GOAL PROGRAMMING - A CRITIQUE

RICHARD E. ROSENTHAL
THE UNIVERSITY OF TENNESSEE
KNOXVILLE, TN 37916

SUMMARY

Goal programming is a well-defined approach to an ill-defined problem, multiobjective optimization. It is suggested that goal programming's power to resolve this problem may not be as great as its numerous proponents have claimed in the literature.

1. MULTIOBJECTIVE OPTIMIZATION

Goal programming is one of several well-defined analytical approaches to the very important, but ill-defined, multiobjective optimization problem. To pose an instance of this problem there must be defined: a decision variable \( x \in \mathbb{R}^n \), a set of feasible decisions \( X \subseteq \mathbb{R}^n \) and a set of real-valued objective functions \( f_i(x), i=1,\ldots,k \). The multiobjective optimization (or vector optimum) problem is then: find an \( x \in X \) such that the most preferred vector of objective function values \( F(x) = (f_1(x), \ldots, f_k(x)) \) is attained. This problem is ill-defined for two reasons: first, the word "preferred" is obviously vague, and more importantly, there exists no natural ordering of the vectors \( F(x) \in \mathbb{R}^k \). The lack of a natural ordering means there may be no general agreement as to whether \( F(x) \) is "greater than", "less than" or "equal" to \( F(y) \). There can then be no guarantee of a generally agreed upon "vector maximum" or "vector minimum".

As obvious as the preceding point may be, it seems rather important to emphasize ill-definition at the outset of a discussion of multiobjective optimization. It is discomfiting to see that many papers in the goal programming literature present models that purport to "maximize" or "minimize" vector quantities. Indeed, some goal programming authors have adopted an unfortunate notation that explicitly states "minimization of a vector" as the object of the model. This point of criticism is probably for the most part stylistic; it is not suggested that the goal programming authors labour under the misconception that "vector minimization" is possible.

The multiobjective optimization problem is much too interesting and important to be dismissed entirely because of its ill-definition. The vast array of fascinating areas in which it arises attests to this [Ignizio, 1977]. The point of this critique is that goal programming may not be as powerful and substantial an addition to the management scientist's tool kit as the numerous proponents of goal programming suggest.

Introduction to the multiobjective optimization problem elicits from most people one or more of the following common sense approaches:

- **Weighting**: defining a weighted sum of the objective functions as the scalar-valued function to be optimized in the well-defined sense.
- **Prioritizing**: performing a sequence of single-objective optimizations in a priority order.
- **Setting targets or aspiration levels** for the objective functions and attempting to meet them or, in some sense, come as close as possible.
- **Efficiency** (also known as Pareto optimality, noninferiority and nondominance): excluding from consideration all solutions that offer no advantage, with respect to any objective function, over some other possible solution.

Much of the literature of multiobjective optimization is founded on these ideas.

2. GOAL PROGRAMMING MODELS

Three classes of goal programming models are briefly discussed here: weighted deviation models, preemptive (also known as lexicographic models), and priority class models. All require the specification of target values, $t_i$, for each objective function $f_i$. Further, all have in common the definition of an over-attainment variable $p_i = \max[0, f_i(x) - t_i]$ and an under-attainment variable $n_i = \max[0, t_i - f_i(x)]$. The constraints of the three classes of goal programming models are, in addition to $x \in X$,

$$f_i(x) + n_i - p_i = t_i$$  \hspace{1cm} (1)
$$n_i p_i = 0$$  \hspace{1cm} (2)
$$n_i, p_i \geq 0$$  \hspace{1cm} (3)

for $i = 1, \ldots, k$.

The objective function for the

$$\min \sum_i (v_i n_i + w_i p_i)$$  \hspace{1cm} (4)

where $v_i, w_i$ are known scalar "weights". The weights represent "penalties" for nonattainment of targets. In some cases a weight may be negative representing a "premium" instead of a "penalty".

In a variation of the weighted deviation model, known as goal interval programming, a target range $[t_i - d_i, t_i + d_i]$ rather than a target value is specified. Assuming $v_i$ and $w_i$ are positive penalties for values of $f_i$ outside the range, then this variation amounts to replacing $p_i$ by $p_i^+ + p_i^-$ and $n_i$ by $n_i^+ + n_i^-$ in constraint (1), with only $p_i^+$, $n_i^-$ in the objective function and with upper bounds of $d_i$ on $p_i^+, n_i^-$.
In the preemptive or lexicographic goal programming model, the objective functions are prioritized in the strict sense that attainment of \( t_1 \) far outweights in importance attainment of \( t_2 \), which far outweights \( t_3 \), etc. This corresponds in theory to an instance of (4) with \( v_1 >> v_2 >> v_3 >> \ldots \) and \( w_1 >> w_2 >> \ldots \) (but not in practice because digital implementations of mathematical programming algorithms cannot handle such large ranges of data). (See the Appendix for discussion of computational aspects.)

The priority class goal programming model is a combination of the previous two models: the weighted sum of one subset of target deviation variables is given highest priority, the weighted sum of another subset is given second priority, etc.

3. IMPLICATIONS OF PRIORITIES

The use of priorities in a goal programming model implies a sequence of minimizations of the form

\[
\min_{i \in I_j} \left( v_i n_i + w_i p_i \right)
\]

where \( I_j \) (possibly a singleton) is the set of objective functions in the \( j \)th priority class. The sequence of minimizations is designed to find the best possible solution with respect to the first priority class; and then, if ties exist, to find among these alternate optima a best solution with respect to the second priority class. If ties still exist, the third priority class is used to discriminate further, etc.

Two evident limitations of prioritization should be pointed out. First of all, the successive tie-breaking approach is applicable only when massive tying (dual degeneracy) occurs. In general, there is little reason to expect the process to continue much beyond the first minimization. Thus, some, if not most, of the objective functions will be totally left out of consideration in the preemptive and priority class models. A second weakness in the prioritizing approach is that it disallows the very reasonable practice of trading off a small degradation in a high priority objective for a large improvement in a low priority objective.

4. A LESSON FROM UTILITY THEORY

No rational person would dispute that a correct approach to the multiobjective optimization problem, if implementable, would be to solve

\[
\max_{x \in X} U(f_1(x), \ldots, f_k(x))
\]

where \( U: \mathbb{R}^k \to \mathbb{R} \) is a utility function whose significance is that \( U(F(x)) > U(F(y)) \) if the vector of objective attainments \( F(x) \) is "preferred" to the vector \( F(y) \). Unfortunately, the difficulty with this approach, and the reason for disagreement among
rational people, is the frequent impossibility of specifying an accurate, explicit mathematical form of $U$.

In spite of the possible impracticality, some consideration of utility theory can shed a beneficial light on multiobjective optimization. In particular, an implicit assumption of goal programming is seen in this light to be inconsistent with well-established economic principles. The goal programming model (1-4) is equivalent to a utility maximization formulation in which

$$\frac{\partial U}{\partial f_i(x)} = \begin{cases} v_i & \text{if } f_i(x) < t \\ -w_i & \text{if } f_i(x) > t \end{cases}$$  \hspace{1cm} (7)

i.e., the marginal rate at which utility increases as attainment of $f_i$ increases is assumed constant on either side of the target. This conflicts, however, with many generations of economic thought (including Bernoulli's 1730 St. Petersburg paradox), which established that these marginal rates should not be constant [Baumol, 1977]. (Would your second million dollars have as much utility for you as your first?)

The idea of a marginal rate of substitution makes the economic inconsistency of goal programming even more apparent. This quantity, defined as

$$\text{MRS}_{ij}(x) = \frac{\partial U/\partial f_i(x)}{\partial U/\partial f_j(x)}$$ \hspace{1cm} (8)

is the rate at which a decision-maker, starting with attainment vector $F(x)$, would be willing to trade off attainments of $f_i$ for $f_j$. These trade off rates vary with the attainment levels of $f_i$ and $f_j$. There would generally be a drastic change in $\text{MRS}_{ij}$ as the decision-maker goes from having, say, a small surplus of $f_i$ and a large surplus of $f_j$ to having the opposite sized surpluses. Plentiful commodities are always easier to trade away than scarce ones. Goal programming, however, disregards this phenomenon. It implicitly assumes the marginal rates of substitution are constant in each quadrant of the $[n_i, p_i]$ space.

5. CONCLUSION, SUGGESTED ALTERNATIVES, AND A PEDAGOGICAL REMARK

Goal programming is viewed as a well-defined analytical approach to the important ill-defined multiobjective optimization problem. Unfortunately, it suffers the weakness of insufficiently modeling the interactions among objectives that are necessary in a resolution of the problem. This weakness emerges in three ways: (i) Because of their insistence on lexicographic optimality, preemptive and priority class models may totally ignore lower priority objectives. (ii) Preemptive and priority class models disallow tradeoffs of small losses in high priority objectives for large gains in low priority objectives. (iii) Both weighted and prioritized goal programming models ignore the nonconstancy of the rates at which benefits from objective attainments increase and they ignore the nonconstancy of the rates at which decision-makers will trade off attainments.
One of the most promising techniques for multiobjective optimization, which has received much less acceptance than goal programming, is *implicit utility maximization* as typified by Geoffrion, Dyer and Feinberg's interactive approach [1972]. Their extremely clever idea is to attempt a utility maximization without explicitly specifying a utility function. For the direction-finding step of a nonlinear programming algorithm (e.g., the Frank and Wolfe method or the reduced gradient method), the gradient of U is estimated by interactive determination of the marginal rates of substitution. Then further interaction is used to approximate optimal step-sizes. Under the assumption of convex utility (consistent with economic theory [Baumol, 1977]), this procedure terminates optimally. This is quite remarkable, considering the limited information available about the function being optimized.

Another promising area of multiobjective analysis is the development of algorithms for identifying efficient solutions [Ecker et al., 1980; Gal, 1977]. The question of determining a most preferred solution from a set of efficient solutions is left to the decision-maker's subjective processes.

It may be the case in practice that the set of efficient solutions is too large to generate (as well as too large to have any managerial value). An approach frequently taken is *partial separation of the efficient set*. Several different but reasonable variations of this idea have been developed. Soland [1979] suggests that the decision maker should have access to an interactive program that allows him or her to probe, sample and wander through the efficient set without any limiting guidelines imposed by the analyst. Soland proves a powerful, general theorem on the characterization of efficient points, which makes this idea practical. (See Hultz et al. [1981] for an application.) A much different variation on the theme of partial generation of the efficient set is the approach taken by Ecker and Shoemaker [1982]. Their idea is to analytically define and algorithmically generate "preferred" subsets of the efficient set. One such subset is the *efficient compromise set* which is based on minimizing the maximum deviation of the objective functions from their ideal values.

Formally, an efficient compromise point is a point which solves

\[
\text{minimize } \max_{e \in E} \left( \max_{i} \left( M_i - f_i(x) \right) \right) \tag{9}
\]

where \( E \) is the set of efficient points and

\[
M_i = \max_{x \in X} f_i(x) \tag{10}
\]

Other techniques for partial generation of the efficient set lie somewhere between Soland's highly unstructured approach and Ecker and Shoemaker's highly structured approach. See Rosenthal [1982] for further discussion.

Finally, a third suggestion is offered -- after implicit utility maximization and partial generation of the efficient set -- as an alternative to goal programming for solving multiobjective
problems. This suggestion is to actually attempt to exploit utility maximization using techniques from decision theory for assessing the utility function (see Keeney and Raiffa, 1976). Past practice of this approach has been mainly limited to problems in which the feasible region \( X \) is discrete and small enough to be totally enumerated prior to the analysis. This limitation may however not be intrinsic to the approach.

In closing, it is often said that goal programming's great virtue is that managers understand it easily. It is also said that many of the alternate approaches suggested here are difficult to understand. This may be true in some instances, but should not be a guiding principle in the evolution of this field. It is especially important for teachers not to be influenced by past successes and failures in explaining multiobjective techniques to managers. If we bring into the classroom only what comes out of the boardroom, then the limitations of today's managers will be imposed on tomorrow's.

REFERENCES


Goal programming models in the literature generally do not include the nonlinear constraint equations (2). It is safe to omit these constraints, with the assurance they will hold automatically, provided the proper convexity conditions hold on $X, f_1, \ldots, f_k$ and provided

$$v_i + w_i \geq 0$$

(11) for all $i$. If $v_i + w_i < 0$, then simultaneous increase of $p_i$ and $n_i$ can feasibly decrease the goal programming objective function (4) without bound. These limitations on the validity of omitting (2) are sometimes overlooked in the literature.

Another computational caveat missing from the goal programming literature is the inadvisability of choosing extreme values of $f_i(x)$ as the targets. This practice may seem natural to a modeler or a decision-maker, but it renders the goal program into an ordinary (single-objective) optimization model with superfluous structural appendages. For example, if

$$t_i = \max_{x \in X} f_i(x)$$

(12)

for all $i$, then $p_i = 0$, $n_i = t_i - f_i(x)$ and (4) is equivalent to

$$\min_{x \in X} \sum_i v_i(t_i - f_i(x)).$$

(13)

This case of model (1-4) is then equivalent to

$$\max_{x \in X} \sum_i v_i f_i(x) + \text{constant}. $$

(14)

The procedure recommended by several goal programming authors for the sequential minimizations in preemptive and priority class models is to let the objective function values from the $j$th minimization define an added constraint in the $(j+1)^{st}$ minimization. Instead of adding the constraint, however, it is evidently preferable to delete the variables with nonzero final reduced cost in the $j$th minimization before commencing the $(j+1)^{st}$ minimization. Students at the University of Tennessee have demonstrated that this strategy can be implemented via the control language of the IBM linear programming package MPSX.