GRADUATE SEMINAR REPORT

ON

DYNAMIC CONSTRUCTION OF APPROXIMATE SUBDIFF ERNTIALS: DUAL FORM OF BUNDLE METHODS

SCIENCE LIBRARY
ADDIS ABABA UNIVERSITY

BY

ZINAB ABA-OLI

ADVISOR

PROF. DR. RER. NAT. HABIL. R. DEUMLICH

SCHOOL OF GRADUATE STUDIES

ADDIS ABABA UNIVERSITY

July, 2003

ACKNOWLEDGEMENT

First of all, I would like to express my deep appreciation to my advisor and instructor Prof. Dr. R. Deumlich for his helpful suggestions, critism, generous advice and devoted assistance while working on this seminar report.

I would like to thank my father Aba-oli Abadiga and my wife W/o Meseret Denkech-a for their devoted economical and moral assistance while working on the seminar.

Zinab Aba-oli

Table of Contents

1.	Introduction		1
2.	Constructing the bundle of information		3
3.	Computing the descent direction		7
	3.1	The quadratic program	7
	3.2	Minimality conditions	10
	3.3	Directional derivative estimates and the role of the cutting plane function	17
4.	The implementable algorithm		20
	4.1	Derivation of the line-search	20
	4.2	The implementatble line-search and its convergence	22
	4.3	Derivation of the descent algorithm	27
	4.4	The implementable algorithm and its convergence	31
	References		36

Introduction

In the seminar report, we will develop a minimization algorithm, which is able to solve the optimization problem:

$$(P) \qquad f(x) \to \min,\ x \in \Re^n,$$

where $f: \Re^n \to \Re$ is a convex function.

As always done when dealing with numerical algorithms, we assume the existence of a system U_l characterizing a given function f which, given $x \in \Re^n$ computes the value f(x) and a subgradient $s(x) \in \partial f(x)$.

In the ε-descent methods, the procedures for finding $x_{x+1} \in \Re^n$ such that $f(x_{k+1}) < f(x_k) - \varepsilon$, at the iteration point x_k have more or less the following form: (as shall be seen in the algorithm given as an example in section 2). The set $\{s_1, \ldots, s_k\}$ of subgradients obtained via suitable line-searches is on hand, compute $\hat{s} := \text{Proj}\ \{0\}/S_k := \text{conv}\ \{s_1, \ldots, s_k\} \subseteq \partial_\varepsilon f(x)$ to obtain the direction $d = -\hat{s}$, then perform a line-search along this d with its two possible exites: either move x_k to a better x_{k+1}(descent-step) or obtain $s_{k+1} \in \partial_\varepsilon f(x)$ to improve the bundle (null-step).

Some of the deficiencies of the ε-descent methods are listed below.

- The choice of ε in advance with out any real behavior of the function and requiring it to decrease by ε at each iteration is suspect.
- The line search is based not on reducing the function $t \to f(x + td)$ but the difference quotient $q_\varepsilon(t) := \dfrac{f(x+td) - f(x) + \varepsilon}{t}\ (t > 0)$ along the search direction.
- All efficient minimization algorithms accumulate and exploit information about f from previous iterations. But in the ε-descent methods once x^p is moved to the next iteration point x^{P+1}, all the past is forgotten.

The aim of this seminar report is to introduce minimization methods, which try to resolve most of the deficiencies encountered by the ε-descent methods. In the following minimization method, what we have to do is replace the convex polyhedron S_k by some more elaborate convex set which still gives an inner approximation of $\partial_\varepsilon f(x)$ and then reproduce the above strategy.

To be self-contained, we give below a short glossary of symbols and terms appearing throughout.

- A set $K \subseteq \Re^n$ is said to be **convex** if and only if $x, y \in K$ implies $\lambda x + (1-\lambda)\, y \in K$ for all $\lambda \in (0, 1)$.
- Let $K \subseteq \Re^n$ be convex and $f: K \to \Re$. f is said to be **convex** if and only if $f(\lambda x + (1-\lambda)y) \leq \lambda f(x) + (1 - \lambda)\, f(y)$, $\forall_{x,y} \in K$, $\forall_{\lambda \in} (0, 1)$

Let $f: \Re^n \to \Re$ be convex. Then:

- $d \in \Re^n$ is said to be an **ε-descent direction** for f at $x \in \Re^n$ if there is a $t > 0$ such that $f(x + td) < f(x) - \varepsilon$
- Given $x \in \Re^n$ and $\varepsilon \geq 0$, an element $s \in \Re^n$ is said to be an **ε-subgradient** of f at x if and only if $f(y) \geq f(x) + \langle s, y - x \rangle - \varepsilon$, $\forall_{y \in} \Re^n$.
- The set of all ε-subgradients of f at x is called of f of f at x and is denoted by $\partial_\varepsilon f(x)$.
- For $x \in \Re^n$, $f'(x, d) := \lim_{t \to 0} \dfrac{f(x+td) - f(x)}{t}$ denotes the Gateaux derivative (directional) of f at x in the direction of d, where $d \in \Re^n$
- $\Delta_\ell := \{\alpha \in \Re^\ell : \sum_{i=1}^{\ell} \alpha_i = 1, \alpha_i \geq 0 \text{ for } i = 1, \ldots, \ell\}$ is the unit simplex of $\Re^\ell$
- Let $\phi \neq S \subseteq \Re^n$. The function $\sigma_s: \Re^n \to \Re \cup \{\infty\}$ given by $\sigma_s(x) := \sup \{\langle s_i x \rangle : s \in S\}$, $x \in \Re^n$ is called the support function of S.

2. Constructing the Bundle of Information

Generally speaking, in most minimization methods, the descent direction d_k such that $x_{k+r} = x_k + t_k d_k$ is computed at the current iterate x_k in view of the information on hand concerning the function f which is obtained from (U_1).

Definition 2.1 Let $f:\Re^n \to \Re$ be convex. At the current iteration of the minimization algorithm, let $y_1, y_2, \ldots, y_n$ be points in $\Re^n$ at which $f(y_i)$ and $s(y_i)$ has been computed by (U_1) from the beginning where $i = 1, \ldots, \ell$. Then the **raw bundle** is the set of triples

$$\{y_i \in \Re^n, f_i := f(y_i), s_i := s(y_i) \in \partial f(y_i): i = 1, 2, \ldots, \ell\} \qquad (2.1)$$

collecting all the information available concerning f.

In the definition since ℓ increase by the number of cycles needed by the line search at each iteration, it is difficult (for even impossible) to keep (store all the points y_i, f_i and s_i for collecting information. So one must do selection and/or compression.

Definition 2.2 A **bundle** is the bank of information that is explicitly obtained from the raw bundle, actually stored and used to compute the current direction.
There are several possibilities for selecting-compressing the raw bundle depending on the role of the bundle interms of the resulting minimization algorithm.

To illustrate this let us consider the following example.

Example 2.1 (ε-Descent Method). In these methods the role of the bundle is to identify $\partial_\varepsilon f(x_k)$, at the current iterate x_k.

Algorithm of ε-descent: Given the starting point $x_1 \in \Re^n$, compute $s_1 \in \partial f(x_1)$. Choose the descent criterion $\varepsilon > 0$, the convergence parameter $\delta > 0$, the tolerance for the line-search $m' \in (0, 1)$ and the maximal bundle size $\bar{\ell} \geqq 2$.

Initialize the iteration index $k = 1$, the descent index $k_0 = 0$, and the bundle size $\ell = 1$.

Step 1. (Direction-finding and stopping criterion)

Solve the minimization problem on the variable $\alpha \in \Re^{\ell}$:

i) $\min \frac{1}{2} \left\| \sum_{i=k_0+1}^{k_0+\ell} \alpha_i s_i \right\|^2$

ii) $\sum_{i=k_0+1}^{k_0+\ell} \alpha_i = 1$, $\alpha_i \geqq 0$ for $i = 1 + k_0, 2+k_0, \ldots, \ell + k_0$.

and set $d_k := \sum_{i=k_0+1}^{k_0+\ell} \alpha_i s_i$.

If $\|d_k\| \leqq \delta$, then stop.

Step2(Line-search)

Minimize $r_\varepsilon(u) := \begin{cases} q_\varepsilon\left(\frac{1}{u}\right) = u\left[f(x + \frac{1}{u} d) - f(x) + \varepsilon \right], \text{if } u > 0 \\ f'_\infty(d) = \lim_{t \to \infty} q_\varepsilon, \text{if } u = 0 \\ \infty, \qquad \text{if } u < 0 \end{cases}$

along d_k accurately enough to obtain a step size $t > 0$ and a new iterate $y_+ = x_k + td_k$ with $s_+ \in \partial f(y_+)$ such that:

- either $f(y_+) < f(x_k) - \varepsilon$; then go to step 3.
- or $\langle s_+, d_k \rangle > -m' \|d_k\|^2$ and $s_+ \in \partial_\varepsilon f(x_k)$; then go to step 4.

Step 3 (descent – step) Set $x_{k+1} = y_+$, $k_0 = k$, $\ell = 1$; replace k by k + 1 and loop to step 1.

Step 4 (Null – step) set $x_{k+1} = x_k$, replace k by k + 1. If $\ell < \bar{\ell}$, then set $Sk_{0+\ell+1} = s_+$ from step 2, replace ℓ by $\ell + 1$ and loop to step 1. Else go to step 5.

Step 5 (Compression) Delete at least two vectors from the list $\{s_{k0}, \ldots s_{k0+\ell})$.

with ℓ denoting the number of elements in the new list, set $s_{k0+\ell+1} = -d_k$ from step 1, set $s_{k0+\ell+2} = s_+$ from step 2; replace ℓ by $\ell + 2$ and loop to step 1.

Examining the ε-descent method in its form as in example 2.1 suggests a weak point: why should the actual bundle be reset entirely at k_0^2. Some of the elements $\{s_1,\ldots, s_{k_0}\}$ might well be in $\partial_\varepsilon f(x_k)$. Then why not use them to compute d_k? The answer is contained in the following proposition.

Proposition 2.1 (Transportation formula) Let x and x″ be in the domain of the function $f:\Re^n \to \Re$, $\varepsilon > 0$ and let $s' \in \partial f(x')$. Then $s' \in \partial_\varepsilon f(x)$ if and only if

$$f(x') \geqq f(x) + \langle s', x' - x\rangle - \varepsilon \qquad (2.2)$$

Proof: Suppose that $s' \in \partial_\varepsilon f(x)$.

Then by the definition of ε-subgradient of f at x, we have $f(y) \geqq f(x) + \langle s', y-x\rangle - \varepsilon$ for all $y \in \Re^n$. Hence it holds for particular $y = x'$. Conversely, for $s' \in \partial f(x')$, we have for all $y \in \Re^n$.

$f(y) \geqq f(x') + \langle s', y - x'\rangle = f(x) + \langle s', y - x\rangle + [f(x') - f(x) + \langle s', x - x'\rangle]$

Hence if (2.2) holds, $s' \in \partial_\varepsilon f(x)$

Definition 2.3 (Lineralization error) For x, x′, $s' \in \Re^n$, the linearlization error made at x, when f is linearlized at x′ with slope s′, is the nonnegative number

$$e(x, x', s') = f(x) - f(x') - \langle s', x - x'\rangle \qquad (2.3)$$

From proposition 2.1 and definition 2.3, it follows that s′ is an e(x, x′, s′) subgradient of f at x. Conversely, for $s' \in \partial f(x')$, $s' \in \partial_\varepsilon f(x)$ if and only if

$e(x. x',s') \leqq \varepsilon$

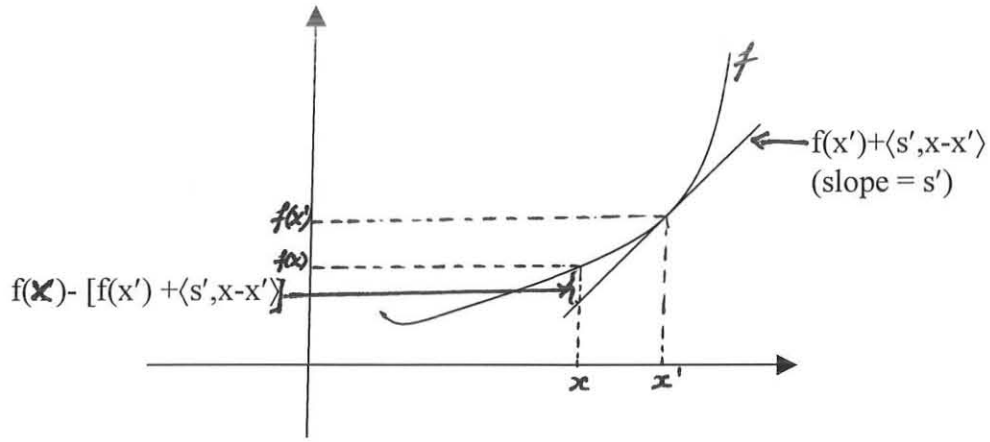


Fig 2.1 Linearlization error and the transportation formula.

Let $s' \in \partial f(x')$ and $\bar{f}_{x',s'}(y) := f(x') + \langle s',\ y - x' \rangle$ be the corresponding affine approximation of f. Then, there holds the relations

$$\bar{f}_{x',s'}(x) = f(x) - e(x, x', x'), \text{ and}$$

$$\bar{f}_{x',s'}(y) \leq f(y) \text{ for all } y \in \Re^n.$$

Remark 2.1 The value s(y) computed by (U_1) at y can be considered as a function of y alone; so the notation e(x, y) could be preferred to e(x, y, s). An important point here is that for given x and y, e(x, y) can be explicitely computed whenever f(x), f(y) and s(y) are known.

Definition 2.4 For each i in definition 2.1, the linearization error at the iterate x_k is the nonnegative number

$$e_i^k := f(x_k) - [f(y_i) + \langle s_i, x_k - y_i \rangle] \qquad (2.4)$$

In this definition, the notation e_i can be used for e_i^k. The bundle is now reduced to

$$\{(s_i, e_i) \mid i = 1, \ldots, \ell\} \qquad (2.5)$$

where $s_i \in \partial_{e_i} f(x_k)$ for $i = 1, \ldots \ell$.

Proposition 2.2 For any two iterates x_k and x_k', and sampling point y_i, there holds

$$e_i^{x'} = e_i^k + f(x_{k'}) - [f(x_k) + \langle s_i, x_{k'} - x_k \rangle]$$

Proof: By definition 2.4, we have

$$e_i^{k'} = f(x_{k'}) - [f(y_i) + \langle s_i, x_{k'} - y_i \rangle \text{ and}$$

$$e_i^k = f(x_k) - [f(y_i) + \langle s_i, x_k - y_i \rangle.$$

From this we get $e_i^{k'} - e_i^k = f(x_{k'}) - [f(y_i) + \langle s_i, x_{k'} - y_i \rangle] - f(x_k)$

$$+ [f(y_i) + \langle s_i, x_k - y_i \rangle].$$

If follows $e_i^{k'} - e_i^k = f(x_{k'}) - f(x_k) + \langle s_i, s_k - x'_k \rangle$

$$= f(x_{k'}) - [f(x_k) + \langle s_i, s_{k'} - x_k \rangle]$$

Hence $e_i^{k'} = e_i^k + f(x_{k'}) - [f(x_k) + \langle s_i, x_{k'} - x_k \rangle]$

At each descent iteration, each e_i in (2.5) must be recomputed according to proposition 2.2 so as to maintain the bundle without storing the y_i.

3. Computing the descent Direction

3.1 The Quadratic Program

Hereafter $\{s_i, e_i\}$ is the bundle, the current iterate and the bundle size are respectively denoted by x and ℓ.

The starting idea of the algorithm in question lies in the following result.

Proposition 3.1.1 With $s_i \in \partial f(y_i)$ for $i = 1, \ldots, \ell$, e_i defined by (2.4) at $x_k = x$, and $\alpha = (\alpha_1, \ldots, \alpha_\ell) \in \Delta_\ell$, there holds

$$f(z) \geq f(x) + \left\langle \sum_{i=1}^{\ell} \alpha_i s_i, z - x \right\rangle - \sum_{i=1}^{\ell} \alpha_i e_i \quad \text{for all } z \in \Re^n.$$

In particular, for $\varepsilon \geq \min_i \{e_i\}$, the set

$$S(\varepsilon) := \{s = \sum_{i=1}^{\ell} \alpha_i s_i : \alpha \in \Delta_\ell, \sum_{i=1}^{\ell} \alpha_i e_i \leq \varepsilon\} \qquad (3.1.1)$$

is contained in $\partial_\varepsilon f(x)$, where $\Delta_\ell := \{\alpha \in \Re^\ell : \sum_{i=1}^{\ell} \alpha_i = 1, \alpha_i \geq 0 \text{ for: } = 1, \ldots \ell\}$.

Proof: By subgradient inequality written at y_i, $f(z) > f(y_i) + \langle s_i, z - y_i \rangle$ for all $z \in \Re^n$, and each $i = 1, \ldots, \ell$ (1)

Again by definition 1.2.4, $e_i = f(x) - [f(y_i) + \langle s_i, x - y_i \rangle]$, for $i = 1, \ldots, \ell$ (2)

From (1) and (2) if follows, $f(20 \geq f(x) + \langle s_i y_i - x \rangle + \langle s_i, z - y_i \rangle - e_i$

$= f(x) + \langle s_i, z - x \rangle - e_i$, for all $z \in \Re^n$ and $i = 1, \ldots \ell$.

That is $f(z) - f(x) \geq \langle_i, z - x \rangle - e_i$, $\forall z \in \Re^n$ and $i = 1, \ldots, \ell$ (3)

Multiplying both sides of (3) by $\alpha_i \geq 0$, we get

$$\alpha_i(f(z) - f(x)) \geq \alpha_i \langle s_i, z - x \rangle - \alpha_i e_i \qquad (4)$$

for $i = 1, \ldots, \ell$, $\alpha_i \geq 0$, $\sum_{i=1}^{\ell} \alpha_i = 1$

By adding the result in (4), we have

$$\sum_{i=1}^{\ell} \alpha_i (f(z) - f(x)) \geq \sum_{i=1}^{\ell} \langle \alpha_i s_i, z - x \rangle - \sum_{i=1}^{\ell} \alpha_i e_i$$

This implies $f(z) - f(x) \geq \langle \sum_{i=1}^{\ell} \alpha_i s_i, z - x \rangle - \sum_{i=1}^{\ell} \alpha_i e_i$

Hence it follows $f(z) \geq f(x) + \langle \sum_{i=1}^{\ell} \alpha_i s_i, z\text{-}x \rangle - \sum_{i=1}^{\ell} \alpha_i e_i$, $\forall_{z \in} \Re^n$.

In particular for $s \in S(\varepsilon)$, we have $s = \sum_{i=1}^{\ell} \alpha_i s_i$, $\alpha \in \Delta_\ell$, and $\sum_{i=1}^{\ell} \alpha_i e_i \leq \varepsilon$ (6)

Using (6) in (5) we get, $f(z) \geq f(x) + \langle s, z - x \rangle - \sum_{i=1}^{\ell} \alpha_i e_i \geq f(x) + \langle s, z - x \rangle - \varepsilon$,

$\forall_{z \in} \Re^n$.

Hence by the definition of ε-subgradient of f at x, we have $s = \sum_{i=1}^{\ell} \alpha_i s_i \in \partial_\varepsilon f(x)$.

Therefore, $S(\varepsilon) \subseteq \partial_\varepsilon f(x)$



From proposition 2.1, it follows that

$\varepsilon < \varepsilon'$ implies $S(\varepsilon) \subseteq S(\varepsilon') \subseteq$ conv $\{s_1, \ldots, s_\ell\}$.

Denotation: For $\alpha \in \Delta_\ell$, we set

$$s(\alpha) := \sum_{i=1}^{\ell} \alpha_i s_i \,,\ e(\alpha) := \sum_{i=1}^{\ell} \alpha_i e_i \quad \text{and}$$

$$S(\varepsilon) = \{s(\alpha) : \alpha \in \Delta_\ell,\ e(\alpha) \leq \varepsilon\} \qquad (3.1.2)$$

The above result gives way to a strategy for computing the direction merely copying the procedure of the ε-descent as described in section 1. The convex set S_k being replaced by the more elaborate set $S(\varepsilon)$, all one has to do is project the origin on to $S(\varepsilon)$: Choose $\varepsilon > 0$ and compute the particular ε-subgradient

$$\hat{s} := \text{proj } \{0\}/S'(\varepsilon) \qquad (3.1.3)$$

The focus of attention in this section is the following direction finding problem.

$$\frac{1}{2} \|S(\alpha)\|^2 \to \min,\ \alpha \in S,$$

$$S := \{\alpha \in \Re^\ell / \sum_{i=1}^{\ell} \alpha_i = 1,\ \alpha_i \geq 0 \text{ for each}$$

$$i = 1, \ldots, \ell \text{ and } \sum_{i=1}^{\ell} \alpha_i e_i \leq \varepsilon\}. \qquad (3.1.4)$$

where $s(\alpha) := \sum_{i=1}^{\ell} \alpha_i s_i$ and e_i, s_i are as given by (2.5)

Once we solve (3.1.4), the descent direction d from the iteration point x is defined by

$$d = -\hat{s} = -\hat{s}_\varepsilon := -s(\alpha) \qquad (3.1.5)$$

for some α solving (3.1.4).

Proposition 3.1.2 The direction finding problem (3.1.4) has an optional solution if and only if $\varepsilon \geq \min \{e_i : i = 1, \ldots, \ell\}$ (3.1.6)

Proof: Suppose (3.1.6) holds. Then the function v: $\Re^{\ell} \to \Re$ given by $v(\alpha) = \frac{1}{2}\|s(\alpha)\|^2$ is continuous, the set S given in (3.1.4) and $S(\varepsilon)$ given by (3.1.2) are compact. By (3.1.6) holds, we have $s_i \in S(e_i) \subseteq S(\varepsilon)$ implies $S(\varepsilon) \neq \phi$. That means there is a $\alpha \in \Delta_{\ell}$ such that $e(\alpha) \leqq \varepsilon$ and $s(\alpha) = \sum_{i=1}^{\ell} \alpha_i s_i \in S(\varepsilon)$. Hence it follows that $S \neq \phi$. Thus v attains its minimum on $S \neq \phi$. Conversely, if the direction finding problem (3.1.4) has an optimal solution say α_0, then by $\alpha_0 \in S$, $e(\alpha_0) = \sum_{i=1}^{\ell} \alpha_i e_i \leqq \varepsilon$.

Therefore we get $\min_i e_i \leqq \sum_{i=1}^{\ell} \alpha_i e_i \leqq \varepsilon$.

Remark 3.1.2 In the sense of proposition 3.1.2, the direction d is well defined independently of the optimal α.

3.2 Minimality Conditions

For each $\alpha \in \Re^{\ell}$, $s(\alpha) := \sum_{i=1}^{\ell} \alpha_i s_i \in \Re^n$, where s_i are given by (2.5). Now consider the function v:$\Re^{\ell} \to \Re$ given by $v(\alpha) = \frac{1}{2}\|s(\alpha)\|^2$, $\alpha \in \Re^{\ell}$. Below we show

$$\frac{\partial v}{\partial \alpha_j} = \langle s_j, s(\alpha) \rangle, \text{ for } j = 1, \ldots, \ell \qquad (3.2.1)$$

By definition of the pre-Hilbert space norm,

$$\|s(\alpha)\|^2 = \langle s(\alpha), s(\alpha) \rangle = \langle \sum_{i=1}^{\ell} \alpha_i s_i , \sum_{i=1}^{\ell} \alpha_i s_i \rangle$$

Hence $$\frac{\partial v}{\partial \alpha_j}(\alpha) = \frac{\partial}{\partial \alpha_j}\left(\frac{1}{2}\left\langle \sum_{i=1}^{\ell} \alpha_i s_i, \sum_{i=1}^{\ell} \alpha_i s_i \right\rangle\right), \text{ for } j = 1, \ldots, \ell.$$

But $$\left\langle \sum_{i=1}^{\ell} \alpha_i s_i, \sum_{i=1}^{\ell} \alpha_i s_i \right\rangle = \left\langle \sum_{\substack{i=1 \\ i\neq j}}^{\ell} \alpha_i s_i, +\alpha_j s_j \sum_{\substack{i=1 \\ i\neq j}}^{\ell} \alpha_i s_i, +\alpha_j s_j \right\rangle$$

$$= \left\langle \sum_{\substack{i=1 \\ i\neq j}}^{\ell} \alpha_i s_i, \sum_{\substack{i=1 \\ i\neq j}}^{\ell} \alpha_i s_i \right\rangle + 2\left\langle \alpha_j s_j, \sum_{\substack{i=1 \\ i\neq j}}^{\ell} \alpha_j s_j \right\rangle + \langle\, \alpha_j s_j,\, \alpha_j s_j \,\rangle$$

$$= \left\langle \sum_{\substack{i=1 \\ i\neq j}}^{\ell} \alpha_i s_i, \sum_{\substack{i=1 \\ i\neq j}}^{\ell} \alpha_i s_i \right\rangle + 2\alpha_j \left\langle s_i, \sum_{\substack{i=1 \\ i\neq j}}^{\ell} \alpha_i s_i \right\rangle + \alpha_j^2 \langle s_j, s_j \rangle$$

Hence $$\frac{\partial v}{\partial \alpha_j}(\alpha) = \left\langle s_i, \sum_{\substack{i=1 \\ i\neq j}}^{\ell} \alpha_i s_i \right\rangle + \alpha_j \langle s_j, s_j \rangle$$

$$= \left\langle s_i, \sum_{i=1}^{\ell} \alpha_i s_i \right\rangle$$

$$= \langle\, s_j,\, s(\alpha) \,\rangle, \text{ for } j = 1, \ldots, \ell.$$

The Lagrange function L for the problem (3.1.4) associated with $\lambda \in \Re$ and $\mu \in \Re^+$ is given by

$$L(\alpha, \lambda, \mu) = \frac{1}{2}\|s(\alpha)\|^2 + \lambda\left(1 - \sum_{i=1}^{\ell} \alpha_i\right) + \mu\,(e(\alpha) - \varepsilon),\ \alpha \in \Re^{\ell} \qquad (3.1.2)$$

Definition 3.2.1 Let the following convex minimization problem be given.

$$f(x) \to \min,\ x \in S,$$

$$S = \{x \in \Re^n \mid \langle\, a_i, x \,\rangle = b_i \text{ for } i=1,2,\ldots,m \text{ and } c_j(x) \leqq 0 \text{ for } j = 1, \ldots p\} \qquad (3.2.3)$$

And let $J_a := \{i \in: \{1, \ldots, p\}: c_i \text{ is affine function}\}$.

Then the constraint set $S = \{x \in \Re^n: A_x = b \in \Re^m, c(x) \leq 0\}$

Satisfies the weak Slater assumption WSA) if there is $x_0 \in S$ such that

(i) $Ax_0 = b$ (ii) $C_i(x_0) \leqq 0$, for $i \in J_a$

(iii) $C_i(x_0) < 0$, for $i \notin J_a$

Theorem 3.2.1 Let the WSA hold. Then at any $\bar{x} \in S$, $\bar{x}$ minimizes the convex function f on S′ if and only if there exist $\lambda \in \Re^m$ and $\mu \in (\Re^+)^p$ such that

$$\text{i)} \qquad 0 \in \partial f(\bar{x}) + \sum_{i=1}^{m} \lambda_i a_i + \sum_{j=1}^{p} \mu_j \partial c_j(\bar{x}),$$

$$\text{ii)} \qquad \mu_j \geq 0 \text{ and } \mu_j c_j(\bar{x}) = 0 \text{ for } j = 1, \ldots, p$$

In theorem 3.2.1, all the notations are as in definition 3.2.1.

Theorem 3.2.2 Let $\bar{x}$ and $\tilde{x}$ be two solutions of the constrained minimization problem (3.2.3), $M(\bar{x})$ and $M(\tilde{x})$ being their corresponding sets of Lagrange multipliers. Then $M(\bar{x}) = M(\tilde{x})$. Based on the Lagrange function (3.2.2), the minimality conditions for the direction finding problem (3.1.4) can be formulated in the following.

Theorem 3.2.3 Let $s(\alpha) := \sum_{i=1}^{\ell} \alpha_i s_i$, $e(\alpha) := \sum_{i=1}^{\ell} \alpha_i e_i$, and $\varepsilon > 0$ where ℓ denotes the current bundle size and $\alpha \in \Re^\ell$. The, α is a solution of the direction finding problem (3.1.4) if and only if the following three condition hold.

i) $\alpha \in \Delta_\ell$,

ii) $e(\alpha) \leqq \varepsilon$,

iii) There is a $\mu \geqq 0$ such that

$$\mu = 0 \text{ if } e(\alpha) < \varepsilon \text{ and } \langle s_i, s(\alpha) \rangle + \mu e_i \geqq \| s(\alpha) \|^2 + \mu\varepsilon, \qquad (3.2.4)$$

with equality if $\alpha_i > 0$.

Besides, the set of such μ is actually independent of the particular solution α.

Proof: All the constraints functions being affine and $S = \{\alpha \in \Re^\ell \mid \alpha \in \Delta^\ell, e(\alpha) \leq \varepsilon\}$ being nonempty, each $\alpha \in S'$ satisfies the WSA.

Hence by theorem 3.2.1 $\bar{\alpha} \in$Sis a solution of (3.1.4) iff and only if there exist $\lambda^* \in \Re$ and $\mu^* \in \Re^+$ such that conditions (i) and (ii) of theorem 3.2.1 hold. Now let $\bar{\alpha} \in$S be a solution of (3.1.4). Then by the above, there exist $\lambda^* \in \Re$ and $\mu^* \in \Re^+$ such that conditions (i) and (ii) of theorem 3.2.1 hold.

By the Lagrange function of (3.1.4), given by (3.2.2) is differentiable for each $\lambda \in \Re$ and $\mu \in \Re^+$ and also convex, we have $\partial L(\bar{\alpha}, \lambda^*, \mu^*) = \{\nabla L(\bar{\alpha}, \lambda^*, \mu^*)\}$. Hence we get by (i) of theorem 3.2.1 that

$$\frac{\partial L}{\partial \bar{\alpha}_j}\left(\bar{\alpha}, \lambda^*, \mu^*\right) = 0 \text{ for } j = 1, \ldots, \ell$$

Hence it follows that $\mu^* e_j - \lambda^* = 0 \geqq 0$, $j = 1, \ldots, \ell$.

This is equivalent to saying

$$\bar{\alpha}_j[\langle s_j, s(\bar{\alpha}) \rangle + \mu^* e_j - \lambda^*] = 0 \text{ for } j = 1, \ldots, \ell.$$

Adding the ℓ equalities, we get

$$0 = \sum_{j=1}^{\ell} \bar{\alpha}_j [\langle s_j, s(\bar{\alpha}) \rangle + \mu^* e_j - \lambda^*]$$

$$= \langle \sum_{j=1}^{\ell} \bar{\alpha}_j s_j, s(\bar{\alpha}) \rangle + \mu^* \underbrace{\sum_{j=1}^{\ell} \bar{\alpha}_j e_j}_{=\varepsilon} - \left[\underbrace{\sum_{j=1}^{\ell} \bar{\alpha}_j}_{=1} \right] \lambda^*$$

$$= \langle s(\bar{\alpha}), s(\bar{\alpha}) \rangle + \mu^* \varepsilon - \lambda^*$$

$$= \|s(\bar{\alpha})\|^2 + \mu^* \varepsilon - \lambda^*$$

Therefore $\lambda^* = \|s(\bar{\alpha})\|^2 + \mu^* \varepsilon$.

Consequently conditions (i), (ii) and (iii) of the theorem follow.

Conversely, suppose there is a $\mu^* \geqq 0$ such that conditions (i), (ii) and (iii) hold for $\bar{\alpha} \in$S.Then we have: (a) $1 - \sum_{j=1}^{\ell} \bar{\alpha}_i = 0$,

(b) $\mu^* [e(\bar{\alpha}) - \varepsilon] = 0$, $e(\bar{\alpha}) \varepsilon \leqq 0$, $\mu^* \geqq 0$,

(c) $\frac{\partial L}{\partial \bar{\alpha}_j}\left(\bar{\alpha}, \lambda^*, \mu^*\right) = 0$

Hence the Khun-Tucker-conditions are satisfied and L is convex.

Thus Kuhn-Tucker-conditions are sufficient for $\overline{\alpha}$ to be a solution of (3.1.4).

Therefore $\overline{\alpha}$ is a solution of (3.1.4). Moreover, by theorem 3.2.2, it follows that μ^* does not depend on $\overline{\alpha}$.

On the other hand if we consider a special Lagrange function

$$\alpha \rightarrow L_\mu(\alpha) := \frac{1}{2}\|s(\alpha)\|^2 + \mu[e(\alpha) - \varepsilon].\ \alpha \in \Re^\ell,$$

we get the dual function $\mu \rightarrow \theta(\mu) = \min_{\beta \in \Delta\ell} L_\mu(\alpha)$ (3.2.5)

which is closed and convex.

The set $M_\varepsilon = \text{Argmin}\ \{\theta(\mu)/\mu > 0\} \neq \phi$: (3.2.6)

it is the set of those μ described by theorem 3.2.1 for some

$\alpha \in A_\varepsilon := \{\alpha \in \Delta_\ell / e(\alpha) \leqq \varepsilon\ ,\ s(\alpha) = \hat{s}_\varepsilon\}$ (3.2.7)

Now it is convenient to state the following theorem whose proof is given in [3].

Theorem 3.2.4 Let the convex minimization problem (3.2.3) has a solution and a nonempty set of Lagrange Multipliers. Let also ψ be defined by

$$(\lambda,\mu) \rightarrow \psi(\lambda,\mu) = \begin{cases} \inf L(x,\lambda,\mu),\ if\ (\lambda,\mu) \in \Re_x^m \left(\Re^+\right)^p \\ -\infty \qquad\qquad ,\ if\ (\lambda,\mu) \notin \Re_x^m \left(\Re^+\right)^p \end{cases}$$

where $(\lambda,\mu) \in \Re_x^m \Re^p$.

Then $-\psi \in \overline{conv}\,(\Re_x^m \Re^p)$ and the maximal of ψ are the Lagrange multipliers.

Thus we get an other formulation of the minimality conditions for the direction finding problem (3.1.4) given in the following theorem.

Theorem 3.2.5 Let $\alpha \in \Delta_\ell$, $e(\alpha) = \sum_{j=1}^{\ell} \overline{\alpha}_j s_j$ and $\varepsilon > 0$ where ℓ denote the current bundle size. Then, α solves (3.1.4) if and only if it solves the minimization problem in (3.2.5) for some $\mu \geqq 0$ and $e(\alpha) \leqq \varepsilon$ such that $e(\alpha) = \varepsilon$ if $\mu \geqq 0$.

Proof: Follows from theorem 3.2.4

The following proposition gives a converse property.

Proposition 3.2.6 For a given $\mu \geqq 0$, all optimal solution α in (3.2.5) make up the same $s(\alpha)$ in (3.1.2). As a consequence of duality theory again, it can easily be checked that the optimal value in (3.1.4) i.e. $v(\varepsilon) = \frac{1}{2}\|\hat{s}_\varepsilon\|^2$, is a convex function of ε, and $\mu \in M_\varepsilon$ if an only if $-\mu \in \partial v(\varepsilon)$. We also have $0 \leqq \varepsilon < \varepsilon'$ only if $\mu \geqq \mu'$ for all $\mu \in M_\varepsilon$ and $\mu' \in M_{\varepsilon'}$. (i.e the multifunction $\varepsilon \to -M_\varepsilon$ is monotone)

Proposition 3.2.7 Suppose $S(0) \neq \phi$ (i.e. $\min_i e_i = 0$)

Then $\mu < 2.\frac{k}{\underline{e}}$ for all $\varepsilon > 0$ and $\mu \in M_\varepsilon$, where

$$k := \max\{\|s_i\|^2 \mid i = 1, \ldots. \ell\},\ \underline{e} := \min\{e_i \mid e_i > 0\}.$$

Proof: If $\mu = 0$, there is nothing to prove. Assume $\mu > 0$/.

Then for optimal solution α of (3.1.4, we have by theorem 3.2.5 that $e(\alpha) = \varepsilon$.

Then for $\varepsilon \in (0, \underline{e})$: by $\sum_{i=1}^{\ell} \alpha_j e_j = \varepsilon$, there must be some j with $\alpha_j > 0$ and $e_j \geqq \underline{e} > \varepsilon$.

For this j, $\|\hat{s}\|^2 + \mu\varepsilon = \langle s_j,\ \hat{s}\rangle + \mu e_j \geqq \langle s_j,\ \hat{s}\rangle + \mu\underline{e}$ (this follows by theorem 3.2.3 and $\underline{e} \leqq e_j$). Hence $\mu(\underline{e} - \varepsilon) < \|\hat{s}\|^2 + \langle -s_j,\ \hat{s}\rangle \underset{\uparrow}{\leqq} \|\hat{s}\|^2 + \|-s_j\|\|\hat{s}\| \leqq \|\hat{s}\|^2 + \|-s_j\|\|\hat{s}\|$

(*by Cauchy–Schewarz*)

$\underset{\uparrow}{\leqq} \|\hat{s}\|^2 + \|s_j\|^2 \leqq 2k$ That is, $\mu(\underline{e} - \varepsilon) \leqq 2k$ (*)

(*by* $\|\hat{s}\| \leqq \|s_j\|$)

Multiplying both sides of (*) by $\frac{1}{\underline{e} - \varepsilon} > 0$, we get $\mu \leqq 2.\frac{k}{\underline{e} - \varepsilon}$

Since ε was arbitrary in $(0, \underline{e})$, (letting $\varepsilon \to 0^+$, we get $\mu \leqq 2.\frac{k}{\underline{e}}$. And by $0 \leqq \varepsilon'$ implies $\mu > \mu'$ for all $\mu \in M_\varepsilon$ and all $\mu' \in M_{\varepsilon'}$, it holts for all $\varepsilon > 0$ that $\mu \leqq 2.\ \frac{k}{\underline{e}}$

This theorem puts an upper bound on μ. When $\varepsilon \to \infty$, the constraint (3.1.4) (iii) become inactive; μ reaches the value 0 and stays there. The treshold where this happens is

$$\bar{\varepsilon} := \min \{e(\alpha): \alpha \in \Delta_\ell\ ;\ S(\alpha) = \hat{s}_\infty\}$$

Hence the function $\varepsilon \to \frac{1}{2}\|\hat{s}_\varepsilon\|^2$ is minimal for $\varepsilon \in [, \infty)$ and $0 \in M_{\bar{\varepsilon}}$. Then we have $\varepsilon > \bar{\varepsilon}$ implies $M_\varepsilon = \{0\}$ and $\varepsilon < \bar{\varepsilon}$ implies $0 \notin M_\varepsilon$. Finally, an interesting question is, whether M_ε has exactly one element.

Proposition 3.2.8 Suppose ε is such that for some α solving (3.1.4), there is a j with $\alpha_j > 0$ and $e_j \neq \varepsilon$. Then M_ε is a singleton set containing the single element $\mu_\varepsilon = \dfrac{\langle s_j - \hat{s}, \hat{s} \rangle}{\varepsilon - e_j}$.

Proof: For j such that $\alpha_j > 0$ and $e_j \neq \varepsilon$, we have by (3.2.4) that

$\langle s_j, \hat{s} \rangle + \mu e_j = \|\hat{s}\|^2 + \mu\varepsilon$. This implies $\langle s_j - \hat{s}, \hat{s} \rangle = \mu(\varepsilon - e_j)$ (*)

Dividing (*) by $\varepsilon - e_j \neq 0$, we get

$$\mu = \frac{\langle s_j - \hat{s}, \hat{s} \rangle}{\varepsilon - \varepsilon_j}$$

In summary, the function $\varepsilon \to \frac{1}{2}\|\hat{s}_\varepsilon\|^2$ has the following properties

Corollary The function $q(\varepsilon) := \frac{1}{2}\|\hat{s}_\varepsilon\|^2$ has the following properties.

i) q is convex with domain $[0, \infty)$

ii) q has finite right slope at 0.

iii) q is ordinarily smooth for $\varepsilon > 0$

iv) q is nowhere affine except for $\varepsilon \geqq \bar{\varepsilon}$

v) q is constant on $[0, \infty)$ if $\varepsilon = 0$.

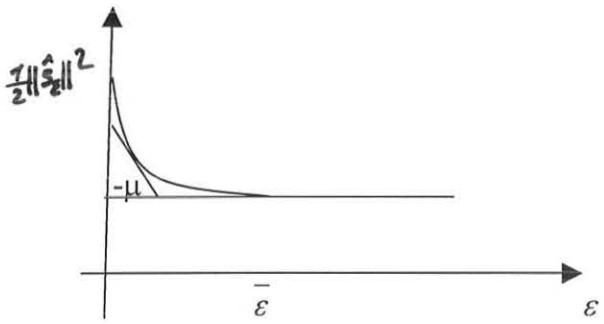


Fig 3.2.1 Typical graph of the squared norm of the direction.

3.3 Directional derivative estimates and the role of the cutting plane function

Proposition 3.3.1 $f_\varepsilon'(x, -\hat{s}_\varepsilon) > \sigma_{S(\varepsilon)}(-\hat{s}_\varepsilon) = \langle \hat{s}_\varepsilon, -\hat{s}_\varepsilon \rangle = -\|\hat{s}_\varepsilon\|^2$

Proof: $\langle s(\alpha), -\hat{s} \rangle \leqq -\|\hat{s}\|^2 + \underbrace{\mu(e(\alpha) - \varepsilon}_{\leq 0} \leqq -\|\hat{s}\|^2$

If α is optimal solution of (3.1.4), then $s(\alpha) = \hat{s}_\varepsilon$.

This implies $\sigma_{S(\varepsilon)}(-\hat{s}_\varepsilon) = -\|\hat{s}_\varepsilon\|^2$

Proposition 3.3.2 Suppose that

i) $e_j = 0$ for some $j \in \{1, \ldots, \ell\}$

ii) the corresponding α_j is positive for some solution of (3.1.4).

Then for any multiplier $\mu \geqq 0$ described by Theorem 3.2.1,

$$f'(x, -\hat{s}_\varepsilon) > -\| \hat{s}_\varepsilon \|^2 - \mu\varepsilon \qquad (3.3.1)$$

If in addition all the extreme points of $\partial f(x)$ appear among the element of bindle, then equality holds in (3.3.1).

Proof: From $e_j = 0$, we get $s_j \in \partial f(x)$.

This implies $f'(x, -\hat{s}) \geqq \langle s_j, -\hat{s} \rangle$

If $\alpha_j > 0$, by theorem 3.2.3, $\langle s_j, -\hat{s} \rangle - \underbrace{\mu e_j}_{=0} = -\|\hat{s}\|^2 - \mu\varepsilon$

Hence $f'(x, -\hat{s}_\varepsilon) \geqq \langle s_j, -\hat{s} \rangle = -\|\hat{s}\|^2 - \mu\varepsilon$. (*)

If all the extreme points of $\partial f(x)$ appear among the ℓ elements of the bundle, then $\partial f(x) = \text{conv}\ \{s_1, \ldots, s_\ell\} \subseteq S(0)$.

Thus for each j, $\langle s_j, -\hat{s} \rangle < -\|\hat{s}\|^2 - \mu\varepsilon$,

This implies $\left\langle \underbrace{\sum_{j=1}^{\ell} \alpha_j s_j, -\hat{s}}_{1} \right\rangle \leqq \underbrace{\sum_{j=1}^{\ell} \alpha_j}_{1} (-\| \hat{s} \|^2 - \mu\varepsilon) = -\| \hat{s}\|^2 - \mu\varepsilon.$

That is $\langle s, -\hat{s} \rangle \leq -\| \hat{s}\|^2 - \mu\varepsilon$ for all $s \in \partial f(x)$. Hence we get $f'(x, -\hat{s}) = \sup \langle s, -\hat{s} \rangle$ $s \in \partial f(x) \} < -\| \hat{s}\|^2 - \mu\varepsilon$ (**)

From (*) and (**), we get $f'(x, -\hat{s}) = -\| \hat{s}\|^2 - \mu\varepsilon$.

The Role of the Cutting plane Function.

We try to find a function having $S'(\varepsilon)$ as approximate sub differential so that $d = -\hat{s}_\varepsilon$ would be an ε-descent direction for it.

Definition 3.3.1 The cutting plane function associated with the bundle is given by

$$\hat{f}(z) = f(x) + \max\{-e_i + \langle s_i, z - x\rangle : i \in \{1, \ldots, \ell\}\}$$

$$= \max\{f(y_i) + \langle s_i, z - y_i\rangle : i \in \{1, \ldots, \ell\}\}, \forall z \in \Re^n.$$

For each $i \in \{1, \ldots, \ell\}$, if we set

$$\hat{f}(z) = f(x) - e_i + \langle s_i, z - x\rangle = f(y_i) + \langle s_i, z - y_i\rangle \ \forall z \in \Re^n.$$

then $f(z) \geqq f(y_i) + \langle s_i, z - y_i\rangle = \hat{f}_i(z) \ \forall z \in \Re^n$

i.e $\hat{f}_i \leqq f$ for each $i = 1, \ldots, \ell$.

Hence $\hat{f} = \max\{\hat{f}_i \mid i = 1, \ldots, \ell\}$ is a piece wise affine minorization of f and we have $\partial_\varepsilon \hat{f}(x) = S(\varepsilon)$ for all $\varepsilon \geqq 0$.

Theorem 3.3.3. For $\mu > 0$, we have $f(x) - \varepsilon - \frac{1}{\mu}\|\hat{s}\|^2 = \hat{f}(x - \frac{1}{\mu}\hat{s}_\varepsilon)$.

Proof: For $0 < \mu \in M_\varepsilon$, by (3.2.4) we have

$$\langle s_i, \hat{s}\rangle + \mu e_i \geqq \|\hat{s}\|^2 + \mu\varepsilon \qquad (*)$$

Multiplying (*) by $-\frac{1}{\mu}$, we get

$$\langle s_i, \frac{1}{\mu}\hat{s}\rangle - e_i \leqq -\frac{1}{\mu}\|\hat{s}\|^2 - \varepsilon.$$

i.e $-\frac{1}{\mu}\|\hat{s}\|^2 - \varepsilon \geqq -e_i + \langle s_i, -\frac{1}{\mu}\hat{s}\rangle = \bar{f}_i(x - \frac{1}{\mu}\hat{s}) - f(x)$

This implies $f(x) = -\frac{1}{\mu}\|\hat{s}\|^2 - \varepsilon \geqq \bar{f}_i(x - \frac{1}{\mu}\hat{s})$.

But $\alpha \in \Delta_\ell$ implies there is at least one i such that $\alpha_i > 0$ and hence equality results in (3.2.4).

i.e $f(x) - \frac{1}{\mu}\|\hat{s}\|^2 - \varepsilon \geqq \bar{f}_i(x - \frac{1}{\mu}\hat{s})$.

Hence $\hat{f}(x-\frac{1}{\mu}\hat{s}_\varepsilon) = \max\{\bar{f}_i(x-\frac{1}{\mu}\hat{s}): i = 1, \ldots, \ell\}$

$$= f(x) - \frac{1}{\mu}\|\hat{s}\|^2 - \varepsilon$$

$$= f(x) - \left[\frac{1}{\mu}\|\hat{s}\|^2 + \varepsilon\right]$$

4. The Implementable Algorithm

4.1 Derivation of the Line-Search

Suppose that at the iteration point $x = x_k \in \Re^n$, we are given the current direction $d = d_k = \hat{s}_\varepsilon$ (which is obtained from the minimization problem (3.1.4), $\varepsilon = \varepsilon_k > 0$ and the multiplier $\mu = \mu_k \geq 0$ of the constraint (iii) in (3.1.4). Below we describe the line-search in the direction d, which will produce a suitable step size $t_k > 0$ such that $y_+ := x_{k+1} = x_k + t_k d_k$ and $s_+ \in \partial f(y_+)$.

As we shall see later, the line-search has a double role which must either find a **descent-step** or **a null-step** yielding an $S_+(\varepsilon)$ richer than $S(\varepsilon)$ but with no better y_+.

Just as any iterative algorithm, the line-search has a rule for iteration and a break off criterion. The break off criterion has to be organized in such a way that it also gives some indications for the next choice of stepsize in case the present stepsize is not suitable. Accordingly, we must test each given stepsize $t > 0$ with the following three possible cases.

Case (O) The given t is suitable and the line-search can be stopped.

Case (R) The given t is not suitable and no suitable $\bar{t}$ should be searched in (t, ∞).

Case (L) The given t is not suitable and no suitable $\bar{t}$ should be search in $(0, t)$

The alternatives suitable descent – suitable null-steps are included in case (0). So, we must answer the following two questions.

i) When do we accept a given stepsize $t > 0$ as a suitable descent-step?

ii) When do we accept a given stepsize $t > 0$ as a suitable null-step?

To answer these questions, we require at least a reasonable estimate of $f'(x, d)$, and also that of $f'(y_+, d)$. Since f is convex, the difference quotient $\frac{f(x+td)-f(x)}{t}$ is increasing and $\frac{f(x+td)-f(x)}{t} < 0$. Thus

$f'(x, d) = \lim_{t \downarrow 0} \frac{f(x+td)-f(x)}{t}$ is less than zero.

hence for the moment, suppose that $\tilde{v} < 0$ be a number playing the role of $f'(x, d)$ and let $m \in (0, 1)$. (We shall see how $\tilde{v}$ can be computed later)

Then, if $t > 0$ does not satisfy the descent test

$$f(x + td) \leqq f(x) + m\, t\tilde{v} \qquad (4.1.1)$$

then case (R) occurs.

A null-step is useful when f can hardly be decreased in the direction d because of the fact that $S(\varepsilon)$ approximate $\partial_\varepsilon f(x)$ prooly. Then the role of the line-search is to improve this approximation with a new element (s_+, e_+) so as to get $S_+(\varepsilon) \supset S(\varepsilon)$ where $e_+ := e(x, y_+, s_+)$. In this case a new projection $\hat{s}_+$ will be computed, a new line-search will be performed from the same x. From theorem 3.2.3, we must have

$$\langle s_+, \hat{s} \rangle + \mu e_+ < \| \hat{s} \|^2 + \mu\varepsilon \qquad (4.1.2)$$

It is clear that for (4.1.2) to hold, e_+ must be sufficiently small. Thus, if no descent direction is obtained, in addition to $\tilde{v}$, another parameter $\tilde{\varepsilon} > 0$ is required which must at least produce an $s_+ \in \partial_\varepsilon f(x)$ to enrich the bundle. For this it suffices to take

$$e_+ := e(x, y_+, s_+) = f(x) - f(y_+) + t\langle s_+, d\rangle \leqq \tilde{\varepsilon} \qquad (4.1.3)$$

For (4.1.2) to hold, we must also have $\langle s_+, \hat{s} \rangle = -\langle s_+, d\rangle$ sufficiently small. So, we require

$$\langle s_+, d \rangle \geqq m'\tilde{v}, \qquad (4.1.4)$$

for some $m' \in (m, 1)$.

The choice of $\tilde{v}$ and $\tilde{\varepsilon}$ has to be in such a way that

$$\langle s_+, \hat{s} \rangle < -m'\tilde{v} \text{ and } e_+ < \tilde{\varepsilon} \text{ implies } \langle s_+, \hat{s} \rangle + \mu e_+ < \|\hat{s}\|^2 + \mu\varepsilon.$$

For this to hold, it suffices to choose $\tilde{v}$ and $\tilde{\varepsilon}$ such that

$$-m'\tilde{v} + \mu\tilde{\varepsilon} < \|\hat{s}\|^2 + \mu\varepsilon. \qquad (4.1.5)$$

The inequality (4.1.5) is an important inequality, which we have to keep in mind when choosing the tolerances. Finally, we come to case (L). A stepsize $t > 0$ is considered as "not too small" when the inequality (4.1.3) does not hold. In the case (4.1.3) dos not hold, a stepsize $t > 0$ is accepted as a descent-step if the inequality (4.1.1) holds; otherwise case (L) occurs.

To sum up, the break off criterion for the line-search will be as follows. Given the tolerances $\tilde{v} < 0$, $\tilde{\varepsilon} > 0$ satisfying (4.1.5) and the coefficients $0 < m < m' < 1$, accept the stepsize $t > 0$ with $y_+ = x + td$, $s_+ \in \partial f(y_+)$ and $e_+ = e(x, y_+, s_+)$ when:

$$f(y_+) \leqq f(x) + mt\tilde{v} \text{ and } e_+ > \tilde{\varepsilon} \text{ (descent-step)} \qquad (4.1.6)$$

$$\text{or} \quad e_+ \leqq \tilde{\varepsilon} \text{ and } \langle s_+, d \rangle \geqq m'\tilde{v} \text{ (null-step)} \qquad (4.1.7)$$

4.2 The Implementable Line-Search and its convergence

The general organization of the line-search algorithm will be as follows.

(0) t is convenient when it satisfies (4.1.6) or

(4.1.7) with an appropriate exit-case;

(R) t is called t_R when (3.1.1) does not hold;

(L) t is called t_L in all other Cass.

The following truth table specifies the decision made in each possible combination of (4.1.1), (4.1.3) and (4.1.4) corresponding to the above rules.

(4.1.1)	(4.1.3)	(4.1.4)	Decision
T	T	T	(O) (null)
T	T	F	(L)
T	F	T	(O) (descent)
T	F	F	(O) (descent)
F	T	T	(O) (null)
F	T	F	(R)*
F	F	T	(R)
F	F	F	(R)*

Table 4.2.1 Exhaustive analysis of the test

In the table (R)* means an impossible case, ruled out by convexity.

Algorithm 4.2.1 (Line-search; Non smooth case)

Given the initial point $x \in \Re^n$, stepsize $t > 0$, the direction $d \in \Re^n$, the tolerances $\tilde{v} < 0$, $\tilde{\varepsilon} > 0$ and coefficients $m \in (0, 1)$ $m' \in (m, 1)$. set $t_L = 0$, $t_R = 0$.

STEP 1. Compute $f(x + td)$ and $s = s(x + td)$; compute $e = e(x, x + td, s)$

STEP 2. (test for a null step). If (4.1.3) and (4.1.4) hold, stop the line search with a null-step otherwise go to step 3.

STEP 3. (test for large t). If (4.1.1) does not hold, set $t_R = t$ and go to step 6. Otherwise go to step 4.

STEP 4. (test for a descent-step; t is not too large). If (4.1.3) does not hold, stop the line-search with a descent-step. Otherwise set $t_L = t$ and to go step 5.

STEP 5. (extrapolation). If $t_R = 0$, find a new t by extrapolation beyond t_L and loop to Step 1. Otherwise go to Step 6.

STEP 6 (interpolation). Find a new t by interpolation in (t_L, t_R) and loop to step 1

Property 4.2.1 The update of t by Interpol (t_L, t_R) must be such that:

(i) infinitely many extrapolations would let $t_L \to \infty$,

(ii) infinitely many interpolations would let $t_R - t_L \to 0$

said differently, at each cycle, the extrapolation interpolation formula must guarantee a definite decrease of the bracket $[t_L, t_R]$, or a definite increase of t_L.

For example, one possibility is to replace t by 10t in case of extrapolation and by $\frac{1}{2}(t_L + t_R)$ in case of interpolation.

Algorithm 4.2.1 is illustrated by the following chart

Given: q(t): = f(x + td), q(0) $\tilde{v} < 0 < m < m' < 1$, $\tilde{\varepsilon} > 0$, $t > 0$.

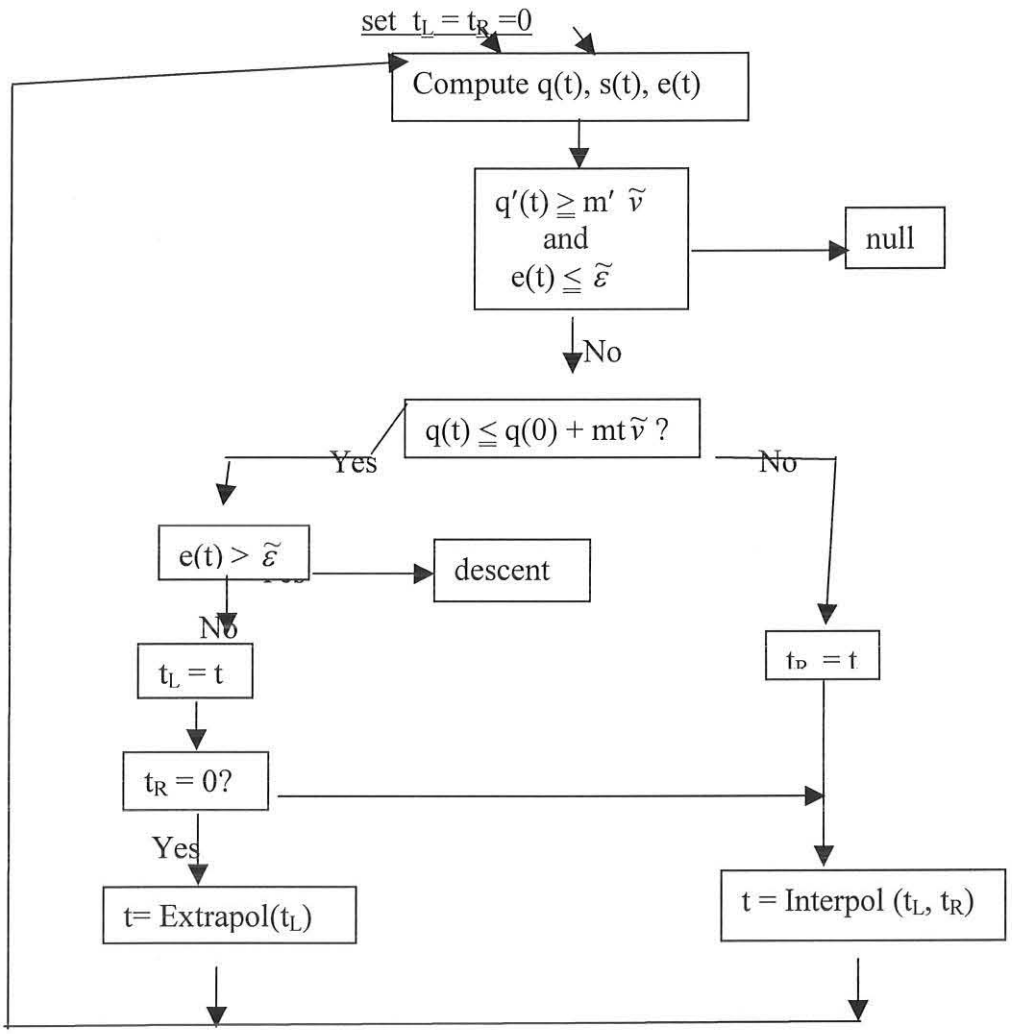


Fig 4.2.2 Line-Search with Provision for null-step

Theorem 4.2.2 Let $f:\Re^n \rightarrow \Re$ be convex, and assume that property 4.2.1 holds. Then Algorithm 4.2.1 either generates a sequence of stepsize $t \rightarrow \infty$ with $f(x + td) \rightarrow -\infty$, or terminates after a finite number of cycles, with a null-or a descent-step. (convergence of the line-search)

Proof: Suppose that the stop never occurs, neither in step 2 nor in step 4. We observe that, at each cycle,

$$f(x + t_L d) < f(x) + mt_L \tilde{v} \qquad (1)$$

[Extrapolations] suppose first that Algorithm 4.2.1 loops indefinitely between step 5 and step 1: no interpolation is even made. Then, by construction, every generated t is a t_L and tends to ∞. Because $\tilde{\nu} < 0$, (1) shows that $f(x + t_L d) \to -\infty$. [Interpolations] Thus, if f(x + td) is bounded from below, t_R becomes positive at some cycle. From then on, the algorithm loops between step 6 and step 1. By construction, at each subsequent cycle,

$$f(x + t_R d) > f(x) + m t_R \tilde{\nu} ; \qquad (2)$$

the sequence $\{t_L\}$ is increasing, the sequence $\{t_R\}$ is decreasing $t_L \leqq t_R$, and property 4.2.1 implies that these two sequences have a common limit, say $\bar{t} \geqq 0$. Since f is continuous, (1) and (2) implies that $f(x + \bar{t}\, d) = f(x) + m\bar{t}\ \tilde{\nu}$ (3)

Case 1: $\bar{t} = 0$

From (2) and (3), we see that $t_R > \bar{t}$.

This implies

$$f'(x, d) = \lim_{t_R \downarrow 0} \frac{f(x + t_R d) - f(x)}{t_R} \geqq m\tilde{\nu} > m'\tilde{\nu}$$

This is a contradiction: indeed, for $t_R \downarrow 0$,

$\langle s_R, d \rangle \to f'(x,d) > m'\tilde{\nu}$ and $e_R \to 0$,

i.e. (4.1.3) and (4.1.4) are satisfied by t_R sufficiently small; the break off criterion for step 2 must eventually be satisfied.

Case 2. $\bar{t} > 0$

In this case t_L becomes positive after some cycle.

From $\langle s_L, d \rangle \geqq \dfrac{f(x + t_L d) - f(x)}{t_L}$, taking the limit as the number of cycles goes to ∞, we see that the right-hand side tend to $m\tilde{\nu} > m'\tilde{\nu}$.

Thus, after some cycle, (4.1.4) is satisfied forever by $s_+ = s_L$. On the other hand, each t_L satisfies (4.1.3) (otherwise, a descent step is found); we therefore obtain again a contradiction: t_L should be accepted as a null-step.

4.3 Derivation of the Descent Algorithm

Based on the developments we made in previous sections, the general organization of the minimization algorithm can be described as follows: At each iteration, for a chosen ε, we solve the direction finding problem (3.1.4) to obtained a primal-dual solution (α, μ), and $\hat{s} = s(\alpha)$. If $\| \hat{s} \|$ is sufficiently small, the inequality $f(z) \geqq f(x) - \|s(\alpha)\| \|z-x\| - e(\alpha)$, $Z \in \Re^n$ gives a minimality estimate. Otherwise the line-search (4.2.1) is done in the direction $d = -\hat{s}$ to obtain the corresponding results. To complete the description of the algorithm, in the following, we specify the line-search parameters, stopping criterion and the management of the bundle.

(a) Line-Search Parameters ($\tilde{\nu}$ and $\tilde{\varepsilon}$ for (4.1.6) and (4.1.7)

As described in section 4.1, the purpose of $\tilde{\varepsilon}$ is to find $s_+ \in \partial_\varepsilon f(x_k) \backslash S(\varepsilon)$. Hence we set

$$\tilde{\varepsilon} = \tilde{m}\ \varepsilon, \tag{4.3.1}$$

for some $\tilde{\varepsilon} \in (0, 1]$.

Concerning the choice of $\tilde{\nu}$, we have the following two strategies which depend on the role of $\tilde{\nu}$ in the algorithm.

i) If the aim is to decrease f by a fraction of its initial slope $f'(x, d)$, from proposition (3.3.4) it can be replaced by

$$\tilde{\nu} = -\| \hat{s} \|^2 - \mu\varepsilon . \tag{4.3.2}$$

In this case if $f'(x, d) < -\| \hat{s} \|^2 - \mu\varepsilon$, a descent step will be easily found, otherwise we get a null step.

ii) If the aim is to force (separate) s_+ away from $S(\varepsilon)$, from proposition (3.3.2), $\tilde{\nu}$ can be replace by

$$\tilde{\nu} = -\| \hat{s} \|^2. \tag{4.3.3}$$

Since the above two choices of $\tilde{\nu}$ are logically equivalent we will keep both possibilities open.

(b) Stopping Criterion

The rational for the stopping criterion is the inequality $f(z) \geq f(x) + \langle \hat{s}_\varepsilon, z - x \rangle - e(\alpha) > f(x) - \|\hat{s}_\varepsilon\| \|z - x\| - \varepsilon$, for all $z \in \Re^n$: when the projection $\hat{s}$ is sufficiently close the zero, x_k minimizes f within ε_k (at least approximately). However, due to the double role played by ε(that is to stop the algorithm as well as to compute the descent direction), the minimization algorithm cannot be stopped directly when $\hat{s} \cong 0$. We must still check that the minimality condition is tight enough, if not, ε must be reduced and the projection recomputed. Accordingly, the general strategy will be as follows. First, a tolerance $\delta > 0$ is chosen, and the event "$\hat{s} \cong 0$" is quantified by "$\|\hat{s}\| \leq \delta$". This is for the reason of implement ability of the algorithm. Concerning ε, the following three values are maintained through out the algorithm.

(i) The Current value denoted by $\varepsilon = \varepsilon_k$, which is used to solve the direction finding problem.

(ii) The value $\underline{\varepsilon} > 0$ is the final tolerance, or the smallest useful valuc for ε_k. It is fixed for the entire algorithm, which is stopped as soon as (3.1.4) is used with $\varepsilon = \underline{\varepsilon}$ and produces an $\hat{s}$ such that $\|\hat{s}\| \leq \delta$.

(iii) The value $\bar{\varepsilon}$ represents the best estimate

$$f(y) > f(x) - \delta\|y - x\| - \bar{\varepsilon}$$

obtained so far.

Each time we obtain a solution $\hat{s}$ of the minimization Problem (3.1.4) with $\|\hat{s}\| \leq \delta$, $\bar{\varepsilon}$ is decreased to the current ε.

In practice, having the current $\bar{\varepsilon} \geq \varepsilon$, the minimization problem (3.1.4) is solved at each iteration with an $\varepsilon \in [\underline{\varepsilon}, \bar{\varepsilon}]$. The successive values $\bar{\varepsilon}$ forming a decreasing sequence $\{\bar{\varepsilon}_k\}$ which is minorized by $\underline{\varepsilon}$.



It is clear that the first direction at the initial point x_1 does not depend on ε. Also when the current iteration results in a null-step, there is no need to choose ε for the next iteration (since the next iteration is performed with the same ε from the same x).

Hence only when the current iteration results in a descent-step that we have to fix $\varepsilon \in [\underline{\varepsilon}, \bar{\varepsilon}]$ for the next iteration. In this case, the choice of $\varepsilon \in [\underline{\varepsilon}, \bar{\varepsilon}]$ has a great influence on the numerical behavior of the algorithm as we shall see later.

Thus, after a descent-step, a new $\varepsilon = \varepsilon_{k+1} \in [\underline{\varepsilon}, \bar{\varepsilon}]$ is chosen to compute the direction d_{k+1} issuing from the new $x_{k+1} \neq x_k$. (This choice of ε will not be specified here)

c) Management of the Bundle

At every 'normal" iteration, the line-search has produced either a null-step or a decent-step with $y_+ = x_k + t_k d_k$ and $s_+ \in \partial f(y_+)$ found by the k^h line-search. Thus a new element (s_+, e_+) is appended to the bundle. Concerning e_+, we have the following two cases.

Case 1. The line-search had produced a descent-step. Then $x_{k+1} = y_+$ and $e_+ = (x_{k+1}, y_+, s_+) = 0$ since $s_+ \in \partial f(y_+)$. In this case because x_k is moved to $x_{k+1} \neq x_k$, we must also update the old linearlization errors in the bundle by the help of proposition 2.2: for each index i in the old bundle the old linearization error is changed to $e_i + f(x_{k+1}) - [f(x_k + t_k d_k) + \langle s_i, d_k \rangle]$.

This appending process can not go forever as the mamimal bundle size $\bar{\ell}$ is finite. hence at some iteration point x_k, we must find a place in the bundle for the new element (s_+, e_+) as well as the element $(\hat{s}, \hat{e})$ where $\hat{s} := \sum_{i=1}^{\ell} \alpha_i s_i \in \partial_\varepsilon f(x_k) \subseteq \partial_\varepsilon f(x)$

and $\hat{e} := \sum_{i=1}^{\ell} \alpha_i e_i \leqq \varepsilon$, for some optimal α in (3.1.4). The purpose of appending the element $(\hat{s}, \hat{e})$ is to extract essential information from the bundle as much as possible and also to ensure the convergence of $\|\hat{s}\|$ to 0, which is based on the fact that $\hat{s} \in S_+(\varepsilon)$ at least in the case of a null-step. (cf. [2])

To find place in the bundle for (s_+, e_+) and $(\hat{s}, \hat{e})$ (once we fix the maximal bundle size $\bar{\ell} \geqq z$), whenever the current bundle size $\ell = \bar{\ell}$, at least two elements have to be removed from the current bundle. Then we append $(s_{\ell-1}, e_{\ell-1}) = (\hat{s}, \hat{e})$, and $(s_\ell, e_\ell) = (s_+, e_+)$ (4.3.4)

(with possible renumbering)

Remark 4.3.1 If all the elements removed from the bundle have their α components $\alpha_i = 0$ from the quadratic problem (3.1.4), then $(\hat{s}, \hat{e})$ brings nothing to the definition of $S(\varepsilon)$. In that case, no aggregation is necessary, only one element has to be removed from the bundle.



Lemma 4.3.1 Let $s' \in \partial_\varepsilon f(x')$. Then $s' \in \partial_\varepsilon f(x)$ if $e(x, x', s') + \varepsilon' \leqq \varepsilon$

Proof: Fir $s' \in \partial_\varepsilon f(x')$, we have for all $y \in \Re^n$,

$$f(y) \geqq f(x') + \langle s', y - x' \rangle - \varepsilon'.$$

Then if $e(x,x',s') + \varepsilon' \leqq \varepsilon$, then we get

$$f(x') \geqq f(x) + \langle s', x', -x \rangle - \varepsilon + \varepsilon'$$

Hence it follows that

$$f(y) \geqq f(x') + \langle s', y-x' \rangle - \varepsilon'$$

$$\geqq f(x) + \langle s', y-x \rangle - \varepsilon \quad \text{for all } y \in \Re^n$$

This implies that $s' \in \partial_\varepsilon f(x)$.

Lemma 4.3.2 Let $(\hat{s}_\varepsilon, \hat{e}^k) = (\hat{s}, \hat{e})$ be the aggregate element at the k^{th} iteration. Then for all $x' \in \Re^n$, we have $\hat{s}_k \in \partial_\varepsilon f(x')$,
where $\hat{e}' := \hat{e}^k + f(x') - f(x^k) - \langle \hat{s}_k, s' - x_k \rangle$.

Proof: Follows directly from Lemma 4.3.1.
By this Lemma, the aggregate weight $\hat{e}$ will be updated according to proposition 2.2, since it is identical to any other element of the bundle coming directly from (U_1).

4.4 The Implementable Algorithm and its convergence.

Now we come to the main objective of the seminar report; that is the algorithm refered to in section 1. For this we made all the necessary developments in previous sections. Based on these developments, we are now in a position to give a detailed description of the minimization algorithm. We start the description with the following remark concerning the line-search Parameters.

Remark 4.4.1 The line-search parameters $\tilde{\varepsilon}$ (i.e $\tilde{m}$) and $\tilde{v}$ are not totally independent of each other: They should satisfy (4.1.5); otherwise (4.1.2) which is essential in the case of a null-step, many not hold. If $\tilde{v} = -\|\hat{s}\|^2$ as it was given in (4.3.3), then we have

$$\begin{aligned} -m'\tilde{v} + \mu\tilde{\varepsilon} &= -m'(-\|\hat{s}\|^2) + \mu\tilde{\varepsilon} \\ &= m'\|\hat{s}\|^2 + \mu\tilde{m}\,\varepsilon, \text{ since } \tilde{\varepsilon} = \tilde{m}\,\varepsilon \text{ by } (4/\varepsilon 1) \\ &\leqq \max\{m', \tilde{m}\}\,[\|\hat{s}\|^2 + \mu\varepsilon]. \end{aligned}$$

Hence if we choose simply $0 < m', \tilde{m} < 1$, the inequality $-m'\tilde{v} + \mu\tilde{\varepsilon} < \|\hat{s}\|^2 + \mu\varepsilon$ as it was given by (4.1.5) holds. This implies the inequality $\langle s_+, \hat{s} \rangle + \mu e_+ < \|\hat{s}\|^2 + \mu\varepsilon$, which was given by (4.1.2) does hold.

On the other hand, if $\tilde{v} = -\|\hat{s}\|^2 - \mu\varepsilon$ as it was given by (4.3.2), then we have

$$\begin{aligned} -m'\tilde{v} + \mu\tilde{\varepsilon} &= -m'(-\|\hat{s}\|^2 - \mu\varepsilon) + \mu\tilde{\varepsilon} \\ &= m'\|\hat{s}\|^2 + m'\mu\varepsilon + \mu\tilde{m}\,\varepsilon. \\ &= m'\|\hat{s}\|^2 + (m' + \tilde{m})\mu\varepsilon \end{aligned}$$

To see that the case $m' + \tilde{m} > 1$ is dangerous: the new subgradient s_+ may belong to the old $S(\varepsilon)$ (that is to mean (4.1.5) may not hold). This may also imply that the inequality (4.1.2), which is necessary to obtain an $S_+(\varepsilon)$ richer than $S(\varepsilon)$ may not hold; in the case of a null step, $\hat{s}$ and μ will stay the same: the minimization algorithm enters a disastrous loop.

Hence to be on the safe side, if $\tilde{v}$ is given by (4.3.2), we take $m' + \tilde{m} \leqq 1$. (4.4.1)
Now the algorithm is given in the following. Notations Such as (1) in the algorithm refers to explanations following the algorithm.

Algorithm 4.4.1 (Bundle Method in Dual Form)[1]

Given that the initial point $x_1 \in \Re^n$ and the value $\bar{\varepsilon} > 0$[2] described in section 4.3(b) (iii); the maximal bundle size $\bar{\ell} \geqq 2$; the tolerances $\underline{\varepsilon} > 0$ as given in section $4.3_{(b)}$(ii) and $\delta > 0$.

Fix the parameters m, m′, $\tilde{m}$ such that $0 < m < m' < 1$ and $0 < \tilde{m} \leqq 1$[3].

Initialize the bundle with $s_1 = s(x_1)$, $e_1 = 0$, the iteration index $k = 1$, the bundle size $\ell = 1$; set $\varepsilon = \bar{\varepsilon}$.

STEP 1 (Computing the direction).

Replace ε by $\max\{\underline{\varepsilon}, \min(\varepsilon, \bar{\varepsilon})\}$[4]. Solve the direction finding Problem (3.1.4) to obtain an optimal solution α and multiplier μ[5]. Set

$$\hat{s} := \sum_{i=1}^{\ell} \alpha_i s_i \,,\quad \hat{e} := \sum_{i=1}^{\ell} \alpha_i s_i \,.$$

STEP 2. (Stopping criterion).

If $\| \hat{s} \| > \delta^{(6)}$ go to step 3. Otherwise:

If $\hat{e} \leqq \underline{\varepsilon}$ stop[7]; or else diminish $\bar{\varepsilon}$ [8] and go to step 1.

STEP 3 (line-search).

Given the current iterate x_k, the current direction $d = d_k = -\hat{s}$, the descent parameter $\tilde{v} = \tilde{v}_k < 0$ and the null-step parameter $\tilde{\varepsilon} = \tilde{\varepsilon}_k = \tilde{m}\,\varepsilon_k = \tilde{m}\,\varepsilon$, perform the line-search (4.2.1) along d.

Obtain a step-size $t > 0$, realizing either a descent-step or a null-step.

STEP 4. (Managing the bundle size)

If the current bundle size ℓ is equal to the maximal bundle size $\bar{\ell}$: delete at least two elements from the bundle; append $(\hat{s}, \hat{e})$ coming from step 1; call $\ell' < \bar{\ell}$ the length of the new list thus obtained.

STEP 5 (appending the new element)

Append $(s_{\ell+1}, e_{\ell+1}) := (s_+, e_+)$ to the bundle where

– in case of a null-step, $e_{\ell+1} := f(x_k + t_d) + t\langle s_t, d\rangle - f(x_k)$

- in case of a descent –step $e_{\ell+1} := 0$ and, for each $i = 1, \ldots, \ell$, change each e_i to

$e_i + f(x_k + t_d) - [f(x_k) + t\,\langle s_i, d\rangle]$.

Choose a new $\varepsilon \in [\underline{\varepsilon}, \bar{\varepsilon}]$[9], replace k by k + 1 and loop to step 1.

Explanation Notes:

1. This name is used because the algorithm has been conceived via a development in the dual space, to construct approximate sub differentials.

2. We can use for $\bar{\varepsilon}$ an estimate of $f(x_1) - \bar{f}$ where $\bar{f} \geq \inf_{x \in \Re^n} f(x)$ is a best approximation of the in fimal value.

3. Suggested values are: $m = \tilde{m} = 0.1$, $m' = 0.2$.

4. The aim of this operation is to make sure that $0 < \underline{\varepsilon} \leq \varepsilon \leq \bar{\varepsilon}$.

5. $\mu \geq 0$ is the Lagrange multiplier associated with the inequality constraint in the quadratic problem (3.1.4).

6. We want to detect $\hat{e}$-optimality of the current x. For this it is a good idea to let δ vary with ε.

   For example if there is an $M > 0$ such that

   $f(y) < f(x) - \hat{e}$ implies $\|y - x\| \leq M$, then we have

   $f(y) \geq f(x) - \hat{e} - M \|\hat{s}\|$ for all $y \in \Re^n$.

   This means that roughly $\hat{e}$-optimality if $M\|\hat{s}\|$ is small compared to $\hat{e}$, say $M\|\hat{s}\| \leq 0.1\,\hat{e}$.

   Hence we can take at each iteration $\delta = \dfrac{0.1}{M} \max\{\underline{\varepsilon}, \hat{e}\}$.

7. It is better to take $\hat{e} = \min\left\{\sum_{i=1}^{\ell} \alpha_i e_i : \alpha \in \Delta_\ell, \sum_{i=1}^{\ell} \alpha_i s_i = \hat{s}\right\}$

8. A sensible strategy is to take $\bar{\varepsilon} = \max\{0.1\,\hat{e}, \underline{\varepsilon}\}$.

9. Between $\underline{\varepsilon}$ and $\bar{\varepsilon}$, a possibly wide range of values are available for ε. Each of them will give a different direction, thus conditioning the efficiency of the algorithm. Knowing a rough idea of sensible values for ε is not enough to use full efficiency of the algorithm, as full efficiency needs more accurate values.

On Convergence of the algorithm

Lemma 4.4.1 Let f: $\Re^n \rightarrow \Re$ be convex and assume that the line-search terminates at each iteration. Then, either $f(x_k) \xrightarrow{k\to\infty} -\infty$, or the number of descent steps is finite. Cf. [3]

Lemma 4.4.2 The assumptions are those of Lemma; assume also $m' + \widetilde{m} \leq 1$. Then, between two descent – steps, only finitely many consecutive null-steps can be made without reducing $\bar{\varepsilon}$ in setp 2. Cf. [3].

Theorem 4.4.3 Let $f : \Re^n$ be convex. Assume for example that $\bar{\varepsilon}$ is divided by 10 at each loop from step 2 to step 1 of Algorithm; if $\tilde{v}$ is given by ( ), assume also $m' + \widetilde{m} \leq 1$. Then, either of the following is true.

i) $f(x_k) \xrightarrow{k\to\infty} -\infty$,

ii) or the line-search detects at some iteration k that f is unbounded from below,

iii) or the stop occurs in step 2 for some $k \in N$, at which there holds

$$f(y) \geq f(x_k) - \underline{\varepsilon} - \delta\|y - x_k\| \text{ for all } y \in \Re^n .$$

Proof: Follows from Lemma 4.4.1 and Lemma 4.4.2.

References

[1] Deumlich, R. :Optimization and Theory of Approximation.
Textbook for Lecture (Math 651), Addis Ababa, 1997

[2] Lemaréchal, C. : Convex Analysis and Minimization Algorithms I.
Springer, Berlin Heidelberg, 1993

[3] Lemaréchal, C. : Convex Analysis and Minimization Algorithms II.
Springer, Berlin Heidelberg, 1993