Operations Research

May-June 1970

PRIMAL RESOURCE-DIRECTIVE APPROACHES FOR OPTIMIZING NONLINEAR DECOMPOSABLE SYSTEMS

Arthur M. Geoffrion

University of California, Los Angeles, California

(Received October, 1969)

This study presents some new results on three primal-feasible computational approaches for optimizing a system composed of interrelated subsystems. The general structure treated is the same as the principal one of the classic paper by Dantzig and Wolfe, except that convex nonlinearities are permitted, provided that the overall criterion function and coupling constraints are separable by subsystem. Each approach decentralizes the optimization by iteratively allocating system resources to the subsystems, with each subsystem computing its own optimal utilization of the given resources at each iteration. The chief obstacle to directing the resource allocation centrally toward an overall optimum is that the optimal response of each subsystem, as a function of its allowed resources, is not available explicitly. All three procedures therefore approximate or generate the optimal response functions 'as needed.'

THE OPTIMIZATION of 'decomposable' systems comprised of a number of interrelated subsystems is an important and frequently occurring topic in economic planning, engineering, and operations research. A large and rapidly growing literature on the subject bears witness to this fact. How the present study fits into this literature can perhaps best be indicated by considering briefly each of the words in the title.

The term *decomposable system* here means an optimization problem of the form

$$\text{maximize}_x \sum_{i=1}^{i=k} f_i(x_i) \quad \text{subject to} \qquad x_i \epsilon X_i, \quad i=1, \cdots, k, \quad \text{and} \quad \sum_{i=1}^{i=k} g_i(x_i) \geqq b, \tag{1}$$

where x_i is an n_i-dimensional vector of real-valued variables associated with the ith subsystem, X_i is a permissible region of R^{n_i} associated with x_i, f_i is a real-valued payoff function, $g_i=(g_{i1}, \cdots, g_{im})$ is a real vectorvalued function specifying the role of x_i in the m overall 'system' constraints, and b is a given m-dimensional real vector. The system constraints account for the scarce resources available to the system, the tasks assigned to it, and the interconnections or coupling between the subsystems. The f_i and g_i functions are assumed throughout to be concave on the convex sets X_i, and (1) is assumed to possess a feasible solution.

Notice that the criterion function and system constraints are linearly separable in the subsystem variables. This leads to the possibility of achieving an overall optimum in a 'hierarchical' or 'multilevel' fashion by coordinating the individual optimization of the subsystems. Two principal types of coordination are possible: resource-direction and price-direction. *Resource-directive* coordination typically involves the iterative determination of m-vectors $y_1, \cdots, y_k$ such that optimal solutions of the k subproblems

$$\text{maximize}_{x_i}\, f_i(x_i) \quad \text{subject to} \quad x_i \epsilon X_i \quad \text{and} \quad g_i(x_i) \geqq y_i \tag{2}$$

also solve (1). Price-directive coordination, on the other hand, typically involves the iterative determination of m-vectors $\lambda_1, \cdots, \lambda_k$ such that the optimal solutions of the k subproblems

$$\text{maximize}_{x_i}\, f_i(x_i) + \lambda_i^t g_i(x_i) \quad \text{subject to} \quad x_i \epsilon X_i \tag{3}$$

also solve (1). Both types of coordination are aimed at decomposing (1) into essentially k separate optimization problems. Since the computational burden of optimization usually rises much faster than linearly with the size of the problem, the potential saving could be quite substantial. The semiautonomous treatment accorded the subsystems also enables each subsystem to be optimized in the manner best suited to its structure. Of course, there are many applications in which semiautonomy of the subsystems is desirable or even necessary for economic, political, or social reasons.

The distinction between resource-directive and price-directive methods really amounts to the distinction between *'primal'* and 'dual' methods. With resource direction, it is easy and natural to guarantee that the iteratively generated sequence of y_i vectors leads via (2) to feasible solutions of (1). With price direction, however, a feasible solution of (1) typically is not produced by (3) until the very end. The term 'dual' is used in connection with price-directive methods because, as we note in Section 6.1, most such methods are naturally viewed as solving a dual problem associated with (1). A primal method is customarily preferred to a dual method in practical applications, because it enables a known good initial feasible solution to be utilized, and because a good feasible solution is usually available if the coordinating iterations are stopped prior to optimality. Hence, consideration will be limited to primal resource-directive approaches, although it should be evident that there is an entirely analogous development of dual price-directive approaches.

Three primal resource-directive approaches will be developed for solving (1): the so-called tangential approximation, large-step subgradient, and piecewise approaches. We use the term *approaches* to emphasize that the main concern is with fundamental strategies and natural classes of algorithms, rather than with tactics or particular organizations of the computations. Each approach can spawn many algorithmic variants, depending on how one handles the tactical questions. Some of the tactical questions involve a selection among known algorithms for solving intermediate linear, convex, or parametric programs such as (2); in this sense, one may regard the present results as a framework extending the domain of applicability of known algorithms. Thus the present development, while rigorous as far as it goes, must leave open many of the details prerequisite to computational implementation. For similar reasons, not much attention is given to convergence questions, although each approach offered here should converge to an optimal solution of (1) if one properly attends to details.

Many of the important ideas applicable to (1) were born in the rich literature treating the linear case. The nonlinear literature is not nearly as rich. For this reason, and also because most real applications involve nonlinearities of one kind or another, the scope of this study is limited to methods workable for *nonlinear* as well as linear systems.

It is assumed that the reader is familiar with the foundations of linear and nonlinear programming. He will also find it helpful, but not necessary, to be familiar with GEOFFRION,[22] which surveys the fundamental concepts of large-scale mathematical programming.

1. ORGANIZATION, CONTENT, NOTATION

SECTION 2 PRESENTS and justifies the manipulation of (1) that permits it to be optimized via resource direction. This manipulation, called 'projection,' is used to rephrase (1) in terms of the subproblems (2) as the fundamental resource allocation problem (P), which can be interpreted as the highest level of a hierarchical or multilevel formulation of (1).

Sections 3 through 5 may be read in any order.

Section 3 presents the tangential approximation approach to (P). The object being approximated tangentially is the maximand of (P), which is composed of the sum over i of the optimal value of (2) as a function of y_i. It turns out that a tangential approximation is automatically available from the optimal multipliers of (2) whenever it is optimized for a given y_i. The roots of this approach are to be found in BENDERS,[6] CHENEY AND GOLDSTEIN,[9] DANTZIG AND MADANSKY,[13] and KELLEY.[26] This section deals at length with ways of maintaining primal feasibility when this approach is used, an important matter not adequately dealt with in previous studies. Primal feasibility is much easier to maintain with the other two approaches.

Section 4 contains the large-step subgradient approach, so-called because it extends the ordinary large-step gradient approach via subgradients and directional derivatives to cope with the nondifferentiability of the maximand of (P). The latter difficulty is the central obstacle to the direct solution of (P) by the powerful extant large-step gradient algorithms. Our principal contribution is the derivation of an explicit linear program for finding the feasible direction yielding the greatest rate of increase in the maximand of (P). Previous studies along these lines, largely confined to the completely linear case,[1, 43] have generally settled for approximations to such 'best' feasible directions. The optimal multipliers of (2) play a central role.

Section 5 presents the piecewise approach, which takes advantage of the relative simplicity of the maximand of (P) over certain regions of its domain. The roots of this approach are due to ROSEN.[33, 34] Separate treatments are given for the case where (1) is a linear program and the case where it is a quadratic program. The treatment of the latter case appears to be novel, as does the use of improving feasible directions (drawing on the results of Section 4) to guide the transition between adjacent 'regions of simplicity.'

Some discussion is given in Section 6 which includes an indication of how the three approaches considered can also be used to obtain price-directive solution procedures for (1); instead of projection, the appropriate manipulation of (1) is dualization with respect to the coupling constraints.

With a little ingenuity, a wide variety of decomposable problems can be cast in the form (1). Linear interconnection constraints between subsystems, for example, can be expressed as linear inequalities to be included among the coupling constraints (some of the g_{ij} would then be identically zero). And the linear separability property of (1) in the subsystem variables can sometimes be obtained as a result of simple manipulations applied to a nonseparable problem. For instance, the constraint $\varphi_1(x_1)\varphi_2(x_2)\geq 0$ can be expressed as $\varphi_1(x_1)\theta\geq 0$ and $\theta=\varphi_2(x_2)$, where θ would be treated as an additional component of x_1 and the second constraint would be treated as a coupling constraint. Related manipulations have been suggested in the context of 'separable programming.'[31] One can also apply the ideas in reference 21 to treat the important case in which the maximand of (1) has the form $F[f_1(x_1), \cdots, f_k(x_k)]$, where F is quasiconcave and increasing in each argument (i.e., the system objective functional on the performance of the individual subsystems is nonlinear rather than a simple sum). A special instance of this means of achieving separability was also noted by BELL.[5] [*Note added in proof.* It has recently been found that, for $k>2$, the generalization to accommodate F is simpler and more efficient if done directly rather than via the approach suggested by reference 21.]

The notation is quite standard. Matrices and sets are written in the upper case, and variables and functions in the lower. All vectors are columnar unless transposed by the superscript "t." The symbol x denotes a vector composed of the vectors $x_1, \cdots, x_k$, and a similar interpretation is accorded the symbols w, y, z, α, and σ. Vector inequalities are written "$\geqq$," and scalar inequalities "$\leq$." The symbol "$\triangleq$" means "equals by definition."

2. THE RESOURCE-ALLOCATION PROBLEM (*P*)

THE MOST natural way of deriving resource-directive procedures for solving (1) is first to recast it in terms of the resource-allocation problem

$$\text{maximize}_y \sum_{i=1}^{i=k} v_i(y_i) \quad \text{subject to} \quad \sum_{i=1}^{i=k} y_i \geqq b, \tag{P}$$

where $v_i(y_i)$ is defined as the supremal value of the following parameterized subproblem:

$$\text{maximize}_{x_i \epsilon X_i} f_i(x_i) \quad \text{subject to} \quad g_i(x_i) \geqq y_i. \tag{$P_y{}^i$}$$

Although $(P_y{}^i)$ depends only on y_i, the notation $(P_y{}^i)$ rather than $(P_{y_i}^i)$ is used because it is simpler.

One may derive (P) from (1) in two steps as follows. First introduce the m-vectors $y_1, \cdots, y_k$ and rewrite (1) as

$$\begin{gathered}\text{maximize}_{x,y} \sum_{i=1}^{i=k} f_i(x_i) \quad \text{subject to:} \\ x_i \epsilon X_i, \quad i=1, \cdots, k; \qquad g_i(x_i) \geqq y_i, \quad i=1, \cdots, k; \qquad \sum_{i=1}^{i=k} y_i \geqq b.\end{gathered} \tag{4}$$

In effect, this changes (1) from a problem with 'coupling constraints' to a problem with 'coupling variables,' since (4) separates into k separate problems if y is held fixed temporarily. One may interpret y_i as the 'resources' allocated to the ith subsystem. Next, (4) is 'projected onto y' to yield

$$\begin{gathered}\text{maximize}_y \sum_{i=1}^{i=k} \sup_{x_i} \{f_i(x_i) \quad \text{subject to} \quad x_i \epsilon X_i \quad \text{and} \quad g_i(x_i) \geqq y_i\} \\ \text{subject to} \quad \sum_{i=1}^{i=k} y_i \geqq b,\end{gathered} \tag{5}$$

where the convention is made that the supremum (least upper bound) of an empty set is $-\infty$. This convention reflects the fact that a choice of y_i is useless if there is no $x_i \epsilon X_i$ such that $g_i(x_i) \geqq y_i$. For a similar and obvious reason, $\infty - \infty$ is defined to be $-\infty$. In other words, by convention one need not explicitly introduce into (5) the constraints

$$y_i \epsilon Y_i \triangleq \{y_i \epsilon R^m : g_i(x_i) \geqq y_i \quad \text{for some} \quad x_i \epsilon X_i\}. \tag{6}$$

The maximand of (5) is evaluated for fixed y by solving k independent maximization problems.

Projection, as more fully explained in reference 22, is an often used and rigorously defensible problem manipulation sometimes known as 'partitioning.' It is a straightforward matter to demonstrate that (1) and (P) are equivalent in the following sense (even in the absence of convexity).

THEOREM 1. A. (1) *is infeasible if and only if* (P) *has optimal value* $-\infty$.

B. (1) *has unbounded optimal value if and only if* (P) *has unbounded optimal value.*

C. *If* (1) *has an optimal solution* x^0, *then* (P) *has an optimal solution* y^0 *and* x_i^0 *is an optimal solution of* $(P_{y^0}^i)$, *where* y^0 *is any vector satisfying* $g_i(x_i^0) \geqq y_i^0 (i=1, \cdots, k)$ *and* $\sum_{i=1}^{i=k} y_i^0 \geqq b$ [*e.g.*, $y_i^0 = g_i(x_i^0)$ *for* $i=1, \cdots, k$].

D. *If* (P) *has an optimal solution* y^0 *and* x_i^0 *is optimal in* $(P_{y^0}^i)$, *then* $(x_1^0, \cdots, x_k^0)$ *is optimal in* (1).

If the constraints (6) are explicitly incorporated into (P), the only change required in the theorem is in part A; (1) is infeasible if and only if (P) with (6) appended is infeasible.

It is important to realize that (P), like (1), is a concave program.[22] The following result is not difficult to demonstrate.

THEOREM 2. *The set* Y_i *is nonempty and convex, and* v_i *is concave and nonincreasing on* Y_i *and* $-\infty$ *outside of* Y_i.

Thus, (P) is a concave program equivalent for all practical purposes to (1). It might seem that to consider (P) in place of (1) would be to give up convenient optimality conditions, for the Kuhn-Tucker optimality conditions of (1) can be conveniently expressed in terms of gradients if all functions are differentiable, whereas this is not true of (P), since the functions v_i are not everywhere differentiable.[11,43] Fortunately, however, for present purposes the optimality conditions of (1) suffice for (P) as well, and when they are satisfied an optimal solution of (1) is automatically available. For, suppose that a feasible point y^0 is offered as possibly optimal in (P). Let x_i^0 be any optimal solution of $(P_{y^0}^i)$, and test x^0 for optimality in (1). It follows from Theorem 1 that x^0 is optimal in (1) if and only if y^0 is optimal in (P).

How might one go about solving (P)? Certainly any approach that requires the supremal value functions v_i to be found explicitly first is doomed to impracticality except in very special cases. Probably the only situation in which v_i can be determined explicitly is the case $m=1$. Then (P_y^i) entails but a single parameter and it may be possible to obtain v_i via parametric programming techniques. The three approaches below are of more general applicability in that they evaluate or approximate v_i only as needed.

3. THE TANGENTIAL APPROXIMATION APPROACH TO (*P*)

THE KEY TO the tangential approximation approach for solving (P) is the observation that a linear support to v_i is a natural by-product of its evaluation. This implies that (P) can be solved by building up an adequate approximation to each v_i in terms of its linear supports.

A *linear support* to a concave function is any linear function whose value is always greater than or equal to that of the concave function, with equality holding at some point of the domain. Evaluating v_i at the point $\bar{y}_i$, say, requires solving the concave program $(P\bar{y}^i)$. There will usually exist an optimal multiplier vector $\bar{\lambda}_i$ associated with the constraints $g_i(x_i) \geqq \bar{y}_i$; in fact, most concave programming algorithms produce $\bar{\lambda}_i$ automatically. It is well known and not difficult to show that $\bar{\lambda}_i$ is the normal of a linear support to v_i at $\bar{y}_i$.[23]

THEOREM 3. *Let $\bar{y}$ be such that $(P_{\bar{y}}^i)$ has an optimal solution $\bar{x}_i$ and an optimal multiplier vector $\bar{\lambda}_i$. Then the function $f_i(\bar{x}_i)-\bar{\lambda}_i^t(y_i-\bar{y}_i)$ is a linear support to v_i at $\bar{y}_i$, so that*

$$v_i(y_i) \leq f_i(\bar{x}_i)-\bar{\lambda}_i^t(y_i-\bar{y}_i) \quad \text{for all} \quad y_i. \tag{7}$$

To be sure that the hypotheses of Theorem 3 hold whenever $(P_{\bar{y}}^i)$ is feasible (i.e., whenever $\bar{y}_i \epsilon Y_i$), we shall impose throughout this section the following additional assumptions:

A1: f_i and g_{ij} are all upper semicontinuous functions, and X_i is compact.

A2: For every $y_i \epsilon Y_i$, (P_y^i) has an optimal multiplier vector $\lambda_i(y_i)$ associated with the constraints $g_i(x_i) \geqq y_i$.

These assumptions are really quite mild. Upper semicontinuity is a weaker property than continuity, and, for concave functions, only concerns boundary behavior.

The boundedness of X_i can be enforced, if necessary, by restricting x_i to some suitably large but bounded region within which an optimal solution of (1) must lie if it is to be meaningful. Assumption A2 can be replaced by any of a number of constraint qualification assumptions, but we shall suggest in Section 3.2 how it can be eliminated altogether.

With these additional assumptions, one can assert not only that v_i has a linear support at every point in Y_i, but also that v_i actually coincides on Y_i with the envelope of these supports. More precisely, we have:

COROLLARY. *If the additional assumptions* A1 *and* A2 *hold, then*

$$v_i(y_i) = \min_{\bar{y}_i \epsilon Y_i} \{v_i(\bar{y}_i) - \lambda_i^t(\bar{y}_i)(y_i-\bar{y}_i)\} \quad \text{for all} \quad y_i \epsilon Y_i.$$

Proof. From Theorem 3 and the additional assumptions, $\lambda_i(y_i)$ exists as defined on Y_i and

$$v_i(y_i) \leq v_i(\bar{y}_i) - \lambda_i^t(\bar{y}_i)(y_i-\bar{y}_i) \quad \text{for all} \quad y_i, \bar{y}_i \epsilon Y_i.$$

Taking the infimum of this inequality over all $\bar{y}_i$ in Y_i, we obtain

$$v_i(y_i) \leq \inf_{\bar{y}_i \in Y_i} \{v_i(\bar{y}_i) - \lambda_i^t(\bar{y}_i)(y_i - \bar{y}_i)\} \quad \text{for all} \quad y_i \in Y_i.$$

But equality must hold and the infimum must be achieved, for one may take $\bar{y}_i = y_i$.

This result suggests that (P) can be solved by building up a better and better approximation of v_i in terms of its linear supports. If an optimal multiplier vector λ_i^j has been found at each of a number of points y_i^j in Y_i $(j=0, \cdots, \nu)$, then the approximation to v_i will be the piecewise-linear function

$$v_i^\nu(y_i) \triangleq \text{minimum}_{j=0,1,\cdots,\nu} \{v_i(y_i^j) - (\lambda_i^j)^t(y_i - y_i^j)\}. \tag{8}$$

We are thus led to the:

TANGENTIAL APPROXIMATION APPROACH

Step 1. Let a feasible solution y^0 of (P) be given such that the corresponding subproblems $(P_{y^0}^i)$ are feasible. Solve $(P_{y^0}^i)$ for an optimal solution by any suitable concave programming algorithm and recover an optimal multiplier vector λ_i^0, $i=1, \cdots, k$. Put $\nu=0$.

Step 2. Solve the current tangential approximation to (P), namely

$$\begin{aligned} &\text{maximize}_y \textstyle\sum_{i=1}^{i=k} v_i^\nu(y_i) \quad \text{subject to:} \\ &y_i \in Y_i, \quad i=1, \cdots, k; \qquad \textstyle\sum_{i=1}^{i=k} y_i \geqq b, \end{aligned} \tag{9}$$

by any suitable algorithm. Let $y^{\nu+1}$ be an optimal solution.

Step 3. Recover an optimal solution and an optimal multiplier vector $\lambda_i^{\nu+1}$ for $(P_{y^{\nu+1}}^i)$, $i=1, \cdots, k$.

Step 4. If $y^{\nu+1}$ is sufficiently near optimal in (P), terminate. Otherwise, increase ν by 1 and return to step 2.

Note that the constraints $y_i \in Y_i$ are introduced explicitly in (9), since a *primal* resource-directive method is desired.

It is not difficult to see that problem (9) is equivalent in the obvious sense to the concave program

$$\begin{aligned} &\text{maximize}_{y,\sigma} \textstyle\sum_{i=1}^{i=k} \sigma_i \quad \text{subject to:} \\ &\sigma_i \leq v_i(y_i^j) - (\lambda_i^j)^t(y_i - y_i^j), \quad j=0, \cdots, \nu, \quad i=1, \cdots, k; \\ &y_i \in Y_i, \quad i=1, \cdots, k; \qquad \textstyle\sum_{i=1}^{i=k} y_i \geqq b. \end{aligned} \tag{10}$$

Assumption A1 implies that (10) is sure to have an optimal solution. Certainly (10) is a far more tractable problem than (P) itself, although the ease of solution depends upon how conveniently the convex sets Y_i can be handled. Ways of handling Y_i will be discussed in Section 3.1. It is likely that in practice the number of linear constraints involving σ_i can be kept to a reasonable number less than $k\nu$ by dropping certain of the constraints that are amply satisfied at $(\sigma_i^{\nu+1}, y_i^{\nu+1})$. This corresponds to coarsening the piecewise-linear approximation to v_i away from the current trial solution to (P).

[*Note added in proof.* Rigorous justification for dropping constraints is provided by two recent studies: B. C. EAVES and W. I. ZANGWILL, "Generalized Cutting Plane Algorithms," Working Paper No. 274, Center for Research in Management Science, University of California, Berkeley, July 1969; and DONALD M. TOPKIS, "Cutting-Plane Methods without Nested Constraint Set," which appears later in this issue of OPERATIONS RESEARCH, pages 404–413.]

Note that the subproblems need be solved *ab initio* only once, at Step 1. Thereafter, an appropriate solution-recovery technique—possibly of a parametric variety[12,20,24,35]—can be used at Step 3. Of course, post-optimality techniques would also be used to reoptimize (10) after the first execution of Step 2 (because of the simple manner in which σ appears, recovery techniques that are primal rather than dual with respect to (10) can be used if desired).

Since v_i^ν never underestimates v_i, the optimal value of (9) or (10) at each iteration provides an upper bound on the optimal value of (P). These upper bounds are a monotone decreasing sequence. This, coupled with the lower bounds provided by the y^ν's [which are feasible in (P)], leads to a natural termination criterion for Step 4: stop if

$$\sum_{i=1}^{i=k} v_i^\nu(y_i^{\nu+1}) - \text{maximum}_{0\le j\le \nu}\{\sum_{i=1}^{i=k} v_i(y_i^{j+1})\} \le \epsilon,$$

where ϵ is a suitably small positive number.

3.1 Dealing with Y_i

We must consider now how the convex sets

$$Y_i \triangleq \{y_i \epsilon R^m : y_i \leqq g_i(x_i) \quad \text{for some} \quad x_i \epsilon X_i\}$$

appearing in (9) and (10) ought to be handled. Three possibilities will be discussed:

(a) Obtain Y_i explicitly.

(b) Build up an adequate outer (containing) polyhedral approximation as needed based on supporting hyperplanes.

(c) Build up an adequate inner (contained) polyhedral approximation as needed based on points in Y_i.

Of course, it may be desirable to treat Y_i differently for different i so as to exploit different structures of (P_y^i).

It might seem that possibilities (b) and (c) would be undesirable in that they may require an extra infinitely convergent process within Step 2. The following two ameliorating factors, however, should be kept in mind. First, any approximation to Y_i is cumulative in benefit since Y_i does not change between successive executions of Step 2. Second, v_i and hence v_i'', is a nonincreasing function, and therefore the optimal solution of (9) tends to stay away from that portion of the boundary of Y_i to which, as we shall see, the approximations of (b) and (c) are mainly addressed.

Only in quite special circumstances can one obtain Y_i once and for all as a finite collection of explicit inequalities in y_i. In the case $m=1$ (i.e., y_i is a scalar rather than a vector variable), $y_i \epsilon Y_i$ is obviously equivalent to the single inequality $y \leq$ maximum $g_i(x_i)$ over X_i. The case $m=2$ is almost as obvious, since it is easily shown that Y_i is completely determined by the image under g_i of the g_i-efficient points of X_i. That is, under assumption A1, $y_i \epsilon Y_i \Leftrightarrow y_i \leqq g_i(x_i)$ for some x_i in X_i that is g_i-efficient. A point $x_i \epsilon X_i$ is said to be 'g_i-efficient' if there exists no other point x_i' in X_i such that $g_i(x_i') \geqq g_i(x_i)$ with strict inequality holding for at least one

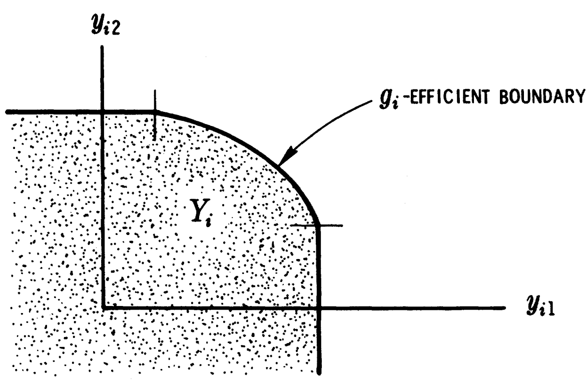

Figure 1

component function of g_i. *See* Fig. 1. The g_i-efficient boundary can be found with considerable computational efficiency under various additional assumptions on g_i and X_i by using special parametric programming techniques to solve the parametric program

$$\text{maximize}_{x_i \epsilon X_i} \ \theta g_{i1}(x_i) + (1-\theta) g_{i2}(x_i)$$

for values of θ in the unit interval.[19,20,21,40] Once the g_i-efficient boundary is known, it is a relatively simple matter to translate this into one or more inequality constraints on y_i to as great a degree of approximation as desired.

For $m \geq 3$, extensions of the above approach become computationally unattractive. If X_i is convex polyhedral and g_i is linear, however, the following result shows that Y_i can be specified without approximation by a finite collection of linear inequalities. In this case, (10) becomes a block-diagonal linear program with coupling constraints.

THEOREM 4. *Suppose that* $X_i = \{x_i \geqq 0 : A_i x_i \leqq b_i\}$ *and* $g_i(x_i) = B_i x_i$, *where*

A_i and B_i are given matrices and b_i is a given vector. Then Y_i can be expressed as the finite collection of linear inequalities $b_i^t\mu_i^j - y_i^t\lambda_i^j \geq 0, j=1, \cdots, J_i$, where $\{(\mu_i^1, \lambda_i^1), \cdots, (\mu_i^{J_i}, \lambda_i^{J_i})\}$ is a set of vectors spanning the cone

$$C_i \triangleq \{(\mu_i, \lambda_i): \mu_i \geqq 0, \lambda_i \geqq 0, A_i^t\mu_i - B_i^t\lambda_i \geqq 0\}.$$

Proof. The elementary theory of linear programming provides a simple and constructive proof as follows. Clearly, $y_i \epsilon Y_i$ if and only if the linear program

$$\text{maximize}_{x_i \geqq 0}\, 0^t x_i \quad \text{subject to} \quad A_i x_i \leqq b_i \quad \text{and} \quad B_i x_i \geqq y_i$$

is feasible and has optimal value 0. By the Dual Theorem of linear programming, this is true if and only if the dual linear program

$$\text{minimize}_{\mu_i, \lambda_i}\, b_i^t\mu_i - y_i^t\lambda_i \quad \text{subject to} \quad (\mu_i, \lambda_i) \epsilon C_i \tag{11}$$

is also feasible and has optimal value 0. Now C_i is a pointed convex polyhedral cone no matter what y_i is and, consequently, (11) will have optimal value 0 if and only if its minimand is nonnegative along every extreme ray. If we select any nonzero vector (μ_i^j, λ_i^j) from the jth ray, we obtain the desired conclusion.

Thus one could, if desired, generate Y_i explicitly by generating a complete set of spanning vectors for the associated convex polyhedral cone. Algorithms for doing this are known,[27,43] but the computational burden is likely to be excessive. Because only a small proportion of the corresponding linear constraints will probably ever be binding at an optimal solution of (9), it would be natural to solve (9) by the strategy of relaxation:[22] violated constraints could be generated as needed by linear programming as implicitly suggested by the proof of Theorem 4. That is, one would temporarily ignore all but a subset of the constraints defining each Y_i, solve the corresponding approximation to (9), and then introduce a violated constraint if the resulting solution $\bar{y}$ is not feasible in (9). Testing the ignored constraints for feasibility at $\bar{y}$ would be done by solving the linear programs (11) for each i with $y_i = \bar{y}_i$, and a violated constraint would be produced automatically if the optimal value is <0, i.e., if $\bar{y}_i \notin Y_i$. An optimal solution of (9) will be obtained in a finite number of iterations. This manner of handling implicit constraint sets like Y_i is due to Benders.[6]

The relaxation strategy for solving (9) can be generalized to apply even when Y_i is not polyhedral. This requires a method of testing whether a given trial solution $\bar{y}$ is feasible in (9), and of obtaining a 'violated' supporting hyperplane to Y_i if $\bar{y}_i \notin Y_i$. Again, the successive approximations to (10) become block diagonal linear programs with coupling constraints. Of course an optimal solution of (9) may not be obtained in a finite number of iterations, since in general it takes an infinite number of supporting hyperplanes to characterize Y_i completely. *See* Fig. 2, in which the approximating polytope is sketched (shaded) for a hypothetical case in which $m=2$ and 3 supports to Y_i are available. The fundamental characterization theorem, and therefore the main underpinning of the relaxation strategy, is as follows.

THEOREM 5. *A point y_i is in Y_i if and only if it satisfies the system of linear constraints*

$$\lambda_i^t y_i \leq \text{maximum}_{x_i \epsilon X_i} \lambda_i^t g_i(x_i), \quad \text{all} \quad \lambda_i \epsilon \Lambda, \tag{12}$$

where $\Lambda \triangleq \{\lambda_i \epsilon R^m : \lambda_i \geqq 0$ *and* $\sum_{j=1}^{j=m} \lambda_{ij} = 1\}$. *Furthermore, every constraint among* (12) *is a supporting half-space for* Y_i.

Proof. It is obvious that the system $g_{ij}(x_i) \geq y_{ij}$, $j=1, \cdots, m$, has a solution in X_i only if every convex linear combination of these constraints also has a solution in X_i. The converse also holds, by a fundamental re-

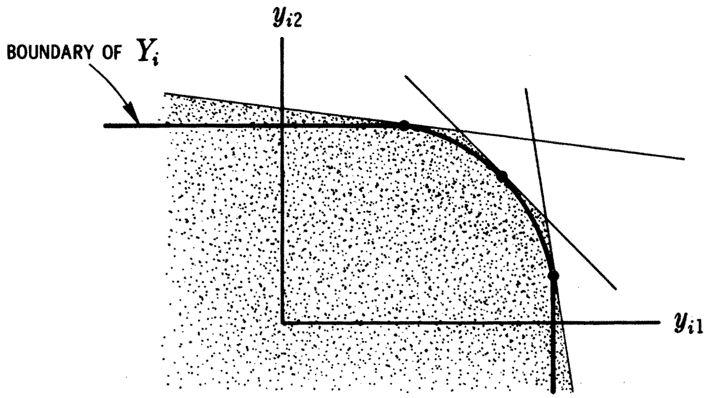

Figure 2

sult in the theory of convexity (p. 64 of reference 7). The first assertion now follows directly. To see the second assertion, we note that the first implies that every constraint from (12) is a containing half-space for Y_i, and must actually be supporting, because equality holds if we take $y_i = g_i[x_i(\lambda_i)]$, where $x_i(\lambda_i)$ maximizes $\lambda_i^t g_i(x_i)$ over X_i.

This characterization suggests several possible methods for testing whether a trial point $\bar{y}_i$ is in Y_i and generating a violated constraint from (12) if not. In the interest of computational efficiency, however, the choice of such a method should be coordinated with the choice of an algorithm for solving $(P_{y^{\nu+1}}^i)$ at Step 3. The chosen method ought to be able to make use of the available optimal solution to $(P_{y^\nu}^i)$, and ought to furnish a feasible (if not optimal) solution to $(P_{\bar{y}}^i)$ if $\bar{y}_i \epsilon Y_i$. Rather than attempt a detailed discussion here, we shall be content to suggest the following general requirement:

Requirement for Step 3. The algorithm chosen to implement Step 3 must be able to commence if necessary without the benefit of a known feasible solution of $(P_{\bar{y}}{}^i)$ and, if $(P_{\bar{y}}{}^i)$ has no feasible solution, must determine a convex combination of the constraints $g_i(x_i) \geqq \bar{y}_i$ that has no solution in X_i (such a convex combination exists by virtue of Theorem 5).

This requirement is not unduly stringent: it can be shown that most 'dual' methods addressed to $(P_{\bar{y}}{}^i)$ satisfy it automatically, as do most 'primal' methods when fitted with a 'Phase One' procedure designed to find an initial feasible solution if one exists. It certainly enables the relaxation strategy for (9) to be carried out using the machinery of Step 3, for if $\bar{y}_i \notin Y_i$, then $\bar{\lambda}_i \epsilon \Lambda$ is produced such that the constraint $\bar{\lambda}_i{}^t g_i(x_i) \geq$

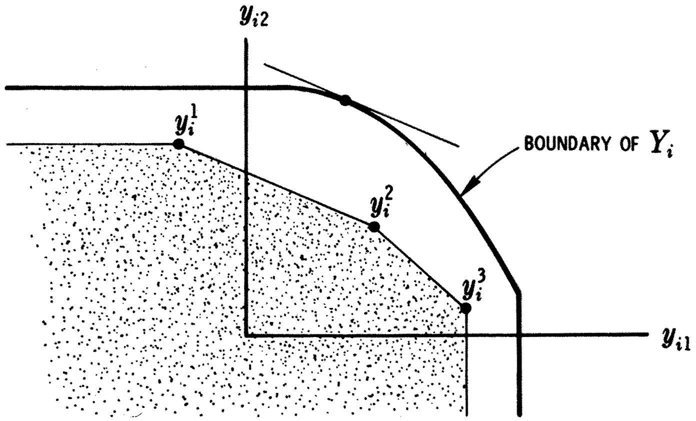

Figure 3

$\bar{\lambda}_i{}^t \bar{y}_i$ has no solution in X_i—that is, such that $\bar{\lambda}_i{}^t y_i \leq \max_{x_i \epsilon X_i} \bar{\lambda}_i{}^t g_i(x_i)$ is a violated constraint from (12).

In addition to relaxation methods for handling Y_i in (9) by outer convex polyhedral approximation, there are also inner convex polyhedral approximation methods to be considered. If $\langle y_i^1, \cdots, y_i^{p_i} \rangle$ are known points in Y_i, the polytope

$$\{y_i \epsilon R^m : y_i \leqq \textstyle\sum_{j=1}^{j=p_i} \alpha_{ij} y_i{}^j \quad \text{for some} \quad \alpha_i \geqq 0, \ \sum_{j=1}^{j=p_i} \alpha_{ij} = 1\}$$

is contained in Y_i. *See* Fig. 3, where the approximating polytope is sketched (shaded) for a hypothetical case in which $m=2$, $p_i=3$. The corresponding approximation to (9) would obviously be

$$\begin{aligned} &\text{maximize}_{y,\alpha} \textstyle\sum_{i=1}^{i=k} v_i^{\nu}(y_i) \quad \text{subject to:} \\ &\textstyle\sum_{i=1}^{i=k} y_i \geqq b; \qquad y_i \leqq \sum_{j=1}^{j=p_i} \alpha_{ij} y_i{}^j, \qquad \sum_{j=1}^{j=p_i} \alpha_{ij} = 1, \quad \text{and} \qquad (13) \\ &\alpha_i \geqq 0, \quad i=1, \cdots, k. \end{aligned}$$

When $v_i^{\nu}(y_i)$ is handled as in (10), (13) becomes a block-diagonal linear program (k blocks) with coupling constraints. One appealing way of refining the approximation is to make use of the optimal dual variables of (13) corresponding to the constraints $y_i \leqq \sum_{j=1}^{j=p_i} \alpha_{ij} y_i^j$. Denote the values of these variables by $\hat{u}_i$. Solve the concave program

$$\text{maximize}_{x_i \epsilon X_i} \ \hat{u}_i^t g_i(x_i) \tag{14}$$

for an optimal solution $\hat{x}_i$, and introduce the following new point (which must lie on the boundary of Y_i): $y_i^{p_i+1} \triangleq g_i(\hat{x}_i)$.

The motivation behind this choice of $y_i^{p_i+1}$ rests upon the fact that $\hat{u}_i$ can be shown to be the normal of a supporting hyperplane to the current inner polyhedral approximation to Y_i at $\hat{y}$, where $\hat{y}$ solves (13). In Fig. 3, if the optimal solution to (13) lies strictly between y_i^1 and y_i^2, y_i^4 would be the point on the boundary of Y_i indicated by the heavy dot. Thus the polyhedral approximation to Y_i would be improved in the region of interest until no further significant improvement is possible, at which point (9) would be adequately optimized.

It is interesting to observe that this method for handling Y_i is equivalent to rewriting (9) as

$$\text{maximize}_{y,x} \sum_{i=1}^{i=k} v_i^{\nu}(y_i) \quad \text{subject to:}$$
$$x_i \epsilon X_i, \quad i=1, \cdots, k; \qquad g_i(x_i) \geqq y_i, \quad i=1, \cdots, k; \qquad \sum_{i=1}^{i=k} y_i \geqq b, \tag{15}$$

and then applying DANTZIG AND WOLFE's nonlinear decomposition method (Chapter 24 of reference 12) only to the g_i functions (in the terminology of reference 22, one would describe this as inner-linearizing g_i over X_i and then applying the strategy of restriction).

3.2 Alternative to Assumption A2

Let $y_i \epsilon Y_i$ be fixed such that v_i is finite. It is known[23,32] that an optimal multiplier vector for (P_y^i) exists if and only if the one-sided directional derivative

$$v_i'(y_i; z_i) \triangleq \lim_{t \to 0+} [v_i(y_i + t z_i) - v_i(y_i)]/t \tag{16}$$

is not $+\infty$ in any direction $z_i \epsilon R^m$; and (*see* reference 16, p. 80) that $v_i'(y_i; z_i)$ is finite for all z_i if y_i is in the interior of $Y_i(\text{int}Y_i)$. Thus *an optimal multiplier vector must automatically exist everywhere on* $\text{int}Y_i$, and assumption A2 is needed only for the boundary of Y_i.

This observation suggests that one might dispense with assumption A2 altogether, provided that y_i is restricted to $\text{int}Y_i$. This can be done by perturbing y_i^{ν} slightly, if necessary, so that it is sure to lie in $\text{int}Y_i$. Any m-vector with all negative components will do as a perturbation direction. Or, if a point $\tilde{y}$ satisfying $\sum_{i=1}^{i=k} \tilde{y}_i \geqq b$ and $\tilde{y}_i \epsilon \text{int}Y_i$ is known, then the perturbation direction $(\tilde{y}_i - y_i^\nu)$ will point into the interior of Y_i from every boundary point and avoid even miniscule violations of the constraints $\sum_{i=1}^{i=k} y_i \geqq b$ after perturbation. If the amount of perturbation tends to 0 as $\nu \to \infty$, then the perturbed sequence $\langle y_i^\nu \rangle$ can converge to a boundary point of Y_i if an optimal solution to (P) is, in fact, on the boundary.

An additional argument for dispensing with assumption A2 comes from the observation that optimal solutions of (P) tend not to occur at points y^* where an optimal multiplier vector does not exist (i.e., where $v_i'(y_i^*; z_i) = +\infty$ for some z_i), for then $v_i(y_i)$ increases at an infinite rate as y_i departs from y_i^* in direction z_i into the interior. (It is known [reference 16, p. 79] that if $v_i'(y_i^*; z_i) = \infty$ for *some* z_i, then it is ∞ for *all* z_i pointing into the interior of Y_i.) This implies a natural repulsion away from any points on the boundary of Y_i at which assumption A2 fails.

4. LARGE-STEP SUBGRADIENT APPROACH TO (*P*)

SOME OF THE most powerful algorithms for concave programming with differentiable functions are so-called 'large-step gradient methods.' [10,41,42] (These methods are known to produce near-optimal solutions quickly, although to obtain high accuracy it may be preferable to resort to second-order methods.) Such methods produce a sequence of feasible solutions, each determined from the previous one by the choice of an improving feasible direction followed by a step in this direction. The gradient of the extremal function usually guides the choice of an improving feasible direction, and the step size is usually determined by maximizing the extremal function along the chosen direction subject to the constraints. Unfortunately, these algorithms do not apply directly to (P), because its maximand need not be differentiable even if all of the functions defining (1) are.

In this section we shall suggest how such algorithms can be modified so as to overcome the nondifferentiability obstacle. The optimal multipliers of (P_y^i) will play a central role in obtaining a complete local characterization of v_i via subgradients, thereby enabling an 'optimal' choice of an improving feasible direction (or proof that none exists). These developments, conveyed in Sections 4.1 and 4.2, will enable Step 2 of the following procedure to be carried out successfully.

LARGE-STEP SUBGRADIENT APPROACH

Step 1. Let a feasible solution y^0 of (P) be given such that the corresponding subproblems are feasible. Solve each $(P_{y^0}^i)$ for an optimal solution by any suitable algorithm for concave programming; if any subproblem has an unbounded optimal value, then so does (P) and the procedure terminates.

Step 2. Determine a 'good' improving feasible direction z^6 for (P) at y^0; if no improving feasible direction exists, then y^0 is optimal in (P) and the procedure terminates.

Step 3. Determine a step size θ^0 in the direction z^0 by solving the following one-dimensional restricted version of (P):

$$\text{maximize}_{\theta \geq 0} \sum_{i=1}^{i=k} v_i(y_i^0 + \theta z_i^0) \quad \text{subject to} \quad \sum_{i=1}^{i=k} (y_i^0 + \theta z_i^0) \geqq b. \tag{17}$$

If (17) has unbounded optimal value, then so does (P), and the procedure terminates. Denote $y^0 + \theta^0 z^0$ by y', and recover an optimal solution to each $(P_{y'}^i)$. Return to Step 2 with y^0 replaced by y'.

It is implicitly assumed that $(P_y^{\,i})$ has an optimal solution whenever it is feasible and v_i is not infinite.

Problem (17) at Step 3 involves maximizing a concave function of a single variable over a closed interval [the general constraints of (17) are equivalent to a readily computed upper bound on θ]. Perhaps the easiest way of performing this maximization when parametric programming techniques[19,20,24,35] are available for the parametric subproblems $(P_{y^0+\theta z^0}^i)$ is to increase θ by small increments above 0, all the while maintaining optimal solutions of the subproblems, until the maximand ceases to increase or the upper bound for θ is reached, whichever occurs first. Note that the parametric programming techniques are applied to each subproblem independently of the others, and that the required optimal solution of $(P_{y'}^i)$ is produced with no extra work. If parametric programming techniques are not available, (17) could be executed using Fibonacci search to economize on the number of evaluations of v_i.

We must now grapple with the central technical difficulty inherent in this approach, namely how to carry out Step 2. The importance of being able to find 'good' improving feasible directions can hardly be overemphasized, for the chosen directions largely determine the sequence of trial resource allocations.

4.1 Step 2

An improving feasible direction for (P) at y^0, of course, is given by any z such that $y^0 + \theta z$ is feasible in (P) for some $\theta > 0$ and $\sum v_i(y_i^0 + \theta z_i) > \sum v_i(y_i^0)$ for all $\theta > 0$ sufficiently small. It is easy to see that such a direction fails to exist if and only if y^0 is actually optimal in (P).

It is essential to recognize that we must reckon not only with the problem of finding an improving feasible direction if one exists, but also with the problem of choosing one from among the many when more than one exists. The criterion by which this choice is made is what mainly distinguishes the various large-step gradient methods referred to above. Similarly, we have our choice here. We shall work with the following appealing criterion (cf. reference 41):

Criterion. Choose the improving feasible direction that maximizes the initial rate of improvement of the value of the maximand.

The initial rate of improvement is, of course, measured by the directional derivative $\sum_{i=1}^{i=k} v_i'(y_i^0;z_i)$ [*see* (16)]. The direction-finding problem is therefore

$$\text{maximize}_z \sum_{i=1}^{i=k} v_i'(y_i^0;\, z_i) \quad \text{subject to}$$
$$\sum_{i=1}^{i=k} z_{ij} \geq 0 \quad \text{for } j \text{ such that} \quad \sum_{i=1}^{i=k} y_{ij}^0 = b_j, \tag{18}$$
$$-1 \leq z_{ij} \leq 1 \quad \text{for all } i \text{ and } j.$$

The first set of linear constraints simply ensures that z is a feasible direction. The normalization constraint $-1 \leq z_{ij} \leq 1$ accounts for the fact that v_i' is homogeneous of degree one (obviously norms other than l_∞ could be used to define the normalization).

Problem (18) is a concave program, for it is easy to see that v_i' is concave in z_i for each fixed y_i such that $v_i(y_i)$ is finite. The optimal value of (18) is nonnegative, for $z=0$ is feasible, and equals 0 if and only if no improving feasible direction exists. Any feasible solution with value exceeding 0 provides an improving feasible direction, and conversely. An optimal solution of (18), of course, yields (provided its value exceeds 0) an improving feasible direction that is best according to the chosen criterion.

Variants of the above criterion would lead to variants of (18), many of which could be handled by the techniques to follow, perhaps combined with the techniques of Section 3.1. One close variant was treated approximately in the completely linear case by Zschau[43] (*see also* references 1 and 39). We shall see that under mild assumptions (18) can be expressed as an explicit linear program.

[Since completing this work, it has come to the author's attention that G. Silverman[35] has extended Zschau's treatment[43] to a concave case almost as general as the one considered here. His 'reallocation' problem corresponds to our problem (18), but he proposes a solution procedure for it that seems considerably more complicated. Also of interest is his extension of the results in reference 20 to obtain a parametric programming procedure for what corresponds to Step 3.]

4.2 Solving the Direction-Finding Problem

Consider now how one might solve (18). The difficulty, of course, is that an explicit expression for v_i' seems to be as elusive as an explicit expression for v_i itself. Fortunately, it turns out that one can indeed express v_i' in useful terms. To do this, one must make use of the concavity of v_i (Theorem 2), the theory of subgradients of concave functions, and the connection between subgradients of v_i and optimal multipliers of (P_y^i). Fundamental background material can be found in references 16, 23, 32.

A *subgradient* of v_i at a point $\bar{y}_i$ for which v_i is finite is any m-vector p_i with the property $v_i(y_i) \leq v_i(\bar{y}_i) + p_i^t(y_i - \bar{y}_i)$ for all y_i. That is, the subgradients of v_i at a given point are simply the outer normals of the linear supports at this point. The essential fact relating subgradients of v_i and optimal multipliers of (P_y^i) is that they are precisely the negatives of one another, as the following key theorem shows.

THEOREM 6. *Let $\bar{x}_i$ be any optimal solution of $(P_{\bar{y}}^i)$. Then $\bar{\lambda}_i$ is an optimal multiplier vector associated with the constraints $g_i(x_i) \geqq \bar{y}_i$ of $(P_{\bar{y}}^i)$ if and only if $-\bar{\lambda}_i$ is a subgradient of v_i at $\bar{y}_i$; that is, the pair $(\bar{x}_i, \bar{\lambda}_i)$ satisfies the Kuhn-Tucker conditions*

(i) *$\bar{x}_i$ maximizes $f_i(x_i) + \bar{\lambda}_i^t[g_i(x_i) - \bar{y}_i]$ over X_i,*

(ii) $\bar{\lambda}_i^t[g_i(\bar{x}_i) - \bar{y}_i] = 0$,

(iii) $\bar{\lambda}_i \geqq 0$,

if and only if

$$v_i(y_i) \leq v_i(\bar{y}_i) - \bar{\lambda}_i^t(y_i - \bar{y}_i) \quad \textit{for all} \quad y_i. \tag{19}$$

Proof. We suppress the subscript i. The 'only if' part of the conclusion is available from Theorem 3. To demonstrate the 'if' part, let $-\bar{\lambda}$ be a subgradient of v at $\bar{y}$. We need to show that $(\bar{x}, \bar{\lambda})$ satisfies (i)–(iii). To establish (iii), suppose to the contrary that $\bar{\lambda}_j < 0$ for some j. Then by taking $y = \bar{y}$ except for the jth component, which we let equal $\bar{y}_j - 1$, (19) yields a contradiction (since v is obviously a nonincreasing function). To establish (ii), suppose to the contrary that $\bar{\lambda}_j > 0$ for some j such that $g_j(\bar{x}) > \bar{y}_j$. Then by taking $y = \bar{y}$ except for the jth component, which we set equal to $g_j(\bar{x})$, (19) yields the contradiction

$$v(\bar{y}) = v(y) \leq v(\bar{y}) - \bar{\lambda}_j(y_j - \bar{y}_j) < v(\bar{y}).$$

To establish (i), we must show [in view of (ii)] that $x' \epsilon X$ implies

$$f(x') \leq f(\bar{x}) - \bar{\lambda}^t[g(x') - y].$$

But this follows from (19) upon letting $y = g(x')$:

$$f(x') \leq v[g(x')] \leq v(\bar{y}) - \bar{\lambda}^t[g(x') - \bar{y}] = f(\bar{x}) - \bar{\lambda}^t[g(x') - \bar{y}].$$

The proof is now complete.

We shall use the concavity of v_i, the theory of subgradients, and Theorem 6 to obtain an expression for v_i' under mild hypotheses. Before getting to the main result, let us illustrate our pattern of reasoning with a preliminary result. The concavity of v_i implies that it is differentiable almost everywhere on any open set on which it is finite, and that it is differentiable at a point $\bar{y}_i$ at which it is finite if and only if it has a unique subgradient there;

and, in this case, that $v_i'(\bar{y}_i; z_i)$ is given by $\bar{p}_i^t z_i$, where $\bar{p}_i$ is the unique subgradient. Making use of Theorem 6, this result translates as follows: if $(P_{\bar{y}}^i)$ has an optimal solution, then v_i is differentiable at $\bar{y}_i$ if and only if $(P_{\bar{y}}^i)$ has a unique optimal multiplier $\bar{\lambda}_i$, and in this case

$$v_i'(\bar{y}_i; z_i) = -\bar{\lambda}_i^t z_i \quad \text{for all} \quad z_i. \tag{20}$$

This is a nice result, but is not general enough, because it turns out that points of nondifferentiability tend to arise often. A sufficiently general result for our purposes is the following.

THEOREM 7. *Let $\bar{y}_i$ be interior to Y_i, $\bar{x}_i$ be any optimal solution of $(P_{\bar{y}}^i)$, X_i be given by the concave constraints $h_{ij}(x_i) \geq 0$ $(j=1, \cdots, m_i)$, and the functions f_i, g_{ij}, and h_{ij} be continuously differentiable. Assume that the constraint set $\{x_i : h_i(x_i) \geqq 0\}$ admits a solution that satisfies all nonlinear constraints with strict inequality (this is but one of a number of constraint qualifications of X_i that would suffice here). Then $v_i'(\bar{y}_i; z_i)$ equals, for all z_i, the optimal value of the linear program*

$$\begin{aligned} &\text{minimize}_{\lambda_i \geqq 0, \mu_i \geqq 0} -\lambda_i^t z_i \quad \text{subject to} \\ &\lambda_i^t[g_i(\bar{x}_i) - \bar{y}_i] = 0, \qquad \mu_i^t h_i(\bar{x}_i) = 0, \\ &\nabla f_i(\bar{x}_i) + \lambda_i^t \nabla g_i(\bar{x}_i) + \mu_i^t \nabla h_i(\bar{x}_i) = 0, \end{aligned} \tag{21}$$

where μ_i is an m_i-vector. [Here $\nabla f_i(\bar{x}_i)$ is the gradient of f_i at $\bar{x}_i$ expressed as a row vector; $\nabla g_i(\bar{x}_i)$ is a matrix whose jth row is the gradient of g_{ij} at $\bar{x}_i$; a similar definition holds for $\nabla h_i(\bar{x}_i)$.]

Proof. It follows directly from the concavity of v_i and the fact that $\bar{y}_i \epsilon \operatorname{int} Y_i$ that $v_i'(\bar{y}_i; z_i)$ equals (for any z_i) the minimum of $p_i^t z_i$ over the (necessarily nonempty compact convex) set of subgradients at $\bar{y}_i$. Applying Theorem 6, therefore, we have

$$v_i'(\bar{y}_i; z_i) = \min\{-\lambda_i^t z_i : \lambda_i \quad \text{is an optimal multiplier vector of} \quad (P_{\bar{y}}^i)\}.$$

Now λ_i is an optimal multiplier vector of $(P_{\bar{y}}^i)$ if and only if it satisfies conditions (i)–(iii) of Theorem 6. Conditions (ii) and (iii) appear directly in (21), while condition (i) appears in an equivalent form derived by applying the Kuhn-Tucker Theorem to the concave program therein and explicitly introducing the multipliers associated with the constraints $h_i(x_i) \geqq 0$.

Before using this characterization of v_i' in (18), let us write it in equivalent form. First observe that the first two groups of constraints in (21) express 'complementary slackness.' Since $g_i(\bar{x}_i) - \bar{y}_i \geqq 0$ and $h_i(\bar{x}_i) \geqq 0$ by the feasibility of $\bar{x}_i$ in $(P_{\bar{y}}^i)$, these constraints merely assert the following:

$\lambda_{ij}=0$ whenever $g_{ij}(\bar{x}_i)-\bar{y}_{ij}>0$, $\mu_{ij}=0$ whenever $h_{ij}(\bar{x}_i)>0$.

Hence (21) can be rewritten as

$$\begin{aligned}&\text{minimize}_{\lambda_{ij}\geq 0,\mu_{ij}\geq 0}\textstyle\sum_j \lambda_{ij}z_{ij} \quad \text{subject to}\\ &\nabla f_i(\bar{x}_i)+\textstyle\sum_j \lambda_{ij}\nabla g_{ij}(\bar{x}_i)+\sum_j \mu_{ij}\nabla h_{ij}(\bar{x}_i)=0,\end{aligned} \tag{22}$$

where the index j on λ_i or μ_i is understood in each case to range over the corresponding constraints which are taut (rather than amply satisfied). Now dualize (22), a manipulation justified by the Dual Theorem of linear programming, since $\bar{y}_i \epsilon \operatorname{int} Y_i$ implies that $v_i'(\bar{y}_i; z_i)$ is finite for all z_i. Consequently, the conclusion of Theorem 7 asserts that $v_i'(\bar{y}_i; z_i)$ equals, for all z_i, the optimal value of the linear program:

$$\begin{aligned}&\text{maximize}_{w_i}\ \nabla f_i(\bar{x}_i)w_i \quad \text{subject to}\\ &\nabla g_{ij}(\bar{x}_i)w_i\geq z_{ij}, \quad j \text{ such that } g_{ij}(\bar{x}_i)-\bar{y}_{ij}=0,\\ &\nabla h_{ij}(\bar{x}_i)w_i\geq 0, \quad j \text{ such that } h_{ij}(\bar{x}_i)=0,\end{aligned} \tag{23}$$

where w_i is an n_i-dimensional vector of dual variables.

At last we obtain, using (23) in (18) and identifying y^0 with $\bar{y}$, the desired linear programming equivalent of (18) under the hypotheses of Theorem 7:

$$\begin{aligned}&\text{maximize}_{w,z}\ \textstyle\sum_{i=1}^{i=k} \nabla f_i(x_i^0)w_i \quad \text{subject to:}\\ &\nabla g_{ij}(x_i^0)w_i-z_{ij}\geq 0, \quad i=1,\cdots,k, \quad j \text{ such that } g_{ij}(x_i^0)=y_{ij}^0;\\ &\nabla h_{ij}(x_i^0)w_i\geq 0, \quad i=1,\cdots,k, \quad j \text{ such that } h_{ij}(x_i^0)=0;\\ &\textstyle\sum_{i=1}^{i=k} z_{ij}\geq 0, j \ \text{ such that } \sum_{i=1}^{i=k} y_{ij}^0=b_j; \ -1\leq z_{ij}\leq 1, \ \text{all } i \text{ and } j.\end{aligned} \tag{24}$$

Note that this linear program is block-diagonal with coupling constraints, and is therefore amenable to special solution techniques. In addition, observe that the only essential z_{ij} variables are those appearing in the first group of constraints; the rest can be fixed at their upper bounds and dropped from the problem. The remaining z_{ij} variables can be translated by an obvious substitution so that they satisfy the usual nonnegativity convention of linear programming, and the upper bounding constraints can be handled implicitly using special techniques.

It is interesting and perhaps surprising that (24) can be interpreted as a direction-finding problem addressed to (4). From this viewpoint, w_i is interpreted as an *unnormalized* displacement away from x_i^0, and of course z_i is a normalized displacement away from y_i^0.

Theorem 7, and consequently (24), assumes $\bar{y}_i \epsilon \operatorname{int} Y_i$. To enforce this assumption, one can restrict y_i to the interior of Y_i by starting in the interior and then always using a step size slightly shorter than the value θ^0 computed at Step 3 of Procedure 2. If the amount of perturbation approaches 0 as the number of iterations increases, an optimal solution of (P) can be approached arbitrarily closely even if it should happen to be on the boundary of some Y_i.

Using the results of Section 3.2 along with results analogous to the above for $\bar{y}_i$ on the boundary of Y_i, however, it can be shown that (24) is still equivalent to (18) for any point $\bar{y}$ that could be generated at Step 3, provided only that the initial y^0 is such that each $(P_{y^0}^i)$ has an optimal multiplier vector.

5. PIECEWISE APPROACH TO (P)

THE FUNDAMENTAL observation underlying the so-called 'piecewise' approach is that the functions v_i usually have a significantly simpler structure over certain regions of R^m. This is because different subsets of the constraints of (P_y^i) are active at different values of y_i. The piecewise approach attempts to capitalize on this fact by implicitly subdividing R^m into the regions on which all of the v_i have a relatively simple structure, and then seeking an optimal solution of (P) in a piecemeal fashion, each time with y restricted to one of these regions.

We present two applications of this approach: one for the case in which (1) is a linear program, and one for the case in which (1) is a quadratic program. The first case illustrates the fundamental nature of the approach clearly without any technical clutter, while the second illustrates the technical methods required for more general application. *See* reference 22 for additional discussion of this approach.

5.1 The Linear Case

Let (1) be a linear program. Then (P_y^i) is a linear program parameterized in the 'right-hand side,' and we know from the theory of linear programming that v_i is a piecewise-linear function, that the regions of R^m on which v_i is linear are precisely the subsets of R^m on which a given optimal basis for (P_y^i) remains feasible, and that these subsets are convex polytopes. One is thus led to the following procedure for solving (P).

PIECEWISE APPROACH (LINEAR CASE)

Step 1. Let a feasible solution y^0 of (P) be given such that the corresponding subproblems are feasible. Solve each $(P_{y^0}^i)$ for an optimal solution by a linear programming algorithm; if any subproblem has an unbounded optimal value, then so does (P), and the procedure terminates. Denote by $\mathcal{R}_i^0$ the (convex polyhedral) region of R^m on which the current optimal basis for $(P_{y^0}^i)$ remains feasible, and denote $v_i(y_i)$ on this region by $l_i^0(y_i)$ (l_i^0 is a *linear* function).

Step 2. Solve (P) with the additional constraint that y_i be restricted to $\mathcal{R}_i^0$, namely,

$$\begin{gathered} \text{maximize}_y \textstyle\sum_{i=1}^{i=k} l_i^0(y_i) \quad \text{subject to:} \\ y_i \in \mathcal{R}_i^0, \qquad i=1, \cdots, k; \qquad \textstyle\sum_{i=1}^{i=k} y_i \geqq b, \end{gathered} \tag{25}$$

by a linear programming algorithm (y^0 is a feasible solution). Let y' be an optimal solution; if (25) has an unbounded optimal value, then so does (P) and the procedure terminates.

Step 3. If y' is optimal in (P), terminate. Otherwise, identify and change to an alternate optimal basis in some of the subproblems $(P_{y'}^i)$ so that y' is free to move in an improving feasible direction for (P) without requiring a further basis change in any of the subproblems. Determine the corresponding new regions $\mathcal{R}_i'$ and functions l_i', and return to Step 2 with y^0, $\mathcal{R}_i^0$, and l_i^0 replaced by y', $\mathcal{R}_i'$, and l_i'.

This procedure owes a great debt to Rosen[34] (*see also* references 4, 18, 33, 38), although our use of improving feasible directions to guide the transition between adjacent regions (*see* the discussion below) seems novel.

Note that the subproblems are solved from scratch only once, at Step 1. Thereafter a simple parametric linear programming procedure suffices to maintain an optimal solution as y varies. Post-optimality techniques can also be used to reoptimize (25) each time Step 2 is executed. It is also important to recognize that the entities $\mathcal{R}_i$ and l_i required by (25) are all explicitly available from the current tableaux associated with the subproblems. We shall not take the space to develop the details here.[22,34] In practice one could, of course, work with the dual of (25), or with the duals of the subproblems.

If y' is not optimal in (P) at Step 3, then there must exist an improving feasible direction (cf. Section 4.1). Such a direction can be identified by the methods of Section 4—or alternatively, by making use of the theory of linear programming. In either case, as one might expect, the dual variables of the linear programs $(P_{y'}^i)$ play a central role. Let us suppose that an improving feasible direction z' has been identified. Displacement away from y' in the direction z' is not permissible in (25), for it would lead to a violation of some $\mathcal{R}_i^0$, i.e., a current optimal basis would become infeasible. To permit such a displacement one must change to an alternate basis optimal at y' in one or more of the subproblems according to the dictates of parametric linear programming theory applied to the parametric problems $(P_{y'+\theta z'}^i)$, where θ is a nonnegative parameter. In this way parametric linear programming theory guides the basic changes necessary to make the piecewise approach work.

The optimality test in Step 3 can be integrated with the determination of an improving feasible direction; y' is optimal in (P) if and only if such a direction fails to exist.

5.2 The Quadratic Case

Let (1) be a strictly concave quadratic program. That is, let each f_i be a negative definite quadratic polynomial and let all constraints be linear. It will prove convenient to introduce the constraints defining X_i explicitly as $h_{ij}(x_i) \geq 0, j=1, \cdots, m_i$.

To apply the piecewise approach, it is necessary to take explicit account of the fact that different subsets of the constraints of (P_y^i) are 'active' for different values of y_i. We shall denote a subset of constraint indices for the ith subproblem by S_i. Assuming that S_i coincides with the set of active constraints, we shall see that (P_y^i) is equivalent to

$$\begin{gathered} \text{maximize}_{x_i} f_i(x_i) \quad \text{subject to} \\ h_{ij}(x_i)=0, \quad j \epsilon S_i; \qquad g_{ij}(x_i)=y_{ij}, \quad j \epsilon S_i. \end{gathered} \tag{26}$$

Note that the constraints in S_i have been written as equalities rather than inequalities, and that the constraints not in S_i have been dropped entirely.

It turns out that it is sufficient to consider only index sets that are *allowable*. We say that S_i is allowable at y_i if (26) has a feasible solution and if the constraints indexed by it have linearly independent normals (remember that all constraints are linear). It is easy to see that (26) admits a unique optimal solution $x_i^s(y_i)$ when S_i is allowable at y_i, and furthermore that corresponding Lagrange multipliers $\lambda_i^s(y_i)$ and $\mu_i^s(y_i)$ must exist and be unique (λ_{ij}^s is associated with g_{ij}, and μ_{ij}^s with h_{ij}). The triple $[x_i^s(y_i), \lambda_i^s(y_i), \mu_i^s(y_i)]$ is the unique solution to the usual Lagrange multiplier equations associated with (26).

The relation between (26) and (P_y^i), when S_i is allowable at y_i, can be deduced from the standard optimality conditions for (P_y^i) and a fundamental theorem of linear programming stating that any set of linear equations with a nonnegative solution also has a basic nonnegative solution (e.g., reference 12). Let us call S_i *optimal* at y_i if $x_i^s(y_i)$ is feasible in (P_y^i) (i.e., satisfies the constraints *not* indexed by S_i), and $\lambda_i^s(y_i)$ and $\mu_i^s(y_i)$ are both nonnegative. The central fact about optimal index sets is that $x_i^s(y_i)$ must be optimal in (P_y^i) if S_i is optimal at y_i; and, moreover, to every y_i such that (P_y^i) admits a feasible solution, there corresponds at least one optimal index set. Thus (P_y^i) can be replaced by (26) so long as S_i is an index set optimal at y_i.

Let us define $\mathcal{R}_i(S)$ as the set of values for y_i at which a given index set S_i is optimal:

$$\begin{aligned} \mathcal{R}_i(S) \triangleq \{y_i \epsilon R^m : S_i \quad &\text{is allowable at } y_i, \quad h_{ij}[x_i^s(y_i)] \geq 0 \quad \text{for} \quad j \notin S_i, \\ g_{ij}[x_i^s(y_i)] \geq y_{ij} \quad \text{for} \quad j \notin S_i, \quad &\lambda_i^s(y_i) \geqq 0, \quad \text{and} \quad \mu_i^s(y_i) \geqq 0\}. \end{aligned} \tag{27}$$

If a particular $\mathcal{R}_i(S)$ is not empty, then from the linearity of the Lagrange multiplier equations associated with (26) and the consequent linearity of the functions $x_i^s(\cdot)$, $\lambda_i^s(\cdot)$, and $\mu_i^s(\cdot)$, we see that $\mathcal{R}_i(S)$ is a convex polytope. Furthermore, v_i must be quadratic on $\mathcal{R}_i(S)$ {since $v_i(y_i) = f_i[x_i^s(y_i)]$ holds}. Let us therefore denote $v_i(y_i)$ on $\mathcal{R}_i(S)$ by $q_i(y_i)$.

At last we are ready to detail the piecewise approach for the quadratic case. Any available quadratic programming algorithm (e.g., the one in reference 40) can be used in Steps 1 and 2.

Piecewise Approach (Quadratic Case)

Step 1. Let a feasible solution y^0 of (P) be given such that the corresponding subproblems are feasible. Solve each $(P_{y^0}^i)$ for an optimal solution by a quadratic programming algorithm. Determine an index set S_i^0 that is optimal at y_i^0, and determine the corresponding convex polytopes $\mathcal{R}_i(S^0)$ and the quadratic functions q_i^0.

Step 2. Solve (P) with the additional constraint that y_i be restricted to $\mathcal{R}_i(S^0)$, namely

$$\begin{aligned} &\text{maximize}_y \textstyle\sum_{i=1}^{i=k} q_i^0(y_i) \quad \text{subject to:} \\ &y_i \in \mathcal{R}_i(S^0), \quad i=1, \cdots, k; \qquad \textstyle\sum_{i=1}^{i=k} y_i \geqq b, \end{aligned} \tag{28}$$

by a quadratic programming algorithm (y^0 is an initial feasible solution). Let y' be an optimal solution.

Step 3. If y' is optimal in (P), terminate. Otherwise, identify and change to an alternate optimal index set S_i' in some of the subproblems so that y' is free to move in an improving feasible direction for (P) while still being feasible in the corresponding version of (28). Determine the corresponding new regions $\mathcal{R}_i(S')$ and functions q_i', and return to Step 2 with y^0, $\mathcal{R}_i(S^0)$, and q_i^0 replaced by y', $\mathcal{R}_i(S')$, and q_i'.

As before, parametric and postoptimality techniques (this time for quadratic programming) can be used to reoptimize the subproblems and (28). The constraints specifying $\mathcal{R}_i(S)$ and the functions q_i required by (28) are all explicitly available from the unique solution to the (linear) Lagrange multiplier equations associated with (26), and optimal index sets for (P_y^i) are easily determined from its optimal solution. The optimality test in Step 3 can be posed as the question of the nonexistence of an improving feasible direction, and the search for such a direction can be carried out by the methods of Section 4 or by the theory of quadratic programming. Parametric programming theory[20,24] can be used to guide the determination of appropriate new optimal index sets in Step 3.

6. DISCUSSION

We close with a brief discussion of price-directive approaches to (1) and some opportunities for further research.

6.1 Price-Directive Approaches

The tangential approximation, large-step subgradient, and piecewise approaches can all be applied so as to yield *price-directive* rather than resource-directive procedures for solving (1). The essential difference is that, instead of projecting (1) onto the space of its subsystem resources to obtain (P), one dualizes (1) with respect to the system constraints to obtain the dual problem

$$\text{minimize}_{\lambda \geqq 0} \sum_{i=1}^{i=k} w_i(\lambda) - \lambda^t b, \qquad (D)$$

where $w_i(\lambda)$ is defined as the supremal value of the parameterized subproblem

$$\text{maximize}_{x_i \epsilon X_i} f_i(x_i) + \lambda^t g_i(x_i). \qquad (D_\lambda{}^i)$$

Note that we have not introduced multipliers associated with the constraints defining X_i. The only difference between $(D_\lambda{}^i)$ and (3) is that here the same vector of 'prices' (or 'incentives') is used by all subsystems. Under a mild assumption, from the duality theory of nonlinear programming[23,32] it can be shown that (D) must have an optimal solution, and that an optimal solution of (1) can be recovered by solving the corresponding subproblems $(D_\lambda{}^i)$. Thus we may address ourselves to (D) rather than to (1) or (P). (D) is a convex program since w_i is convex (it is the supremum of a collection of linear functions).

Application of these three approaches to (D) is quite analogous to their application to (P). Some simplification even arises for the tangential approximation approach, since (i) the analog of Y_i, namely $\{\lambda \epsilon R^m : w_i(\lambda) < \infty\}$, is simply all of R^m under the additional assumption A1 (*see* Section 3), and (ii) a linear support to w_i, namely $w_i(\bar{\lambda}) + g_i{}^t(\bar{x}_i)(\lambda - \bar{\lambda})$, where $\bar{x}_i$ is optimal in $(D_{\bar{\lambda}}{}^i)$, is an immediate by-product of its evaluation at a point $\bar{\lambda}$. The tangential approximation approach to (D) generalizes the 'global approach' discussed by TAKAHASHI[36] and also, it can be shown, the Dantzig-Wolfe Decomposition approach to (1).

The closest relatives of the large-step subgradient approach to (D) found in the literature appear to be what are sometimes known as Lagrangean decomposition methods [*see* references 8 (Sec. 3.2), 15, 25 (Sec. III), 29, 36 ('local approach'), and 37]. These methods typically work with just the immediate subgradient implicit in (ii) above, and sometimes take steps of nonoptimal length.

The piecewise approach to (D) has not yet appeared in the literature except by dual interpretation in the completely linear case.

6.2 Research Opportunities

There are numerous opportunities for research into computational, economic, and theoretical aspects.

From the computational viewpoint, the central unresolved issue is the relative efficiency of the three approaches developed here (or six approaches, if we include the price-directive ones too). It seems quite plausible to expect tangential approximation to be most efficient when the v_i functions have little curvature and the sets Y_i can be handled more or less directly; the large-step subgradient approach to be most efficient when the parameterized subproblems $(P^i_{y^0+\theta z^0})$ can be handled effectively; and the piecewise approach to be most efficient when (1) is a quadratic program. But only extensive computational experience will tell. One important object should be to establish broad classes of problems for which a particular approach is superior in some sense. This would help practitioners choose an effective algorithm and guide further theoretical development. In pursuing this goal, one should pay special attention to the growth of solution time as a function of problem size and difficulty parameters such as k, m, and $\sum_{i=1}^{i=k} n_i$ within various given classes of problems. In large-scale programming it is primarily this kind of behavior, rather than comparative absolute solution times for a given problem, that distinguishes the durable algorithms from the transitory ones. Of course, studies of convergence rate and numerical stability are also required. Of particular interest is stability with respect to ϵ-optimal solutions of the subproblems when they are not linear or quadratic programs (cf. reference 17 and Sec. 7 of reference 23).

Another aim of computational studies should concern how to make the most effective use of the parallel-processing opportunities presented by the essentially uncoupled nature of the subproblems (P_y^i) and (D_λ^i). Computers with substantial parallel processing capabilities are already available. The Control Data 6600, for example, has ten peripheral processors (each with a separate magnetic core memory) in addition to the central processor; thus up to ten subsystems could be optimized simultaneously at the appropriate steps in each of the suggested procedures.

Many aspects of the resource-directive and price-directive approaches mentioned here have as yet to be systematically studied from the economic viewpoint. The economic interpretations and implications of information flows and the resource- and price-adjusting mechanisms are quite interesting from the point of view of the decentralized organization or the planned economy.[2,3,8,28,30,43]

Finally, we mention a few theoretical questions that deserve further study. At the head of the list is the general subject of convergence, especially the *rate* of convergence. It would also be useful to explore direct means of dealing with the boundary of Y_i in Sections 3 and 4, and to devise specialized solution-recovery techniques for the problems and subproblems that must be solved repeatedly. And there are a number of potentially useful extensions that could be pursued, having to do with certain kinds of nonseparability in the system criterion function or constraints; the presence of 'system' as well as subsystem variables in (1); the piecewise approach for more general classes of problems; and choice criteria for an improving feasible direction other than the one of Subsection 4.1 (computational studies would be especially helpful here).

ACKNOWLEDGMENTS

SUPPORT FOR this work was provided by the Ford Foundation under a Faculty Research Fellowship, by the National Science Foundation, and by the United States Air Force under Project Rand. It is a pleasure to acknowledge this support gratefully, and also helpful comments received from B. FOX, J. R. JACKSON, J. L. MIDLER, and D. A. WISMER.

REFERENCES

1. ABADIE, J., AND M. SAKAROVITCH, "Two Methods of Decomposition for Linear Programs," presented at the International Symposium on Mathematical Programming, Princeton University, August, 1967.
2. ARROW, K. J., AND L. HURWICZ, "Decentralization and Computation in Resource Allocation," in R. W. PFOUTS (ed.), *Essays in Economics and Econometrics*, University of North Carolina Press, Chapel Hill, 1960.
3. BAUMOL, W. J., AND T. FABIAN, "Decomposition, Pricing for Decentralization and External Economies," *Management Sci.* **11,** 1–32 (1964).
4. BEALE, E. M. L., "The Simplex Method Using Pseudo-Basic Variables for Structured Linear Programs," in R. L. GRAVES AND P. WOLFE (eds.), *Recent Advances in Mathematical Programming*, McGraw-Hill, New York, 1963.
5. BELL, E. J., "Primal-Dual Decomposition Programming," in *Preprints of the Proceedings of the Fourth International Conference on Operations Research*, I.F.O.R.S., Boston, 1966.
6. BENDERS, J. F., "Partitioning Procedures for Solving Mixed-Variables Programming Problems," *Numerische Mathematik* **4,** 238–252 (1962).
7. BERGE, C., AND A. GHOUILA-HOURI, *Programming, Games and Transportation Networks*, Wiley, New York, 1965.
8. BRADLEY, S. P., "Decomposition Programming and Economic Planning," Operations Research Center Report ORC-67-20, University of California, Berkeley, June, 1967.
9. CHENEY, E. W., AND A. A. GOLDSTEIN, "Newton's Method for Convex Programming and Tchebycheff Approximation," *Numerische Mathematik* **1,** 253–268 (1959).
10. COLVILLE, A. R., "A Comparative Study on Nonlinear Programming Codes," Tech. Report No. 320–2949, I.B.M. New York Scientific Center, New York, June, 1968.
11. DANSKIN, J. M., *The Theory of Max-Min*, Springer-Verlag, New York, 1967.

12. DANTZIG, G. B., *Linear Programming and Extensions*, Princeton University Press, Princeton, N. J., 1963.
13. ——— AND A. MADANSKY, "On the Solution of Two-Stage Linear Programs Under Uncertainty," in *Proc. Fourth Symposium on Math. Stat. and Prob.* Vol. I, University of California Press, Berkeley, 1961.
14. ——— AND P. WOLFE, "Decomposition Principle for Linear Programs," *Opns. Res.* **8,** 101–111 (1960).
15. FALK, J. E., "A Constrained Lagrangian Approach to Nonlinear Programming," Doctoral Dissertation, Dept. of Mathematics, University of Michigan, Ann Arbor, March, 1965.
16. FENCHEL, W., Lecture notes on *Convex Cones, Sets, and Functions*, Logistics Research Project, Princeton University, Princeton, N. J., 1953.
17. FOX, B. L., "Stability of the Dual Cutting Plane Algorithm for Concave Programming," RM-6147-PR, The Rand Corporation, Santa Monica, California, February 1970.
18. GASS, S. I., "The Dualplex Method for Large-Scale Linear Programs," Operations Research Center Report ORC-66-15, University of California, Berkeley, June, 1966.
19. ——— AND T. L. SAATY, "The Computational Algorithm for the Parametric Objective Function," *Naval Res. Log. Quart.* **2,** 39–45 (1955).
20. GEOFFRION, A. M., "Strictly Concave Parametric Programming, Parts I and II," *Management Sci.* **13,** 244–253 (1966) and 359–370 (1967).
21. ———, "Solving Bi-Criterion Mathematical Programs," *Opns. Res.* **15,** 39–54 (1967).
22. ———, "Elements of Large-Scale Mathematical Programming," Western Management Science Institute WP 144, University of California, Los Angeles, December, 1968. To appear in *Management Sci.*
23. ———, "Duality in Nonlinear Programming: A Simplified Applications-Oriented Development," Western Management Science Institute WP 150, University of California, Los Angeles, August, 1969.
24. GRIGORIADIS, M. D., AND K. RITTER, "A Parametric Method for Semi-Definite Quadratic Programs," *SIAM J. Control* **1,** 559–577 (1969).
25. GRINOLD, R. C., "Steepest Ascent for Large-Scale Linear Programs," Center for Research in Management Science Working Paper No. 282, University of California, Berkeley, September, 1969.
26. KELLEY, J. E., JR., "The Cutting-Plane Method for Solving Convex Programs," *J. Soc. Indust. and Appl. Math.* **8,** 703–712 (1960).
27. KOHLER, D. A., "Projections of Convex Polyhedral Sets," Operations Research Center Report ORC 67-29, University of California, Berkeley, August, 1967.
28. KORNAI, J., AND T. LIPTAK, "Two-Level Planning," *Econometrica* **33,** 141–169 (1965).
29. LASDON, L. S., "Duality and Decomposition in Mathematical Programming," *IEEE Trans. Sys. Sci. and Cyb., SSC-4,* **2,** 86–100 (1968).
30. MALINVAUD, E., "Decentralized Procedures for Planning," in E. MALINVAUD

AND M. O. L. BACHARACH (eds.), *Activity Analysis in the Theory of Economic Growth*, Macmillan, New York, 1967.

31. MILLER, C. E., "The Simplex Method for Local Separable Programming," in R. L. GRAVES AND P. WOLFE (eds.), *Recent Advances in Mathematical Programming*, McGraw-Hill, New York, 1963.
32. ROCKAFELLAR, R. T., "Convex Functions and Duality in Optimization Problems and Dynamics," in H. W. KUHN AND J. P. SZEGÖ (eds.), *Mathematical Systems Theory and Economics*, Springer-Verlag (to appear).
33. ROSEN, J. B., "Convex Partition Programming," in R. L. GRAVES AND P. WOLFE (eds.), *Recent Advances in Mathematical Programming*, McGraw-Hill, New York, 1963.
34. ———, "Primal Partition Programming for Block Diagonal Matrices," *Numerische Mathematik* **6,** 250–260 (1964).
35. SILVERMAN, G. J., "Primal Decomposition of Mathematical Programs by Resource Allocation," Tech. Memo. No. 116, Operations Research Dept., Case-Western Reserve University, Cleveland, August, 1968.
36. TAKAHASHI, I., "Variable Separation Principle for Mathematical Programming," *J. Opns. Res. Soc.* (Japan) **6,** 82–105 (1964).
37. UZAWA, H., "Iterative Methods for Concave Programming," Chap. 10 in K. ARROW, L. HURWICZ, AND H. UZAWA (eds.), *Studies in Linear and Nonlinear Programming*, Stanford University Press, Stanford, Calif., 1958.
38. VARAIYA, P., "Decomposition of Large-Scale Systems," *J. SIAM Control* **4,** 173–178 (1966).
39. WETS, R., "Programming Under Uncertainty: The Complete Problem," *Z. Wahrscheinlichkeitstheorie und Verw.* **4,** 316–339 (1966).
40. WOLFE, P., "The Simplex Method for Quadratic Programming," *Econometrica* **27,** 382–398 (1959).
41. ZOUTENDIJK, G., *Methods of Feasible Directions*, Elsevier, Amsterdam, 1960.
42. ———, "Nonlinear Programming: A Numerical Survey," *J. SIAM Control* **4,** 194–210 (1966).
43. ZSCHAU, E. V. W., "A Primal Decomposition Algorithm for Linear Programming," Graduate School of Business Working Paper No. 91, Stanford University, Stanford, Calif., January, 1967.