CHAPTER XI: MULTI-OBJECTIVE PROGRAMMING

Optimization of a single objective oversimplifies the pertinent objective function in some potential mathematical programming application situations. Arguments can also be made following Simon that optimization is not as appropriate as statisficing. These two statements introduce the general topic of multiobjective programming. Multiobjective programming formally permits formulations where: a) solutions are generated which are as consistent as possible with target levels of goals; b) solutions are identified which represent maximum utility across multiple objectives; or c) solution sets are developed which contain all nondominated solutions. Multiple objectives can involve such considerations as leisure, decreasing marginal utility of income, risk avoidance, preferences for hired labor, and satisfaction of desirable, but not obligatory, constraints.

A discussion of this area requires some definitions. An objective is a measure that one is concerned about when making a choice among the decision variables (something to be maximized, minimized or satisfied like leisure, risk, profits, etc.). A goal implies that a particular goal target value has been chosen for an objective.

We will use "multiple objective programming" to refer to any mathematical program involving more than one objective regardless of whether there are goal target levels involved. Note, the literature contains conflicting definitions (see Blake and McCarl; Ignizio [1978,1983]; Romero [1989, 1991]). For example: a) goal programming has been used to refer to multiple objective problems with target levels; b) multiobjective programming has been used to refer to only the class of problems with weighted or unweighted multiple objectives; c) vector maximization has been used to refer to problems in which a vector of multiple objectives are to be optimized; and d) risk programming has been used to refer to multiobjective problems in which the objectives involve income and risk.

Multiobjective programming involves recognition that the decision maker is responding to multiple objectives. Generally, objectives are conflicting, so that not all objectives can simultaneously arrive at their

optimal levels. An assumed utility function is used to choose appropriate solutions. Several fundamentally different utility function forms have been used in multiobjective models. These may be divided into three classes: lexicographic, multi-attribute utility and unknown utility.

The lexicographic utility function specification assumes the decision maker has a strictly ordered preemptive preference system among objectives with fixed target levels. For example, a lexicographic system could have its first priority goal as income of not less than $10,000; the second priority as leisure of no less than 20 hours a week; the third as income of no less than $12,000, etc. This formulation is typical of "goal programming models." (Charnes and Cooper (1961); Lee). The various goals are dealt with in strict sequential order - higher goals before lower order goals. Once a goal has been dealt with (meeting or failing to meet the target level), its satisfaction remains fixed and the next lower order goal is considered. Consideration of the lower level goals does not alter the satisfaction of higher level goals and cannot damage the higher level goals with respect to target level attainment.

Multi-attribute utility approaches allow tradeoffs between objectives in the attainment of maximum utility. The most common form involves maximization of the sum of linearly weighted objectives. This type of formulation has been used by Candler and Boeljhe; and Barnett, Blake and McCarl.

The third utility approach involves an unknown utility function assumption. Here the entire Pareto efficient (nondominated) solution set is generated so that every solution is reported wherein one of the multiple objectives is as satisfied as it possibly can be without making some other objective worse off (Steur, Geoffrion (1968)).

11.1 Formulations

All of the above utility functions can be expressed in terms of the following problem. Assume there are multiple objectives which are given by

GX

where there are J decision variables (X) and R objectives. Thus, the matrix G is of dimension R by J while X is J by 1. These objectives can also be expressed in summation notation as

$$\sum_j g_{rj} X_j \qquad \text{for all } r$$

When target levels are added, the objectives become

$$GX \geq T$$

The general goal programming problem, then, is as follows:

Select X so that we

optimize or achieve	GX
s.t.	$AX \leq b$
Possibly subject to	$GX \geq T$
	$X \geq 0$

Here the normal LP objective function is replaced by a more general function which permits use of different utility function forms (it is difficult to write the Pareto utility function in this form). The problem involves selection of the X's. The selection is driven by either optimization of some weighted tradeoff of objectives or through lexicographic achievement of various goal target levels. The specific formulations used for each of the above utility function specifications are given below.

11.1.1 Lexicographic Utility - Target Values

Perhaps the first application of multiobjective programming was the Charnes and Cooper goal programming formulation. The formulation is: Select X so that

$$AX \leq b$$
$$X \geq 0$$

and so that the goals are handled in the following priority order:

$$\sum_j g_{1j} X_j \geq T_1$$

then

$$\sum_j g_{2j} X_j \geq T_2$$

on through to

$$\sum_j g_{Rj} X_j \geq T_R$$

for the R^{th} and last goal.

The lexicographic multiple objective formulation is not precisely a LP problem. It has many structural characteristics in common with a LP problem; however, a conventional objective function is not defined, nor can a single LP formulation reflect imposition of the sequential ordering of the goals. Rather, an iterative procedure is needed (Lee). Essentially, the approach is to solve problems for each of the goals sequentially. When considering the i^{th} goal solve the problem

$$\begin{aligned}
&\text{Min} \quad w_i \\
&\text{s.t.} \quad w_r + \sum_j g_{rj} X_j \geq T_r \qquad \text{for all } r \\
&\qquad\quad \sum_j a_{mj} X_j \leq b_m \qquad \text{for all } m \\
&\qquad\quad w_r \leq w_r^* \qquad \text{for all } r < i \\
&\qquad\quad w_r \leq \infty \qquad \text{for all } r \geq i \\
&\qquad\quad x_j \geq 0 \qquad \text{for all } j \\
&\qquad\quad w_r \geq 0 \qquad \text{for all } r
\end{aligned}$$

The new variable w_r gives the amount that the goal level ($\sum g_{rj} X_j$) is less than the target value (T_r). When i = 1 the problem minimizes the shortfall from the first goal target level, subject to the LP constraints. One of two solution situations will then occur. Either the optimum value of w_1 (denoted w^*_1) equals 0, indicating full satisfaction of the first goal, or $w^*_1 \neq 0$, indicating the goal cannot be fully satisfied. Subsequently, a second problem is solved. This problem is virtually identical to the first, except w_2 is minimized and a constraint is appended indicating that w_1 cannot be any worse than the optimum value obtained at the end of the solution of the first stage (w_1^*). This requires: 1) if goal 1 was met before, then goal 1 will continue to be met (i.e., w_1 must be less than or equal to zero); or 2) if goal 1 was not met, then the deviation from goal 1 will not get bigger than the minimum deviation obtained at the previous iteration. Thus, the prior objective is constrained

to be no worse off than it was before. This problem, in effect, explores alternative optimums where we hold the prior objectives at their optimum values, then try to optimize the satisfaction of the subsequent objectives.

This procedure is executed for all R goals where different deviation variables are minimized at each stage and a constraint is added holding all previous deviations to maximum values prohibiting the earlier objectives from becoming worse off. Lee presents a more comprehensive discussion of the procedure while the example below gives an empirical application.

11.1.2 Utility Tradeoff Model

The second utility function type involves tradeoffs between various objectives. Such problems can be formulated as conventional linear programs. There have been two alternative formulations of this problem. They differ in their assumptions about target levels. The first formulation (appearing for example in Candler and Boeljhe) does not take into account target levels, maximizing the weighted sum of the quantities of each objective. This is

$$\begin{aligned} \text{Max} \quad & \sum_r c_r q_r \\ & -\sum_j g_{rj} X_j + q_r = 0 \qquad \text{for all } r \\ & \sum_j a_{mj} X_j \le b_m \qquad \text{for all } m \\ & X_j, \; q_r \ge 0 \qquad \text{for all } j \text{ and } r \end{aligned}$$

where c_r is the weight which expresses the importance of the r^{th} objective in the context of the decision maker's total utility. The c_r coefficients would be in utility units per unit of the r^{th} objective achieved; q_r is the amount of r^{th} objective in the optimal solution and may often be multiplied by a magnitude normalizing factor.

The objective function, maximizes multi-dimensional utility summed across all objectives. Each objective is weighted. The second equation sums the level of each objective into the variable q_r. The third represents resource availability limitations, and the fourth expresses nonnegativity constraints.

The second weighted tradeoff formulation embodies goal target levels. The formulation is

$$\text{Max} \quad \sum_r c_r^+ q_r^+ + \sum_r c_r^- q_r^-$$

$$\text{s.t.} \quad \sum_j g_{rj} X_j - q_r^+ + q_r^- = T_r \qquad \text{for all } r$$

$$\sum_j a_{mj} X_j \leq b_m \qquad \text{for all } m$$

$$X_j, \quad q_r^+, \quad q_r^- \geq 0 \qquad \text{for all } j \text{ and } r$$

where T_r is the goal target value for objective r, c_r^+ is the weight attached to overachieving objective r relative to its target, c_r^- is the weight attached to underachieving objective r relative to its target, q_r^+ is the amount the target for objective r is overachieved, and q_r^- is the amount the target for objective r is underachieved.

This is again a linear program. The formulation is adapted from Lee and is used in Barnett, et al. (1982).

11.1.3 Unknown Utility Function

The other approach to multiobjective programming involves an unknown utility function assumption. Instead, the entire nondominated set of alternatives is generated. The formulation for this approach is exactly like the first one under the weighted tradeoff section above except that all possible weights are utilized in the problem. This particular approach has been studied extensively, (see, for example, the bibliographies in Steuer; and Ignizio, 1983) but does not appear to be very empirically useful.

11.2 Examples

A common example is used to demonstrate the above formulations. However, we will omit coverage of the unknown utility function model as its solutions would be rather extensive and its use has been limited.

The example builds upon the chair example used in the resource allocation section of the linear programming chapter. Suppose that the firm is interested in profit, idle labor, and idle lathe time in formulating its goals. Thus, the firm values leisure and slack lathe time as well as profits.

11.2.1 Lexicographic Formulation

The lexicographic formulation will be based on four different goals. The first goal is that the profit be greater than $9,000, the second that idle labor be greater than or equal to 30 hours. The third is that idle lathe time be greater than or equal to 25 hours and the fourth, that profit be greater than or equal to $9,500. The formulation of this problem with the deviation variables included is given in Table 11.1 where p gives the profit, S_1, S_2, and S_3 the idle lathe and labor time, and d_1 through d_4 the goal deviations.

The approach in solving this problem is as above. First, the deviation from the $9,000 profit target level will be minimized. This results in a deviation of zero. Subsequently, when minimizing the deviation from the idle labor goal, a constraint is entered allowing zero deviation from a profit level of $9,000. Then we precede to consider idle lathe time holding the profits and idle labor goal achievement constant.

The GAMS implementation is given in Table 11.2 and file LEXICO. We have introduced profit accounting in line 67 through 69. Also, we have introduced slack variables accounting for idle resources (lines 71-73). The four goals are modeled in lines 75 through 79 where the idle large and small lathe time is added and set equal to the goal level in line 79. In turn, line 80 relates the goal levels to their associated targets. Namely, the goal achievement level plus a deviation variable to make up the shortfall is set greater than or equal to the target. The constraints defined in line 82 then restrict the goal shortfall to be less than or equal to an acceptable deviation level.

The lexicographic approach is implemented in lines 85 through 96. The initial allowable deviations are set to a large number in line 30. The loop from lines 86 through 95 sets up the problem for each goal changing the objective function weights then solves. Subsequently, the maximum allowed deviation for future iterations is set equal to the shortfall. Report writing statements (lines 92-94) summarize the solution.

The solutions are presented in Table 11.3. Four solutions are involved, each arising when the goals are individually considered. In the first solution, the $9,000 profit goal is easily attained while the labor and lathe time goals fall short with 4.024 units of idle labor and 17.073 units of idle lathe time. This plan has 12.195 units of functional normal chairs being produced and 108.337 units of fancy normal chairs.

In turn, when the idle labor problem is setup with the profit deviation restricted ($d_1 \leq 0$), the solution exhibits profits of $9,482 full attainment of the idle labor goal but the lathe goal is now 20.641 units short and we are $18.421 short of the $9,500 profit goal. This is achieved with production of 115.296 fancy chairs and 3.289 fancy chairs with maximum large lathe use.

Now turning to the idle lathe time problem, we constrain $d_1 \leq 0$ and $d_2 \leq 0$ then solve. Here we fully attain the $9,000 profit and idle labor goals but fall 4.337 units short of the idle lathe time goal. Profit is now $9,000. This plan is achieved by producing 15.152 units of functional normal chairs and 99.811 units of fancy normal chairs.

Finally, turning our attention to the last goal we find that we can not make any progress on it and have a solution which is equivalent to the solution in the step before.

The above results show the action of a lexicographic solution. Namely, the $9,000 profit goal was satisfied and held satisfied throughout the process while the $9,500 profit was not considered until the last step and the $9,500 profit goal was only pursued when the other goals had been held at their satisfaction levels and as a consequence no progress could be made.

11.2.2 Weighted Tradeoff - No Targets

One version of the weighted tradeoff formulation does not contain targets. We will follow the theoretical formulation but will also include weights and normalizing factors. The normalizing factors are used so the goal magnitudes are approximately equal. Namely, we divide the profit goal through by 10,500, the labor goal by the labor resource availability and the total lathe goal by the lathe resource availability. This then will convert all of these goal numbers that range over 0 to 1 and allow relative weights to be used. In turn, the weight for profit equals one, while the weight for idle labor and lathe time both equal 0.4. The resultant formulation is in Table 11.4 and the GAMS instructions in Table 11.5 (file WEIGHT). Notice in the GAMS formulation the weight and normalization data are defined in lines 23-27, while the goal setup is essentially the same as in the previous example. The main variant is that the goal levels are normalized in lines 67-69 and the objective function is the sum of the goal weights times the goal levels (line 57).

The resultant solution to this problem is that 2.44 functional normal chairs are made while 112.2 fancy normal chairs are made and 6.829 fancy chairs with maximum use of the large lathe. The profit goal is achieved at a .92 level of the 10,500 normalization level ($9,674) while the labor goal is achieved at a 0.217 level leaving 27.06 hours of labor idle. The idle lathe time variable equals 0.

11.2.3 Weighted Objective and Targets

Finally, let us consider the unified target levels and weighted tradeoff formulation. Here we will use the same target levels as in the lexicographic model, but introduce weights where we value profit at a weight of one if it is more than the first goal and at -10 units if it is less than that. Idle labor and lathe time are weighted at 0.1 if they are greater than their target level and -0.4 if less. Finally, the profit in excess of $9,500 is valued at .9 and less than that at -1. The goals are normalized by multiplying the deviation variables by the target value. This formulation is portrayed in Table 11.6 with the GAMS instructions in Table 11.7 (see file WTTAR). Notice in this formulation both positive and negative deviations are defined and the objective function both reflects shortfalls and excesses. The solution shows profit equals $9,000, idle labor 25.08 units, and idle lathe time 25 units. This makes for a labor shortfall of .164 units and a profit shortfall of .05.

11.3 Choice Among Formulations

An important question given the alternative formulations is: which one should be used for a problem? There are several general considerations involved in choosing among these formulations.

The first consideration is solver availability. Traditionally, the undominated approach requires a specially adapted solver. Such adaptations have been implemented (Steuer) although they are not routinely available for more than small problems. In addition, the undominated set approach can be quite expensive computationally (Steuer). Thus, this criteria favors the weighted tradeoff or goal formulations where one could use standard algorithms.

Second, one needs to consider the required amount of decision maker contact, particularly if the

model is being used for predictive purposes. The undominated solution alternative would not be satisfactory in a limited contact setting as it requires active choice of the "best" strategy by the decision maker. This would be particularly troubling in many predictive exercises as the methods would generate a large number of answers, any of which could be the solution depending on decision maker choice.

Third, the treatment of goal target levels is difficult in comparative static exercises. It is difficult to: 1) specify goal target levels and 2) conclude that the goal target levels do not depend upon the resource base. Certainly, an income goal is easier to satisfy if the resource base is augmented. Thus, the lexicographic utility function formulations are not scale neutral. Many LP models are built to do comparative static studies, such as what would happen if larger equipment were available, more land area, labor, etc. This causes difficulties in using comparative statics with the weighted tradeoff model using fixed goal target levels or the lexicographic model.

Fourth, weights are difficult to discover. A complex questioning or calculation procedure is required (we review procedures for establishing weights below). Further, the use of constant weights over the entire domain of goal levels may be questionable. Procedures for including diminishing marginal utility would involve quadratic or separable programs as covered in other chapters.

All things considered, we prefer the weighted tradeoff model due to its consistency with our perceptions that individuals are willing to establish tradeoffs between objectives on the margin and that most models will be used for comparative static analysis.

11.4 Developing Utility Functions

Conceptually, multiobjective programming problems look attractive. However, assuming one knows the objectives, it is difficult to specify the utility structure. Clearly, this is not a problem with the undominated solution procedure as there are no weights. On the other hand, one puts all the alternative solutions in the decision maker's lap, which could involve thousands of solutions. Here we address how to find the utility function for the other formulations.

The easiest system to use is the lexicographic system, where one has to establish goal targets and the preemptive order. Targets such as the minimum amount of debt service plus consumption or the desired length of a vacation can be used. However, one must be careful in using these targets in comparative static analysis, as the relative ability to satisfy the targets changes with alterations in the resource base. Also, one must ask whether tradeoffs are in order.

Weights are more difficult and are the subject of the bulk of the discussion herein. The first way of specifying weights is to take decision makers' past actions and then through a grid search over alternative weights, choose weights so as to minimize deviations of the model solution from observed actions. An example is given by Brink and McCarl for a risk analysis problem. We know of no formal attempt to do this in other than a risk analysis framework. The advantage here is that one obtains weights which are somehow consistent with revealed preferences. However, in a LP problem there is a range of weights which will generate the same solution. It is therefore possible that the proper set of weights is somewhere within the range, but that the wrong set of weights is chosen. In turn, this set of weights could lead to dramatically different behavior in a comparative static study.

A third procedure involves survey techniques. Here decision makers are asked questions about the relative importance of objectives and then through a scaling procedure a set of objective weights is obtained. This was done by Barnett, Blake and McCarl; Smith and Capstick; and Harman, et al. Two difficulties arise with this procedure. First, there is no assurance that the surveys generate results which mimic actions in actual situations. Second, it is difficult to translate the results into the proper specification of the programming model objective.

The fourth procedure we discuss was proposed by Candler and Boehlje and applies to the weighted tradeoff models. The procedure involves interaction with the decision maker and is based on revealed preference. To begin the process, modelers choose an initial set of weights and present the answers to the decision maker. In turn, the decision maker expresses preference for a change in the objective satisfaction levels (i.e., the decision maker could argue that there was insufficient income and excessive risk). The

modeler would then alter the weights on those objectives and rerun the model. The process would continue until the decision maker was satisfied. This approach has the advantage of obtaining goal weights consistent with the decision maker's preferences and the potential disadvantage of obtaining the proper output with the wrong set of goal weights, leading to improper solutions when one is doing further analysis.

Finally, we must comment that there is no real way to abstractly set up a multiple objective model. The weights for the multiple objectives clearly require interaction with the decision maker.

11.5 Shadow Prices

Much discussion has been devoted in previous chapters to shadow prices but little here. In this section we explore the meaning of shadow prices in a weighted multiple objective problem and derive meaningful shadow prices.

The shadow prices for a weighted multiobjective problem nominally give the marginal change in the weighted utility of a marginal right hand side change. The weighted utility is a multi-dimensional utility measure constructed as the sum of the individual objectives times their weights. However, one must ask how useful it is to know the expected change in this multi-dimensional utility function. This ordinarily would probably not be terribly useful as decision makers will be more interested in knowing what happens to the specific objectives. Mathematically, the effect on the specific objectives may be derived as follows. The restatement of the first weighted multiple objective function formulation is

$$\begin{array}{lrcl}
\text{Max} & \sum_r w_r q_r & & \\
\text{s.t.} & -\sum_j g_{rj} X_j + q_r = 0 & & \text{for all } r \\
& \sum_j a_{kj} X_j \leq b_k & & \text{for all } k \\
& X_j, \geq 0 & & \text{for all } j \\
& q_r \lessgtr 0 & & \text{for all } r
\end{array}$$

We may eliminate the equality constraints to get

$$\text{Max} \quad \sum_r w_r \left(\sum_j g_{rj} X_j\right)$$

$$\text{s.t.} \quad \sum_j a_{kj} X_j \leq b_k \quad \text{for all } k$$

$$X_j \geq 0 \quad \text{for all } j$$

Rearranging, we obtain

$$\text{Max} \quad \sum_j \left(\sum_r w_r g_{rj}\right) X_j$$

$$\text{s.t.} \quad \sum_j a_{kj} X_j \leq b_k \quad \text{for all } k$$

$$X_j \geq 0 \quad \text{for all } j$$

Note that each variable X_j has an objective function coefficient which is the sum of the weights times its relative goal contributions. This can be collapsed using

$$C_j = \left(\sum_r w_r \, g_{rj}\right)$$

or, in matrix terms,

$$C = WG$$

We now turn our attention to shadow prices, given by

$$U = C_B B^{-1}$$

The C_B terms within the multiobjective programming model are given by the multiplication of goal weights times the goal levels involved with the basic variables

$$C_B = W G_B = \begin{bmatrix} w_1 & w_2 \ldots w_r \end{bmatrix} \begin{bmatrix} g_{11}^b \ldots\ldots g_{1M}^b \\ g_{21}^b \ldots\ldots g_{2M}^b \\ \ldots\ldots \\ g_{R1}^b \ldots\ldots g_{RM}^b \end{bmatrix}$$

where the superscript b on the g terms refer to the coefficients associated with the basic variable in the various objectives.

The shadow price term can be rewritten as

$$U = C_B B^{-1} = W G_B B^{-1}$$

Here, the term $G_B B^{-1}$ gives an unweighted set of shadow prices, each column of $G_B B^{-1}$ shows how each

objective function is affected by right hand side changes. Practically, these shadow prices could be obtained analytically from small problems and from larger problems by employing the solver starting from the optimal solution which can do a pricing pass but does not optimize (doing zero iterations).

11.5.1 Example

Suppose we were to maximize the following multiple goal objective problem

and we are willing to assume that the weights are each 1. The problem with the composite objective

$$\text{Max}\begin{Bmatrix} 10X_1 + X_2 \\ -7X_1 + X_2 \end{Bmatrix}$$

$$\begin{aligned} \text{s.t.} \quad & X_1 - X_2 \le 5 \\ & X_1 + X_2 \le 10 \\ & X_1 , X_2 \ge 0 \end{aligned}$$

function then becomes

$$\begin{aligned} \text{Max} \quad & 1(10X + X_2) + 1(-7X_1 + X_2) = 3X_1 + 2X_2 \\ \text{s.t.} \quad & X_1 - X_2 \le 5 \\ & X_1 + X_2 \le 10 \\ & X_1 , X_2 \ge 0 \end{aligned}$$

Solving this problem (see file SHADOW) we find that our solution consists X_1=7.5 and X_2=2.5. equals 2.5.

The basis matrix and its inverse are

$$B = \begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix} \qquad B^{-1} = \begin{bmatrix} .5 & .5 \\ -.5 & .5 \end{bmatrix}$$

The composite shadow prices are

$$C_B B^{-1} = \begin{bmatrix} 3 & 2 \end{bmatrix} \begin{bmatrix} .5 & .5 \\ -.5 & .5 \end{bmatrix} = \begin{bmatrix} .5 & 2.5 \end{bmatrix}$$

However, if we break this down we get

$$WG_B B^{-1} = \begin{bmatrix}1 & 1\end{bmatrix} \begin{bmatrix}10 & 1\\ -7 & 1\end{bmatrix} \begin{bmatrix}.5 & .5\\ -.5 & .5\end{bmatrix} = \begin{bmatrix}1 & 1\end{bmatrix} \begin{bmatrix}4.5 & 5.5\\ -4 & -3\end{bmatrix}$$

where the last matrix gives the shadow prices in terms of individual objectives. Thus, the change of one unit in the right hand side will increase the first objective by 4.5 while decreasing the second objective by 4. These shadow prices are more meaningful than the weighted shadow prices as they tell the implications of resource changes for each objective. Note that the weighted problem shadow prices are simply the individual weights times the shadow prices of this problem.

Preckel et al. proposed an alternative where they estimate relevant shadow prices by dividing through the by shadow prices on the individual objective accounting rows. However, this does not work as well as the theory above, as the shadow prices will be strictly proportional across the goals which need not happen in the general case (i.e. in the example above).

References

Burned, D., B. Blake and B.A. McCarl. "Goal Programming via Multidimensional Scaling Applied to Sengalese Subsistence Farms." American Journal of Agricultural Economics . 64 (1982):720-27.

Blake, B.F. and B.A. McCarl. "Goal Programming via Multidimensional Scaling Applied to Sengalese Subsistence Farming: A Reply." American Journal of Agricultural Economics . 65 (1983):832-33.

Brink, L. and B.A. McCarl. "The Expected Income-Risk Trade-Off Among Corn Belt Crop Farmers." American Journal of Agricultural Economics . 60(1978):259-63.

Candler, W. and M. Boehlje. "Use of Linear Programming in Capital Budgeting with Multiple Goals." American Journal of Agricultural Economics . 53(1977):325-330.

Charnes, A. and W.W. Cooper. Management Models and Industrial Applications of Applications of Linear Programming . Vol. 1. New York: John Wiley & Sons, 1961.

Charnes, A., W.W. Cooper and R.O. Ferguson. "Optimal Estimation of Executive Compensations by Linear Programming." Management Science . 1(1955):138-51.

Geoffrion, A.M. "Proper Efficiency and the Theory of Vector Maximization." Journal of Mathematical Analysis and Applications . 22(1968):618-30.

Harman, W., R. Hatch, V. Eidman and P. Claypool. "An Evaluation of Factors Affecting the Hierarchy of Multiple Goals." Technical Bulletin T-134, Oklahoma State Agriculture Experiment Station and USDA, June 1972.

Ignizio. "A Review of Goal Programming: A Tool for Multiobjective Analysis." Journal of Operational Research Society . 29(1978):1109-1119.

_____. "Generalized Goal Programming: An Overview." Computers and Operations Research . 10(1983):277-89.

Lee, S.M. Goal Programming for Decision Analysis . Philadelphia: Auerbach, 1972.

Preckel, P.V., A.M. Featherstone, and T.G. Baker. "Interpreting Dual Variables for Optimization with Nonmonetary Objectives." American Journal of Agricultural Economics . 69(1988):849-851.

Romero, C. Handbook of Critical Issues in Goal Programming . Oxford, New York: Pergamon Press, 1991.

Romero, C. Multiple Criteria Analysis for Agricultural Decisions . Amsterdam, New York: Elsevier, 1989.

Simon, H.A. Models of Man . New York: John Wiley and Sons, 1954.

Smith, D. and D. Capstick. "Establishing Priorities Among Multiple Management Goals." Southern Journal of Agricultural Economics . 8(1972):37-43.

Steuer, R. "Vector Maximization Gradient Core Contraction Techniques." in Multiple Criteria Problem Solving . S. Zionts, ed. Berlin: Springer-Verlag, 1978, pp. 462-81.

Table 11.1. Tableau for Lexicographic Example

Min											w_1					
s.t.	$67X_1$	$+\ 66X_2$	$+\ 66.3X_3$	$+\ 80X_4$	$+\ 78.5X_5$	$+\ 78.4X_6$				$-\ p$					$=$	0
	$0.8X_1$	$+\ 1.3X_2$	$+\ 0.2X_3$	$+\ 1.2X_4$	$+\ 1.7X_5$	$+\ 0.5X_6$	$+\ S_1$								$=$	140
	$0.5X_1$	$+\ 0.2X_2$	$+\ 1.3X_3$	$+\ 0.7X_4$	$+\ 0.3X_5$	$+\ 1.5X_6$		$+\ S_2$							$=$	90
	$0.4X_1$	$+\ 0.4X_2$	$+\ 0.4X_3$	$+\ X_4$	$+\ X_5$	$+\ X_6$									$\leq$	120
	X_1	$+\ 1.05X_2$	$+\ 1.1X_3$	$+\ 0.8X_4$	$+\ 0.82X_5$	$+\ 0.84X_6$			$+\ S_3$						$=$	125
										p	$+\ w_1$				$\geq$	9000
									S_3			$+\ w_2$			$\geq$	30
							S_1	$+\ S_2$					$+\ w_3$		$\geq$	25
										p				$+\ w_4$	$\geq$	9500
											w_1				$\leq$	∞
												w_2			$\leq$	∞
													w_3		$\leq$	∞
														w_4	$\leq$	∞
	X_1 ,	X_2 ,	X_3 ,	X_4 ,	X_5 ,	X_6	S_1 ,	S_2 ,	S_3		w_1 ,	w_2 ,	w_3 ,	w_4	$\geq$	0
										p					$\gtrless$	0

Table 11.2. GAMS Formulation of Lexicographic Example

```
  5   SET             GOALS         GOALS IN THE ORDER THEY ARE TO BE MET
  6                                    /PROFIT1,LABOR,LATHETIME,PROFIT2/
  7                   PROCESS       TYPES OF PRODUCTION PROCESSES
  8                                    /FUNCTNORM , FUNCTMXSML , FUNCTMXLRG
  9                                    ,FANCYNORM , FANCYMXSML , FANCYMXLRG/
 10                   RESOURCE      TYPES OF RESOURCES
 11                                    /SMLLATHE,LRGLATHE,CARVER,LABOR/
 13     ALIAS(GOALS,GOAL) ;
 15   PARAMETER PRICE(PROCESS)       PRODUCT PRICES BY PROCESS
 16                      /FUNCTNORM  82, FUNCTMXSML  82, FUNCTMXLRG  82
 17                      ,FANCYNORM 105, FANCYMXSML 105, FANCYMXLRG 105/
 18             PRODCOST(PROCESS)    COST BY PROCESS
 19                      /FUNCTNORM  15, FUNCTMXSML  16  , FUNCTMXLRG  15.7
 20                      ,FANCYNORM  25, FANCYMXSML  26.5, FANCYMXLRG  26.6/
 21             RESORAVAIL(RESOURCE) RESOURCE AVAILABLITY
 22                      /SMLLATHE 140, LRGLATHE  90,
 23                       CARVER   120, LABOR    125/
 24             TARGET(GOALS)       GOAL TARGET LEVELS
 25                                    /PROFIT1  9000,LABOR  30,LATHETIME 25
 26                                     ,PROFIT2 9500/
 27              DEV(GOALS)   MAXIMUM DEVIATION BY GOAL
 28              WEIGHTS(GOALS)      WEIGHTS BY GOAL  ;
 30     DEV(GOALS)=999999;
 31     WEIGHTS(GOALS)=0.00001;
 33   TABLE RESOURUSE(RESOURCE,PROCESS) RESOURCE USAGE
 35                 FUNCTNORM   FUNCTMXSML  FUNCTMXLRG
 36  SMLLATHE          0.80       1.30        0.20
 37  LRGLATHE          0.50       0.20        1.30
 38  CARVER            0.40       0.40        0.40
 39  LABOR             1.00       1.05        1.10
 40   +                FANCYNORM   FANCYMXSML  FANCYMXLRG
 41  SMLLATHE          1.20       1.70        0.50
 42  LRGLATHE          0.70       0.30        1.50
 43  CARVER            1.00       1.00        1.00
 44  LABOR             0.80       0.82        0.84;
 46   POSITIVE VARIABLES
 47             PRODUCTION(PROCESS) ITEMS PRODUCED BY PROCESS
 48             IDLE(RESOURCE)     SLACK VARIABLES FOR RESOURCES
 49             GOALLEVEL(GOALS)    GOAL LEVELS
 50             PROFIT              TOTALPROFIT
 51             SHORTFALL(GOALS)    GOAL SHORTFALLS;
 52   VARIABLES
 53             GOALOBJ             GOAL OBJECTIVE;
 54   EQUATIONS
 55             OBJT                  OBJECTIVE FUNCTION
 56             PROFITACCT            PROFIT ACCOUNTING
 57             AVAILABLE(RESOURCE)   RESOURCES AVAILABLE
 58             IDLLABGOAL           IDLE LABOR GOAL
 59             PROFITGL1            PROFIT1 GOAL
 60             PROFITGL2            PROFIT2 GOAL
 61             LATHEGOAL             IDLE LATHE GOAL
 62             TARGS(GOALS)          GOAL TARGETS
 63             MAXSHORT(GOALS)       SHORTFALL LIMITS;
 64
 65    OBJT.. GOALOBJ =E= SUM(GOALS,WEIGHTS(GOALS)*SHORTFALL(GOALS)) ;
 66
 67   PROFITACCT..   PROFIT =E=
 68                SUM(PROCESS,(PRICE(PROCESS)-PRODCOST(PROCESS))
 69                              * PRODUCTION(PROCESS)) ;
 70
 71    AVAILABLE(RESOURCE)..
 72        SUM(PROCESS,RESOURUSE(RESOURCE,PROCESS)*PRODUCTION(PROCESS))
 73                      +IDLE(RESOURCE)   =E= RESORAVAIL(RESOURCE);
 74
 75    IDLLABGOAL..    IDLE("LABOR") =E= GOALLEVEL("LABOR") ;
```

Table 11.2. GAMS Formulation of Lexicographic Example (Continued)

```
 76    PROFITGL1..    PROFIT        =E= GOALLEVEL("PROFIT1");
 77    PROFITGL2..    PROFIT        =E= GOALLEVEL("PROFIT2");
 78    LATHEGOAL..     IDLE("LRGLATHE")+IDLE("SMLLATHE")
 79                            =E= GOALLEVEL("LATHETIME");
 80    TARGS(GOALS).. GOALLEVEL(GOALS) + SHORTFALL(GOALS) =G= TARGET(GOALS) ;
 81
 82    MAXSHORT(GOALS)..  SHORTFALL(GOALS) =L= DEV(GOALS);
 83
 84   MODEL RESALLOC /ALL/;
 85   PARAMETER   GOALDATA(GOAL,*,*)
 86   LOOP(GOAL,
 87        WEIGHTS(GOAL)=1.
 88
 89       SOLVE RESALLOC USING LP MINIMIZING GOALOBJ;
 90        DEV(GOAL)=SHORTFALL.L(GOAL);
 91        WEIGHTS(GOAL)=0.00001;
 92        GOALDATA(GOAL,GOALS,"ATTAIN")=GOALLEVEL.L(GOALS) ;
 93        GOALDATA(GOAL,GOALS,"SHORT")=SHORTFALL.L(GOALS) ;
 94       GOALDATA(GOAL,PROCESS,"XLEVEL")=PRODUCTION.L(PROCESS);
 95      );
 96     DISPLAY GOALDATA;
```

Table 11.3. Solution to Lexicographic Example

Goal Being Pursued	Solution Item	Idle	Goal Attainment	Goal Level	Goal Shortfall	Production Level
PROFIT1	PROFIT1	GOAL	9500.000	9500	0	
	.LABOR	GOAL	25.976	30	4.024	
	.LATHETIME	GOAL	7.927	25	17.073	
	.PROFIT2	GOAL	9500.000	9500	0	
	.FUNCTNORM	PROD				12.195
	.FANCYNORM	PROD				108.537
LABOR	.PROFIT1	GOAL	9481.579	9500	0	
	.LABOR	GOAL	30.000	30	0	
	.LATHETIME	GOAL	4.359	25	20.641	
	.PROFIT2	GOAL	9481.579	9500	18.421	
	.FANCYNORM	PROD				115.296
	.FANCYMXLRG	PROD				3.289
LATHETIME	.PROFIT1	GOAL	9000.000	9500	0	
	.LABOR	GOAL	30.000	30	0	
	.LATHETIME	GOAL	20.663	25	4.337	
	.PROFIT2	GOAL	9000.000	9500	500.000	
	.FUNCTNORM	PROD				15.152
	.FANCYNORM	PROD				99.811
PROFIT2	.PROFIT1	GOAL	9000.000	9500	0	
	.LABOR	GOAL	30.000	30	0	
	.LATHETIME	GOAL	20.663	25	4.337	
	.PROFIT2	GOAL	9000.000	9500	500.000	
	.FUNCTNORM	PROD				15.152
	.FANCYNORM	PROD				99.811

Table 11.4. Tableau for Weighted Tradeoff Example

Max											d_1	+ $.4d_2$	+ $.4d_3$		
s.t.	$67X_1$	+ $66X_2$	+ $66.3X_3$	+ $80X_4$	+ $78.5X_5$	+ $78.4X_6$				- p				=	0
	$0.8X_1$	+ $1.3X_2$	+ $0.2X_3$	+ $1.2X_4$	+ $1.7X_5$	+ $0.5X_6$	+ S_1							=	140
	$0.5X_1$	+ $0.2X_2$	+ $1.3X_3$	+ $0.7X_4$	+ $0.3X_5$	+ $1.5X_6$		+ S_2						=	90
	$0.4X_1$	+ $0.4X_2$	+ $0.4X_3$	+ X_4	+ X_5	+ X_6								$\leq$	120
	X_1	+ $1.05X_2$	+ $1.1X_3$	+ $0.8X_4$	+ $0.82X_5$	+ $0.84X_6$			+ S_3					=	125
										- p	+ $10500d_1$			=	0
									- S_3			+ $125d_2$		=	0
							- S_1	+ S_2					+ $230d_3$	=	0
	X_1 ,	X_2 ,	X_3 ,	X_4 ,	X_5 ,	X_6	S_1 ,	S_2 ,	S_3		d_1 ,	d_2 ,	d_3	$\geq$	0
										p				$\gtrless$	0

Table 11.5. GAMS Setup for Weighted Objective Example

```
   4
   5   SET            GOALS            /PROFIT,LABOR,LATHETIME/
   6                  PROCESS       TYPES OF PRODUCTION PROCESSES
   7                                   /FUNCTNORM , FUNCTMXSML , FUNCTMXLRG
   8                                   ,FANCYNORM , FANCYMXSML , FANCYMXLRG/
   9                  RESOURCE      TYPES OF RESOURCES
  10                                   /SMLLATHE,LRGLATHE,CARVER,LABOR/
  11
  12   PARAMETER PRICE(PROCESS)       PRODUCT PRICES BY PROCESS
  13                     /FUNCTNORM  82, FUNCTMXSML  82, FUNCTMXLRG  82
  14                     ,FANCYNORM 105, FANCYMXSML 105, FANCYMXLRG 105/
  15             PRODCOST(PROCESS)    COST BY PROCESS
  16                     /FUNCTNORM  15, FUNCTMXSML  16  , FUNCTMXLRG  15.7
  17                     ,FANCYNORM  25, FANCYMXSML  26.5, FANCYMXLRG  26.6/
  18             RESORAVAIL(RESOURCE) RESOURCE AVAILABLITY
  19                     /SMLLATHE 140, LRGLATHE  90,
  20                      CARVER   120, LABOR    125/
  21             WEIGHTS(GOALS)       WEIGHT FOR GOALS
  22                                  /PROFIT  1,LABOR  0.4,LATHETIME 0.4/
  23             MAGNITUDE(GOALS)       MAGNITUDE FOR GOALS
  24                                  /PROFIT  10500/;
  25             MAGNITUDE("LATHETIME")=RESORAVAIL("SMLLATHE")+RESORAVAIL("LRGL
ATHE");
  26             MAGNITUDE("LABOR")=RESORAVAIL("LABOR");
  27
  28
  29    TABLE RESOURUSE(RESOURCE,PROCESS) RESOURCE USAGE
  30
  31                  FUNCTNORM   FUNCTMXSML  FUNCTMXLRG
  32   SMLLATHE          0.80       1.30        0.20
  33   LRGLATHE          0.50       0.20        1.30
  34   CARVER            0.40       0.40        0.40
  35   LABOR             1.00       1.05        1.10
  36    +              FANCYNORM   FANCYMXSML  FANCYMXLRG
  37   SMLLATHE          1.20       1.70        0.50
  38   LRGLATHE          0.70       0.30        1.50
  39   CARVER            1.00       1.00        1.00
  40   LABOR             0.80       0.82        0.84;
  41
  42    POSITIVE VARIABLES
  43             PRODUCTION(PROCESS) ITEMS PRODUCED BY PROCESS
  44             IDLE(RESOURCE)     SLACK VARIABLES FOR RESOURCES
  45             GOALLEVEL(GOALS)    GOAL LEVELS
  46             PROFIT              TOTALPROFIT;
  47    VARIABLES
  48             GOALOBJ             GOAL OBJECTIVE;
  49    EQUATIONS
  50             OBJT                  OBJECTIVE FUNCTION
  51             PROFITACCT            PROFIT ACCOUNTING
  52             AVAILABLE(RESOURCE)   RESOURCES AVAILABLE
  53             IDLLABGOAL           IDLE LABOR GOAL
  54             PROFITGOAL            PROFIT GOAL
  55             LATHEGOAL             IDLE LATHE GOAL;
  56
  57     OBJT.. GOALOBJ =E= SUM(GOALS,WEIGHTS(GOALS)*GOALLEVEL(GOALS)) ;
  58
  59    PROFITACCT..   PROFIT =E=
  60                SUM(PROCESS,(PRICE(PROCESS)-PRODCOST(PROCESS))
  61                              * PRODUCTION(PROCESS)) ;
  62
  63     AVAILABLE(RESOURCE)..
  64         SUM(PROCESS,RESOURUSE(RESOURCE,PROCESS)*PRODUCTION(PROCESS))
  65                       +IDLE(RESOURCE)   =E= RESORAVAIL(RESOURCE);
  66
  67     IDLLABGOAL..    IDLE("LABOR") =E= GOALLEVEL("LABOR")*MAGNITUDE("LABOR");
  68     PROFITGOAL..    PROFIT        =E= GOALLEVEL("PROFIT")*MAGNITUDE("PROFIT");
  69     LATHEGOAL..     IDLE("LRGLATHE")+IDLE("SMLLATHE")
  70                               =E= GOALLEVEL("LATHETIME")*MAGNITUDE("LATHETIM E");
  71    MODEL RESALLOC /ALL/;
```

```
72   SOLVE RESALLOC USING LP MAXIMIZING GOALOBJ;
```

Table 11.6. Tableau for Weighted Tradeoff with Targets Example

Max											$+\ d_1^+ - 10d_1^-$	$+\ .1d_2^+ - .4d_2^-$	$-\ .1d_3^+ - .4d_3^-$	$+\ 9d_4^+ - 1d_4^-$		
s.t.	$67X_1$	$+\ 66X_2$	$+\ 66.3X_3$	$+\ 80X_4$	$+\ 78.5X_5$	$+\ 78.4X_6$				$-\ p$					$=$	0
	$0.8X_1$	$+\ 1.3X_2$	$+\ 0.2X_3$	$+\ 1.2X_4$	$+\ 1.7X_5$	$+\ 0.5X_6$	$+\ S_1$								$=$	140
	$0.5X_1$	$+\ 0.2X_2$	$+\ 1.3X_3$	$+\ 0.7X_4$	$+\ 0.3X_5$	$+\ 1.5X_6$		$+\ S_2$							$=$	90
	$0.4X_1$	$+\ 0.4X_2$	$+\ 0.4X_3$	$+\ X_4$	$+\ X_5$	$+\ X_6$									$\leq$	120
	X_1	$+\ 1.05X_2$	$+\ 1.1X_3$	$+\ 0.8X_4$	$+\ 0.82X_5$	$+\ 0.84X_6$			$+\ S_3$						$=$	125
										p	$-\ 9000d_1^+ + 9000d_1^-$				$=$	9000
									S_3			$-\ 30d_2^+\ 30d_2^-$			$=$	30
							S_1	$+\ S_2$					$-\ 25d_3^+ + 25d_3^-$		$=$	25
										p				$-\ 9500d_4^+ + 9500d_4^-$		9500
	X_1	, X_2	, X_3	, X_4	, X_5	, X_6	S_1	, S_2	, S_3		d_1^+, d_1^-	, d_2^+, d_2^-	, d_3^+, d_3^-	, d_4^+, d_4^-	$\geq$	0
										p					$\gtrless$	0

Table 11.7. GAMS Setup for Weighted Objective with Target Example

```
 4
 5   SET            GOALS         GOALS IN THE ORDER THEY ARE TO BE MET
 6                                   /PROFIT1,LABOR,LATHETIME,PROFIT2/
 7                  PROCESS       TYPES OF PRODUCTION PROCESSES
 8                                   /FUNCTNORM , FUNCTMXSML , FUNCTMXLRG
 9                                   ,FANCYNORM , FANCYMXSML , FANCYMXLRG/
10                  RESOURCE      TYPES OF RESOURCES
11                                   /SMLLATHE,LRGLATHE,CARVER,LABOR/
12                  DIR           GOAL DEVIATION DIRECTION  /MORETHAN, LESSTHAN/
13
14     ALIAS(GOALS,GOAL) ;
15
16   PARAMETER PRICE(PROCESS)       PRODUCT PRICES BY PROCESS
17                     /FUNCTNORM  82, FUNCTMXSML  82, FUNCTMXLRG  82
18                     ,FANCYNORM 105, FANCYMXSML 105, FANCYMXLRG 105/
19             PRODCOST(PROCESS)    COST BY PROCESS
20                     /FUNCTNORM  15, FUNCTMXSML  16  , FUNCTMXLRG  15.7
21                     ,FANCYNORM  25, FANCYMXSML  26.5, FANCYMXLRG  26.6/
22             RESORAVAIL(RESOURCE) RESOURCE AVAILABLITY
23                     /SMLLATHE 140, LRGLATHE  90,
24                      CARVER   120, LABOR    125/
25             TARGET(GOALS)       GOAL TARGET LEVELS
26                                   /PROFIT1  9000,LABOR  30,LATHETIME 25
27                                    ,PROFIT2 9500/
28             MAGNITUDE(GOALS)       MAGNITUDE FOR GOALS;
29             MAGNITUDE(GOALS)=TARGET(GOALS);
30
31     TABLE            WEIGHTS(GOALS,dir)      WEIGHTS BY GOAL
32
33                                MORETHAN   LESSTHAN
34                 PROFIT1       1         -10
35                 LABOR        0.1         -0.4
36                 LATHETIME    0.1         -0.4
37                 PROFIT2       .9         -1.       ;
38
39   TABLE RESOURUSE(RESOURCE,PROCESS) RESOURCE USAGE
40
41                 FUNCTNORM   FUNCTMXSML  FUNCTMXLRG
42   SMLLATHE         0.80       1.30        0.20
43   LRGLATHE         0.50       0.20        1.30
44   CARVER           0.40       0.40        0.40
45   LABOR            1.00       1.05        1.10
46    +              FANCYNORM   FANCYMXSML  FANCYMXLRG
47   SMLLATHE         1.20       1.70        0.50
48   LRGLATHE         0.70       0.30        1.50
49   CARVER           1.00       1.00        1.00
50   LABOR            0.80       0.82        0.84;
51
52   POSITIVE VARIABLES
53             PRODUCTION(PROCESS) ITEMS PRODUCED BY PROCESS
54             IDLE(RESOURCE)     SLACK VARIABLES FOR RESOURCES
55             GOALLEVEL(GOALS)    GOAL LEVELS
56             PROFIT              TOTALPROFIT
57             SHORTFALL(GOALS)    GOAL SHORTFALLS
58             EXCESS(GOALS)       GOAL EXCESSES;
59   VARIABLES
60             GOALOBJ             GOAL OBJECTIVE;
61   EQUATIONS
62             OBJT                  OBJECTIVE FUNCTION
63             PROFITACCT            PROFIT ACCOUNTING
64             AVAILABLE(RESOURCE)   RESOURCES AVAILABLE
65             IDLLABGOAL           IDLE LABOR GOAL
66             PROFITGL1            PROFIT1 GOAL
```

```
67            PROFITGL2            PROFIT2 GOAL
```

Table 11.7. GAMS Setup for Weighted Objective with Target Example (Continued)

```
68        LATHEGOAL            IDLE LATHE GOAL
69        TARGS(GOALS)         GOAL TARGETS  ;
70
71   OBJT.. GOALOBJ =E= SUM(GOALS,WEIGHTS(GOALS,"LESSTHAN")*SHORTFALL(GOALS)
72                          +WEIGHTS(GOALS,"MORETHAN")*EXCESS(GOALS)) ;
73
74  PROFITACCT..   PROFIT =E=
75            SUM(PROCESS,(PRICE(PROCESS)-PRODCOST(PROCESS))
76                        * PRODUCTION(PROCESS)) ;
77
78   AVAILABLE(RESOURCE)..
79      SUM(PROCESS,RESOURUSE(RESOURCE,PROCESS)*PRODUCTION(PROCESS))
80                  +IDLE(RESOURCE)   =E= RESORAVAIL(RESOURCE);
81
82   PROFITGL1..    PROFIT        =E= GOALLEVEL("PROFIT1");
83   IDLLABGOAL..    IDLE("LABOR") =E= GOALLEVEL("LABOR");
84   LATHEGOAL..     IDLE("LRGLATHE")+IDLE("SMLLATHE")
85                        =E= GOALLEVEL("LATHETIME");
86   PROFITGL2..    PROFIT        =E= GOALLEVEL("PROFIT2");
87
88   TARGS(GOALS)..
89       GOALLEVEL(GOALS) + MAGNITUDE(GOALS)*( SHORTFALL(GOALS) -EXCESS(GOALS))
90                        =E= TARGET(GOALS) ;
91
92
93  MODEL RESALLOC /ALL/;
94
95  SOLVE RESALLOC USING LP MAXIMIZING GOALOBJ;
```