NON-DIFFERENTIABLE OPTIMIZATION VIA ADAPTIVE SMOOTHING

by

D.Q. Mayne and E. Polak

Memorandum No. UCB/ERL M82/87

12 November 1982

ELECTRONICS RESEARCH LABORATORY

College of Engineering
University of California, Berkeley
94720

NON-DIFFERENTIABLE OPTIMIZATION VIA ADAPTIVE SMOOTHING[1]

D Q Mayne[2] and E Polak[3]

ABSTRACT

The problem of minimizing a non-differentiable function $x \mapsto f(x)$ (subject, possibly, to non-differentiable constraints $g^j(x) \leq 0$) is considered. Conventional algorithms are employed for minimizing a differentiable approximation f_ε of f (subject to differentiable approximations of g^j). The parameter ε is adaptively reduced in such a way as to ensure convergence to points satisfying necessary conditions of optimality for the original problem.

---

1 Research supported by the UK Science and Engineering Research Council, the National Science Foundation under grant No.ECS-8121149 and the Joint Services Electronics Program, contract No. F49620-79-C-0178.

2 D Q Mayne is with the Department of Electrical Engineering, Imperial College, London, SW7 2BT, UK.

3 E Polak is with the Department of Electrical Engineering and Computer Science, University of California, Berkeley, CA 94720, USA.

Original: 10.6.82
First Draft: 9.8.82
Second Draft: 26.10.82

1. INTRODUCTION

The development of a calculus for locally Lipschitz continuous functions [1] has been accompanied by a variety of algorithms [2-7] for non-differentiable optimization. If $f : \mathbb{R}^n \to \mathbb{R}$ is locally Lipschitz continuous, it possesses at each x in $\mathbb{R}^n$ a generalised gradient $\partial f(x)$ which is a convex compact subset of $\mathbb{R}^n$. A suitable candidate for a search direction $s(x)$ for the problem of minimizing $f(x)$ on $\mathbb{R}^n$ might appear to be $-g(x)$ where $g(x) \triangleq \operatorname{argmin}\{\|g\| \mid g \in \partial f(x)\}$ since this yields $\langle s(x), g\rangle \leq -\|g(x)\|^2$ for all $g \in \partial f(x)$. Clearly $\|g(x)\|^2 > 0$ if $0 \notin \partial f(x)$. However, such a search direction lacks the continuity properties necessary for convergence and can lead to jamming. Existing algorithms therefore employ a bundle of generalised gradients constructed by exploring an ε-neighbourhood of x. When f is semi-smooth [8] a suitable approximation to this bundle can be obtained by using a special line exploration technique. These bundles have the necessary continuity; algorithms utilising these bundles must, of course, include a procedure for reducing ε to zero in order to ensure that any accumulation point x^* generated by the algorithm satisfies the necessary condition of optimality $0 \in \partial f(x^*)$.

When f is not semi-smooth the computational cost involved in calculating the bundle of generalised gradients is considerable. This paper therefore presents an alternative approach which, it is hoped, will be of use in such situations. A non-differentiable function f is approximated by a differentiable function f_ε which converges to f as ε tends to zero. Conventional algorithms, such as steepest descent or conjugate gradient, can then be employed for minimizing f_ε. A procedure for reducing ε to zero completes the algorithm. A similar approach can be employed for constrained optimization.

Our approach, for unconstrained optimization, is easily illustrated for the simple case when n = 1. Then, for all $\varepsilon > 0$, f_ε is defined by:

$$f_\varepsilon(x) \triangleq (1/2\varepsilon) \int_{x-\varepsilon}^{x+\varepsilon} f(x')dx'. \tag{1.1}$$

It is clear that f_ε is continuously differentiable, its gradient being:

$$\nabla f_\varepsilon(x) = [f(x + \varepsilon) - f(x - \varepsilon)]/2\varepsilon \tag{1.2}$$

which is, of course, an approximation to $\nabla f(x)$ when f is differentiable. To construct an algorithm we require two sequences $\{\varepsilon_i\}$ and $\{\gamma_i\}$ such that $\varepsilon_i \searrow 0$ and $\gamma_i \searrow 0$ as $i \to \infty$. At iteration i the algorithm utilises a standard minimization algorithm (using x_{i-1} as its initial point) to compute an x_i satisfying $\| \nabla f_{\varepsilon_i}(x_i) \| \leq \gamma_i$. Such an x_i can be determined in a finite number of iterations. Hence $\nabla f_{\varepsilon_i}(x_i) \to 0$ and $\varepsilon_i \to 0$ as $i \to \infty$. We prove that any accumulation point $\{x^*\}$ of $\{x_i\}$ satisfies $0 \in \partial f(x^*)$. We also show how the approach may be employed for constrained optimization.

The paper is organized as follows. In Section 2 the approximating function f_ε is defined and some elementary properties established. In Sections 3 and 4 the algorithms are stated and convergence (in the above sense) proven. Computational considerations are discussed in Section 5.

2. THE APPROXIMATING FUNCTION

Let $f : \mathbb{R}^n \to \mathbb{R}$ be locally Lipschitz continuous. Let P denote the unconstrained optimization problem $\min \{f(x) \mid x \in \mathbb{R}^n\}$. For all $\varepsilon > 0$, $x \in \mathbb{R}^n$ let $N_\varepsilon(x)$ denote the set $\{x' \in \mathbb{R}^n \mid \| x' - x \|_\infty \leq \varepsilon\}$. For all $\varepsilon > 0$ the approximating function $f_\varepsilon : \mathbb{R}^n \to \mathbb{R}$ is defined by

$$f_\varepsilon(x) \triangleq a(\varepsilon) \int_{N_\varepsilon(x)} f(x')dx' \tag{2.1}$$

where the 'normalising' constant $a(\varepsilon)$ is the reciprocal of the volume of $N_\varepsilon(0)$, i.e.

$$a(\varepsilon) = \left[\int_{N_\varepsilon(0)} dx \right]^{-1} = 1/(2\varepsilon)^n. \tag{2.2}$$

Our first result concerns the differentiability of f_ε.

Proposition 1 For all $\varepsilon > 0$, f_ε is continuously differentiable. When n is greater than unity, the gradient ∇f_ε of f_ε is given by:

$$[\nabla f_\varepsilon(x)]^i = a(\varepsilon) \left[\int_{D^i_{\varepsilon+}(x)} f(x')dx' - \int_{D^i_{\varepsilon-}(x)} f(x')dx' \right] \tag{2.3}$$

for all $i \in \{1, \ldots, n\}$ where

$$\begin{aligned} D^i_{\varepsilon+}(x) &\triangleq \{y \in \mathbb{R}^n \,|\, \|y - x\|_\infty = \varepsilon;\ y^i = x^i + \varepsilon\} \\ &= \{x + s \,|\, s^i = \varepsilon; |s^j| \le \varepsilon,\ j \ne i\} \end{aligned} \tag{2.4}$$

and where

$$\begin{aligned} D^i_{\varepsilon-}(x) &\triangleq \{y \in \mathbb{R}^n \,|\, \|x - y\|_\infty = \varepsilon;\ y^i = x^i - \varepsilon\} \\ &= \{x + s \,|\, s^i = -\varepsilon; |s^j| \le \varepsilon,\ j \ne i\}. \end{aligned} \tag{2.5}$$

The proof of this result is elementary and is, therefore, omitted. $D^i_{\varepsilon+}(x)$, $D^i_{\varepsilon-}(x)$ are parallel faces of the ε-cube centered at x. They are perpendicular to the i^{th} standard basis vector.

We explore next the relationship between ∇f_ε and ∂f. We need to ensure that the algorithm does not jump up at a non-optimal point for P.

We require the following definitions. For all $\varepsilon > 0$ let $\partial_\varepsilon f(x)$ denote the 'smeared' generalised gradient of f at x, i.e. $\partial_\varepsilon f(x)$ is defined as the convex hull of the set $\{\partial f(x') \mid x' \in N_\varepsilon(x)\}$. Also for all $x, h \in \mathbb{R}^n$, $df_\varepsilon(x; h)$ denotes the directional derivative of f_ε at x in the direction h.

Proposition 2 For all $\varepsilon > 0$, $x \in \mathbb{R}^n$

$$\nabla f_\varepsilon(x) \in \partial_{2\varepsilon} f(x).$$

Proof For all h in $\mathbb{R}^n$, $\|h\| = 1$

$$df_\varepsilon(x; h) = \langle \nabla f_\varepsilon(x), h \rangle$$

$$= \lim_{\lambda \downarrow 0} a(\varepsilon) \int_{N_\varepsilon(0)} \frac{f(x+s+\lambda h) - f(x+s)}{\lambda} \, ds$$

$$= \lim_{\lambda \downarrow 0} a(\varepsilon) \int_{N_\varepsilon(0)} \langle g(s, \lambda), h \rangle \, ds$$

where, by Lebourg's mean value theorem [9], $g(s, \lambda) \in \partial f(x + s + a\lambda h)$ for some $a \in [0, 1]$. Hence $g(s, \lambda) \in \partial_{2\varepsilon} f(x)$ for all $s \in N_\varepsilon(0)$, all $\lambda \in [0, \varepsilon]$ so that

$$\nabla f_\varepsilon(x) \in \partial_{2\varepsilon} f(x). \qquad \square$$

Corollary Suppose $0 \notin \partial f(x)$. Then there exists an $\varepsilon > 0$ such that $\nabla f_{\varepsilon'}(x) \neq 0$ for all $\varepsilon' \in (0, \varepsilon]$.

Proof Since $0 \notin \partial f(x)$ there exists an $\varepsilon > 0$ such that $0 \notin \partial_{2\varepsilon'} f(x)$ for all $\varepsilon' \in (0, 2\varepsilon]$. Since $\nabla f_{\varepsilon'}(x) \in \partial_{2\varepsilon'} f(x)$ it follows that $\nabla f_{\varepsilon'}(x) \neq 0$.

$\square$

Since the algorithm reduces ε to zero it follows that any non-desirable point x ($0 \notin \partial f(x)$) will eventually be detected.

3. UNCONSTRAINED OPTIMIZATION

If, for each x and ε, $f_\varepsilon(x)$ and $\nabla f_\varepsilon(x)$ can be exactly computed, then a suitable algorithm for solving the unconstrained optimization problem P is:

Algorithm 1 (for unconstrained minimization)

Data: $x_0 \in \mathbb{R}^n$; sequences $\{\varepsilon_i\}$, $\{\gamma_i\}$ satisfying $\varepsilon_i \searrow 0$, $\gamma_i \searrow 0$ as $\varepsilon \to \infty$.

Step 0: Set $i = 0$.

Step 1: Compute x_i such that

$$\| \nabla f_{\varepsilon_i}(x_i) \| \leq \gamma_i .$$

Step 2: Set $i = i + 1$. Go to Step 1. □

Any convergent algorithm (i.e. one producing limit points satisfying $\nabla f_{\varepsilon_i}(x) = 0$) may be employed in Step 1. The convergence properties of Algorithm 1 are established in

Theorem 1

Any limit point $\hat{x}$ of an infinite sequence $\{x_i\}$ generated by Algorithm 1 is desirable, i.e. $\hat{x}$ satisfies $0 \in \partial f(\hat{x})$. □

The proof of this theorem requires the following two ancilliary results.

Proposition 4

The map $(\varepsilon, x) \mapsto \partial_\varepsilon f(x)$ is upper semi-continuous at any $(0, x)$.

Proof Let $\delta > 0$ be given. Because ∂f is upper semi-continuous, there exists an $\bar{\varepsilon} > 0$ such that $\partial f(x') \subset N_\delta[\partial f(x)] \triangleq \{y \mid \| y - g\|_\infty \leq \delta$ for some $g \in \partial f(x)\}$ for all $x' \in N_{2\bar{\varepsilon}}(x)$. Now

$$\partial_{\bar{\varepsilon}} f(x') = \text{co}\{\partial f(x'') \mid x'' \in N_{\bar{\varepsilon}}(x')\}$$

for all $x' \in N_{\bar{\varepsilon}}(x)$. Thus $\partial_\varepsilon f(x') \subset N_\delta[\delta f(x)]$ for all $x' \in N_{\bar{\varepsilon}}(x)$ all $\varepsilon \in [0, \bar{\varepsilon}]$. □

The following result holds because $\partial f(\hat{x})$ is compact.

Proposition 5

If $0 \in N_\delta[\partial f(\hat{x})]$ for all $\delta > 0$ then $0 \in \partial f(\hat{x})$. □

Proof of Theorem 1

Suppose $x_i \to \hat{x}$ as $i \to \infty$, $i \in K$. Let $g_i \triangleq \nabla f_{\varepsilon_i}(x_i)$, $i = 0, 1, 2 \ldots$. From Proposition 3, $g_i \in \partial_{2\varepsilon_i} f(x_i)$ for all i. From the upper semi-continuity of $(\varepsilon, x) \mapsto \partial_\varepsilon f(x)$ at $(0, \hat{x})$, for all $\delta > 0$ there exists an integer i_δ such that $g_i \in N_\delta[(\partial f(\hat{x}))]$ for all $i \geq i_\delta$, $i \in K$. From Step 1 of the algorithm, $g_i \to 0$ as $i \to \infty$. Hence for all $\delta > 0$, $0 \in N_\delta[\partial f(\hat{x})]$. By Proposition 5, $0 \in \partial f(\hat{x})$.

4. CONSTRAINED OPTIMIZATION

Consider the constrained optimization problem:

$$P_c : \min\{f(x) \,|\, \psi(x) \leq 0\} \tag{4.1}$$

where $f : \mathbb{R}^n \to \mathbb{R}$ is locally Lipschitz continuous and $\psi : \mathbb{R}^n \to \mathbb{R}$ is defined by:

$$\psi(x) \triangleq \max\{g^j(x) \,|\, j \in \underline{m}\} \tag{4.2}$$

where $g^j : \mathbb{R}^n \to \mathbb{R}$ is locally Lipschitz continuous, $j = 1,\ldots,m$, and $\underline{m}$ denotes the set $\{1,\ldots,m\}$. It is easily shown that ψ is locally Lipschitz continuous. A well known necessary condition of optimality for P_c is

$$\psi(\hat{x}) \leq 0, \; 0 \in \mathrm{co}\{\partial f(x), \partial \psi(x)\}. \tag{4.3}$$

An alternative method of expressing this is to employ an "optimality function" $\theta : \mathbb{R}^n \to \mathbb{R}$ defined by

$$\theta(x) \triangleq - \| h(x) \|^2 \tag{4.4}$$

where $h : \mathbb{R}^n \to \mathbb{R}$ is defined by

$$h(x) \triangleq Nr[A(x)], \tag{4.5}$$

$$A(x) \triangleq co\{\partial f(x), \partial\psi(x)\}, \tag{4.6}$$

and where Nr[A] denotes that point in set A which is closest (in the Euclidean sense) to the origin. It is clear that $\theta(x) \leq 0$ for all x in $\mathbb{R}^n$ and in zero if and only if the origin lies in A(x), thus satisfying the second condition in (4.3).

Let f_ε and $\psi_\varepsilon : \mathbb{R}^n \to \mathbb{R}$ denote the smoothed versions of f and ψ respectively, defined as in (2.1). It follows that f_ε and ψ_ε are continuously differentiable and that:

$$\nabla f_\varepsilon(x) \subset \partial_{2\varepsilon} f(x), \quad \nabla\psi_\varepsilon(x) \subset \partial_{2\varepsilon}\psi(x) \tag{4.7}$$

for all x in $\mathbb{R}^n$ and all positive ε. As for the unconstrained case we replace the hard problem (P_c) by an infinite sequence $\{P_c^{\varepsilon_i}\}$ of easy (smooth) problems which are approximately solved. The sequence $\{\varepsilon_i\}$ is such that $\varepsilon_i \searrow 0$ as $i \to \infty$ and for each ε, P_c^ε is defined by

$$P_c^{\varepsilon} : \min\{f_{\varepsilon}(x) \,|\, \psi_{\varepsilon}(x) \leq 0\}. \tag{4.8}$$

A necessary condition of optimality for P_c^{ε} is :

$$\psi_{\varepsilon}(\hat{x}) \leq 0,\ 0 \in \text{co}\{\nabla f_{\varepsilon}(\hat{x}),\ \nabla\psi_{\varepsilon}(\hat{x})\}. \tag{4.9}$$

For all x in $\mathbb{R}^n$, all positive ε let $A_{\varepsilon}(x)$ denote $\text{co}\{\nabla f_{\varepsilon}(x),\ \nabla\psi_{\varepsilon}(x)\}$ and let $h_{\varepsilon}(x)$ denote $\text{Nr}[A_{\varepsilon}(x)]$. Then $\theta_{\varepsilon} : \mathbb{R}^n \to \mathbb{R}$ defined by:

$$\theta_{\varepsilon}(x) \triangleq -\| h_{\varepsilon}(x) \|^2 \tag{4.10}$$

is non-positive and is zero if and only if the origin lies in $A_{\varepsilon}(x)$. Hence the necessary condition of optimality for P_c^{ε} may be expressed as

$$\psi_{\varepsilon}(\hat{x}) = 0,\ \theta_{\varepsilon}(\hat{x}) = 0. \tag{4.11}$$

Problem $P_{\varepsilon}^{\varepsilon}$ will be said to be approximately solved if $\psi_{\varepsilon}(\hat{x}) \leq \gamma$ and $\theta_{\varepsilon}(\hat{x}) \geq -\gamma$ where γ is a small positive number. A conventional algorithm for solving P_c^{ε} will compute such an $\hat{x}$ in a finite number of iterations.

Our algorithm for solving P_c can now be stated.

Algorithm 2 (for constrained minimization)

Data: $x_0 \in \mathbb{R}^n$, sequences $\{\varepsilon_i\}$ and $\{\gamma_i\}$ satisfying $\varepsilon_i \searrow 0$, $\gamma_i \searrow 0$ as $i \to \infty$.

Step 0: Set $i = 0$.

Step 1: Compute x_i such that

$$\psi_\varepsilon(x_i) \leq \gamma_i, \ \theta_\varepsilon(x_i) \geq -\gamma_i.$$

Step 2: Set $i = i + 1$. Go to Step 1. □

The convergence properties of this algorithm are given by

Theorem 2

Any limit point $\hat{x}$ of an infinite sequence $\{x_i\}$ generated by Algorithm 2 satisfies $\psi(\hat{x}) \leq 0$, $\theta(\hat{x}) = 0$.

Proof

Suppose $x_i \xrightarrow{K} \hat{x}$ where K is an appropriate subsequence of $\{0, 1, 2, \ldots\}$. It follows that $h_{\varepsilon_i}(x_i)$ lies in $A_{\varepsilon_i}(x_i)$ and, therefore, in $\text{co}\{\partial_{2\varepsilon_i} f(x_i), \partial_{2\varepsilon_i} \psi(x_i)\}$ for all i. From the upper semi-continuity of $(\varepsilon, x) \mapsto \text{co}\{\partial_\varepsilon f(x), \partial_\varepsilon \psi(x)\}$ at $(0, \hat{x})$ it follows that for all $\delta > 0$ there exists a integer i_δ such that $h_{\varepsilon_i}(x_i) \in N_\delta[A(\hat{x})]$ for all $i \geq i_\delta$, $i \in K$. Since $\theta_{\varepsilon_i}(x_i) \to 0$ it follows that $h_{\varepsilon_i}(x_i) \to 0$ as $i \to \infty$ and, hence, that $0 \in A(\hat{x})$ i.e. $\theta(\hat{x}) = 0$. It is easily established that $\psi(\hat{x}) \leq 0$.

□

5. IMPLEMENTABLE ALGORITHMS

Algorithm 1 and 2 are very simple. However, they suffer from the severe disadvantage of requiring the evaluation of multidimensional integrals to obtain $f_\varepsilon(x)$ and $\nabla f_\varepsilon(x)$. Any practical algorithm can only compute approximations to these quantities. To implement Step 1 of Algorithm 1, for example, we need therefore a subalgorithm which solves $P_\varepsilon : \min\{f_\varepsilon(x) \mid x \in \mathbb{R}^n\}$ using estimates of $f_\varepsilon(x)$ and $\nabla f_\varepsilon(x)$.

Two kinds of algorithms, deterministic and stochastic, are available. In the deterministic algorithms the multidimensional integrals are approximated by summations over a finite grid. In the stochastic algorithms Monte-Carlo techniques are employed to estimate the integrals.

We consider initially the deterministic algorithms which approximate the integrals $f_\varepsilon(x)$ and $\nabla f_\varepsilon(x)$ by weighted summations over a finite number of points. For any positive integer j let $\tau(j)$ denote the number of points used in the numerical approximation, and let $f_j(x)$ and $\nabla f_j(x)$ denote the corresponding approximations to $f_\varepsilon(x)$ and $\nabla f_\varepsilon(x)$ (ε is fixed in the subproblem of Step 1). The function τ is monotonic increasing ($j_1 > j_2 \Rightarrow \tau(j_i) > (j_2)$) and $\tau(j) \to \infty$ as $j \to \infty$). Let $A_j(x) : \mathbb{R}^n \to 2^{\mathbb{R}^n}$ denote the corresponding algorithm map.

The following algorithm model [12] is appropriate.

Algorithm Model

Data: Integer $j_0 \geq 0$, $\delta_0 > 0$, $\alpha \in (0, 1)$.

Step 0: Set $i = 0$, $j = j_0$, $\delta = \delta_0$.

Step 1: Compute a $y \in A_j(x_i)$.

Step 2: (a) If $f_j(y) - f_j(x_i) > -\delta$, set

$j = j + 1$, $\delta = \alpha\delta$ and go to Step 1.

(b) If $f_j(y) - f_j(x_i) \leq -\delta$, set

$x_{i+1} = y$, set $i = i + 1$ and go to Step 1. □

A corresponding convergence theorem gives conditions on f_j and A_j which ensure that limit points of sequences generated by the model satisfy necessary conditions of optimality. Let $\Delta_\varepsilon \triangleq \{x \in \mathbb{R}^n \mid \nabla f_\varepsilon(x) = 0\}$.

Theorem 3 [12]

Suppose that

(i) There exists a set $M \subset \mathbb{R}^n$ satisfying $M \cap \Delta_\varepsilon \neq \phi$ such that for every

$x \in M$, $x \notin \Delta_\varepsilon$, there exists an $\gamma > 0$, $\delta > 0$ and an integer $N > 0$ satisfying

$$f_j(x'') - f_j(x') \leq -\delta$$

for all $x' \in N_\gamma(x)$, all $x'' \in A_j(x')$, all $j \geq N$.

(ii) There exists a sequence $\{\beta_s\}_{s=0}^{\infty} \subset \mathbb{R}^+$, possibly depending on M, such that

$$\sum_{s=0}^{\infty} \beta_s < \infty \ ,$$

and

$$|f_\varepsilon(x) - f_j(x)| \leq \beta_s \ ,$$

for all $x \in M$, all $j \geq s$.

Let $\{x_i\}$ be an infinite sequence generated by the Algorithm Model such that $\{x_i\} \subset M$.

Then:

If $\{x_i\}$ is finite (because the algorithm jams up between Steps 1 and 2 reducing ε infinitely often) then its last point lies in Δ_ε. If $\{x_i\}$ is infinite, then any accumulation point lies in Δ_ε. □

Condition (i) is a common condition, f_j replacing f_ε, for a "convergent algorithm" [11]; condition (ii) imposes a uniform convergence property on the

numerical approximation f_j.

The simplest deterministic algorithm is a simple modification of the steepest descent algorithm.

Algorithm 3 (for unconstrained optimization)

Data: Integer $j_0 \geq 0$, $\delta_0 > 0$, $\alpha \in (0, 1)$, $\beta \in (0, 1)$, $\lambda_{min} \in (0, 1]$.

Step 1: Set $i = 0$, $j = 0$, $\delta = \delta_0$.

Step 1: Set $\lambda > 1$.

Step 2: (a) If $f_j(x_i - \lambda \nabla f_j(x_i)) - f_j(x_i) > -\lambda \| \nabla f_j(x_i) \|^2 / 2$,

set $\lambda = \beta\lambda$.

If $\lambda \geq \delta\lambda_{min}$, repeat Step 2(a).

If $\lambda < \delta\lambda_{min}$, set $y = x_i$ and proceed.

(b) If $f_j(x_i - \lambda \nabla f_j(x_i)) - f_j(x_i) \leq -\lambda \| \nabla f_j(x_i) \|^2 / 2$,

set $y = x_i - \lambda \nabla f_j(x_i)$.

Step 3(a) If $f_j(y) - f_j(x_i) > -\varepsilon$, set $j = j + 1$,

set $\delta = \alpha\delta$ and go to Step 1.

(b) If $f_j(y) - f_j(x_i) \leq -\delta$, set $x_{i+1} = y$,

set $i = i + 1$, and go to Step 1. □

It can be seen that Algorithm 2 has the same form as the Algorithm Model, $A_j(x_i)$ in the latter corresponding to Steps 2(a) and (2b) in the former. Algorithm 2 is a finite dimensional analog of the Algorithm presented in [12].

We now make the following assumptions:

A1 The sequence $\{x_i\}$ generated by the algorithm is bounded.

A2 The integration formulae and the truncation function τ are such that for any compact subset Q of R^n and any $\delta > 0$ there exists a positive integer J and a $K \in (0, \infty)$ such that

$$\| f_\varepsilon(x) - f_j(x) \| \leq K/2^j$$

$$\| \nabla f_\varepsilon(x) - \nabla f_j(x) \| \leq \delta$$

for all $x \in Q$ and all $j \geq J$.

Because ∇f_ε is continuous (and, hence, uniformly continuous in Q) A2 is satisfied by standard integration formulae for a suitably chosen truncation function τ.

Theorem 3

Suppose that $\{x_i\}$ is a bounded sequence generated by Algorithm 2. If $\{x_i\}$ is finite then its last element lies in Δ_ε. If $\{x_i\}$ is infinite then

any accumulation point lies in Δ_ε. □

The proof of Theorem 3 is omitted since it is essentially the same as that given in Theorem 3.48 in [12].

Hence Algorithm 3 may be used in Step 1 of Algorithm 1 since it will compute (using x_{i-1} as its initial point) an x_i satisfying $\| \nabla f_{\varepsilon_i}(x_i) \| \leq \gamma_i$ in a finite number of iterations.

Whereas the deterministic algorithms estimate $\nabla f_\varepsilon(x)$ to obtain a search direction s and then estimate $f_\varepsilon(x + \lambda s)$ to obtain a step length, stochastic algorithms generally estimate only $\nabla f_\varepsilon(x)$ using Monte-Carlo techniques (to obtain a search direction) and use a pre-determined step length. Thus a standard stochastic approximation algorithm is defined by:

$$x_{i+1} = x_i - \lambda_i \nabla \hat{f}_\varepsilon(x_i)$$

for $i = 0, 1, 2, \ldots$, where $\nabla \hat{f}_\varepsilon(x_i)$ is defined by:

$$\nabla \hat{f}_\varepsilon(x_i) \triangleq [f(\xi_j + \varepsilon e_j) - f(\xi_j - \varepsilon e_j)]/2\varepsilon$$

and ξ_j is a point chosen from a uniform distribution on $D_0^j(0) \triangleq \{x \mid \| x \|_\infty \leq 1, x^j = 0\}$, e_j is the j^{th} standard basis vector $j = 1, \ldots, n$, and the sequence $\{\lambda_i\}$ of step lengths satisfies $\lambda_i > 0$, $\Sigma\lambda_i = \infty$, $\Sigma\lambda_i^2 < \infty$. (The estimate $\nabla \hat{f}_\varepsilon(x_i)$ can alternatively be defined as the average of N estimates, N any positive integer). The almost sure convergence of the algorithm is established in [10]. However, there remains the difficulty of satisfying the condition

$\| \nabla f_{\varepsilon_i}(x_i) \| \leq \gamma_i$ since $\nabla f_{\varepsilon_i}(x_i)$ is not computed. The stochastic approximation algorithm will indeed compute a sequence $\{y_j\}$ such that $\nabla f_{\varepsilon_i}(y_j) \xrightarrow{J} 0$ almost surely for some subsequence J of $\{0, 1, 2, \ldots\}$ but computes at each iteration $\nabla \hat{f}_{\varepsilon_i}(y_j)$ which does <u>not</u> converge to zero since the variance of the the estimate remains finite for fixed ε_i. Hence the test $\| \nabla f_{\varepsilon_i}(x_i) \| \leq \gamma_i$ is not implementable. We cannot therefore, at this stage, propose a suitable stochastic approximation algorithm, further research being required. One possibility is to increase the accuracy of the estimate $\nabla \hat{f}_{\varepsilon_i}(y_j)$ monotonically as j increases. Implementable algorithms for the constrained optimization problem can be similarly constructed.

6. CONCLUSION

The algorithms presented in this paper are conceptually very simple. They approximately minimize a smooth approximation f_ε of f (subject possibly to smooth approximations of the constraints), reducing ε and the accuracy of the solution in such a way as to ensure convergence. Evaluation of f_ε and ∇f_ε requires multi-dimensional integration. Implementable algorithms replace f_ε and ∇f_ε by suitable numerical approximations.

REFERENCES

1. Clarke, F., "Generalized gradients and applications", Trans. Amer. Math. Soc., Vol 205, pp 247-262, 1975.

2. Bertsekas, D.P. and Mitter, S.K., "A descent numerical method for optimization problems with non-differentiable cost functionals", Journal of Control, Vol 11, pp 636-652, 1973.

3. Goldstein, A.A., "Optimization of Lipschitz continuous functions", Mathematical Programming, Vol 13, pp 14-22, 1977.

4. Lemarechal, C., "Extensions Diverses des Méthodes de Gradient et Applications", Thesis, University of Paris VIII, 1980.

5. Demjanov, V.F., "Algorithms for some minimax problems", J.C.S.S., Vol. 2.

6. Polak, E. and Sangiovanni-Vincentelli, A., "Theoretical and computational aspects of optimal design centering, tolerancing and tuning problems", IEEE Trans. on Circuits and Systems, Vol. CAS-26, pp 295-318, 1979.

7. Polak, E., Mayne, D.Q. and Wardi, Y., "On the extension of constrained optimization algorithms from differentiable to non-differentiable problems", SIAM Journal of Control and Optimization, to appear.

8. Mifflin, R., "Semi-smooth and semi-convex functions in constrained optimization", SIAM Journal of Control and Optimization, Vol 15, pp 959-972, 1977.

9. Lebourg, C., "Valeur Moyenne pour Gradient Generalise", C.R. Acad. Sci., Paris, Vol 281, 1975.

10. Kushner, H. and Clark, D.S., "Stochastic Approximation Methods for Constrained and Unconstrained Systems", Springer-Verlag, 1978.

11. Polak E., "Computational Method in Optimization", Academic Press, 1971.

12. Klessig, R. and Polak, E., "An adaptive precision gradient method for optimal control", SIAM J. Control, Vol 11, pp 80-93, February, 1973.