Discussion Paper No. 20

A COLUMN GENERATION ALGORITHM FOR NONLINEAR PROGRAMMING

by

Fredric H. Murphy

October 26, 1972

Graduate School of Management

Northwestern University

Evanston, Illinois

This work was supported in part by National Science Foundation Grant GS-3032.

Abstract

An algorithm using column generation and penalty function techniques is presented. A linear program with a uniformly bounded number of columns, similar to the restricted master in Generalized Programming, is used to reduce the number of constraints included in forming a penalty function. The penalty function is used as a Lagrangian in an unconstrained subproblem.

In solving the convex nonlinear programming problem (NLP):

(1) $$\text{maximize } f(x)$$

subject to

(2) $$g_i(x) \leq 0 \quad \text{for } i = 1,\ldots,m,$$

where $f(x)$ is concave and $g_1(x),\ldots,g_m(x)$ are convex, two types of algorithms that have been developed are the penalty function approaches investigated by Fiacco and McCormick [3] and Dantzig-Wolfe Generalized Programming [2]. We propose an algorithm combining penalty functions and linear programming. This is done in such a way that a penalty function serves as the Lagrangian of Generalized Programming in an unconstrained maximization. The solution of the unconstrained problem is used to determine a new column to add to the linear program. Also the algorithm includes column dropping procedures in this linear program as a means of keeping the number of columns uniformly bounded. The linear program then determines the form of the penalty function for the next iteration. The next section provides a statement of the algorithm after which there is a discussion of the relative merits of the algorithm.

Statement of the Algorithm

In the algorithm, a linear program is used to determine what constraints will be included in the unconstrained maximization. To initialize the algorithm we find an x^0 such that each $g_i(x^0) < 0$ for $i = 1,2,\ldots,m$.

At iteration 0 we form the linear program

$$\text{(1)} \qquad \text{maximize} \quad f(x^0)w_0$$

subject to

$$\text{(2)} \qquad g_i(x^0)w_0 \leq 0 \quad \text{for } i = 1,\ldots,m$$

$$\text{(3)} \qquad w_0 = 1 \ .$$

For an $\epsilon > 0$, which we fix for all iterations, let

$$\text{(4)} \qquad I_1 = \{i \mid g_i(x^0)W_0^1 > -\epsilon\} \ ,$$

(I_1 may be empty) where $W_0^1 = 1$, the optimal solution to (1), (2), and (3).

We then

$$\text{(5)} \qquad \underset{x \,\epsilon\, X}{\text{maximize}} \ Q_1(f(x)) - \sum_{i=I_1} P_1(g_i(x))$$

where X is a compact space containing the feasible region and $Q_1(f(x)) - \sum_{i\epsilon I_1} P_1(g_i(x))$ is an appropriate penalty function, that is, any penalty function for which convergent subsequences of penalty function maximizers solve NLP. Several authors have suggested penalty functions that can be used in (5). With $I_k \subset \{1,\ldots,m\}$ an index set, at iteration k-1, suitable penalty functions are those in Carroll [1]

(6) $$f(x) + r_k \sum_{i\epsilon I_k} \frac{1}{g_i(x)} ,$$

where $r_k \to 0$ as $k \to \infty$; Frisch [6]

(7) $$f(x) + r_k \sum_{i\epsilon I_k} \ln(-g_i(x)) ,$$

where $r_k \to 0$ as $k \to \infty$; Zangwill [12]

(8) $$f(x) - r_k \sum_{i\epsilon I_k} \{\max[0,g_i(x)]\}^2$$

where $r_k \to \infty$; as $k \to \infty$; Fiacco and McCormick [3]

(9) $$\frac{1}{f(x^{k-1})-f(x)} + \sum_{i\epsilon I_k} \frac{1}{g_i(x)} ;$$

and Murphy [8]

$$(10) \qquad f(x) - \sum_{i \epsilon I_k} \frac{1}{s(k)} e^{r(k)g_i(x)} ,$$

where $r(k) \uparrow \infty$ as $k \to \infty$ and $r(k) \geq s(k) \geq 1$. Letting x^1 maximize (5), we add the column $\begin{bmatrix} f(x^1) \\ g_1(x^1) \\ \vdots \\ g_m(x^1) \\ 1 \end{bmatrix}$ to the linear program.

At iteration k we have the linear program (RM), or restricted master in keeping with the Generalized Programming terminology,

$$(11) \qquad \text{maximize} \quad f(x^0)w_0 + \ldots + f(x^{\ell(k)})w_{\ell(k)}$$

subject to

$$(12) \qquad g_i(x^0)w_0 + \ldots + g_i(x^{\ell(k)})w_{\ell(k)} \leq 0 \text{ for } i = 1, \ldots, m$$

$$(13) \qquad w_0 + \ldots + w_{\ell(k)} = 1$$

$$(14) \qquad w_j \geq 0 \qquad \text{for} \quad j = 0,\ldots,\ell(k) ,$$

where $\ell(k) + 1$ is the number of columns at iteration k.

Let $W_0^k,\ldots,W_{\ell(k)}^k$ be an optimal solution to RM; and let

$$(15) \qquad x_k = x^0W_0^k + \ldots + x^{\ell(k)}W_{\ell(k)}^k .$$

Set

(16) $$I_{k+1} = \{i \mid g_i(x^0)W_0^k + \ldots + g_i(x^{\ell(k)})W_{\ell(k)}^k > -\varepsilon\} ,$$

for the same ε as in (4). We then

(17) $$\underset{x \in X}{\text{maximize}} \quad Q_{k+1}(f(x)) - \sum_{i \in I_{k+1}} P_{k+1}(g_i(x)) .$$

Let $x^{\ell(k+1)}$ maximize (17). In RM we consider basic solutions [11] only. We drop all nonbasic columns, except the column associated with x^0.

We then add the column $\begin{bmatrix} f(x^{\ell(k+1)}) \\ g_1(x^{\ell(k+1)}) \\ \vdots \\ g_m(x^{\ell(k+1)}) \\ 1 \end{bmatrix}$ to form a new RM2, with at most $m+3$ nonslack columns, and continue.

The purpose of retaining the column determined by x^0 in RM is to ensure that the dual variables of RM are <u>uniformly</u> bounded. The dual variables are not used in an algorithmic sense in that they are not used as coefficients in the function to be maximized in (17). However, we need to have the dual variables of RM bounded in the limit to ensure convergence.

Since the choice of ε is arbitrary, for computational purposes we let ε be smaller than the smallest number the computer can handle. Then I_{k+1} becomes the set of indices of constraints in RM at iteration k that are binding within the numerical tolerances of the machine.

Comments on the Algorithm

At this moment, there is no computational experience with this algorithm so that all comments are derived from the structure of the algorithm as presented. The usefulness of this algorithm is dependent on which penalty function is used for the unconstrained problem (17). For the sake of generality convergence is proved for all penalty functions even though it is doubtful whether there is any practical use in considering a penalty function such as (9) in an algorithm of this sort. An advantage of this algorithm is that it uses the linear programs to avoid calculations on constraints that are nonbinding in a linear programming optimum. This simplifies the unconstrained maximization. For example in their calculations with penalty functions Fiacco and McCormick [3, P. 167] use a modified Newton technique to solve the unconstrained problem. This means that the matrix of second partial derivatives must be calculated for each constraint function, whether the constraint is binding at the optimal solution or not. The same is also true when using the Davidon technique as in Fletcher and McCann [5]. Here, these sorts of calculations need by done only for the constraints selected by the linear program. With certain penalty functions, however, this can be a disadvantage. With interior penalty functions like (6), (7) and (8), x^{k-1} may not be in $\{x | g_i(x) < 0 \text{ for } i \in I_k\}$. This is because I_k may change at every iteration. As will be shown, for every subset of $\{1,\ldots,m\}$ that is repeated infinitely often, the relaxed nonlinear program consisting of maximizing $f(x)$ over the feasible region $g_i(x) \leq 0$ for i in this subset has an optimal solution of the same value as an optimal solution of NLP. This might, then, reduce significantly the number of I_k faced in applying this algorithm. Exterior penalty functions and exponential penalty functions like (10) are unaffected by this problem. Actually, having x_k as a trial solution to NLP here is an advantage because one of the complaints with exterior penalty functions is that they never provide a feasible solution to NLP if they do not terminate finitely. Another reason for an algorithm of this sort is that the rate of convergence of penalty function algorithms is a function of the parameter r_k. By having an inner linearization of NLP, x_k the solution to RM at iteration k, has the potential of being a better approximation to an optimal solution of NLP than $x^{\ell(k)}$. If $f(x)$ is strictly concave, the limiting convergence rate of x_k to an optimal solution of NLP is at least as good as that of $x^{\ell(k)}$.

In the Generalized Programming algorithm [2] we have upper and lower bounds on the value of an optimal solution of NLP. Here, $f(x_k)$ forms a lower bound as in Generalized Programming. With some of the penalty functions presented we can formulate upper bounds. For example, using (10),

$$(18)\qquad f(x^{\ell(k)}) - \sum_{i\in I_k} \frac{1}{s(k)} e^{r(k)g_i(x^{\ell(k)})} \geq f(x^*) - \sum_{i\in I_k} \frac{1}{s(k)} e^{r(k)g_i(x^*)} \geq f(x^*) - \sum_{i\in I_k} \frac{1}{s(k)}$$

where x^* is an optimal solution to NLP. Letting $m(k)$ be the number of constraints indexed by I_k, we have

$$(19)\qquad f(x^{\ell(k)}) - \sum_{i\in I_k} \frac{1}{s(k)} e^{r(k)g_i(x^{\ell(k)})} + \frac{m(k)}{s(k)} \geq f(x^*).$$

By choosing $s(k)$ so that $s(k) \to \infty$ the inequality in (19) becomes an equality.

Therefore, by an appropriate choice of penalty function we can calculate an effective upper bound to $(f(x^*))$.

As a final comment, the technique of proof is of interest in itself as a useful tool for proving the convergence of nonlinear programming algorithms involving linear programs. It has been used successfully to prove convergence of cutting plane algorithms [10] and Generalized Programming [9] under certain conditions when rows and columns are dropped from the linear programs.

Proof of Convergence

First, the number of nonslack columns in RM not including the column determined by x^0, is $\ell(k)$ at iteration k, where $\ell(k)$ is less than or equal to $m + 2$, the added column plus at most $m + 1$ basic nonslack columns. For all k, $x^{\ell(k)} \epsilon X$, a compact set, with the result that all coefficients in RM are uniformly bounded, since $f(x), g_1(x), \ldots, g_m(x)$ are continuous. Also, $0 \leq W_i^k \leq 1$ for $i = 1, \ldots, \ell(k)$ for all k.

Before continuing, we clarify a point about the construction of RM. At each iteration we drop all nonbasic columns except for the initial column, if it is nonbasic. After doing that we reindex all the columns that remain and add our new column to the rightmost position in the matrix of columns. Doing this means that the x^j that determines column j at iteration k is different from the x^j at iteration $k + 1$, if, for some $h \leq j$, the column determined by x^h is dropped. The reason for making this clear is that we use a subsequence on which the x^j's that determine column j on this subsequence converge to a limit. All that we need to know about these x^j's for the subsequence to exist is that they are in the compact space X. We do not need to know anything as to how they came to determine column j for the purpose of taking a subsequence.

Using the properties of RM mentioned at the beginning of this section, we will state the existence of a subsequence with seven properties. To show that such a subsequence exists, all we must do is take subsequences of subsequences until we have completed our list of properties.

We claim that there exists a subsequence indexed by k_u where

(a) $L_1 = \ell(k_u)$

(b) $L_2 = \ell(k_u + 1)$

(c) $J = \{j | W_j^{k_u} \text{ is basic}\}$

(d) $I = I_{k_u+1}$

(e) x^j at iteration k_u converges to $\bar{x}^j$ as $k_u \to \infty$ for $j = 0,\ldots,L_1$

(f) x^{L_2} at iteration k_u converges to $\bar{x}$ as $k_u \to \infty$

(g) $W_j^{k_u} \to \bar{W}_j$ for $j = 0,\ldots,L_1$, as $k_u \to \infty$.

The purpose behind defining the subsequence indexed by k_u is to have two subsequences of linear programs. On the subsequence indexed by k_u, RM2 has exactly $L_1 + 1$ columns, property a, and the x^j that determine the columns and the $W_j^{k_u}$ that form the optimal solution to RM converge to finite limits, properties e and g. Since the same constraints are binding or within ϵ

of binding at every iteration by d, the penalty function solves the same relaxed problem at iteration k_u. On the subsequence indexed by $k_u + 1$, by property c, we save the same columns from iteration k_u, which means that this RM has exactly $L_2 + 1$ columns. Since by property f the solution to the unconstrained problem converges to a finite limit, and by properties c and e the columns that are saved for iteration $k_u + 1$ converge to finite limits, the sequence of RM's at iterations $k_u + 1$ has the same number of columns with the x^j that determine these columns converging to finite limits.

Since $x^j \to \bar{x}^j$, on our subsequence, by the continuity of $f(x)$ and $g_1(x),\ldots,g_m(x)$ we set

(20) $$f_j = \lim_{k_u \to \infty} f(x^j) \quad \text{for } j = 0,\ldots,L_1$$

and

(21) $$g_{ij} = \lim_{k_u \to \infty} g_i(x^j) \quad \text{for } i = 1,\ldots,m \text{ and } j = 0,\ldots,L_1 .$$

Because we have chosen everything to converge on the subsequence k_u, we analyze RM in the limit. To do this we need a definition from [9].

<u>Definition 1</u> Given a sequence of linear programs with coefficients uniformly bounded

(22) $$\text{maximize} \sum_{j=1}^{n} c_j^k w_j$$

subject to

(23) $$\sum_{j=1}^{n} a_{ij}^{k} w_j \leq b_i^{k} \quad \text{for} \quad i = 1,\ldots,m'$$

where, for $k = 1,2,\ldots,$ $c_j^k \to c_j^\infty$, $a_{ij}^k \to a_{ij}^\infty$, and $b_i^k \to b_i^\infty$, the <u>limiting linear program</u> is defined as

(24) $$\text{maximize} \quad \sum_{j=1}^{n} c_j^\infty w_j$$

subject to

(25) $$\sum_{j=1}^{n} a_{ij}^\infty w_j \leq b_i^\infty \quad \text{for} \quad i = 1,\ldots,m' .$$

Here we have a limiting linear program for our subsequence k_u of RM's:

(26) $$\text{maximize} \quad f_0 w_0 + \ldots + f_{L_1} w_{L_1}$$

subject to

(27) $$g_{i0} w_0 + \ldots + g_{iL_1} w_{L_1} \leq 0 \quad \text{for} \quad i = 1,\ldots,m$$

(28) $$w_0 + \ldots + w_{L_1} = 1$$

$$w_j \geq 0 \qquad \text{for } j = 1,\ldots,L_1 .$$

Since the column indices in an optimal basis are the same for all iterations k_u, column j at iteration k_u either is dropped or translated into a position, either the same or new, that is the same for all RM's indexed by $k_u + 1$. That is, column j becomes column $t(j)$ in the new RM2 if it is retained. This means that $g_i(x^{t(j)}) \to g_{it(j)}$ and $f(x^{t(j)}) \to f_{t(j)}$ on the subsequence indexed by $k_u + 1$ for all i and all

j for which $t(j)$ is defined. Also, by property f we see that $g_i(x^{L_2}) \to \bar{g}_{iL_2}$ for all i and $f(x^{L_2}) \to \bar{f}_{L_2}$ on the subsequence $k_u + 1$. Setting $g_{it(j)} = \bar{g}_{it(j)}$ for $i = 1,\ldots,m$ and for $t(j) = 0,\ldots,L_2-1$ and $f_{t(j)} = \bar{f}_{t(j)}$ for $t(j) = 0,\ldots,L_2-1$, we then have a second limiting linear program for the subsequence of iterations indexed by k_u+1

(29) $$\text{maximize } \bar{f}_0 w_0 + \ldots + \bar{f}_{L_2} w_{L_2}$$

subject to

(30) $$\bar{g}_{i0} w_0 + \ldots + \bar{g}_{iL_2} w_{L_2} \leq 0 \quad \text{for } i = 1,\ldots,m$$

(31) $$w_0 + \ldots + w_{L_2} = 1$$

(32) $$w_j \geq 0 \qquad \text{for } j = 1,\ldots,L_2 .$$

Since an optimal solution at iteration k is feasible at iteration $k + 1$, the values of the optimal solutions to the RM's are monotonically increasing and are bounded above by the value of the optimal solution to the NLP. This means the values of the optimal solutions converge to a finite limit f. Hoffman and Karp [7] prove the result below on the continuity of linear programs.

<u>Theorem 2</u> Consider the following two linear programs

(33) $$\text{maximize } \sum_{j=1}^{n} c_j w_j = p$$

subject to

(34) $$\sum_{j=1}^{n} a_{ij} w_j \leq b_i \quad \text{for } i = 1,\ldots,m,$$

with P the set of optimal solutions to (33) and (34) and D the set of optimal solutions to the dual of (33) and (34), and

(35) $$\text{maximize} \sum_{j=1}^{n} c_j^* w_j = p^*$$

subject to

(36) $$\sum_{j=1}^{n} a_{ij}^* w_j \le b_i^* \quad \text{for} \quad i = 1,\ldots,m \;,$$

with P^* the set of optimal solutions to (35) and (36) and D^* the set of optimal solutions to the dual of (35) and (36). If P, P^*, D, D^* are compact sets, then given any $\varepsilon > 0$, there exists a $\sigma_\varepsilon > 0$ such that, if

(37) $$\max \begin{bmatrix} |a_{ij} - a_{ij}^*|, & \text{all } i,j \\ |c_j - c_j^*|, & \text{all } j \\ |b_i - b_i^*|, & \text{all } i \end{bmatrix} < \sigma_\varepsilon \;,$$

then $|p - p^*| < \varepsilon$.

By retaining the column determined by x^0, we know that the optimal solutions to the duals of the two limiting linear programs and all RM's are bounded [2]. Also, $0 \le W_j^{k_u} \le 1$ for $j = 0,\ldots,\ell(k_u)$.

From this we can conclude that an optimal solution to either of the two limiting linear programs produces an objective function value of f. Since $W_0^{k_u},\ldots,W_{L_1}^{k_u}$ is feasible in RM for all k_u, $\overline{W}_0,\ldots,\overline{W}_{L_1}$ is feasible in the first limiting linear program, because

(38) $$0 \geq \lim_{k_u \to \infty} [g_i(x^0)W_0^{k_u} + \ldots + g_i(x^{\ell(k_u)})W_{L_1}^{k_u}]$$

$$= \lim_{k_u \to \infty} g_i(x^0) \lim_{k_u \to \infty} W_0^{k_u} + \ldots + \lim_{k_u \to \infty} g_i(x^{\ell(k_u)}) \lim_{k_u \to \infty} W_{L_1}^{k_u}$$

for all i. The solution $\bar{W}_0, \ldots, \bar{W}_{L_1}$ is also optimal in the first limiting linear program since

(39) $$f_0\bar{W}_0 + \ldots + f_{L_1}\bar{W}_{L_1} = \lim_{k_u \to \infty} [f(x^0)W_0^{k_u} + \ldots + f(x^{\ell(k_u)})W_{L_1}^{k_u}] = f \, .$$

In the second limiting linear program we set

(40) $$\hat{W}_{t(j)} = \bar{W}_j \quad \text{for} \quad t(j) = 0, \ldots, L_2 - 1$$

and

(41) $$\hat{W}_{L_2} = 0 \, ,$$

that is, we translate the $\bar{W}_j$'s to match the translation of the columns. This solution is optimal in the second limiting linear program, because the value of an optimal solution in (29, (30), (31) and (32) is f and

(42) $$\bar{f}_0\hat{W}_0 + \ldots + \bar{f}_{L_2}\hat{W}_{L_2} = \sum_{t(j)=0}^{L_2-1} \bar{f}_{t(j)}\hat{W}_{t(j)}$$

$$= f_0\bar{W}_0 + \ldots + f_{L_1}\bar{W}_{L_1}$$

$$= f \, ;$$

and

(43)
$$0 \geq g_{i0}\overline{W}_0 + \dots + g_{iL_2}\overline{W}_{L_2}$$
$$= \sum_{t(j)=0}^{L_2-1} \overline{g}_{it(j)}\hat{W}_{t(j)}$$
$$= \overline{g}_{i0}\hat{W}_0 + \dots + \overline{g}_{iL_2}\hat{W}_{L_2} ,$$

for $i = 1,\dots,m$.

We must observe what happens in the subproblem maximization. Since $I_{k_u+1} = I$ for all k_u, this subproblem is the subproblem in the penalty function method of solving the nonlinear programming problem

(44)
$$\underset{x \, \varepsilon \, X}{\text{maximize}} \quad f(x)$$

subject to

(45)
$$g_i(x) \leq 0 \quad \text{for} \quad i \,\varepsilon\, I .$$

Now by the properties of penalty function methods [4, 6, 8] extended to the case of X constrained subproblems, $\overline{\mathbf{x}} = \lim_{k_u \to \infty} x^{\ell(k_u+1)}$ is an optimal solution to (44) and (45). This means that for x^* an optimal solution to the NLP

(46)
$$f(x^*) \leq \lim_{u \to \infty} f(x^{\ell(k_u+1)}) = f(\overline{x}) .$$

<u>Theorem 3</u> With x_{k_u} defined as in (15), $\hat{x} = \lim_{k_u \to \infty} x_{k_u}$ is an optimal solution to NLP.

<u>Proof</u> Since $\hat{x}$ is feasible in NLP,

(47) $$f(\hat{x}) \leq f(x^*) \leq f(\bar{x}) \;,$$

where x^* is an optimal solution to NLP. Now,

(48) $$g_i(\bar{x}) \leq 0 \text{ for } i \in I \;,$$

and with ϵ being fixed at the first iteration of the algorithm in defining I_1 at (4),

(49) $$\bar{g}_{i0}\hat{W}_0 + \ldots + \bar{g}_{iL_2-1}\hat{W}_{L_2-1} \leq -\epsilon \quad \text{for} \quad i \notin I \quad \text{because}$$

(50) $$g_{i0}\bar{W} + \ldots + g_{iL_2}\bar{W}_{L_2} \leq -\epsilon \qquad \text{for} \quad i \notin I \;.$$

From the concavity of $f(x)$ and (47) we know

(51) $$\bar{f}_0\hat{W}_0 + \ldots + \bar{f}_{L_2-1}\hat{W}_{L_2-1} \leq f(\hat{x}) \leq f(\bar{x}) \;.$$

Assume

(52) $$\bar{f}_0\hat{W}_0 + \ldots + \bar{f}_{L_2-1}\hat{W}_{L_2-1} < f(\bar{x}) \;.$$

Then

(53) $$\bar{f}_0\hat{W}_0 + \ldots + \bar{f}_{L_2-1}\hat{W}_{L_2-1} < (1-\lambda)(\bar{f}_0\hat{W}_0 + \ldots + \bar{f}_{L_2-1}\hat{W}_{L_2-1}) + \lambda f(\bar{x}) \quad \text{for } 0 < \lambda \leq 1.$$

If $\{i \mid g_i(\bar{x}) > 0\}$ is empty let $\lambda = 1$. Otherwise let

(54) $$\lambda = \min_{1 \leq i \leq m} \left\{ \frac{-[\bar{g}_{i0}\hat{W}_0 + \ldots + \bar{g}_{iL_2-1}\hat{W}_{L_2-1}]}{g_i(\bar{x}) - [\bar{g}_{i0}\hat{W}_0 + \ldots + \bar{g}_{iL_2-1}\hat{W}_{L_2-1}]} \;\middle|\; g_i(\bar{x}) > 0 \right\}$$

We have $\lambda > 0$ because for all i with $g_i(\bar{x}) > 0$

$$\bar{g}_{i0}\hat{W}_0 + \ldots + \bar{g}_{iL_2-1}\hat{W}_{L_2-1} \leq -\epsilon \ , \tag{55}$$

that is, $i \notin I$ because $\bar{x}$ is feasible in (45).

By our choice of λ, $[(1-\lambda)\hat{W}_0, \ldots, (1-\lambda)\hat{W}_{L_2-1}, \lambda]$ is a feasible solution to the second limiting linear program. However, by (53) since $\lambda > 0$, this solution strictly improves on the optimal solution $\hat{W}_0, \ldots, \hat{W}_{L_2}$, contradicting the Hoffman and Karp result. This means our assumption (52) is false, that is,

$$\bar{f}_0\hat{W}_0 + \ldots + \bar{f}_{L_2}\hat{W}_{L_2} = f(\bar{x}) \ . \tag{56}$$

This implies, by (47) and (51)

$$f(\hat{x}) = f(x^*) = f(\bar{x}) \ , \tag{57}$$

Which implies the optimality of $\hat{x}$.

Note that

$$f(\lambda\hat{x} + (1-\lambda)\bar{x}) = f(\bar{x}) \quad \text{for} \quad 0 \leq \lambda \leq 1 \tag{58}$$

because $\hat{x}$ is feasible in the relaxed nonlinear programming problem making $\lambda\hat{x} + (1-\lambda)\bar{x}$ for $0 \leq \lambda \leq 1$ feasible in the relaxed problem. This is because

$$\begin{aligned} f(\lambda\hat{x} + (1-\lambda)\bar{x}) &\geq \lambda f(\hat{x}) + (1-\lambda)f(\bar{x}) \\ &= f(\bar{x}) \\ &\geq f(\lambda\hat{x} + (1-\lambda)\bar{x}) \end{aligned} \tag{59}$$

by the concavity of $f(x)$ and the optimality of $\bar{x}$ in the relaxed problem. We know, then, if $f(x)$ is strictly concave, $\hat{x} = \bar{x}$. That is, for any subset of indices that are repeated infinitely often, an optimal solution to the relaxed problem involving only the constraints indexed by this set is optimal in the NLP. Therefore, the limiting convergence rate to an optimal solution to the NLP is the same as the convergence rate of the original penalty function used.

Because the trial solution to RM at iteration k is feasible at iteration k+1, the values of the optimal solutions to the RM's are monotonically increasing to $f(x^*)$. With x_k defined as in (15), $f(x_k) \to f(x^*)$. However, this is not necessarily monotonic. Setting $f^k = \underset{1<h<k}{\text{maximum}} f(x_h)$, we have $f^k \uparrow f(x^*)$.

References

(1) Carroll, C. W., "The Created Response Surface Technique for Optimizing Nonlinear Restrained Systems," Operations Research, Vol. 9 (1961), pp. 169-184

(2) Dantzig, G., Linear Programming and Extensions, Princeton University Press, Princeton, New Jersey, 1963.

(3) Fiacco, A. and McCormick, G., "The Sequential Unconstrained Minimization Technique Without Parameters," Operations Research, Vol. 15 (1967), pp. 820-827.

(4) Fiacco, A. and McCormick, G., Nonlinear Programming, Sequential Unconstrained Minimization Techniques, Wiley, New York, 1968

(5) Fletcher, R. and McCann, A.P. "Acceleration Techniques for Nonlinear Programming," in Optimization, Fletcher (ed), Academic Press, 1969.

(6) Frisch, K. R., "The Logarithmic Potential Method of Convex Programming," Memorandum of May 13, 1955, University Institute of Economics, Oslo

(7) Hoffmann, A. J. and Karp, "On Nonterminating Stochastic Games," Management Science, Vol. 12 (1966), pp. 349-370.

(8) Murphy, F., "A Class of Exponential Penalty Functions" in preparation.

(9) Murphy, F., "Column Dropping Procedures for the Generalized Programming Algorithm" submitted to Management Science.

(10) Murphy, F., "Row Dropping Procedures for Cutting Plane Algorithms" submitted to Operations Research.

References, Cont.

(11) Wagner, H. M., *Principles of Operations Research With Applications to Managerial Decisions* , Prentice-Hall, 1969.

(12) Zangwill, W., "Nonlinear Programming via Penalty Functions", Management Science, Vol. 13 (1967), pp. 344-358.