Finding Everett's Lagrange Multipliers
by Generalized Linear Programming,
Part II: Tactical Options†

Harvey J. Greenberg
Theodore C. Robbins

Computer Science/Operations Research Center
Southern Methodist University
Dallas, Texas 75222

Technical Report CP-70008

August, 1970
Revised June, 1972

†This research was supported by a National Science Foundation Seed Grant through the Office of Research Administration, Southern Methodist University (No. 88-48).
1. Introduction

In Part I of this three-part paper, we developed geometrical foundations for the GLM problem in response-space, employing the theory of conjugate functions. Analyzed within the context of a generalized semi-infinite linear programming model, the strategy of GLM was presented as an iterative search for the optimal multiplier vector (minimizing the Lagrangian).

Our attention now, in Part II, is turned to tactics for accelerating the GLM-LP procedure. Beginning with the case of a single constraint, we consider the "Noose Method" of Furman and Weinstein [20] within the previously described general framework. A counterexample to their initialization procedure is demonstrated, and an alternative procedure is presented.

The issue of gap resolution is considered next. While many problems can be "adequately solved" by GLM, there are problems for which the terminal solution is not sufficiently close to the optimum. In such cases, it is desirable to be able to resolve the gap. Several gap-resolution tactics are presented, and it is shown how the GLM-LP procedure can be continued following gap detection.

Pricing tactics are also considered, and it is shown that the theory permits "early" revision of the multiplier values. The practical significance of this tactic is underscored for the important case of separable programs, and GLM is related to the classical Charnes-Lemke methods [11] (λ or δ form).

In particular, we prove that the Dantzig-Wolfe method is uniformly better than either of the Charnes-Lemke forms.

2. The Noose Method

The GLM-LP strategy chooses the Lagrange multipliers iteratively as the optimal dual variables to a finite approximation of our semi-infinite linear program. Using the Revised Simplex Method, the basis inverse (of the primal) is updated by maximizing the Lagrangian constructed with the prescribed multiplier values, generating the new vector to enter the basis at each iteration. However, there are modifications to the Simplex Method intended to accelerate the algorithm.
We shall describe a particular variation suggested by Brooks and Geoffrion [6] (for finite $S$) and subsequently developed geometrically by Furman and Weinstein [20].

For convenience we shall assume $k = 0$ (i.e., no equalities). It is also convenient to think of our mathematical program as a resource allocation problem, although other interpretations are possible.

Let us begin with $m = 1$. Greenberg [28] developed a method which can be considered a special case of the Furman-Weinstein "Noose Method" (although the work was developed independently). It should serve to tie together the linear programming concepts and the conjugate function theoretical descriptions.

Greenberg's "reduced GLM-LP" algorithm is as follows:

(0) Choose $x^0$ and $x^1$ to satisfy

$$g(x^0) \leq \bar{b} < g(x^1)$$

and

$$f(x^0) = \text{Max} \{f(x)|x \in S \text{ and } g(x) \leq g(x^0)\}$$

$$f(x^1) = \text{Max} \{f(x)|x \in S \text{ and } g(x) \leq g(x^1)\}$$

Set $b_{\text{LOW}} = g(x^0)$, $f_{\text{LOW}} = f(x^0)$

$$b_{\text{HIGH}} = g(x^1), f_{\text{HIGH}} = f(x^1)$$

(If $b_{\text{LOW}} = \bar{b}$, then stop; otherwise, note that $b_{\text{LOW}} < \bar{b}$.)

(1) Choose $\lambda = \frac{(f_{\text{HIGH}} - f_{\text{LOW}})}{(b_{\text{HIGH}} - b_{\text{LOW}})}$ 

(Note: $\lambda > 0$). Solve for $L^*(\lambda)$ and $\Omega_L(\lambda)$. If the termination conditions are met, i.e., if $\Psi(\lambda)$ contains $\bar{b}$, then stop. Otherwise, update as follows:

Let $X^1 = \{x \in \Omega_L(\lambda) | g(x) < \bar{b}\}$

$X^2 = \{x \in \Omega_L(\lambda) | g(x) > \bar{b}\}$

(Note that $X^1 \cup X^2 \neq \emptyset$ and $X^1$ and $X^2$ are compact when $\bar{b} \notin \Psi(\lambda)$). If $X^1 \neq \emptyset$, let

$$b_{\text{LOW}} = \text{Max} \{g(x)|x \in X^1\}.$$ If $X^2 \neq \emptyset$, let

$$b_{\text{HIGH}} = \text{Min} \{g(x)|x \in X^2\}.$$ Return to (1).

Now let us relate this first to GLM-LP, then to the geometry of response space. If we initialize GLM-LP with a nondegenerate basis consisting of $w(x^0)$ and the slacks, then we must choose $x^0 \in S$ such that $g(x^0) < \bar{b}$. This will
yield \( \lambda^0 = 0 \), in which case \( x^1 \) is the maximum of \( f \) on \( S \).

A possible difficulty with step (0) is obtaining \( x^0 \) such that
\[
f(x^0) = f(g(x^0)), \text{ i.e., } (f(x^0), g(x^0)) \text{ is on the response surface in response space.}
\]
Eventually, the GLM-LP algorithm generates such an \( x^0 \). Moreover, if \( f \) and \( g \) are nondecreasing (coordinate-wise) and if \( S = \{ x \in \mathbb{R}^n | 0 \leq x \leq \bar{a} \} \), then we can let \( x^0 = 0 \), (where \( \bar{b} > g(x^0) \) in order to have a nontrivial problem).

We can now prove convergence of Greenberg's method by relating it to GLM-LP.

**Theorem 2.1:** If \( x^0 \) satisfies (0) in GLM-LP, where \( g(x^0) < \bar{b} \) and \( LP^* \)
is initialized with \( w(x^0) \) and \( m \) slacks, then the GLM-LP sequence, \( \{ \lambda^t \} \), is the same as Greenberg's sequence with
\[
f(x^1) = \max \{ f(x) | x \in S \}.
\]

**Proof:** Starting with \( x^0 \) in GLM-LP, \( x^1 \) is chosen to be the unconstrained maximum of \( f \) on \( S \) (since \( \lambda^0 = 0 \) by complementary slackness). Therefore, to obtain \( \lambda^1 \) GLM-LP solves
\[
\begin{align*}
\min v + \lambda^0 \\
v + \lambda g(x^0) & \geq f(x^0) \\
v + \lambda g(x^1) & \geq f(x^1)
\end{align*}
\]
Consider
\[
\lambda = (f(x^1) - f(x^0))/(g(x^1) - g(x^0))
\]
(Note: If \( g(x^1) \leq \bar{b} \), then \( x^1 \) solves \( \overline{F} \). Hence, to have a nontrivial problem we shall assume \( g(x^1) > \bar{b} \).)

Choose
\[
\bar{v} = f(x^0) - \lambda g(x^0)
\]
and note that
\[
\bar{v} = f(x^1) - \lambda g(x^1)
\]
Hence \((\bar{v}, \lambda)\) is dual feasible.
Now consider

\[ \bar{\omega}(x^0) = (g(x^1) - \bar{b})/(g(x^1) - g(x^0)) \]

and

\[ \bar{\omega}(x^1) = (\bar{b} - g(x^0))/(g(x^1) - g(x^0)) \]

Observe that \( w \geq 0, \bar{\omega}(x^0) + \bar{\omega}(x^1) = 1 \)

and

\[ \bar{\omega}(x^0)g(x^0) + \bar{\omega}(x^1)g(x^1) = \bar{b} \]

Therefore, \( \bar{\omega} \) is optimal in the primal and \((\bar{\nu}, \bar{\lambda})\) is optimal in the dual, so

\[ \lambda^1 = \bar{\lambda}. \]

More generally, at the \( t \)th iteration choose \( \lambda^t \) as in Greenberg's method, and suppose \( r \) and \( q \) are defined to satisfy

\[ b_{LOW} = g(x^r) \text{ and } b_{HIGH} = g(x^q) \]

If \( b_{HIGH} > \bar{b} > b_{LOW} \), then choose

\[ w_r = (b_{HIGH} - \bar{b})/(b_{HIGH} - b_{LOW}) \]

\[ w_q = (\bar{b} - b_{LOW})/(b_{HIGH} - b_{LOW}) \]

\[ w_j = 0 \text{ for } j \neq r, q \]

Note that

\[ w > 0, \sum_{j=0}^{t} w_j = 1 \]

and

\[ \sum_{j=0}^{t} w_j g(x^j) = \nu_r b_{LOW} + w_q b_{HIGH} = \bar{b} \]

Therefore, \( w \) is primal feasible. Finally, choose

\[ \nu = \text{Max} \{ f(x^j) - \lambda^t g(x^j) \} \]

\[ j < t \]

so \((\nu, \lambda^t)\) is dual feasible.

Since the points \( x^r \) and \( x^q \) were generated by GLM, \((f(x^r), g(x^r))\) and \((f(x^q), g(x^q))\) lie on the response surface. Then it can be easily shown that

\[ \nu = (f_{LOW} b_{HIGH} - f_{HIGH} b_{LOW})/(b_{HIGH} - b_{LOW}) \]

(i.e., \( r \) and \( q \) are maximizing indices). Now, from duality theory we can conclude that \((\nu, \lambda^t)\) solves the dual; hence our proof is complete.
Figure 2.1 illustrates the procedure in R-space. Starting at points \( P_0 \) and \( P_1 \) (corresponding to \( f(x^0), g(x^0) \) and \( f(x^1), g(x^1) \) respectively), the multiplier is chosen to be the slope of the line segment joining the two points. Then \( P_2 \) is generated by GLM, replacing \( P_0 \); i.e., \( w(x^0) \) leaves the basis and \( w(x^2) \) enters. As the procedure is repeated, we observe the zig-zag sequence of line segments through points on the response surface generated (in succession) by GLM. Also, note that the projection into the abscissa of the successive line segments is a sequence of intervals of decreasing length containing \( b \).

Consider the Furman-Weinstein "Noose Method" (NM) which is similar to the earlier Brooks-Geoffrion formulation. The Lagrange multipliers of NM are obtained at each iteration of the procedure as the dual variables of an LP in which a convex combination of extreme points of the response surface is maximized. The difference between the two formulations is in the way the inequality constraints are treated. In the NM, the relation between the corresponding convex combination of resource vectors and the requirements vector (i.e., constraint right-hand sides, \( b \)) is maintained with equality.

Since the multipliers are not restricted to nonnegative values in NM, it is assumed (implicitly) that in R-space, the hull of the envelope function is a nondecreasing function of the constraints. In that case, a solution to NM solves the original (inequality constrained) problem. In the previous discussion of Greenberg's reduced GLM-LP procedure for the case \( m=1 \) we noted that if \( x^1 \) is chosen to be the unconstrained maximum of \( f \) on \( S \) and if \( g(x^1) > b \), then GLM-LP will choose \( \lambda^t > 0 \) for all \( t > 0 \). This means we could be solving the equality problem, i.e., solving for \( F(b) \) now becomes equivalent to solving for \( z^e(b) \).

Generalizing the foregoing to where \( m > 1 \) is not a trivial matter.
Figure 2.1. Illustration of climbing the response curve in R-space.
However, when the second conjugate, \( z^c \), is nondecreasing, we note that any support for hypothesis \( z^* \) in PR space must also be a support for hypothesis \( F \) since the slopes of the support \( (\lambda) \) must be nonnegative. This is the Furman-Weinstein monotonicity assumption, under which we can consider an equality constraint formulation. In general, (without their assumption holding) this can yield negative multipliers, but as Brooks and Geoffrion pointed out, we can use any "column improvement" technique. The Noose Method solves LP in that way—with the inequalities replaced by equalities. Hence, we might want to let LP be stated as an equality problem, and thus not allow slack variables. In that case, there would be no iterations used to exchange basic columns in LP except those using GLM, i.e., by maximizing the Lagrangian. For a particular class of resource-allocation problems, the relative efficiency of inequality- and equality-constraint formulations are compared in Part III, in which computational results are reported.

Figure 2.2 illustrates the geometrical interpretation provided by Furman and Weinstein[20] and is an extension to two constraints of the one-constraint case described in Figure 2.1. In R-space, the response surface is climbed while keeping \( \bar{b} \) (the desired resource level) in the projection of the simplex defined by the set of generated points. At each iteration, one vertex of the simplex is exchanged for a new one generated by maximizing the Lagrangian whose multiplier values are the slopes of the contour passing through the vertices of the simplex. In the "resource plane" \( \pi \equiv (\bar{z}_1, \bar{z}_2) \), the projected simplex shrinks around \( \bar{b} \) until a solution is found (e.g., \( \bar{b} \) becomes a vertex) or until a vertex of the simplex is regenerated by GLM, in which case a gap is detected [27].
Projection into "resource plane"

Figure 2.2: Moving from simplex (1) to simplex (2).
Referring to the iteration illustrated in Figure 2.2, we start with vertices A, B and C on the response surface (convex hull of hypo z and of hypo F); these points correspond to solutions of previous GLM problems (i.e., from finding \( L^*(\lambda) \) and \( \Omega_L(\lambda) \) for previous values of \( \lambda \)). Hence, they have supports. Using the slopes defined by this triangle we have the next \( \lambda \) to use in GLM; that is, we place A, B and C on the same contour as \( f(x^i) - \lambda g(x^i) = \nu \) for \( i = A, B, C \). This need not have a solution with \( \lambda \geq 0 \), as will be discussed subsequently. (Note that these are merely the LP constraints holding with equality, i.e., for \( \lambda \) unrestricted.) The point D is then generated, replacing A; the new simplex is smaller, and the point in the plane through B, C and D at \( g(x) = \bar{b} \) is higher than the corresponding point on the plane through A, B and C. Repeating the procedure, we climb the response surface around \( \bar{b} \). As implied by the name "Noose Method", the sequence of generated simplices tends to "tighten" around the desired right-hand-side vector, \( \bar{b} \).

A proof of finite convergence to a simplex containing \( \bar{b} \) (for finite \( S \)) appears in Furman and Weinstein [20], but more must be said about initializing the Noose Method. Namely, we must be able to find a simplex with vertices on the response surface whose projection into the resource plane contains \( \bar{b} \). That is, we must find \( x^1, \ldots, x^{m+1} \) such that \[ \bar{b} = \sum_{i=1}^{m+1} \alpha_i g(x^i) \] for some \( \alpha > 0 : \sum_{i=1}^{m+1} \alpha_i = 1 \).

Furman and Weinstein presented the following procedure:

Let \( M \) be the largest real value the computer can store, so that \( M \) represents infinity. Then, let
\[\lambda^0 = (M, M, \ldots, M)\]
\[\lambda^1 = (-M, M, \ldots, M)\]
\[\lambda^2 = (M, -M, \ldots, M)\]
\[\vdots\]
\[\lambda^i = \lambda^0 - 2M e_i\]
\[\vdots\]
\[\lambda^m = (M, M, \ldots, -M)\]

where \(e_i\) is a vector of zeros except for the \(i\)th component, which is 1.

However, this initialization procedure does not always work, as demonstrated by the following counter-example:

\[P: \text{max } x_1^2 + x_2^2.\]
\[x_1 + 2x_2 \leq b_1\]
\[2x_1 + x_2 \leq b_2\]
\[x_1, x_2 \in \{0, 1, \ldots, 10\}\]

The GLM problem is

\[\text{GLM: Max } x_1^2 - x_1(\lambda_1 + 2\lambda_2) + x_2^2 - x_2(2\lambda_1 + \lambda_2):\]
\[x_1, x_2 \in \{0, 1, \ldots, 10\}\]

The initial points generated are:

<table>
<thead>
<tr>
<th>(\lambda_1)</th>
<th>(\lambda_2)</th>
<th>(x^*)</th>
<th>(g_1(x^*))</th>
<th>(g_2(x^*))</th>
<th>(f(x^*))</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1) M</td>
<td>M</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>(2) -M</td>
<td>M</td>
<td>0</td>
<td>10</td>
<td>10</td>
<td>100</td>
</tr>
<tr>
<td>(3) M</td>
<td>-M</td>
<td>10</td>
<td>10</td>
<td>20</td>
<td>100</td>
</tr>
</tbody>
</table>
In Figure 2.3, the value of $F(b)$ (which equals $z^*(b)$) is indicated for each $b \in \pi$ (the resource plane). Clearly, the monotonicity assumption is satisfied since the convex hulls of $z^*(b)$ and $F(b)$ coincide for $b \in \pi$. However, the projection of the initial simplex onto the resource plane does not contain all of $\pi$; e.g., $b = (18, 18)$, which is generated by $x = (6, 6)$, lies outside the triangle formed by the initial points.

If we set $\lambda = (0, 0)$, obtaining $(30, 30)$ in the resource plane, then all points of $\pi$ lie within the convex hull of the four points, as illustrated. However, although $b = (5, 20)$ is not in $\pi$, it could be the specified requirements vector (i.e., $\bar{b} \in B$), and in general, the constraints need not have an equality solution. Hence a more general procedure is required.

Consider the initialization obtained by specifying

$$
\lambda^* = 0
$$

$$
\lambda_i = \bar{x}_i = (0, ..., 0, M, 0, ..., 0), \quad i = 1, ..., m
$$

$$
\lambda_{m+1} = M_1 = (M, ..., M).
$$

Under certain conditions for $m=2$, we can obtain the following four points in the resource plane:

$$
g_1^* \leq \bar{b}_1 \quad g_2^* \leq \bar{b}_2 \quad g_1^* \geq \bar{b}_1 \quad g_2^* \geq \bar{b}_2
$$

Regarding the aforementioned conditions, we note that:

1. For $\lambda = \lambda^*$, we are not guaranteed that $g_i^* \geq \bar{b}_i$, $i = 1, 2$. However, if $g_i^* < \bar{b}_i$ for some $i$, then we can apply the Brooks-Geoffrion weakening of GLM termination. That is, if $x^*$ maximizes $f(x) - \lambda g(x)$ and $\lambda_i = 0$, then $x^*$ is optimal for all $g_i^* \geq \bar{b}_i(x^*)$. Hence for $\lambda = 0$ we can consider an "equivalent" point $\hat{b}^* \geq b$. Referring to Figure 2.4 in which the resource plane is shown partitioned into four regions about $\hat{b}$, $\hat{b}^*$ lies in region I.
Figure 2.3. Illustration of unsuccessful initialization by the Furman-Weinstein procedure.
Figure 2.4. Illustration of initialization for $m = 2$. 
2. Similarly, for $\lambda = \lambda^1$, it is possible, in general, to have $g_2 < \bar{g}_2$ with $\lambda = \lambda^1$ (or $g_1 < \bar{g}_1$ with $\lambda = \lambda^2$). Consider, for example, that both $g^1$ and $g^2$ lie in regions II and III above $L_{03}$ (the line segment joining $g^0$ and $g^3$ in Figure 24). That is,

$$g_1 < \bar{b}_1$$

and $\bar{b}$ is not contained in any simplex of the generated points.

Again, however, we can employ Brooks Geoffrion weakening to obtain an equivalent $\bar{b}_2 > \bar{b}_1$, in which case $\bar{b}_2$ would lie in region IV.

3. If $g^3 < \bar{b}$, then $g^3$ lies in region III, as illustrated in Figure 24.

However, with $\lambda = \lambda^3 = \frac{M_1}{2}$ we are not guaranteed that $g^3 < \bar{b}$; e.g., linear dependence between constraints could lead to arbitrarily large values of some constraint functions. If $g^3 < \bar{b}$, however, then we have a point in each of the four "quadrants" relative to the "origin" at $\bar{b}$, and thus an initial simplex containing $\bar{b}$.

Now in the general case ($m \geq 2$), we see that for $M$ sufficiently large,

$$g_i < \bar{b}_i, \quad i = 1, \ldots, m$$

Further, if

$$g_j < \bar{b}_j$$

for some $j \neq i$

then we can use the Brooks-Geoffrion weakening to let

$$\hat{b}_j = \bar{b}_j$$

Thus, if we set

$$\hat{b}_j = \max \{ g_j, \bar{b}_j \} \quad \text{for } j \neq i$$

and

$$\hat{b}_i = \max \{ g_i, \bar{b}_i \}$$

Then,

$$\bar{b}_j = \max \{ g_j, \bar{b}_j \} \quad \text{for } j \neq i$$

and

$$\bar{b}_i = \max \{ g_i, \bar{b}_i \}$$
for \( i = 1, \ldots, m \), and if we set
\[
\hat{b}^o = g^o = b
\]
\[
\hat{b}^{m+1} = g^{m+1} \leq b
\]
then, for the general case we have \( m + 2 \) initial \( b \)-vectors satisfying
\[
\begin{align*}
\hat{b}^o \geq b, \\
\hat{b}^i_j > b_j, & \quad j \neq i \\
\hat{b}^i \leq b^i, \\
\hat{b}^{m+1} \leq b
\end{align*}
\]
\( i = 1, \ldots, m \)

Unfortunately, a counterexample exists for this procedure for \( m=3 \). However, note that if each of the functions \( f(x), g_i(x), \) \( i=1, \ldots, m \) is strictly monotonic (which is where the Noose Method has been applied to date), then the above procedure generates \( m+1 \) points in \( \mathbb{R} \)-space given by
\[
\begin{align*}
(z,b)_0 &= (f^{\text{MAX}}, \bar{b}_1 + \gamma_1, \bar{b}_2 + \gamma_2, \bar{b}_3 + \gamma_3, \ldots, \bar{b}_m + \gamma_m) \\
(z,b)_1 &= (0, 0, \bar{b}_2, \bar{b}_3, \ldots, \bar{b}_m) \\
(z,b)_2 &= (0, \bar{b}_1, 0, \bar{b}_3, \ldots, \bar{b}_m) \\
& \vdots \\
(z,b)_m &= (0, \bar{b}_1, \bar{b}_2, \bar{b}_3, \ldots, 0)
\end{align*}
\]
where
\[
\gamma = g(x^*) - b > 0
\]
(since if \( g_i(x^*) \leq \bar{b}_i \), the \( i \)th constraint can be dropped from the problem).

We shall prove that these points are generated by our procedure (assuming monotonicity), so that we can begin our search with these known \( m+1 \) points without explicitly solving any initial problems. Of course, we must also show that these \( m+1 \) points provide a feasible initialization.

First, note that the point \((z,b)_0\) is generated by \( \lambda = 0 \) because of
The functions $f$, $g_1$, and $g_2$ are tabulated in Table 2.1.

Table 2.1. Data for the Modified Noose Method Example

<table>
<thead>
<tr>
<th>$x_1$</th>
<th>$x_2$</th>
<th>$x_3$</th>
<th>$f(x)$</th>
<th>$g_1(x)$</th>
<th>$g_2(x)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>6</td>
<td>3</td>
<td>2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>7</td>
<td>7</td>
<td>6</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>9</td>
<td>7</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>13</td>
<td>12</td>
<td>3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>14</td>
<td>16</td>
<td>7</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>2</td>
<td>0</td>
<td>9</td>
<td>14</td>
<td>9</td>
</tr>
<tr>
<td>1</td>
<td>15</td>
<td>17</td>
<td>11</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>16</td>
<td>21</td>
<td>15</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>9</td>
<td>7</td>
</tr>
<tr>
<td>1</td>
<td>7</td>
<td>12</td>
<td>9</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>8</td>
<td>16</td>
<td>13</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>8</td>
<td>18</td>
<td>8</td>
</tr>
<tr>
<td>1</td>
<td>14</td>
<td>21</td>
<td>10</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>15</td>
<td>25</td>
<td>14</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>0</td>
<td>10</td>
<td>23</td>
<td>16</td>
</tr>
<tr>
<td>1</td>
<td>16</td>
<td>26</td>
<td>18</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>17</td>
<td>30</td>
<td>22</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>0</td>
<td>11</td>
<td>10</td>
<td>11</td>
</tr>
<tr>
<td>1</td>
<td>17</td>
<td>13</td>
<td>13</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>18</td>
<td>17</td>
<td>17</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>0</td>
<td>18</td>
<td>19</td>
<td>12</td>
</tr>
<tr>
<td>1</td>
<td>24</td>
<td>22</td>
<td>14</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>25</td>
<td>26</td>
<td>18</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>0</td>
<td>20</td>
<td>24</td>
<td>20</td>
</tr>
<tr>
<td>1</td>
<td>26</td>
<td>27</td>
<td>22</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>27</td>
<td>31</td>
<td>26</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Figure 2.5 illustrates the points in PR space projected into the resource plane; each of the 27 points corresponds to a distinct triple, $(x_1, x_2, x_3)$, and is plotted as the value of $f(x)$ at coordinates $(g_1(x), g_2(x))$. The progression of the Modified Noose Method (MNM) is indicated by the simplices defined by the numbered squares. The initial simplex (points #1, #2 and #3) consists of the "artificial points" obtained by Brooks Geoffrion weakening, (0,7) and (2,0), and the unconstrained maximum at (31,26). Subsequent points generated by MNM are labelled sequentially.

Since the Noose Method demands the equality
\[ \sum_{i=0}^{t} w_i g(x_i) = b \]
(i.e., $b$ is maintained in a simplex), it is possible to have negative multipliers generated. In our example, a gap is detected when point #5 is regenerated as point #7. The support defining the gap, in this case, has a negative slope (i.e., a negative multiplier) as can be deduced from Figure 2.5. Our concern with negative multipliers is principally because the objective upper bound given by the conjugate relation:
\[ F(b) \leq L^*(\lambda) + \lambda b \]
(inherently obtained with GLM) is valid only for nonnegative multipliers.

When the second conjugate of the envelope function is nondecreasing, negative multipliers are not obtained. In that case, the above mentioned relation always yields a valid upper bound on the optimum objective value. However, as will be demonstrated shortly, monotonicity of $z^{cc}$ is not implied by the monotonicity of the objective and constraint functions for $m > 1$.

The source of the negative multiplier in our example is the use of "artificial points" (i.e., points in $B \sim \pi$) in initializing the Modified Noose Method. In the example (Figure 2.5), point #1 is $(0, b_2)$ and is not
Figure 2.5. Resource plane illustration of the Modified Noose Method simplex progression
in $\pi$. However, we can establish the following:

**Theorem 2.3:** Assume $z^{*\overline{\mathbf{c}}}c$ is nondecreasing on the convex hull of $\pi$. Then, if NM has no artificial points at termination with $\overline{\lambda}$, then $\overline{\lambda} \geq 0$. In this case, $\overline{\lambda}$ solves $\text{DP}^\circ$.

**Proof:** With no artificial points, our final solution is a support for hypo $z^{*\overline{cc}}$. Since $z^{*\overline{cc}}$ is assumed to be nondecreasing, $\overline{\lambda} \geq 0$. Moreover, we have at termination: $\{x_j^r\}_{j=0}^r \subseteq \Omega_L(\overline{\lambda})$, $r \leq m$, and

$$\overline{b} = \sum_{j=0}^r w_j g(x_j), \quad w \geq 0 \text{ and } \sum_{j=0}^r w_j = 1.$$ 

Since $\lambda \geq 0$, $\overline{b} \epsilon \psi(\overline{\lambda})$; thus $\overline{b} \epsilon \psi(\overline{\lambda}) \cup B_G$ and $\overline{\lambda}$ minimizes $\{L^*(\lambda) + \lambda \overline{b}\}$ on $E^m_+$.

If (in our example) the negative multiplier, $\overline{\lambda}_2$, were made zero, then in the resource plane, the point $\overline{b}$ could be translated parallel to the axis of the corresponding constraint function; i.e., $g_2(x)$ could take any value in $[0, \overline{b}_2]$, by Brooks- Geoffrion weakening. In this way, $\overline{b}$ could be expressed as a convex combination of extreme points (e.g., points #1 and #7). Then a gap would be detected with $\lambda = (\overline{\lambda}_1, 0) > 0$ instead of $\overline{\lambda}$ (obtained with NM), for which $\overline{\lambda}_2 < 0$), and a valid upper bound on the objective would be obtained as $L^*(\lambda) + \lambda \overline{b}$.

As we have seen, the use of artificial points can cause difficulties, resolvable by recalling their significance. Our concern is when negative multipliers are obtained when the NM terminates with a gap indication. The problem is overcome by forcing the negative multipliers to assume zero values; operationally (since the multipliers are the dual variables of an LP), the offending multipliers are made zero by introducing the corresponding slack activities into the basis (exchanging each slack for one of the points defining the gap). Executing NM in this manner, a valid GLM bound on the objective is obtained when the terminal gap is detected.
Operationally, the Noose Method and the Brooks-Geoffrion procedure are equivalent if slack variables are introduced in NM in order to maintain nonnegative multipliers, or slacks are not introduced in the Brooks-Geoffrion method. Our computational experiences with the algorithms have shown that when negative multipliers are allowed (except at termination), several gap detections (with negative multipliers) may occur prior to terminating with nonnegative multipliers and the desired least upper bound on $f^*$. Implicit in the above comment that slacks need not be introduced in the Brooks-Geoffrion method is the fact that it is not necessary that the Lagrangian always be maximized. As observed by Brooks and Geoffrion, it is clear from the theory of the Simplex method that it is sufficient that an entering column be "improving" (i.e., that the value of the Lagrangian be increased at each iteration). From the viewpoint of efficiency, the total effort required to solve a particular problem may be less for some strategy in which Lagrangian maximizations are not always carried to completion. We shall return to this point later.

Before presenting some computational results of GLM-LP and MNN, one more point deserves elaboration, namely the monotonicity assumption. For $m = 1$, $z^{cc}$ is clearly nondecreasing when $f$ and $g$ are (coordinate wise) nondecreasing. However, for $m>1$, this need not be so. To illustrate, consider the following:

**Example:** $S = \{x \in \mathbb{E}^3 | x_j = 0 \text{ or } 1 \text{ for each } j = 1,2,3\}$

<table>
<thead>
<tr>
<th>$f(x)$</th>
<th>$g_1(x)$</th>
<th>$g_2(x)$</th>
<th>$x_1$</th>
<th>$x_2$</th>
<th>$x_3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>2</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>4</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
<td>3</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>25</td>
<td>5</td>
<td>5</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

Observe that $f$, $g_1$ and $g_2$ are nondecreasing. Figure 2.6 shows the resource plane with values of $z^*$ given. Note that $z^{cc}$ is not nondecreasing.
Figure 2.6. R Space for Example Showing $z^{*cc}$ not nondecreasing
3. Gap Resolution

Let us suppose GLM-LP has terminated with a gap, and our final basis consists of certain slack variables plus \( w(x^1), \ldots, w(x^r) \), where \( 2 \leq r \leq m+1 \).

(Note: In order to have a gap, it is necessary that \( r \geq 2 \).) There are several ways to attempt to resolve the gap, \(^{[27]}\) or at least improve the bounds on \( f^* \). We shall describe certain ways in which the linear programming approach can be extended.

First we could replace one or more constraints with a new constraint given by:

\[
G_i(g_i(x)) \leq G_i(0)
\]

where \( G_i \) is an increasing function. This does not change \( P^o \), but it does change the \( R \)-space. The primal LP then becomes

Max \( \sum_{x \in S_w} w(x)f(x) \):

(1) \( \sum_{x \in S_w} w(x) = 1 \), \( |S_w| \leq m+1 \)

(2) \( \sum_{x \in S_w} w(x) G(g(x)) \leq G(0) \)

(3) \( w(x) \geq 0 \) for all \( x \in S \)

Our terminal basis need no longer be optimal or feasible. However, if we choose each \( G_i \) \((1 \leq i \leq m)\) to be concave (as well as increasing), then our terminal basis is feasible, but to no avail. We shall prove that the Lagrangian is at least as strong as any concave \( G \) (in the sense described by Greenberg\(^{[29]}\)).

Let us suppose that we solve

Max \( f(x) - \lambda G(g(x)) \)

\( x \in S \)

where \( \lambda \geq 0 \). If \( G \) is convex on \( E^m \), then it is subdifferentiable everywhere\(^{[42]}\).

Let \( \delta(-\lambda G) \) denote its subdifferential, where

\[
\delta(-\lambda G(0)) = \{ \lambda^* | \lambda G(0) = \text{Sup} \{ \lambda^* b + \lambda G(b) | b \in E^m \} \} \neq \emptyset
\]
Then, by choosing any Lagrange multiplier $\lambda^*$, such that $-\lambda^* e \delta(-\lambda G(0))$, it follows that $\lambda^* \geq 0$ and $\lambda^* b \geq \lambda G(b) - \lambda G(0)$ for all $b \in B$.

Greenberg's Theorem 2.4 of Reference 29 then implies that the Lagrangian is stronger, so that no lower upper bound on $f^*$ is possible with a concave $G$.

There are certain choices of $G$ that are "stronger" than the Lagrangian in the sense that they provide a lower upper bound on $f^*$ and may resolve our gap entirely. For example, one possible choice is:

$$G_i(g_i) = e^{g_i(x)}$$

But this choice of $G$ is convex, resulting in the terminal basis becoming infeasible. However if we wish to continue with GLM-LP, we can introduce the activity vector corresponding to the best feasible solution currently available and the set of $m$ slack activities. We can then continue the linear programming procedure, finding new multipliers which solve:

$$\min \max \{ f(x) - \lambda^* G(g(x)) + \lambda G(0) \}$$

$$\lambda^* \geq 0 \quad x \in S$$

The reason we do not start out with this transformation is because the Lagrangian has certain desirable properties that facilitate its maximization (for fixed $\lambda$). In particular, if $f$ and $g$ are sum separable, and if $S$ is a product $\prod S_j$, where $\{ S_j \}_{j=1}^n \subseteq \mathbb{R}^1$, then the maximization of the Lagrangian decomposes into $n$ univariate problems.

A second method of gap resolution is to delete certain members of $S$ known to be infeasible in $P^*$. In particular, suppose we order the values of $\{ \lambda^* g(x^k) \}_{k=1}^r$ so that

$$\lambda^* g(x^1) \leq \ldots \leq \lambda^* g(x^r)$$

It is easily shown that we have

$$\lambda^* g(x^1) \leq 0 \leq \lambda^* g(x^r)$$
Let $k^*$ be the largest index such that 
$$\lambda^* g(x^{k^*}) \leq 0$$

Clearly, all feasible solutions in $P^*$ satisfy 
$$\lambda^* g(x) \leq 0$$

Therefore, we can add the following constraints to $LP^*$:

$$w(x) = 0 \text{ for } x \in S \text{ such that } \lambda^* g(x) > 0$$

In order to not increase the size of the basis, we alternatively proceed as follows. Define 
$$\sigma(\lambda) = \{x \in S | \lambda g(x) \leq 0\}$$

For $\lambda = \lambda^*$, we have $\sigma(\lambda^*) \subseteq \{x \in S | x^{k^*+1}, \ldots, x^{r} \in T^*\}$. If $T^* \neq \emptyset$ we can modify $LP^*$ as:

$$LP^{\lambda^*}: \text{Max } \sum_{x \in S} w(x)f(x) - M \sum_{x \notin T^*} w(x):$$

1. $\sum_{x \in S} w(x) = 1$, $|S_\lambda| \leq m + 1$
2. $\sum_{x \in S} w(x)g(x) \leq 0$
3. $w(x) \geq 0$ for all $x \in S$

The dual is given by:

$$DP: \text{Min } v:$$

1. $\lambda \geq 0$
2. $v + \lambda g(x) \geq f(x)$ for $x \in T^*$
3. $v + \lambda g(x) \geq f(x) - M$ for $x \notin T^*$

We choose $M$ very large; e.g., if the range of $f$ is in $[f^{\text{MIN}}, f^{\text{MAX}}]$, then choose $M > \text{Max } \{|f^{\text{MIN}}|, f^{\text{MAX}}\}$.

Note:
$$v^* \leq L^*(\lambda^*)$$

and the primal basis may no longer be optimal.

To generate our next column, we thus solve

$$\text{Max } L(x, \lambda^*): x \in S \text{ and } \lambda g(x) \leq 0$$
The multipliers are subsequently changed, and we continue the procedure. Note that instead of solving an "unconstrained" problem (i.e., finding $L^*(\lambda)$), we solve a one-constraint problem. This is of course more difficult, and we may not proceed for very many additional iterations. If $f$ and $g$ are sum separable and if $S$ is a product $\prod_{j=1}^n S_j$, where $\{S_j\}_{j=1}^n \subseteq \mathbb{R}^l$, then dynamic programming may be applied (at least theoretically).

The above procedure is related to, but different from, the surrogate model[31] given by:

$$F(\lambda) = \max \{ f(x) | x \in S \text{ and } \lambda g(x) \leq 0 \}$$

In particular, $v^*(\lambda) \leq F(\lambda)$, with equality if and only if a maximal $x^*$ satisfies the orthogonality condition: $\lambda^* g(x^*) = 0$. (The first author is presently studying the surrogate model in relation to generalized LP and searching the multipliers.)

When $P^0$ is an integer program (not necessarily linear), then a method of cuts can be applied with an increase in the size of the basis by only one row. That is, suppose $f$ and $g$ are integer-valued for each $x \in S$. Suppose GLM-LP has terminated with a gap at $\lambda^*=\lambda^*$. This implies

$$f^* \leq L^*(\lambda^*)$$

If this bound is not integer valued, then we require

$$f^* \leq [L^*(\lambda^*)]$$

where $[z]$ denotes the largest integer not exceeding $z$. If $L^*(\lambda^*)$ is integer valued, then we require

$$f^* \leq L^*(\lambda^*) - 1$$

In either case we can introduce an upper bound on $f$ as follows:
\[ \text{LP}^C: \text{Max} \sum_{x \in S_w} w(x)f(x): \]

(1) \[ \sum_{x \in S_w} w(x) = 1 \text{ and } |s_w| \leq m+2 \]

(2) \[ \sum_{x \in S_w} w(x)g(x) \leq 0 \]

(3) \[ w(x) \geq 0 \text{ for all } x \in S \]

(4) \[ \sum_{x \in S_w} w(x)f(x) \leq c \]

The value of \( c \) can be any upper bound on \( f^* \) strictly less than the GLM bound. In particular, we have suggested using the greatest integer strictly less than \( L^*(\lambda^*) \).

The current basis with \( \{x^k\} \subset S_w \text{ for } 2 \leq r \leq m+1 \) satisfies (by construction):

\[ f(x^k) - \lambda^* g(x^k) = L^*(\lambda^*) \text{ for all } k = 1, \ldots, r \]

Upon multiplying by \( w(x^k) \) and summing, we obtain

\[ \sum_{k=1}^r w(x^k)f(x^k) = L^*(\lambda^*) \]

(since \( \lambda^* \sum w(x^k)g(x^k) = 0 \) by duality theory). Therefore, the current basis becomes infeasible in the new problem, \( \text{LP}^C \). By augmenting a feasible solution in \( P^o \) if necessary, we can change the values of \( \{w(x^k)\} \) to obtain a feasible basis for \( \text{LP}^C \) (possibly bringing slacks into the basis).

**Example:** Max \( x_1 + x_2: 2x_1 + 4x_2 = 5 \leq 0 \)

\( x_1, x_2 \in \{0,1,2\} \)

The minimizing multiplier is \( \lambda^* = 1/4 \) and our final basis is:

\[ B = \begin{bmatrix} 1 & 1 \\ g(2,0) & g(2,1) \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ -1 & 3 \end{bmatrix} \]

The two basic variables \( w_1, w_2 \) correspond to \( x^1 = (2,0) \) and \( x^2 = (2,1) \). (We could also have \( w(2,2) \) in the basis instead of \( w(2,1) \).) Further, we have the inherent upper bound from GLM that

\[ f^* \leq L^*(\lambda^*) = 2.25 \]
We thus incorporate the upper bound $f^* \leq 2$ by adding the inequality

$$\sum_{x} w(x)f(x) \leq 2$$

Since $f(x^1) = 2$ and $f(x^2) = 3$, we can use the degenerate basis given by

$$\hat{B} = \begin{bmatrix} 1 & 1 & 0 \\ -1 & 3 & 1 \\ 2 & 3 & 0 \end{bmatrix}$$

where column 3 corresponds to our original slack variable. Now consider the problem:

$$\text{LP}^2: \text{Max } \sum_{j=1}^{3} w_j f(x^j):$$

\begin{align*}
(1) \sum_{j=1}^{3} w_j &= 1 \\
(2) \sum_{j=1}^{3} v_j g(x^j) &\leq 0 \\
(3) w &\geq 0 \\
(4) \sum_{j=1}^{3} w_j f(x^j) &\leq 2
\end{align*}

Our tableau is given by:

<table>
<thead>
<tr>
<th></th>
<th>$w_1$</th>
<th>$w_2$</th>
<th>$s_1$</th>
<th>$s_2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$2w_1$</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>$3w_2$</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>$0s_1$</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>$2$</td>
<td>0</td>
<td>1</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Letting $\lambda_0$ denote the multiplier associated with (4), this yields new multipliers: $\lambda = 0$, $\lambda_0 = 1$ and $v = 2$. To see if a better column exists we solve

$$\text{Max}\{(1-\lambda_0)f(x)-\lambda g(x)-v\}$$

$x \in S$
for $\lambda_0=1$, $\lambda=0$ and $v=2$. Since the above maximum is negative (viz., $-2$), the basis is optimal. Further, since our solution is integer-valued, we deduce that $x^*=(2,0)$ solves $P^*$.

Thus, if upper bounds on $f^*$ are readily available, such as in the integer case, then gaps can be resolved in a finite number of iterations while keeping the size of the basis at $m+2$.

As a generalization of the above procedure, we can augment any redundant constraint to $P^*$ as:

$$a(x) \leq a_0$$

and thus augment to $LP^*$:

$$\sum_{x \in S_w} w(x)a(x) \leq a_0$$

with $|S_w| \leq m+2$ replacing $|S_w| \leq m+1$ in (1). This is a "cut" if the optimal solution to $LP^*$ is no longer feasible. We note that when each cut uses a different function $a(\cdot)$, then the size of the basis may become too large.

In summary, the method of cuts can certainly be used in a natural way without disturbing the GLM-LP procedure. Details of such procedures and the relative computational effectiveness of gap resolution methods are left for future research.

Also note that gaps may be resolve by certain branch and bound techniques. We may, for example, partition $S$ as $S=TVQ$, where $Q=S-T$. For example,\[27\] we may use

$$T = \{x \in S | \sum_{j=1}^{n} x_j \leq c\}$$
The GLM-LP procedure can also deal with such methods in a natural way. When solving the problem associated with $T$, we simply add the constraint: $w(x) = 0$ for $x \notin T$. In order not to increase the size of the basis, it is more convenient to simply drop the offensive columns and generate a new policy (if necessary) which is in $T$.

The multiplier values will change, and we proceed until we solve:

$$\text{LP}^T: \text{Max } \sum_{x \in T_w} w(x)f(x):$$

1. $\sum_{x \in T_w} w(x) = 1, \quad |T_w| \leq m+1 \quad \text{(} T_w = \{x \in T | w(x) > 0\}\text{)}$
2. $\sum_{x \in T_w} w(x)g(x) \leq 0$
3. $w(x) \geq 0$ for all $x \in T$

Thus, various methods of gap resolution can be applied while continuing the GLM-LP procedure. In theory, we can solve any mathematical program satisfying A1-A3, but whether or not this approach is effective remains an open question. In the authors’ experience, for separable problems arising from resource allocation we can obtain optimal solutions for resource limits "near" the prescribed limits rather quickly even when $P$ is not a convex program.

4. Separable Programs

There are certain advantages that accrue when $P^o$ is a separable program of the following form:

$$P^o: \text{Max } \sum_{j=1}^{n} f_j(x_j); \sum_{j=1}^{n} g_j(x_j) \leq 0, \quad x \in S$$

where $S = \prod_{j=1}^{n} S_j$, and each $S_j$ is typically an interval or a finite set.
When, in addition, $P^o$ is a convex program, then it should be noted that GLM-LP is an alternative to the Charnes-Lemke formulation. An advantage of the GLM-LP algorithm is that break points are generated by the algorithm rather than specified a'priori. Thus, the grid generally becomes finer around the solution and one need not worry about choosing a grid in advance and then refining it to achieve greater accuracy. This fact renders the "classical" separable programming method obsolete. Not only do we avoid a'priori specification of grid points, but in fact we avoid redundant dual constraints (cf. Theorem 3.3 in Part I) which would slow convergence.

Our point is that GLM-LP viewed properly is uniformly better than the "$\delta$-form" or "$\lambda$-form" generally used for separable, convex programs. Now let us see how to exploit separability in two ways.

The first advantage of separability is the well-known result that the Lagrangian maximization decomposes into $n$ univariate problems, called "cell problems". The jth cell problem is defined by:

$$C_j: \text{Max } \{f_j(x_j) - \lambda g_j(x_j) | x_j, \lambda \}$$

Let us now consider exploiting separability further, in a way that might be useful when $n$ is large.

Employing GLM-LP as originally described, we would solve all $n$ of the cell problems before revising our choice of $\lambda$. If $n$ is large, this might be a relatively inefficient procedure. That is, in terms of the computational effort required to solve the original problem, it may be advantageous not to solve all the cell problems at a particular iteration before revising $\lambda$. As noted previously, it is sufficient that the Lagrangian be increased, obtaining an "improving column" for LP$^o$.

Turning now to a mathematical programming formulation of the separable program, a dual is given by:
DP: $\min v$:

(1) $\lambda \geq 0$

(2) $v + \lambda \sum_{j=1}^{n} g_j(x_j) \geq \sum_{j=1}^{n} f_j(x_j)$ for all $x \in S$

Let us introduce $n$ dual variables $\{\alpha_j\}_{j=1}^{n}$ in lieu of $v$, where each $\alpha_j$ measures the cell value in $C_j$. The "separable dual" is thus defined to be:

CDP: $\min \sum_{j=1}^{n} \alpha_j$:

(1) $\lambda \geq 0$

(2) $\alpha_j + \lambda g_j(x_j) \geq f_j(x_j)$ for all $x \in S_j$ and all $j=1,\ldots,n$

The following theorem is easily verified:

Theorem: $(v^*, \lambda^*)$ solves DP if and only if $(\alpha^*, \lambda^*)$ solves CDP, where

$v^* = \sum_{j=1}^{n} \alpha_j$. Moreover,

$\alpha_j^* = \max \{f_j(x_j) - \lambda^* g_j(x_j) | x_j \in S_j\}$

Now let us look at the new primal for CDP:

CLP: $\max \sum_{j=1}^{n} \sum_{x_j \in S_j \in w_j} w_j(x_j)f(x_j)$:

(1) $w_j(x_j) \geq 0$ for all $x_j \in S_j$ ($1 \leq j \leq n$)

(2) $S_{w_j} \equiv \{x_j \in S_j | w_j(x_j) > 0\}$, $|s_{w_j}| \leq \lambda + 1$ ($1 \leq j \leq n$)

(3) $\sum_{x_j \in S_j \in w_j} w_j(x_j) = 1$ ($1 \leq j \leq n$)

(4) $\sum_{j=1}^{n} \sum_{x_j \in S_j \in w_j} w_j(x_j) g_j(x_j) \leq 0$

In words, the $w(\cdot)$ in LP is expressible as a sum of univariate functions.

To obtain an entering column we thus solve a sequence of cell problems, $\{C_j\}$. Define the value of the $j$th cell to be:
\[ h_j(x_j; \lambda) \equiv \{ f_j(x_j) - \lambda g_j(x_j) \mid x_j \in S_j \} \]

and the optimal value of the jth cell problem to be

\[ h^*_j(\lambda) = \max_{x_j \in S_j} h_j(x_j; \lambda) \]

We could solve for \( j = 1, \ldots, k \), where \( k \) is the first occurrence of \( \alpha_k < h^*_k(\lambda) \).

(\( \alpha \geq h^*(\lambda) \) signals termination.) Alternatively, choose \( k \) such that

\[ \frac{\sum_{j=1}^{k} \alpha_j}{\sum_{j=1}^{k} h^*_j(\lambda)} < 1. \]

Our goal at each iteration is an improving column. As previously noted, it is not necessary to actually maximize \( h_j(x_j; \lambda) \) in the jth cell problem; if the cell value exceeds \( \alpha_j \) for any \( x_j \in S_j \), an improving column for CLP has been determined. (In practice we might demand some minimal increase, assuring convergence.)

Since it is not necessary to solve all cell problems (or any one cell problem, completely) before revising the multipliers, it appears reasonable to assume that the total amount of computation required to solve a problem generally can be reduced if the multipliers are revised more frequently than when all cell problems are solved at every iteration. However, the order in which cells should be considered (at any iteration), how much the Lagrangian value should be increased before the multipliers are revised, and similar questions are left for future research.

The trade-off to be considered is the number of iterations versus the computation-time per iteration. When premature revision of the multipliers is employed, then one may only obtain arithmetic convergence. (Even this is an open question since \( L^* \) is generally not strongly convex.)

We have described an option of not maximizing the Lagrangian for separable programs because we think there is an analytical way to approach the decision
process: whether to stop and revise the multiplier or to continue the 
Lagrangian maximization. The principle applies to the general problem, but 
some \( \epsilon \)-improvement would be required to avoid the possibility of dual con-
vergence to a non-optimum point. Also, if a parametric study on resource levels 
is desired, then one would not want to exercise the option.

In summary, the theory permits tactics to reduce the time per iteration by 
not maximizing the Lagrangian. Whether or not it is desirable to exercise 
this option is left for further research.

Several points should probably be reiterated before quitting this 
topic, however. It is important to note, for instance, that the upper 
bound on \( f^* \) (i.e., the dual objective) is not obtained if all the cell 
problems are not maximized. The value of the primal objective function 
does converge to \( \text{Sup } f^* \), in either case, though, and the change in objective 
(as a function of computational effort) is suggested as a measure that 
might be considered in developing a strategy for terminating Lagrangian 
maximizations at improved (but suboptimal) solutions.

It is pointed out, also, that although (3) of CLP represents \( n-1 \) 
new constraints that did not appear in \( \text{LP}^0 \), this fact is of no real 
consequence since the coefficients are either 1 or 0. That is, the 
special structure of the constraints (3) permits an implicit treatment 
as in generalized upper bounding methods. In fact, the dual programs 
presented are not unique, and other formulations might illustrate other 
points more clearly. However, the value of alternative formulations is 
primarily conceptual; in practice, we would implement \( \text{LP}^0 \) and maintain 
a basis of \( m+k+1 \) columns.
5. A Relaxation Procedure for Large m

When the number of constraints is very large, we might expect only a "small" percentage of them to actually be binding at a solution. We could then proceed as described in Geoffrion\textsuperscript{[21]} (see also Lasdon\textsuperscript{[36]}).

Starting with $k$ constraints in our linear program, we solve the dual as prescribed in GLM-LP (but with $(g_1, \ldots, g_m)$ replaced by $(g_{i_1}, \ldots, g_{i_k})$). This is equivalent to fixing some of the multipliers at zero. If the optimal primal basis $w^*$ violates one or more (of the ignored) constraints, the problem is augmented by including one of the violated constraints, thus increasing the size of the basis by 1. In some cases, a dual solution to our original problem may be reached before the size of the basis grows "too large". If $P^*$ is a convex program, then inactive constraints can be ignored, so that the size of the basis need not grow.
REFERENCES


