Discussion Paper No. 308

"The Fixed Point Approach to Nonlinear Programming"*

by

Ramesh Saigal

December, 1977

*This research was partially supported by Grant No. MCS 77-03472 from the National Science Foundation.
The Fixed Point Approach to Nonlinear Programming

R. Saigal
Northwestern University
Evanston, Illinois

ABSTRACT

In this paper we consider the application of the recent algorithms that compute fixed points in unbounded regions to the nonlinear programming problem. It is shown that these algorithms solve the inequality constrained problem with functions that are not necessarily differentiable. The application to convex and piecewise linear problems is also discussed.
1. Introduction

In this paper we consider the problem

\[ \min_t \eta_0(t) \]  

\[ \eta_i(t) \geq 0 \quad i = 1, \ldots, m \]  

where \( \eta_0 \) and \( \eta_i \) are arbitrary functions on a set \( X \subset \mathbb{R}^n \), the n-dimensional Euclidean space. The set \( X \) may be discrete, but for simplification, we assume that the convex hull of \( X \) is \( \mathbb{R}^n \), and that there exists a subdivision of \( \mathbb{R}^n \) with vertices in \( X \). One such example of \( X \) is the grid of integers, for which efficient triangulation procedures exist. See Todd [16].

Our approach is to consider piecewise linear approximations \( \eta_i^k \), \( i = 0, \ldots, m \) instead, and then to solve the continuous problem by the fixed point algorithms. We thus obtain an approximate solution to (1.1-2). Such an approach has been successfully used in [8]. We note that \( \eta_i^k \) are not differentiable. Since they are piecewise linear, a notion of a generalized subdifferential can be readily defined. This is the same as the generalized gradient of Clarke [1], and for convex \( \eta_i^k \), the same as the subdifferential of convex functions, Rockafellar [2].

Since the fixed point algorithms of Eaves and Saigal [3] and Merrill [6]
can successfully find fixed points of certain point-to-set mappings, we formulate this problem as such a point-to-set mapping problem which can then be solved by these algorithms.

Hansen [4], Hansen and Scarf [5], and Eaves [2] had recognized the potential of applying these methods to nonlinear programming, but the full potential was explored by Merrill [6]. In section 3, we present extensions of several of his results for the convex case. Traditional descent type methods for solving this problem are summarized in Mifflin [7]. Also, in [7], a steepest descent type algorithm, using the mapping of [6], is presented.

In section 2, we present a brief overview of the fixed point algorithms; in section 3 these algorithms are applied to the constrained and unconstrained convex problems; in section 4 we introduce piecewise linear mappings and establish the necessary and sufficient conditions for local minimization of the constrained and the unconstrained problems; in section 5 the application of the algorithms is discussed; and in section 6 we discuss the computational aspects. Finally, in the appendix, we present the computational experience of solving some fairly large nondifferentiable problems.
2. The Algorithms

We now give a brief description of the algorithm of Eaves and Saigal [3] implemented on the subdivision J3 of IR^n x (D,D). We will assume that the nonlinear programming problem is being solved by this algorithm.

The triangulation J3 of IR^n x (D,D) has vertices in IR^n x (D,D) for k = 0, 1, 2, ... Also, v = (v_0, ..., v_{n+1}) is a vertex if v_{n+1} = D.ε^{-k} for some integer k and v_i / v_{n+1} is an integer for each i. In case, for a vertex v, v_i / v_{n+1} is an odd integer, it is called a central vertex. Any simplex in J3 then has a unique representation by a triplet (v,x,s) where v is a central vertex, v is a permutation of (i, ..., n+1) and s is an n-vector with s_i \epsilon \{-1, +1\}. A complete description of J3 can be found in Saigal [11], and Todd [16].

Given a point set mapping k from IR^n into nonempty subsets of IR^n, and a 1-1 linear mapping r from IR^n into IR^n, we say a n-simplex \sigma = (v^1, ..., v^{n+1}) is

(a) r - complete if 0 \epsilon hull(r(\sigma)),
(b) for - complete if 0 \epsilon hull (r(\sigma)u(\sigma)),
(c) k - complete if 0 \epsilon hull (t(\sigma)).

These algorithms, starting with a unique r-complete simplex \sigma_0 containing the unique zero of r, generate a sequence of for-complete

simplexes \sigma_0, \sigma_1, \sigma_2, ..., \sigma_k, ... In case these simplexes lie in a bounded region, it can be readily shown that if they are from J3, there is a subsequence \sigma_{i_1}, \sigma_{i_2}, \sigma_{i_3}, ..., \sigma_{i_k}, ... of i- complete simplexes such that \sigma_k \epsilon IR^n x (D,D) and so the diameter of \sigma_k approaches zero as k approaches \infty.

Thus, the failure of the algorithm implies that it has generated for-complete simplexes sufficiently far from \eta, the unique zero of r.
In applications to nonlinear programming, we will frequently choose
\[ t(x) = x - x_0 \]
and define a point-to-set mapping \( \mathcal{L} \) such that if, for some \( x \), \( \mathcal{O}(x) \),
then \( x \) is a solution to our problem.

We now prove the following result:

**Theorem 2.1** Let \( s_1, s_2, \ldots, s_k, \ldots \) be a sequence of \( \mathcal{L} \)-complete
simplexes which lie in a bounded set and the diameter \( \epsilon_k \) of \( s_k \) approaches \( 0 \). In addition, let \( \mathcal{L} \) be a upper semi-continuous point-to-set mapping
with \( \mathcal{L}(x) \) nonempty, compact and convex subsets of \( \mathbb{R}^n \). Then, if \( x \) is a
cluster point of \( \{x_k\}_{k=0}^{\infty} \) with \( x_k \in s_k \), then \( \mathcal{O}(x) \).

**Proof:**

Since \( \{s_k\}_{k=0}^{\infty} \) lie in a bounded region, say \( B \), under the hypothe-
sis on \( \mathcal{L} \), \( \mathcal{L}(B) \) is compact, hence \( \mathcal{L}(B) \) is bounded (\( B \) is the closure
of \( B \)). Now, as \( s_k = (s_k^1, s_k^2, \ldots, s_k^n) \) is \( \mathcal{L} \)-complete, there
exist \( y_{1,k}^1 \in \mathcal{L}(s_k^1) \) and \( \lambda_{1,k} \geq 0 \), \( \sum_{i=1}^n \lambda_{i,k} = 1 \) such that
\[ \lambda_{1,k} y_{1,k}^1 + \sum_{i=1}^n \lambda_{i,k} y_{i,k}^i = 0. \]
Since \( 0 \leq \lambda_{i,k} \leq 1 \), \( y_{i,k}^i \in \mathcal{L}(s_k) \) on some common subsequence \( \lambda_{1,k} \rightarrow \lambda_1 \) for all \( i \) and \( y_{i,k}^i \rightarrow y_i^i \) for all \( i \). Thus
\[ \lambda_{1,k} y_{1,k}^1 = 0 \]
\[ \sum_{i=1}^n \lambda_{i,k} = 1 \]
and \( \lambda_{1,k} = 0 \).

Also, as \( \text{diam}(s_k) \) approaches \( 0 \), on some subsequence \( \lambda_{1,k} \rightarrow \lambda_1 \) for all \( i \).

Since \( y_{i,k}^i \in \mathcal{L}(s_k^i) \), using the upper semi-continuity of \( \mathcal{L} \) we have \( y_{i}^i \in \mathcal{L}(x) \),
and since \( \mathcal{L}(x) \) is convex, we have our result.
3. Convex Case

In this section, we will consider the applications of fixed point algorithms for solving (1.1-2) when the underlying functions are convex, not necessarily differentiable. We will make the simplifying assumption that the functions are defined over all of \( \mathbb{R}^n \), and that they are finite.

3.1 Unconstrained Case

We now consider the problem of minimizing \( g_0 \) when the set \( (x : g_0(x) \leq g_0(x_0)) \) is bounded for some \( x_0 \), and the function \( g_0 \) is convex.

The subdifferential set \( \partial g_0(x) \) of a convex function \( g_0 \) at \( x \) is the set of all vectors \( x^\ast \) in \( \mathbb{R}^n \) such that

\[
  g_0(y) \leq g_0(x) + \langle x^\ast, y-x \rangle
\]

for all \( y \) in \( \mathbb{R}^n \) (3.1)

and under our assumption this set is nonempty, closed and bounded,
[9, Theorem 23.4].

A trivial consequence of (3.1) is the following theorem.

**Theorem 3.1:** Under the above conditions on \( g_0 \), \( x \) solves (1.1) if and only if \( 0 \in \partial g_0(x) \).

We now show that the algorithms of Section 2 implemented with

\[
  r(x) = x - x_0
\]

\[
  s(x) = \partial g_0(x)
\]

for an arbitrary starting point \( x_0 \) and initial grid size \( c_0 \) will converge to a solution of (1.1). Let \( B(x,c) = \{ y : ||y-x|| < c \} \), and

\[
  \mathcal{M}(x,c) = \sup \{ g_0(y) : y \in B(x,c) \}.
\]
Theorem 3.2: Starting with the unique \( r \)-complete simplex containing \( x_0 \),
the fixed point algorithms will succeed in generating \( \ell \)-complete simplices of diameters \( \varepsilon < \varepsilon_0 \).

Proof:
Assume that the algorithm does not compute a \( \ell \)-complete simplex
of diameter \( \varepsilon < \varepsilon_0 \). Now, define
\[
D = \{ x : \mathcal{E}_0(x) \geq M(x_0, \varepsilon_0) \}
\]
Since \( D \) is bounded, we can find a \( \ell \)-complete simplex \( \sigma = (v^1, \ldots, v^{\ell+1}) \)
sufficiently far from \( x_0 \) such that for some \( \kappa \in \sigma \), \( v^1 - x_0, x - x_0 > 0 \),
and such that \( v^1 \notin D \) for each \( i = 1, \ldots, \ell+1 \). Now, as \( \mathcal{E}_0 \) is convex, by
(3.1) for every \( y \in \mathcal{E}_0(v^1) \) and every \( i \)
\[
\mathcal{E}_0(v^1) + x_0 \geq \mathcal{E}_0(v^1) + \langle x_0, y \rangle
\]
Since \( v^1 - x_0 \in D \), we have \( \langle x_0, x_0 \rangle > 0 \) for all \( y \in \mathcal{E}_0(v^1) \) and every \( i \).
Hence, using Farkas, we have that \( \sigma \) cannot be \( \ell \)-complete, and
we have a contradiction.

Now, let \( \sigma = (v^1, \ldots, v^{\ell+1}) \) be a \( \ell \)-complete simplex of
diameter \( \varepsilon > 0 \). Then there exist \( y^*_1, x \in \mathcal{E}_0(v^1) \), \( x_1 = 0, 1, \ldots, \ell+1 \)
\( \Sigma_{i=1}^{\ell+1} \) such that \( \Sigma_{i=1}^{\ell+1} y^*_i = 0 \). We can then prove that:

Theorem 3.3: Let \( x \) be a solution to (3.1). Then there is some \( x \in \sigma \) such
that
\[
\mathcal{E}_0(x) = \mathcal{E}_0(x) = \bigoplus_{i=1}^{\ell+1} \langle v^i, y^*_i \rangle
\]
and
\[
M(x, \varepsilon) \geq \mathcal{E}_0(x).
\]

Proof:
Let \( x = \Sigma_{i=1}^{\ell+1} v^i \). From (3.1)
\[
\mathcal{E}_0(x) = \mathcal{E}_0(v^1) + \langle x_0 - v^1, y^*_1 \rangle
\]
Hence
\[
\begin{align*}
g_0(x) & \leq \sum_{i=1}^{n+1} \lambda_i g_i(y_i^1) - \sum_{i=1}^{n+1} \lambda_i <y_i^1,y_i^1> \\
& \leq g_0(x) - \sum_{i=1}^{n+1} \lambda_i y_i^1\cdot y_i^1
\end{align*}
\]
and the first part follows. Also,
\[
g_0(v^1 + x + \bar{u}) \geq g_0(v^1) + \langle \bar{x} - x, y_i^1 \rangle
\]
and so
\[
\sum_{i=1}^{n+1} \lambda_i g_i(v^1 - x + \bar{u}) \geq \sum_{i=1}^{n+1} \lambda_i g_i(v^1) \geq g_0(x).
\]
But, as \(v^1 - x + \bar{u} \in \mathcal{B}(\bar{x},\bar{u})\), we have our result.

Note that in Theorem 3.3, (3.2) gives a computable lower bound on the minimum value of \(g(x)\) and can thus be used as a stopping rule. Also, (3.3) shows that the algorithm is converging to a minimum.

### 3.2 Constrained Case

We now consider the problem (1.1-2) when the functions \(g_i\) are convex functions. Define,
\[
s(x) = \max \{ g_i(x) : 1 \leq i \leq m \}
\]
and we note the \(s\) is also a convex function. We now assume that the set \(\{x : s(x) \geq s(x_0)\}\) is bounded for some \(x_0\). Now, define the mapping
\[
s(x) =
\begin{cases} 
    s_i(x) & \text{if } s(x) < 0 \\
    s_i(x) + 3s(x) & \text{if } s(x) = 0 \\
    3s(x) & \text{if } s(x) > 0
\end{cases}
\]

**Theorem 3.4:** Let \(\bar{x}\) be such that \(0 \in \mathcal{I}(\bar{x})\). Then \(\bar{x}\) solves (1.1-2) or indicates that (1.2) has no solution.
Proof:

There are three cases.

Case (i). $s(\tilde{x}) > 0$. In this case $0 \in \partial s(\tilde{x})$, and thus $\tilde{x}$ is a global minimizer of $s$, and hence the constraint set (1.2) is empty.

Case (ii). $s(\tilde{x}) < 0$. In this case $0 \in \partial g_0(\tilde{x})$ and hence $\tilde{x}$ is a global minimizer of $g_0$. Since it also satisfies (1.2), $\tilde{x}$ solves (1.1-2).

Case (iii). $s(\tilde{x}) = 0$. In this case, there is a $z^* \in \partial g_0(\tilde{x})$ and $y^* \in \partial s(\tilde{x})$ such that $z^* + y^* = 0$. Let $1/\tilde{x} = (\lambda_1: g_1(\tilde{x}) = 0)$. Then, $\partial s(\tilde{x}) = \text{hull} \{ \bigcup_{i \in I(\tilde{x})} \partial g_i(\tilde{x}) \}$ and so there are numbers $\lambda_i \geq 0$, $i \in I(\tilde{x})$, $\lambda_1 = 1$, and $y^*_i \in \partial g_i(\tilde{x})$ such that $y^* = \lambda_1 y^*_1$. Hence, $z^* + \lambda_1 y^*_1 = 0$. Now, let $y$ satisfy (1.2). Then, from (3.1),

$$g_0(y) = g_0(\tilde{x}) + \langle z^*, y - \tilde{x} \rangle$$

$$g_1(y) = g_1(\tilde{x}) + \langle y^*_1, y - \tilde{x} \rangle \quad i \in I(\tilde{x})$$

Hence

$$g_0(y) \geq g_0(\tilde{x}) + \lambda_1 g_1(y)$$

$$= g_0(\tilde{x}) + \langle z^* + \lambda_1 y^*_1, y - \tilde{x} \rangle$$

$$= g_0(\tilde{x})$$

and hence $\tilde{x}$ solves (1.1-2).

We now show that the algorithm initiated with $r(x) = e \cdot x_0$

for arbitrary $x_0$ and $r(x)$ as in (3.4) will find a $\varepsilon$-complete simplex.

Theorem 3.5: Let $\varepsilon > 0$ be arbitrary, and let the algorithm implement the mapping $r$ above. Then, for each $\varepsilon > 0$, the algorithm will find a $\varepsilon$-complete simplex of diameter $\varepsilon$.
Proof:

Let
\[ M(x_0, c) = \max \{ s(x) : x \in B(x_0, c)^\circ \}, 0 \]
and \( F = \{ x : s(x) \leq M(x_0, c) \} \). By assumption, \( D \) is bounded. Now, assume that the algorithm fails. Hence, it generates a simplex \( \sigma = (v_1^\top, \ldots, v_{n+1}^\top) \) of diameter \( c > 0 \) such that \( \sigma \) is \( \nabla \) complete, and sufficiently far from \( D \), i.e., \( 0 \notin D \) and for every \( x \in c, \langle v_i^\top - x_0, x - x_0 \rangle > 0 \). Also \( s(v_i^\top) > 0 \) for all \( i \). Now, consider the point \( v_i^\top - x + x_0 \in D \). Then
\[ s(v_i^\top - x + x_0) \leq s(v_i^\top) + \langle x_0 - x, y_i^\top \rangle \] for all \( y_i^\top \in c s(v_i^\top) \).

Since \( s(v_i^\top) \notin D \), we have \( \langle x - x_0, y_i^\top \rangle > 0 \) for all \( y_i^\top \in c s(v_i^\top) \) and all \( i \); and, from Farkas' lemma, \( \sigma \) cannot be \( \nabla \) complete, a contradiction.

We now assume that there is no solution to (1.2). Hence \( s(x) > 0 \) for all \( x \), and thus (3.4) reduces to \( f(x) = M(x) \). A consequence of Theorem 3.3 is the following.

**Theorem 3.6:** Let \( s(x) > 0 \) for all \( x \), and that \( \{ x : s(x) \leq s(x_0) \} \)
is bounded for some \( x_0 \). Then, the algorithm will detect the infeasibility of (1.2) in a finite number of iterations.

**Proof:**

For each \( c > 0 \), the algorithm computes a \( \nabla \) complete simplex in a finite number of iterations. Also, since \( s(x) > 0 \), it will attempt to minimize \( s(x) \). Now, let \( \sigma = (v_1^\top, \ldots, v_{n+1}^\top) \) be an \( \nabla \) complete simplex of size \( c > 0 \) found by the algorithm. Then, there are \( y_i^\top \in c s(v_i^\top) \) such that \( \Pi_{i=1}^n y_i^\top = 0, c_{i=1}^n \lambda_i = 1, \lambda_i = 0 \) has a solution. Also, from Theorem 3.3, if \( x \) minimizes \( s \),
\[
\begin{align*}
\tilde{s}(x) &= s(x) + \Pi_{i=1}^n \langle v_i^\top, y_i^\top \rangle \\
&= s(x) + \Pi_{i=1}^n \langle v_i^\top - x_0, y_i^\top \rangle \\
&> s(x_0),
\end{align*}
\]
Now, define $D = \{ x : a(x) \neq \emptyset, x \in \mathcal{G}(c) \}$

where $\mathcal{G}(c) = \{ x \in \mathcal{G} : x \neq \emptyset \}$. From Theorem 1.3, $x \in D$ and $a(x) \neq \emptyset$. Define $\mathcal{G}(c) = \{ x \in D(x) \}$ for $x \in \emptyset, x \in \mathcal{G}(c)$. Then

$$ ||x|| = ||x||_\mathcal{G}(c) \leq ||x||_a = ||x||_\mathcal{G}(c) \leq ||x||_a $$

as $\lambda \to 0$, and $a(x) \neq \emptyset$, for some sufficiently small $\epsilon > 0$.

In addition, we can obtain a lower bound on the optimal value of the objective function in this case as well. Let $a = \{ v^1, \ldots, v^m \}$

be $t$-complete and let $w^1, \ldots, w^m$ be labeled by $v^1 \in \emptyset, v^2 \in \mathcal{G}(c)$, and

$v^m \in \mathcal{G}(c)$ be labeled by $v^m_1 \in \emptyset, a(v^m)$. Hence $a(v^m) < 2, 1 = 1, \ldots, t$ and $a(v^m) = 0$ for $v = v^1, \ldots, v^m$. Also, let $2a(v^m) = 0$, $\lambda_1 = 0$, $\lambda_2 = 1$, and $\lambda_3 = \frac{1}{2} \lambda_4 + \frac{1}{2} \lambda_4 v^m$. Then we can prove:

Moreover, if $\lambda = \lambda_1 = \lambda_3 = 0$, then

$$ \lambda_2 = \lambda_1 = 0 $$

Thus:

Using $\mathcal{G}(c)$, we get, for $L = 1, \ldots, t$

$$ \lambda_2 = \lambda_1 = \lambda_3 \geq \lambda_4 = 0 $$

and for $L = 1, \ldots, t$, we get

$$ \lambda_2 = \lambda_3 \geq \lambda_1 = 0 $$

Hence, we have $\lambda = 0$

$$ \lambda_2 = 0 \leq \lambda_1 = 0 \leq \lambda_3 $$

and we verify our result.
4. Piecewise Linear Functions and Nonlinear Programming

In this section we establish the notation and prove some basic results for nonlinear programs with piecewise linear functions.

Cells and Manifolds

A cell is the convex hull of a finite number of points and half lines (half lines are sets of the type \( \{ x : x = a + t b, t \geq 0 \} \)) where \( a \) and \( b \) are fixed vectors in \( \mathbb{R}^n \).

The dimension of a cell is the maximum number of linearly independent points in the cell. We will call an \( n \)-dimensional cell an \( n \)-cell.

Let \( \tau \) be a subset of an \( n \)-cell \( \sigma \). If \( x, y \in \sigma \) \( 0 < \lambda < 1 \),

\[(1-\lambda)x + \lambda y \in \tau \quad \text{implies that} \quad x, y \in \tau \quad \text{then} \quad \tau \quad \text{is called a face of a cell} \quad \sigma .\]

A simple fact is that faces are cells. Also faces that are \( (n-1) \)-cells are called facets of the cell, and that are \( 0 \)-cells are called vertices of the cell.

\[ \emptyset \neq m \quad \text{be a collection of n-cells in} \quad \mathbb{R}^n. \quad \text{Let} \quad M = \bigcup_{m}^{} \]

We call \((M, m)\) a subdivided \( n \)-manifold if

\begin{align*}
(4.1) & \quad \text{Any two n-cells of \( m \) that meet, do so on a common face.} \\
(4.2) & \quad \text{Each (n-1)-face of a cell lies in at most two n-cells.} \\
(4.3) & \quad \text{Each} \quad x \quad \text{in} \quad M \quad \text{lies in a finite number of n-cells in} \quad m .
\end{align*}

If \((M, m)\) is a subdivided \( n \)-manifold for some \( m \), we call \( M \) a \( n \)-manifold.

Piecewise Linear Functions

Let \( M \) be a \( n \)-manifold, then the function

\[ g : M \rightarrow \mathbb{R} \]

is called piecewise linear on a subdivision \( m \) of \( M \) if

\begin{align*}
(4.4) & \quad g \quad \text{is continuous}
\end{align*}
(4.5) Given a cell $\sigma$ in $m$, there exists an affine function $g_0 : \mathbb{R}^n \to \mathbb{R}$ such that $g|\sigma(x) = g_0(x)$ (i.e., $g$ restricted to $\sigma$ is $g_0$).

**Generalized Subdifferentials**

Let $M$ be a $n$-manifold, and $m$ be its subdivision. Let $g : M \to \mathbb{R}$ be a piecewise linear function. Then, for each $x \in M$, we define a generalized subdifferential set $\partial g(x)$ as follows:

From (4.3), $x$ lies in a finite number of $n$-cells, $\sigma_1, \sigma_2, \ldots, \sigma_r$ in $m$ say. Let

$$\nabla g_{\sigma_1} = a_1$$

(where $\nabla$ is the gradient vector of $f$). Then, we define

$$\partial g(x) = \hull\{a_1, \ldots, a_r\}$$

and we note that if $g$, in addition, is convex, then $\partial g(x)$ is the subdifferential of $g$ at $x$. Rockafellar [5]; and, as $g$ is locally Lipschitz continuous, $\partial g(x)$ is the generalized gradient of Clarke [1]. In that case, the theorem below is known, but we will use the piecewise linearity of $g$ to establish it.

**Theorem 4.1**: If $x$ is a local minimum of $g_0$, then $0 \in \partial g_0(x)$.

**Proof:**

Assume $\overline{x}$ is a local minimum but $0 \notin \partial g_0(\overline{x})$. Now, let $\overline{x} \in \sigma_1 \cap \ldots \cap \sigma_r$. Then $\partial g_0(\overline{x}) = \hull\{a_1, \ldots, a_r\}$. Hence, from Farkas' lemma, there is a $\xi \neq 0$ such that $\langle \xi, a_i \rangle > 0$ for $i = 1, \ldots, r$. Let $\varepsilon > 0$ be sufficiently small so that $B(\overline{x}, \varepsilon) \subseteq \bigcup_{i=1}^r$. Then $\overline{x} + \varepsilon \in B(\overline{x}, \varepsilon)$ for sufficiently small $\varepsilon > 0$. Assume $\overline{x} + \varepsilon \in \sigma_j$ for some $j$. Hence

$$g_0(\overline{x} + \varepsilon) = \langle a_j, \overline{x} + \varepsilon \rangle - \gamma_j$$

$$= g_0(\overline{x}) + \varepsilon < a_j, \overline{x} > - \gamma_j$$

Hence

$$g_0(\overline{x} + \varepsilon) < g_0(\overline{x})$$

which contradicts the assumption that $\overline{x}$ is a local minimum. Therefore, $0 \in \partial g_0(\overline{x})$. This completes the proof.
and we have a contradiction to the fact that $\bar{x}$ is a local minimum.

Given a point-to-set mapping $\Gamma$ from $\mathbb{R}^n$ to nonempty subsets of $\mathbb{R}^n$, we say $\Gamma$ is weakly monotone at $\bar{x}$ with respect to $\tilde{y} \in \Gamma(\bar{x})$ on $\mathcal{F}$ if there is an $\epsilon > 0$ such that for all $x \in B(\bar{x}, \epsilon) \cap \mathcal{F}$

$$<\bar{x} - x, \tilde{y} - y> \geq 0$$

for all $y$ in $\Gamma(x)$. We can then prove:

**Theorem 4.4:** $\bar{x}$ is a local minimum of $g_0$ if and only if $0 \in \partial g_0(\bar{x})$ and $\partial g_0$ is weakly monotone at $\bar{x}$ with respect to zero on $\mathbb{R}^n$.

**Proof:**

Let $\bar{x}$ lie in the cells $c_1, c_2, \ldots, c_n$. Then $\partial g_0(\bar{x}) = \text{hull} \{a_1, \ldots, a_n\}$. For some sufficiently small $\epsilon > 0$, let $B(\bar{x}, \epsilon) \subset \bigcup c_i$.

To see the first part, let $0 \in \partial g_0(\bar{x})$ and let $\partial g_0(\bar{x})$ be weakly monotone with respect to zero at $\bar{x}$. Hence, for some $\epsilon > 0$, for all $x \in B(\bar{x}, \epsilon)$ we have

$$<\bar{x} - x, a_i> \geq 0 \quad \text{where} \quad a_i \in \partial g_0(\bar{x}) \subset \partial g_0(\bar{x})$$

Since,

$$g_0(x) - g_0(\bar{x}) = <a_i, x - \bar{x}> - \gamma_i <a_i, \bar{x}> + \gamma_i \geq 0,$$

and so $\bar{x}$ is a local minimum of $g_0$. To see the second part, let $0 \in \partial g_0(\bar{x})$ and $\partial g_0(\bar{x})$ not weakly monotone with respect to zero. Then, for a sufficiently small $\epsilon > 0$ such that $B(\bar{x}, \epsilon) \subset \bigcup c_i$, there is an $x \in B(\bar{x}, \epsilon)$ and a $a_i \in \partial g_0(\bar{x})$ such that $<\bar{x} - x, a_i> < 0$. Since $\partial g_0(\bar{x}) \subset \partial g_0(\bar{x})$, we have

$$g_0(x) - g_0(\bar{x}) = <a_i, x - \bar{x}> - \gamma_i <a_i, \bar{x}> + \gamma_i$$

$$= <\bar{x} - x, a_i> \geq 0$$

which is a contradiction.
The Constrained Problem

Let $\mathbf{x}_j : \mathbb{R}^n \to \mathbb{R}$ be piecewise linear functions on subdivided manifolds $\mathbb{M}_j$, respectively, for each $i = 0, \ldots, m$. We now consider the constrained minimization problem (1.1-2).

For a generic point $\mathbf{x}$ in $\mathbb{R}^n$ we define $\mathbf{e}_j^1, \mathbf{e}_j^2, \ldots, \mathbf{e}_j^m$ as the $m$-cells of $\mathbb{M}_j$ in which $\mathbf{x}$ lies, and $g_j \mid_{\mathbf{e}_j^i}(\mathbf{y}) = \langle a_j^{i,1}, \mathbf{y} \rangle - \gamma_j^{i,1}$, for each $i = 0, \ldots, m$. Also note that, by definition, $r_j^i$ is finite. Also, there exists $\varepsilon > 0$ such that $\mathbb{R}(\mathbf{x}, \varepsilon) \subseteq \bigcup_{j=1}^m \mathbf{e}_j^i$ for each $i$. We are now ready to establish the necessary conditions for $\mathbf{x}$ to be a local minimum of (1.1-2).

Theorem 4.3: Let $\mathbf{x}$ be a local minimum of $F_0$ over all $\mathbf{x}$ satisfying (1.2). Then

(i) There exists $\lambda_j^i > 0$ such that $\lambda_j^i g_j^i(\mathbf{x}) = 0, i = 1, \ldots, m$.

(ii) There exists $\mathbf{y}^* \in \partial F_0(\mathbf{x}), \mathbf{z}^* \in \partial F_0(\mathbf{x})$ such that $m \mathbf{y}^* + \sum_{i=1}^m \lambda_j^i \mathbf{z}^* = 0$.

Proof:

Let $\mathbf{x}$ be a local minimum, and $0 \notin \partial F_0(\mathbf{x}) + \text{cone}(C)$ where

$$C = \bigcup_{i=1}^m \partial g_i^i(\mathbf{x}), \text{cone}(C) = \{ \mathbf{y} : \mathbf{y} = \sum_{i=1}^m \mathbf{x}_i, \mathbf{x}_i \in C, \lambda_i \geq 0 \}. \text{ (It can be readily confirmed that (i) and (ii) hold if and only if } 0 \notin \partial F_0(\mathbf{x}) + \text{cone}(C)\text{). Then, from Farkas lemma, since both } \partial g_0^0(\mathbf{x}) \text{ and } C \text{ are convex combinations of a finite number of vectors, there exists a } \mathbf{z} \text{ such that }$$

$$\langle \mathbf{z}, \mathbf{y} \rangle < 0 \text{ for all } \mathbf{y} \in \partial g_0^0(\mathbf{x})$$

$$\langle \mathbf{z}, \mathbf{y} \rangle 

\leq 0 \text{ for all } \mathbf{y} \in C.$$
Now, consider $x = \bar{x} + \theta \epsilon$ for sufficiently small $\theta > 0$ such that for $i \leq I(\bar{x})$, $g_i(x) < 0$, and $x \in B(\bar{x}, \epsilon)$. Hence, for some $a_i^0 \in \partial g_i(\bar{x})$, $a_i^0 < g_i(x)$. Hence $g_0(x) - g_0(\bar{x}) = a_i^0 x_i + \gamma_i^0 < 0$. Also, for $i \in I(\bar{x})$, there is a $a_j^i \in \partial g_j(\bar{x})$ such that $a_j^i < g_j(x)$, hence $g_j(x) - g_j(\bar{x}) = a_j^i x_i + \gamma_j^i < 0$. Since $g_i(\bar{x}) > 0$, we get a contradiction that $\bar{x}$ is not a local minimum.

We now prove a sufficiency condition.

**Theorem 4.1**: Let $\bar{x}$ be a point such that

(i) There exist $\lambda_i \geq 0$ for which

\[ \lambda_i \partial g_i(\bar{x}) = 0 \quad i = 1, \ldots, m. \]

(ii) Define the map $\Gamma(x) = \partial g_0(x) + \sum_{i=1}^m \lambda_i \partial g_i(x)$. Then $0 \in \Gamma(\bar{x})$.

(iii) If $\Gamma(x)$ is weakly monotone at $\bar{x}$ with respect to $0$ on the set $F = \{x : g_i(x) \leq 0, i = 1, \ldots, m\}$.

Then $\bar{x}$ is a local minimum of $g_0$ on $F$.

**Proof**:

Let $x \in F(\bar{x}, \epsilon) \cap E$, and $\epsilon$ sufficiently small so that $B(\bar{x}, \epsilon) \subseteq \bigcup_{j=1}^m E_j$. Then

\[ g_0(x) - g_0(\bar{x}) = \sum_{i=1}^m \lambda_i g_i(x) - g_0(\bar{x}) = \sum_{j=1}^m a_j^i x_i + \sum_{j=1}^m \lambda_j a_j^i \cdot \epsilon \cdot x_i \cdot a_j^i \]

\[ < 0 \]

since $a_j^i + \sum_{j=1}^m \lambda_j a_j^i = \Gamma(x)$, and $\Gamma(x)$ is weakly monotone with respect to $0$ at $x$, and so $\bar{x}$ is a local minimum.
5. The Fixed Point Approach to PL Nonlinear Programming

We will consider the application of the fixed point algorithm of [3] to the case where $g_0$ is piecewise linear on some subdivision of $R^n$, and $g_q$ are convex functions. In this case, the mapping $\lambda, (3.4)$, is applicable. As is evident from Theorem 3.5, in this case we will prove that the algorithms will find a "stationary point" $\bar{x}$ such that $0 \in \lambda(\bar{x})$. In certain special cases, the progression of the algorithm will indicate if $\bar{x}$ is a local minimum, Sari and Saigal [10], for the general case considered here, $\bar{x}$ may be a local maximum or a saddle point of the function $g_q$.

That the algorithm will compute a stationary point can be established in a manner similar to the proof of Theorem 3.5. Since $s$ is convex, the part of Theorem 3.4 pertaining to the nonexistence of a solution to (1.2) also carries through. The convergence of the algorithms can also be proved under the relaxed hypothesis that $s$ be convex outside some bounded region; i.e., if $D$ is a bounded set containing $x_0$, then for each $x \notin D$, and $y \neq \lambda s(x)$,

$$s(x) = s(x) + <y, x-x> \quad \text{for all } z \notin D.$$ 

Then, starting with $r(x) = x-x_0$ and $z$ as defined in (3.4) we can prove:

Theorem 5.1: For any $\epsilon > 0$, starting with the unique $\epsilon$-complete simplex containing $x_0$, the fixed point algorithm will generate $\lambda$-complete simplices of size $< \epsilon(\epsilon > 0)$, and thus will compute a stationary point of (1.1-2).
Proof:

Let \( N = \max \{0, \sup \{s(x) : x \in B(x_1, \epsilon) \cup \overline{D}\}\} \) for an arbitrary \( x_1 \) such that \( B(x_1, \epsilon) \cap \overline{D} = \emptyset \) and let \( \overline{D} = D \cup \{x : s(x) \leq N\} \). By assumption \( \overline{D} \) is bounded, and \( D \) is convex outside \( \overline{D} \). Now, assume that for some \( \epsilon > 0 \) the algorithm fails to compute a \( \epsilon \)-complete simplex. Then, there is a \( \epsilon \)-complete simplex of diameter \( < \epsilon \) sufficiently far from \( \overline{D} \); i.e., \( \sigma \not\subset \overline{D} \) and for every \( x \in \sigma \), \( <v_i^\perp - x, x_1^\perp > \geq \delta \). Also, \( s(v_i^\perp) > 0 \) for all \( i \). Now consider the point \( v_i^\perp - x + x_1 \in \overline{D} \). Also, by assumption, \( v_i^\perp - x + x_1 \in D \). Hence \( s(v_i^\perp - x + x_1) \geq s(v_i^\perp) + <x_1^\perp - x, y_i^\perp > \) for all \( y_i^\perp \in \partial s(v_i^\perp) \) and as \( v_i^\perp \not\subset \overline{D} \) and \( v_i^\perp - x + x_1 \in \overline{D} \), we have \( <x_1^\perp, y_i^\perp > > 0 \) for all \( y_i^\perp \in \partial s(v_i^\perp) \); for all \( i \). Thus \( \sigma \) is not \( \epsilon \)-complete, a contradiction.
6. Computational Considerations

As is evident from the sections 3 and 5, the convergence of the fixed point algorithms can be established under some general conditions on the problem, and differentiability is not necessary. Computational experience indicates that the computation burden increases when the underlying mappings are not smooth. For smooth mappings, under the usual conditions, the fixed point algorithms can be made to converge quadratically, Saigal [14]. This can be observed by comparing the solution of three nondifferentiable nonlinear programming problems implementing the mapping (3.4) presented in the appendix, Tables A.1-3, with the solution of a smooth problem of eighty variables in Table A.4.

On such a problem, reported in Metravali and Saigal [8], the growth of the number of function evaluations with the number of variables was tested. The results were as anticipated by the works of Saigal [11] and Todd [16]. It was predicted in these works that the function evaluations grow as \( \Theta(n^2) \), where \( n \) is the number of variables. (See [8, 4.1].)
REFERENCES


APPENDIX

We now give some computational experience with solving non-differentiable optimization problems of fairly large number of variables. For comparison purposes, we also give the results of solving an eighty-variable smooth problem (where the convergence has been accelerated).

Problem 1

This is a 7 variable problem. It is a version of the problem considered by Nattrass and Saigal [8]. The value of entropy on the entropy constraint is 2.7, and this is the 19th run in the series of runs done on this problem.

Problem 2

This is a 43 variable problem considered in Elsner, W.B., "A Descent Algorithm for the Multihour Sizing of Traffic Networks," Bell System Technical Journal, 56 (1977), 1405-1430.

This run was made on a piecewise linear version, while the problem formulated by Elsner was piecewise smooth. The function is convex.

Problem 3

This is the following 15 variable convex piecewise smooth problem:

$$\min_{x} \max_{i=1,2,5} f_i(x)$$
where

\[ f_j(x) = \sum_{i=1}^{5} \prod_{j=1}^{5} a_{ij} x_i \]

\[ j = 1, \ldots, 5 \]

and the data \( a_{ij}, c_{ij}, d_{ij}, e_j \) are the same as that for problem 10 in the appendix of Himmelblau, D.M., Applied Nonlinear Programming, McGraw-Hill Book Company, 1972.

Problem 4


The results of the above four problems are summarized in Tables A.1-4, respectively.
<table>
<thead>
<tr>
<th>Grid of Search</th>
<th>Number of Function Evaluations</th>
<th>Number of Simplexes Searched</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0</td>
<td>34</td>
<td>121</td>
</tr>
<tr>
<td>4.0</td>
<td>111</td>
<td>160</td>
</tr>
<tr>
<td>2.0</td>
<td>28</td>
<td>34</td>
</tr>
<tr>
<td>1.0</td>
<td>286</td>
<td>690</td>
</tr>
<tr>
<td>0.5</td>
<td>539</td>
<td>1,034</td>
</tr>
<tr>
<td>0.25</td>
<td>487</td>
<td>918</td>
</tr>
</tbody>
</table>

Constrained minimization problem with piecewise linear objective function and one piecewise linear constraint in seven variables.
<table>
<thead>
<tr>
<th>Grid of Search</th>
<th>Number of Function Evaluations</th>
<th>Number of Simplexes Searched</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.55</td>
<td>951</td>
<td>1,431</td>
</tr>
<tr>
<td>3.27</td>
<td>469</td>
<td>469</td>
</tr>
<tr>
<td>1.65</td>
<td>400</td>
<td>400</td>
</tr>
<tr>
<td>0.81</td>
<td>526</td>
<td>526</td>
</tr>
<tr>
<td>0.42</td>
<td>623</td>
<td>623</td>
</tr>
<tr>
<td>0.20</td>
<td>1,081</td>
<td>1,081</td>
</tr>
<tr>
<td>0.16</td>
<td>885</td>
<td>885</td>
</tr>
</tbody>
</table>

Unconstrained minimization of a piecewise linear convex function of 43 variables.
<table>
<thead>
<tr>
<th>Grid of Search</th>
<th>Number of Function Evaluations</th>
<th>Number of Simplexes Searched</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.87</td>
<td>2,759</td>
<td>3,099*</td>
</tr>
<tr>
<td>1.94</td>
<td>381</td>
<td>381</td>
</tr>
<tr>
<td>0.97</td>
<td>244</td>
<td>244</td>
</tr>
<tr>
<td>0.48</td>
<td>667</td>
<td>667</td>
</tr>
<tr>
<td>0.24</td>
<td>347</td>
<td>347</td>
</tr>
<tr>
<td>0.12</td>
<td>253</td>
<td>253</td>
</tr>
<tr>
<td>0.06</td>
<td>364</td>
<td>364</td>
</tr>
</tbody>
</table>

Unconstrained minimization of a piecewise smooth convex function of 15 variables.
<table>
<thead>
<tr>
<th>Grid of Search</th>
<th>Number of Function Evaluations</th>
<th>Number of Simplexes Searched</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.74</td>
<td>1,573</td>
<td>1,750</td>
</tr>
<tr>
<td>4.47</td>
<td>149</td>
<td>149</td>
</tr>
<tr>
<td>2.24</td>
<td>106</td>
<td>106</td>
</tr>
<tr>
<td>1.18</td>
<td>57</td>
<td>97</td>
</tr>
<tr>
<td>0.56</td>
<td>88</td>
<td>88</td>
</tr>
<tr>
<td>0.24</td>
<td>84</td>
<td>84</td>
</tr>
<tr>
<td>0.14</td>
<td>81</td>
<td>81</td>
</tr>
<tr>
<td>0.03</td>
<td>84</td>
<td>84</td>
</tr>
<tr>
<td>0.002</td>
<td>81</td>
<td>81</td>
</tr>
<tr>
<td>0.000009</td>
<td>82</td>
<td>82</td>
</tr>
</tbody>
</table>

Zero finding problem for a smooth function of 80 variables. The accelerated algorithm has been used to solve this problem.