with multiple objectives that are truly different. For example, suppose that we are trying to determine the best design of a system of dams on a river, with the objectives of promoting "national income," reducing "deaths by flooding," and increasing "employment." Some designs will be more profitable, but less effective at reducing deaths. How can we state which is better when the objectives are so different, and measured in such different terms? How can one state with any accuracy what the relative value of a life is in terms of national income? If one resolved that question, then how would one determine the relative value of new jobs and other objectives? The answer is, with extreme difficulty. The attempts to set values on these objectives are, in fact, most controversial.

To obtain a single global optimum over all objectives requires that we either establish or impose some means of specifying the value of each of the different objectives. If all objectives can indeed be valued on a common basis, the optimization can be stated in terms of that single value. The multiobjective problem has then disappeared and the optimization proceeds relatively smoothly in terms of a single objective.

In practice it is frequently awkward if not indefensible to give every objective a relative value. The relative worth of profits, lives lost, the environment, and other such objectives are unlikely to be established easily by anyone, or to be accepted by all concerned. One then cannot hope to be able to determine an acceptable optimum analytically.

The focus of multiobjective optimization in practice is to sort out the mass of clearly inferior solutions, rather than determine the single best design. The result is the identification of the small subset of the feasible solutions that are worthy of further consideration. Formally, this result is known as the set of noninferior solutions.

8.2 NONINFERIOR SOLUTIONS

To understand the concept of noninferior solutions, it is necessary to look closely at the multiobjective problem. Formally, it consists of several objectives:

$$\mathbf{Y} = (Y_1, \ldots, Y_k)$$

each of which is a different function of the inputs, **X**

$$Y_k = g_k(\mathbf{X})$$

The problem of multiobjective optimization is thus:

Optimize: $\mathbf{Y} = g_1(\mathbf{X}), \ldots, g_k(\mathbf{X})$

Subject to: constraints on **X**

The essential feature of the multiobjective problem is that the feasible region of production of the solutions is much more complex than for a single objective. In single objective optimization, any set of inputs **X** produces a set of results Y that could be represented by a straight line going from bad (typically zero output) to best; the production function is then simply defined as locus of these best or technically efficient solutions (refer back to Figure 2.1). In a multiobjective problem, any set of inputs **X** defines a multidimensional space of feasible solutions, as Figure 8.1 indicates. There is then no exact equivalent of a technically efficient solution.

The *noninferior solutions* are the conceptual equivalents, in multiobjective problems, of a technically efficient solution in a single objective problem. Formally, a noninferior solution $\mathbf{Y}^*$ for a given set of **X** is one such that no other feasible solution for the same **X** is better on all objectives $\mathbf{Y}_k$. If some other feasible solution improves on some objectives, it must degrade on others. Figure 8.2 illustrates the concept with respect to the status quo, $\mathbf{Y}_0$, that exists without the project defined by the inputs **X**. Graphically, the noninferior solutions are the outer shell of the feasible solutions for any set of inputs **X**, in the direction of improving the objectives.

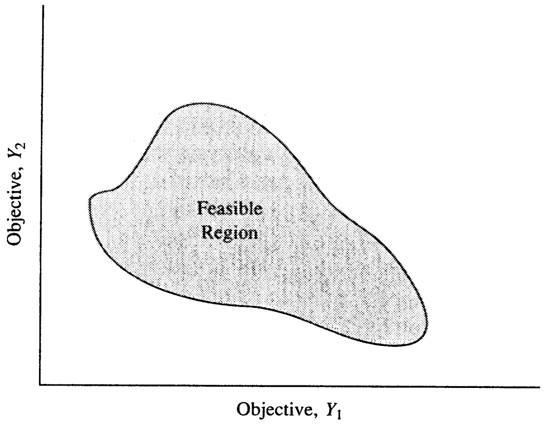

FIGURE 8.1 Feasible region of multidimensional production for a specific set of inputs X.

Semantic caution: The noninferior solutions are sometimes referred to by other names: dominant solutions, Pareto optimal solutions, and the production feasibility frontier. The term "noninferior solutions" lacks elegance but seems preferable to the alternatives in that it is both descriptive and accurate. The term "dominant solutions" is misleading and should be avoided: no solution in the noninferior set dominates any other in the set.

The preferred design for any problem should be one of the noninferior solutions. So long as all objectives worth taking into account have been considered, no design that is not among the noninferior solutions is worthwhile: it is dominated by some designs that are preferable on all accounts. This is the reason multiobjective optimization focuses on the determination of the noninferior solutions.

It is often useful to group the noninferior solutions into major categories. The purpose of this exercise is to facilitate discussions about which solution to select. Indeed, to the extent that it is not possible to specify acceptable relative values for the objectives, and thus impossible to define the best design analytically, it is necessary for the choice of the design to rest on judgement. As individuals find it difficult to consider a large number of possibilities, it is helpful to focus attention on major categories.

The noninferior solutions are best divided into two types of categories: the major alternatives and the compromises. *A major alternative group* of noninferior solutions represents the best performance on some major objective. As Figure 8.3

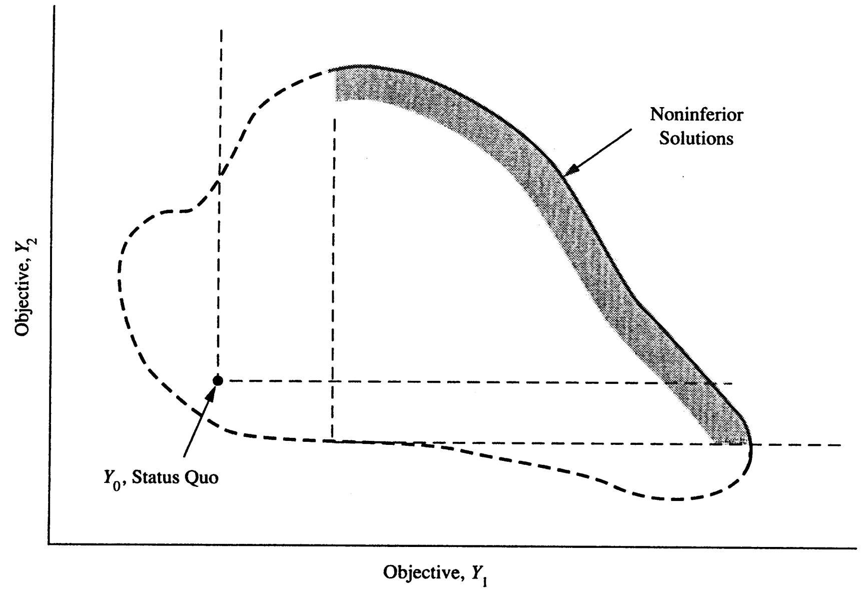

FIGURE 8.2
The noninferior solutions as the outer shell of the feasible region in the direction of improving the objectives.

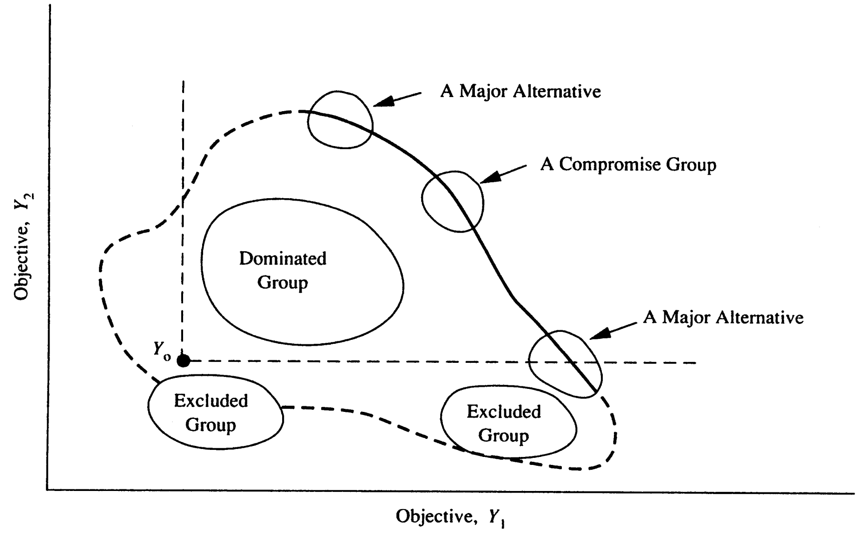

FIGURE 8.3
Groups associated with noninferior and other feasible solutions.

indicates, the major alternatives represent polar extremes. A *compromise group* lies somewhere in between the major alternatives.

The remainder of the feasible region of solutions is likewise usefully categorized into dominated and excluded solutions. *Dominated* solutions are those that are inferior in all essential aspects to other solutions. They can thus be set aside from further consideration. *Excluded* solutions are those that perform so badly on one or more dimensions that they lie beneath the threshold of acceptability. They thus may be dropped from further consideration.

The concepts of noninferior solutions and of major categories are often highly useful in a practical sense. They organize the feasible designs into a small number of manageable ideas and draw attention to the choices that must be made. These ideas can be applied even when the feasible region is not defined analytically. See the box for an example.

Use of Categories in Multiobjective Optimization

The choice of a site for a second major airport for Sydney, Australia was definitely a multiobjective problem. As with most projects centering on the location of a major facility, it involved many distinct issues. Major concerns for an airport are, in general: cost, accessibility, safety, and environmental impact.

The feasible solutions for Sydney were highly constrained by mountains, the city, and the sea. At the time of the study we did in 1984, only 11 distinct locations seemed worth considering, 10 new ones and the existing airport. Their total costs were all roughly comparable, and all were deemed to meet the required threshold of safe operation. Our analysis consequently paid attention to the two remaining major objectives of maximizing accessibility and environmental compatibility.

Close examination of the 11 feasible solutions indicated that they could be placed into each of the groups illustrated in Figure 8.3. Three sites fell into excluded groups: Goulbourn turned out to be too inaccessible, another had unsafe wind considerations, and a third was prohibitively expensive and hazardous since it involved clearing a vast expanse littered with unexploded artillery shells. Five other locations were fairly clearly dominated by other sites. The three remaining locations could then be viewed as two major alternatives and a possible compromise: Wilton represented the most environmentally compatible feasible solution, in great part because it is so far from people whom it might disturb; and Badgery's Creek, somewhere in between, represented a possible compromise. See Figure 8.4.

The division of the possible sites into these categories had the practical benefit of helping the Australian government decide between the different sites, each with their distinct advantages and disadvantages in detail. Based on this analysis the Minister for Aviation selected Badgery's Creek and Wilton for detailed analysis. The existing airport was excluded as being environmentally inappropriate for a major long-term expansion involving many additional runways.

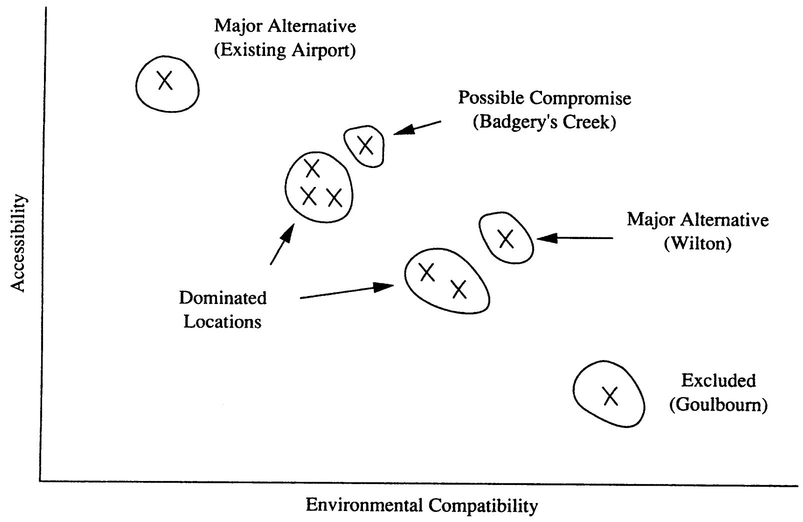

FIGURE 8.4
Application of categories to the feasible locations for a Second Sydney Airport.

8.3 CONSTRAINT METHOD

This and the next section describe the two major ways of defining the noninferior solutions: the constraint and the weighting methods. Both have a common kind of approach to the task. They transform the multidimensional problem, which cannot be solved explicitly, into a series of one-dimensional problems for which solutions can be obtained. The basic difference between the two methods lies in how they make this transformation to one-dimensional problems.

The choice between the constraint and the weighting method is a question of convenience rather than one of principle. Each has distinct operational advantages and disadvantages. Which should be used depends on the specifics of the problem. Guidelines for selecting between the methods occur at the end of the next section.

The *constraint method* generates a systematic exploration of the noninferior solutions. It can define these as precisely and in as much detail as desired, however complex the surface of noninferior solutions may be. The major disadvantage of the constraint method is that it can be tedious and expensive.

The essential idea of the constraint method is to optimize one objective while representing all the other objectives as constraints. This process defines a noninferior solution, if any solution is feasible. Systematic repetition of this process, with different constraints on the objectives, generates the entire set of noninferior solutions.

The representation of objectives as constraints is an explicit expression of what is often done implicitly in practice. In designing a water supply system, for example, there are at least two major objectives: level of cost and level of quality. In practice, the quality objective has generally been expressed as a set of minimal constraints on purity and water pressure. This reduces the design to the single objective problem of minimizing cost (refer to Section 3.1). The constraint method extends this idea by varying the constraints and, thereby, making evident the relationship between the several objectives.

The way the constraint method generates the noninferior solutions is best shown graphically. For this purpose consider the situation with two objectives, Y_1 and Y_2, illustrated by Figure 8.5. If we maximize either objective without placing any restraint on the other, we obtain a noninferior solution. Point A, for example, represents the design with the greatest possible Y_2 and some level of Y_1. Suppose we now set a minimum constraint on the level of objective Y_1, such as $Y_{1B} > Y_{1A}$. If we now maximize Y_2 again, but subject to this minimum constraint on the other objective, we obtain another noninferior solution, Point B. We can repeat this process as often as we care or can afford to and thereby get more and more noninferior solutions. There is no theoretical limit to the precision with which we can define the noninferior solutions by the constraint method.

The constraint method has a noticeable disadvantage when there are three or more objectives. The difficulty is then that as the analysis systematically imposes combinations of constraints on the objectives, many of these combinations lead to infeasible solutions. Figure 8.6 indicates how this may occur. Much computational effort may thus be wasted using the constraint method. Whether this is important depends on its cost.

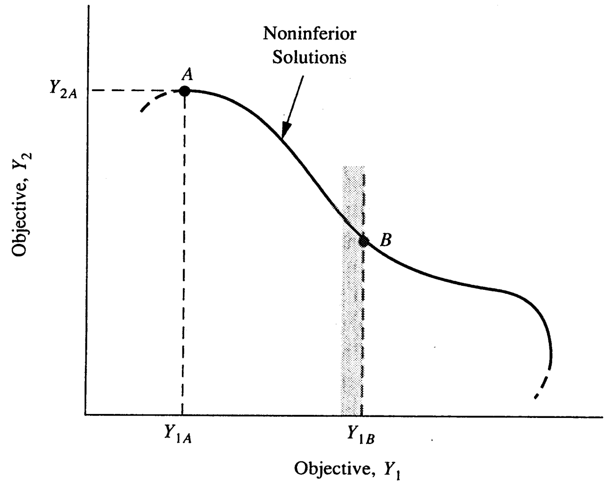

FIGURE 8.5
The constraint method defines the entire set of noninferior solutions by systematically varying the constraints on the objectives.

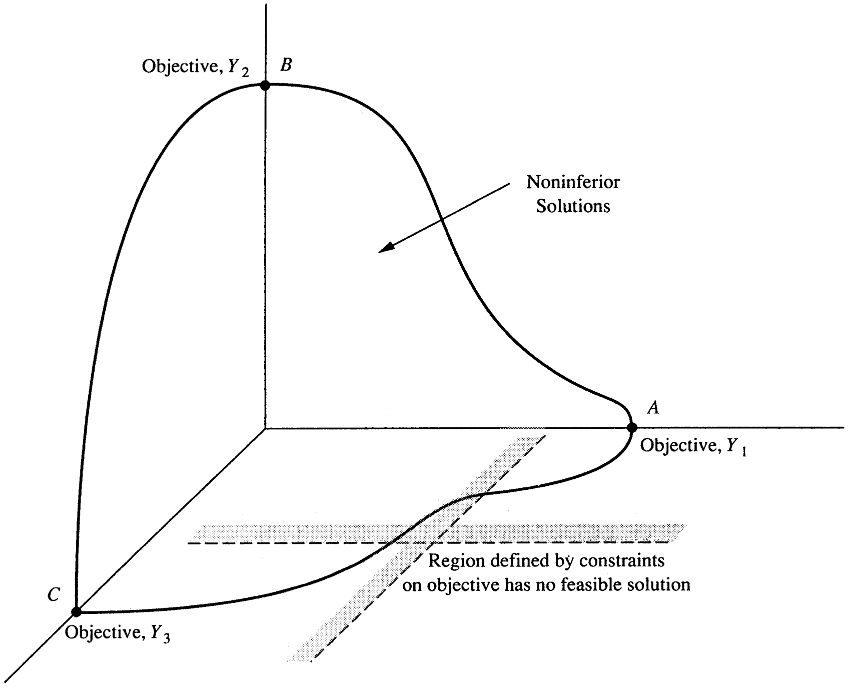

FIGURE 8.6
The constraint method often leads to infeasible solutions when there are three or more objectives.

A practical, step-by-step procedure for applying the constraint method is as follows:

1. Optimize for each objective by itself, without any constraints on the objectives. These results define limits on the range of noninferior solutions as points *A*, *B*, and *C* do in Figure 8.6.
2. Optimize on one objective, placing constraints on only one other objective. Iteratively change this constraint until you reach the end of the maximum feasible value of this objective. Repeat for other objectives. This process traces out noninferior solutions such as those shown by AB, BC and AC in Figure 8.6.
3. Optimize now with two constraints, and proceed similarly to above. This traces out noninferior points in three dimensions.
4. Proceed as before for as many dimensions as desirable or necessary.

The constraint method works both when the solutions can be defined mathematically, or are only defined individually, as was the case for the design of the Second Sydney Airport (see box in Section 8.2). The method implicitly assumes that linear programming will be used in any mathematical analysis, as it is the only method that can readily deal with multiple constraints.

8.4 WEIGHTING METHOD

The *weighting method* also generates a systematic definition of noninferior solutions. It accomplishes this somewhat faster than the constraints method. Its major disadvantage is that it may miss or misrepresent some portion of the set of noninferior solutions, in ways that the constraint method avoids.

The essential idea of the weighting method is to transform the multiobjective problem (see Section 8.2):

Optimize: $\mathbf{Y} = g_1(\mathbf{X}), \ldots, g_k(\mathbf{X})$

Subject to: constraints on $\mathbf{X}$

into a single objective problem. This is done by introducing a set of weights, w_k, for each objective. One then proceeds to optimize the weighted sum of the objectives:

Optimize: $\sum \mathbf{wY} = \sum w_k g_k(\mathbf{X})$

Subject to: constraints on $\mathbf{X}$

Solving this problem defines a noninferior solution. Systematic repetition of the process for different sets of weights defines most of the noninferior solutions—although not all.

The various sets of weights used in the weighting method do not have to be given a meaningful interpretation. Most basically, the sets of weights can simply be thought of as absolutely arbitrary numbers that serve the purpose of generating the set of noninferior solutions.

The weights can also be interpreted as the relative values of each objective. If one really did believe that each unit of achievement of objective Y_k were exactly worth w_k, then maximizing the weighted sum of the objectives, $\sum w_k Y_k$, would maximize total value. Thus, for the automobile factory producing cars and trucks with known degrees of profitability, the multiobjective problem of determining what is the best mix of these objectives can be put as the single objective problem of maximizing profits.

In practice, one generally does not know what the values of different objectives should be. It is also unlikely that these should be constant over the entire range of achievement of these objectives. These facts imply that one should not be particularly concerned about—nor attach much significance to—the numbers used as weights. One need not worry if the numbers do not appear realistic, nor spend effort in the exercise of determining what the weights ought to be; they are simply means that help define the noninferior set.

Kuhn and Tucker demonstrated that the solution to the weighted optimization $\sum w_k Y_k$ defines a noninferior solution to the multiobjective problem, provided some minimal sign conventions were respected. (This result is an extension of their optimality conditions, see Section 3.3). The only limitation is that increasing quantities of each objective Y_k should be desirable, and each weight should correspondingly be positive, $w_k \geq 0$. The point of these conventions is to insure that increases in the weighted sum $\sum w_k Y_k$ always correspond to an increase in desirability. In practice, this limitation poses no difficulty since any objective can be interpreted so that increases are more desirable; for example, if we were concerned with minimizing lives lost through some catastrophe such as an earthquake, one could redefine the objective as maximizing the number of lives saved through protective means (see discussion in Section 5.2).

The way the weighting method generates the noninferior solution is also conveniently shown graphically. Again, consider a situation with two objectives Y_1 and Y_2. As Figure 8.7 shows, isoquants of the quantity to be maximized by the weighting method, that is, the weighted sum $(w_1 Y_1 + w_2 Y_2)$, are simply straight lines. The desired maximum is the feasible design that reaches the highest isoquant, as point E does in Figure 8.7. This design must be a noninferior solution, as can be seen. Repetition of this process for different sets of weights defines additional noninferior solutions.

The weighting method has a couple of computational advantages over the constraint method. Its calculations are somewhat simpler since the constraint method increases the number of constraints, which is a basic determinant of the difficulty of the calculations (see Section 5.4). The weighting method also does not waste calculations since there will always be a feasible solution, whatever the weights used. The constraints method, on the other hand, may cycle through sets of constraints for which there is no feasible solution (see Section 8.3).

The major disadvantage of the weighting method is that it may not provide the analyst with a complete description of the noninferior solutions. Most obviously, this method may miss potentially significant features. As Figure 8.8 illustrates, the weighting method can simply skip over reentrant portions of the feasible region. It may thus portray the set of noninferior solution as a convex region even if it is not, and may suggest feasible solutions where none exist.

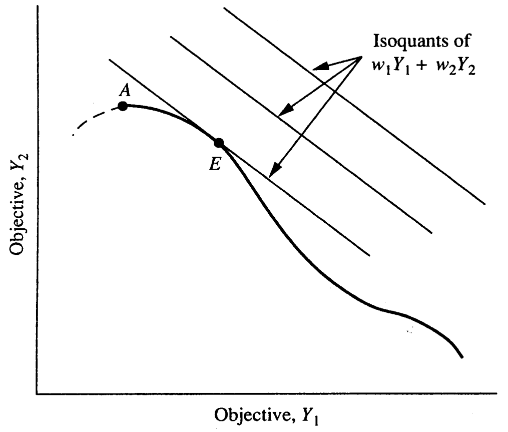

FIGURE 8.7
The weighting method defines a set of noninferior solutions by systematically varying weights on the objective.

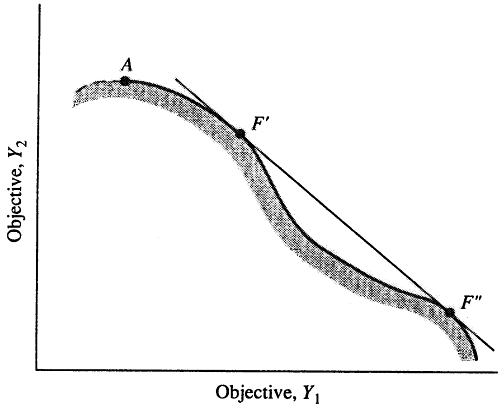

FIGURE 8.8
The weighting method has the disadvantage of skipping over reentrant portions of the feasible region, such as that between F' and F''.

A secondary disadvantage of the weighting method is that it makes it difficult for the analyst to define the noninferior solutions with equal coverage throughout the feasible region. The analyst cannot determine the spacing of the points defined on surface, as can be most easily done with the constraint method. Constant incremental changes in the weights used do not translate into an equal grid across the space of noninferior solutions, as Figure 8.9 suggests.

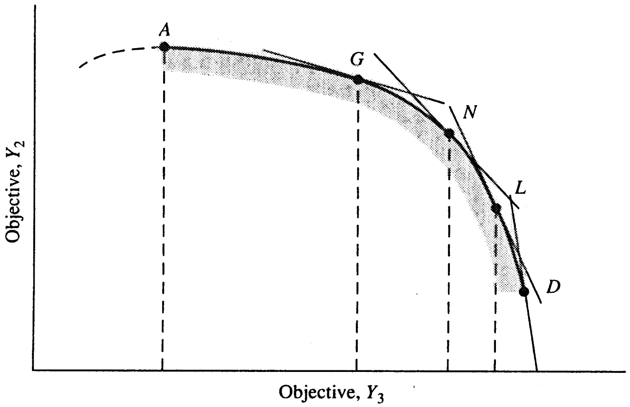

FIGURE 8.9
The weighting method makes it difficult for the analyst to control the spacing with which the noninferior solutions are defined.

The step-by-step procedure for applying weighting method is generally similar to that for the constraint method. In detail:

1. Optimize for each constraint by itself, to determine the range of noninferior solutions. This is conveniently done by solving the weighted sum of the objectives with all except one w_k equal to zero.
2. Optimize over two constraints, with two w_k not equal to zero. This defines the edges of the space of noninferior points.
3. Proceed as above, using successively more w_k not equal to zero, for as many dimensions as desirable or necessary.

Throughout this process it is useful to recognize that it is the relative values of w_k that count, not their absolute size. In two dimensions, for example, the weighted sum $(3Y_1 + 2Y_2)$ defines the same noninferior solutions as $(30Y_1 + 20Y_2)$; graphically the slopes of the isoquants in Figure 8.7 would be the same. This fact means that the analyst can simply vary the w_k between 0 and 1.

The weighting method is particularly easy to apply when the alternative solutions are defined individually, as in the case for the design of the Second Sydney Airport (see box in Section 8.2). This is because the characteristics of the different feasible designs can easily be placed in the spread-sheet programs now commonly available in business computers, and it is then easy to assign relative weights and define the corresponding noninferior solutions. Special programs are also available for this task.

The choice between the weighting and the constraint method depends on the specifics of the system being designed. In general, the constraint method is preferable because it gives a more reliable description of the noninferior solutions. When the computational costs are a big burden, however, the weighting method may be preferred. It may also be selected when a spread-sheet analysis is possible. Whenever a weighting function is chosen, its results should be examined carefully to determine if, within the range of designs that may be chosen, the weighting method has misrepresented the feasible region.

8.5 DISPLAY MECHANISMS

The identification of the noninferior solutions is a crucial element of the analysis of a multiobjective problem. This is so because it focuses attention on the small set of possible designs that are worth considering.

The question is, what does the analyst do with the results? How can they be used to lead to the final selection of the preferred design? How can they be discussed? How should we present the results of a multiobjective analysis? The answers depend on whether the analyst is dealing with two, three, or more objectives.

For the two-objective problems, the most effective presentation is a simple graph of the kind used throughout this chapter. Clients usually have little difficulty understanding such figures. They also typically appreciate the fact that these presentations allow them to see what they would have to give up on one objective to obtain an improvement on the other. These trade-offs help people select the final design.

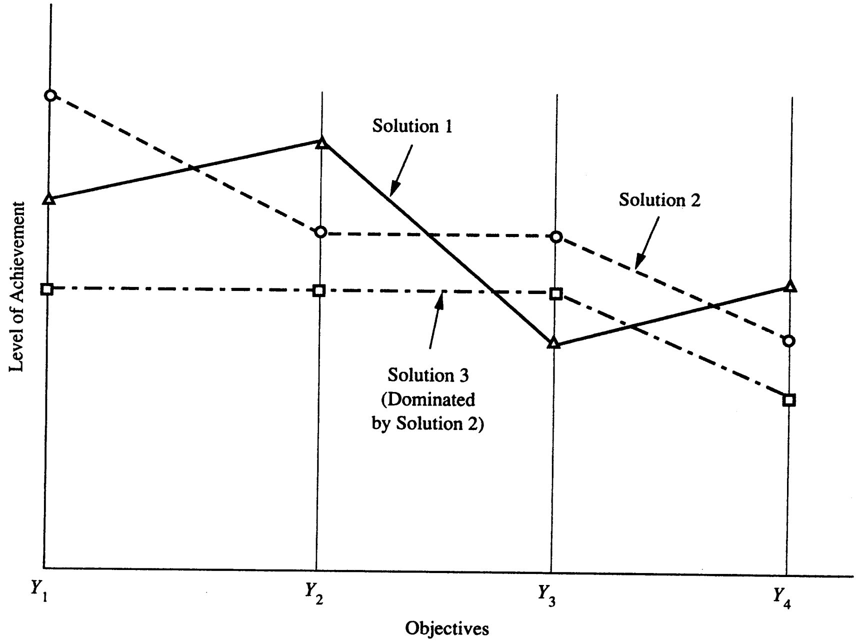

FIGURE 8.10
A profile easily displays the performance of designs on many objectives.

For many dimensions, a simple graph is no longer possible and an entirely different mechanism must be used. The one that appears easiest to understand is the *profile*. This is a simple device in which the level of attainment of each objective is scaled vertically along distinct points on the horizontal axis. This may be done either as points on a line or as a bar chart. The performance of any specific noninferior solution then appears as a wavy horizontal line as Figure 8.10 indicates. Dominated solutions will have lines completely under some other solution; these may be discarded, unless they need to be specifically included for comparison. This kind of figure becomes confusing as more solutions are displayed simultaneously. This presentation is thus most effective when it compares only a few of the possible choices.

For three dimensions, one can either use profiles or an extension of the two-dimensional graph. The graph is best suited to technical audiences, used to this medium. The graph represents a series of slices through the three-dimensional space of possibilities, parallel to the plane of two objectives, and orthogonal to the third. Figure 8.11 illustrates the idea.

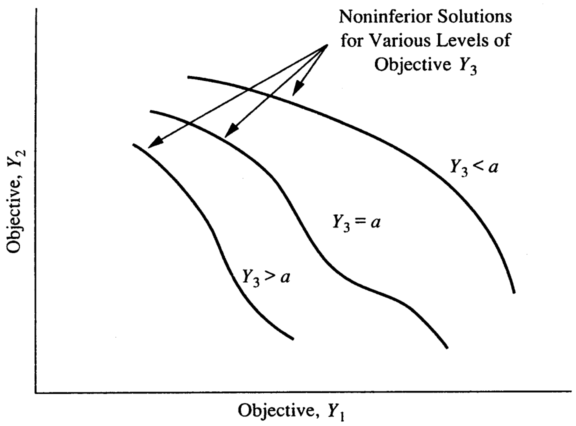

FIGURE 8.11
A graphical representation of three-dimensional, noninferior solutions.

8.6 APPLICATION

The optimal design may in some cases become obvious once the noninferior solutions are defined. This section explains this possibility and illustrates it by an application of multiobjective optimization to the design of a large system.

The application concerns the development of an important river basin in Argentina, the Rio Colorado. This river flows east from the Andes all the way across the country, passing through four distinct regions. The plans for the river involve the whole range of projects conceivable for a river basin: flood-control dams, hydroelectric projects, agricultural diversion canals, and industrial uses. Colleagues at MIT worked on this project in the early 1970s as part of their research into the practice of multiobjective optimization.

Two objectives were central to the discussion. Each had its special constituency and advocates. On the one hand, both the international agencies lending money for the project and the national government wanted to make sure the investments were profitable: they wanted to maximize the contribution to national income. The regional authorities in the river basin were, on the other hand, concerned with fairness in the distribution of the water and its benefits. They were unwilling to let any province have either an excessive or an inadequate share of the water even if that allocation did not maximize national income.

The two objectives could be represented mathematically fairly directly. The contribution to national income was simply the difference between the benefits and costs (with due allowance for interest and similar expenses, see Chapter 11). Fairness, in this context, could be represented by the relative allocation of water to each province: the fairest case being when each province received an equal amount.

The analysis itself centered around a large linear program where the allocations of water to projects, and thus to each province, were the principal decision variables. In this context the constraint method of multiobjective optimization was the better choice. In detail, the analysis maximized contributions to national income with progressively more stringent constraints on the fairness of the water distribution to the provinces.

The noninferior solutions defined by the analysis appear in Figure 8.12. The result was a curve with a distinct *"knee ,"* a salient portion of the feasible region of designs. This phenomenon is actually quite common. It is particularly interesting because it represents a special situation in which the noninferior solutions by themselves define the optimal design.

When the noninferior solutions have a "knee," the optimal design is almost always around this region. This is because all concerned will recognize that they can each get most of what they want. They will also recognize that they can achieve significant gains on one objective at little cost in terms of the others. Individuals can thus feel comfortable with this choice even when they would not feel happy about having to decide on the relative value of different objectives. Groups with a variety of different interests, and relative preference for the objectives, can likewise agree that solutions around the knee are satisfactory, even when they cannot agree among themselves on the relative importance of the several objectives. Thus it was for the plans for the Rio Colorado: as a result of the multiobjective optimization, the regional, federal, and international authorities agreed on a design represented by the knee of the noninferior solutions.

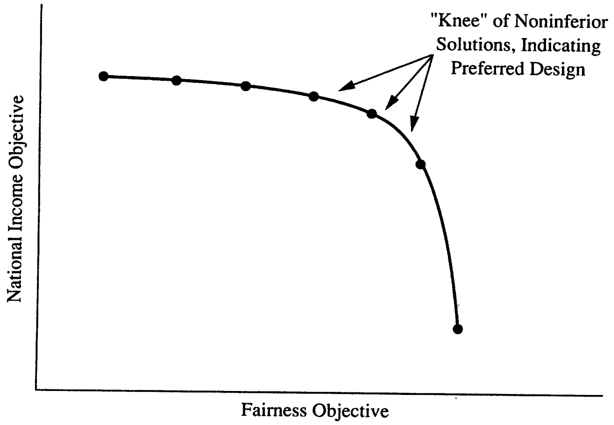

FIGURE 8.12
Noninferior solutions for the development of water resources in the Rio Colorado basin in Argentina.

REFERENCES

Giocoechea, A., Hansen, D. R., and Duckstein, L. (1982). *Multiobjective Decision Analysis with Engineering and Business Applications*, Wiley, New York.

Ignizio, J. P. (1982). "Multiobjective Linear Programming," Part 4, *Linear Programming in Single and Multiple-Objective Systems*, Prentice-Hall, Englewood Cliffs, NJ.

PROBLEMS

8.1. *Personal Objectives*
Imagine that you are in the market for a car. Consult the automobiles for sale section of a major newspaper. Select a popular model for which many offers are available. Plot these alternatives in terms of their mileage and cost. In terms of these objectives alone, which are the dominated, excluded, extreme, and compromise solutions?

8.2. *The Curve*
NIS Consultants has evaluated proposed sites for a municipal sewage facility along two objectives: entire fecal fumigation (EFF) and environmental quality improvement (EQUI). The optimal performance of each site is given by the following (EFF, EQUI) pairs:

A	(20, 135)	B	(75, 120)	C	(90, 100)	D	(35, 1050)
E	(82, 250)	F	(60, −50)	G	(60, 550)	H	(78, 500)
I	(70, 620)	J	(10, 500)	K	(40, 350)	L	(30, 800)
M	(95, −200)	N	(55, 220)	O	(40, −80)	P	(30, 500)
Q	(30, 900)	R	(60, 950)	S	(80, −150)	T	(45, 550)
U	(25, 1080)	V	(70, 800)	W	(63, 450)		

(*a*) Identify the dominated, excluded, and noninferior solutions.
(*b*) Which do you classify as extreme and compromise solutions? Discuss your answer.

8.3. *Optimizing*
Identify the noninferior solutions of Problem 8.2 by
(*a*) Maximizing EFF subject to EQUI $\geq b$ where $b = 200, 400, 600, 800$, and 1000 in turn.
(*b*) The weighting method, assigning relative values to (EFF, EQUI) of (20, 8), (40, 6), (60, 4), and (80, 2).

8.4. *Another Computer Run*
See Problem 6.12. Assume that the customer really thinks in terms of two objectives, minimization not only of cost, but also of impurities. Trace the noninferior solutions for these two objectives by
(*a*) The constraint method.
(*b*) The weighting method.

8.5. *Analysis*
Assuming that the set of noninferior solutions for the objectives X and Y can be described by the function: $Y = 50 - X^2$; describe the noninferior solutions by
(*a*) Maximizing Y subject to $X \geq 1, 2, 3, 4$, and 5
(*b*) Maximizing $Z = w_x X + w_y Y$

CHAPTER 9

SYSTEMS OPTIMIZATION PROCEDURE

9.1 THE PROBLEM

No single procedure can deal completely with all aspects of a system. Any single method must, in order to perform, make some assumptions about the real problem, must simplify it to some degree. It is thus likely to leave out some considerations that may be important overall.

As systems analysts, having the responsibility for a careful investigation of the entire situation, we must incorporate all the important elements. The question is, how can this be done efficiently? This chapter presents a procedure for using all the elements of optimization to achieve a best design.

9.2 DESIGN PROCEDURE

The recommended procedure for systems design consists of four main steps. These are each explained in detail in what follows. They are:

1. *Screening* of the feasible solutions to obtain a small set of noninferior solutions.
2. *Sensitivity Analysis* of these best solutions, to determine their performance in realistic situations.
3. *Dynamic Analysis* to establish the optimal pattern of development over time.
4. *Presentation*, the organization of the final result in a way that makes sense to the client.