4OR manuscript No.
(will be inserted by the editor)

Essentials of numerical nonsmooth optimization

Manlio Gaudioso · Giovanni Giallombardo · Giovanna Miglionico

Abstract Approximately sixty years ago two seminal findings, the cutting plane and the subgradient methods, radically changed the landscape of mathematical programming. They provided, for the first time, the practical chance to optimize real functions of several variables characterized by *kinks*, namely by discontinuities in their derivatives. Convex functions, for which a superb body of theoretical research was growing in parallel, naturally became the main application field of choice.

The aim of the paper is to give a concise survey of the key ideas underlying successive development of the area, which took the name of numerical nonsmooth optimization. The focus will be, in particular, on the research mainstreams generated under the impulse of the two initial discoveries.

Keywords Nonsmooth optimization; Cutting plane; Subgradient method; Bundle method.

1 Introduction

Nonsmooth optimization (NSO), sometimes referred to as Nondifferentiable optimization (NDO), deals with problems where the objective function exhibits *kinks*. Even though smoothness, that is the continuity of the derivatives, is present in most of the functions describing real world decision making processes, an increasing number of modern and sophisticated applications of optimization are inherently nonsmooth. The most common source of nonsmoothness is in the choice of the

M. Gaudioso (✉)
DIMES, Università della Calabria, Italia
E-mail: gaudioso@dimes.unical.it

G. Giallombardo
DIMES, Università della Calabria, Italia
E-mail: giovanni.giallombardo@unical.it

G. Miglionico
DIMES, Università della Calabria, Italia
E-mail: gmiglionico@dimes.unical.it

worst-case analysis as a modeling paradigm. It results in choosing objective functions of the *max* or, alternatively, of the *min* type, thus in stating *minmax* or *maxmin* problems, respectively. Nonsmoothness typically occurs whenever solution of the inner maximization (or minimization) is not unique. Although such phenomenon is apparently rare, nevertheless its occurrence might cause failure of the traditional differentiable optimization methods when applied to nonsmooth problems. Among other areas where nonsmooth optimization problems arise we mention here:

– *Minmaxmin* models, coming from worst–case–oriented formulations of problems where two types of decision variables are considered, "here and now" and "wait and see", respectively, with in the middle the realization of a scenario taken from a set of possible ones.
– *Right-hand-side* decomposition of large scale problems (e.g., multicommodity flow optimization) where the decomposition into subproblems is controlled by a *master* problem which assigns resources to each of them. In such framework, the objective function of the master is typically nonsmooth.
– *Lagrangian relaxation* of Integer or Mixed-Integer programs, where the Lagrangian dual problem, tackled both for achieving good quality bounds and for constructing efficient Lagrangian heuristics, consists in the optimization of a piecewise affine (hence nonsmooth) function of the multipliers.
– *Variational inequalities* and *nonlinear complementarity problems*, which benefit from availability of effective methods to deal with systems of nonsmooth equations.
– *Bilevel problems*, based on the existence, as in Stackelberg's games, of a hierarchy of two autonomous decision makers. The related optimization problems are non-differentiable.

Although the history of nonsmooth optimization dates back to Chebyshëv and his contribution to function approximation [26], it was in the sixties of last century when mathematicians, mainly from former Soviet Union, started to tackle the design of algorithms able to numerically locate the minima of functions of several variables, under no differentiability assumption. The *subgradient* was the fundamental mathematical tool adopted in such context. We recall here the contributions by Shor [137], Demyanov and Malozemov [32], Polyak [126], and Ermolev [44].

Based on quite a different philosophy, as it will be apparent in the following, a general method able to cope with nondifferentiability was devised, independently, by Kelley [79] and by Cheney and Goldstein [28]. Instead of trusting on a *unique* subgradient, the approach consisted in the simultaneous use of the information provided by *many* subgradients. The parallel development of convex analysis, thanks to contributions by Fenchel, Moreau and Rockafellar, was providing, at that time, the necessary theoretical support.

A real breakthrough took place approximately in the mid seventies, when the idea of an iterative process based on information accumulation did materialize in the methods independently proposed by Lemarechal [91] and Wolfe [145]. From those seminal papers an incredibly large number of variants flourished, under the common label of *bundle* type methods. This family of methods, originally conceived for treatment of the convex case, was appropriately enriched by features able to cope with non convexity.

In more recent years, motivated by the interest in solving problems where exact calculation of the objective function is either impossible or computationally costly, several methods based on its approximate computation were devised. At this time the *derivative free* philosophy is successfully stepping in the nonsmooth optimization world.

Establishing a taxonomy of methods in such a rich area is a difficult and somehow arbitrary task. We will adopt the following, imperfect scheme, defining a classification in terms of methods based on *single-point* information and those grounded on *multi-point* models. All subgradient-related methods, ranging from classic fixed step one to recent accelerated versions, belong to the first group, while in the second group we will comprise the cutting-plane related approaches, including bundle methods and their variants. We will see, however, that even in the multi-point approaches, to paraphrase Orwell in his *Animal farm*, all points are equal, but some points are more equal than others.

The methods grounded on inexact function and/or subgradient evaluation will be also cast in the above framework. Some other methods, that can hardly fit the proposed scheme, will be treated separately.

We confine ourselves to the treatment of convex unconstrained optimization problems. When appropriate, we will also focus on the extension of some algorithms to nonconvex Lipschitz functions, or to special classes of nonconvex functions, such as the Difference-of-Convex (DC) ones.

The paper is organized as follows. After stating the main NSO problem, the relevant notation, and some basic theoretical background in Section 2, we introduce the NSO mainstreams in Section 3. In Sections 4 and 5 we discuss, respectively, about the methods based on single-point and multi-point models. Some classes of algorithms hard to classify into the two mainstreams are surveyed in Section 6. Motivations and issues related to the use of inexact calculations are discussed in Section 7, while in Section 8 some possible extensions of convex methods to the nonconvex case are reported. We give only the strictly necessary references in the main body of this survey, postponing to the final Section 9 more detailed bibliographic notes and complementary information, along with few relevant reading suggestions.

2 Preliminaries

We consider the following unconstrained minimization problem

$$\min \big\{ f(\mathbf{x}) : \mathbf{x} \in \mathbb{R}^n \big\}, \tag{1}$$

where the real-valued function $f : \mathbb{R}^n \to \mathbb{R}$ is assumed to be convex and not necessarily differentiable (nonsmooth), unless otherwise stated. We assume that f is finite over $\mathbb{R}^n$, hence it is proper. Besides, in order to simplify the treatment, we assume that f has finite minimum f^* which is attained at a nonempty convex compact set $M^* \subset \mathbb{R}^n$. An unconstrained minimizer of f, namely any point in M^*, will be denoted by $\mathbf{x}^*$. For a given $\epsilon > 0$, an ϵ-approximate solution of (1) is any point $\mathbf{x} \in \mathbb{R}^n$ such that $f(\mathbf{x}) < f^* + \epsilon$. Throughout the paper, the symbol $\|\cdot\|$ will indicate the ℓ_2 norm, while for any given two vectors $\mathbf{a}, \mathbf{b}$, their inner product will be denoted by $\mathbf{a}^\top \mathbf{b}$.

Next, the fundamental tools of nonsmooth optimization are briefly summarized. Further definitions and relevant findings will be recalled at later stages as they will be necessary.

Given any point $\mathbf{x} \in \mathbb{R}^n$, a *subgradient* of f at $\mathbf{x}$ is any vector $\mathbf{g} \in \mathbb{R}^n$ satisfying the following (subgradient-)inequality

$$f(\mathbf{y}) \geq f(\mathbf{x}) + \mathbf{g}^\top (\mathbf{y} - \mathbf{x}) \quad \forall \mathbf{y} \in \mathbb{R}^n. \tag{2}$$

The *subdifferential* of f at $\mathbf{x} \in \mathbb{R}^n$, denoted by $\partial f(\mathbf{x})$, is the set of all the subgradients of f at $\mathbf{x}$, i.e.,

$$\partial f(\mathbf{x}) \triangleq \left\{\mathbf{g} \in \mathbb{R}^n : f(\mathbf{y}) \geq f(\mathbf{x}) + \mathbf{g}^\top (\mathbf{y} - \mathbf{x}) \ \forall \mathbf{y} \in \mathbb{R}^n\right\}. \tag{3}$$

At any point $\mathbf{x}$ where f is differentiable, the subdifferential reduces to a singleton, its unique element being the ordinary gradient $\nabla f(\mathbf{x})$.

Previous definitions are next generalized for any nonnegative scalar ϵ. An ϵ-*subgradient* of f at $\mathbf{x}$, is any vector $\mathbf{g} \in \mathbb{R}^n$ fulfilling

$$f(\mathbf{y}) \geq f(\mathbf{x}) + \mathbf{g}^\top (\mathbf{y} - \mathbf{x}) - \epsilon \quad \forall \mathbf{y} \in \mathbb{R}^n, \tag{4}$$

and the ϵ-*subdifferential* of f at $\mathbf{x}$, denoted by $\partial f_\epsilon(\mathbf{x})$, is the set of all the ϵ-subgradients of f at $\mathbf{x}$, i.e.,

$$\partial_\epsilon f(\mathbf{x}) \triangleq \left\{\mathbf{g} \in \mathbb{R}^n : f(\mathbf{y}) \geq f(\mathbf{x}) + \mathbf{g}^\top (\mathbf{y} - \mathbf{x}) - \epsilon \ \forall \mathbf{y} \in \mathbb{R}^n\right\}. \tag{5}$$

In case $\epsilon = 0$ it obviously holds that $\partial_0 f(\mathbf{x}) = \partial f(\mathbf{x})$.

Since f is convex and finite over $\mathbb{R}^n$, the subdifferential $\partial f(\cdot)$ is a convex, bounded and closed set; hence, for the directional derivative $f'(\mathbf{x}, \mathbf{d})$ at any $\mathbf{x}$, along the direction $\mathbf{d} \in \mathbb{R}^n$, it holds that

$$f'(\mathbf{x}, \mathbf{d}) \triangleq \lim_{t \downarrow 0} \frac{f(\mathbf{x} + t\mathbf{d}) - f(\mathbf{x})}{t} = \max_{\mathbf{g} \in \partial f(\mathbf{x})} \mathbf{g}^\top \mathbf{d}. \tag{6}$$

In particular, at a point $\mathbf{x}$ where f is differentiable, the formula of classic calculus

$$f'(\mathbf{x}, \mathbf{d}) = \nabla f(\mathbf{x})^\top \mathbf{d} \tag{7}$$

easily follows from (6) and recalling that $\partial f(\mathbf{x}) = \{\nabla f(\mathbf{x})\}$.

Any direction $\mathbf{d} \in \mathbb{R}^n$ is defined as a descent direction at $\mathbf{x}$ if there exists a positive threshold $\bar{t}$ such that

$$f(\mathbf{x} + t\mathbf{d}) < f(\mathbf{x}) \quad \forall t \in (0, \bar{t}].$$

Furthermore, we remark that for convex functions the following equivalence holds true

$$f'(\mathbf{x}, \mathbf{d}) < 0 \quad \Leftrightarrow \quad \mathbf{d} \text{ is a descent direction at } \mathbf{x}. \tag{8}$$

At a later stage we will sometimes relax the convexity assumption on f, only requiring that f be locally Lipschitz, i.e., Lipschitz on every bounded set. Under such assumption, f is still differentiable almost everywhere, and it is defined at each point $\mathbf{x}$ the generalized gradient [29] (or Clarke's gradient, or subdifferential)

$$\partial_C f(\mathbf{x}) \triangleq \operatorname{conv}\{\mathbf{g} : \mathbf{g} \in \mathbb{R}^n, \nabla f(\mathbf{x}_k) \to \mathbf{g}, \ \mathbf{x}_k \to \mathbf{x}, \ \mathbf{x}_k \notin \Omega_f\}, \tag{9}$$

Ω_f being the set (of zero measure) where f is not differentiable. Any point $\mathbf{x}$ with $0 \in \partial_C f(\mathbf{x})$ will be referred to as a *Clarke stationary* point.

In the rest of the article, it will be referred to as an *oracle* any black-box algorithm capable to provide, given any point $\mathbf{x}$, the objective function value $f(\mathbf{x})$ and, in addition, a subgradient in $\partial f(\mathbf{x})$ or in $\partial_C f(\mathbf{x})$, depending on whether f is convex or just locally Lipschitz.

3 Nonsmooth optimization mainstreams

In order to understand the main difference between smooth (i.e., differentiable) and nonsmooth functions, in an algorithmic perspective, we focus on comparing equations (6) and (7). On one hand, for smooth functions, at any point $\mathbf{x}$ the gradient $\nabla f(\mathbf{x})$ provides complete information about the directional derivative, along every possible direction, through the formula $f'(\mathbf{x}, \mathbf{d}) = \nabla f(\mathbf{x})^\top \mathbf{d}$, see (7). On the other hand, for nonsmooth functions, at a point $\mathbf{x}$ where f is not differentiable, the directional derivative, along any given direction, can only be calculated via a maximization process over the entire subdifferential, see (6), thus making any single subgradient unable to provide complete information about $f'(\mathbf{x}, \cdot)$. From an algorithmic viewpoint, such a difference has relevant implications that make not particularly appealing the idea of extending classic descent methods to NSO (although some elegant results for classes of nonsmooth nonconvex functions can be found in [33]). In the following remark, we highlight why most of the available NSO methods do not follow a steepest descent philosophy.

Remark 1 Let $\mathbf{x} \in \mathbb{R}^n$ be given, and assume there exists a descent direction at $\mathbf{x}$. The *steepest* descent direction $\mathbf{d}^*$ at $\mathbf{x}$ is the one where the directional derivative is minimized over the unit ball, i.e.,

$$\mathbf{d}^* = \arg\min \left\{ f'(\mathbf{x}, \mathbf{d}) : \|\mathbf{d}\| \le 1,\, \mathbf{d} \in \mathbb{R}^n \right\}.$$

We observe that $\mathbf{d}^*$ is well defined both in the smooth and in the nonsmooth case. As for the former, it simply holds that $\mathbf{d}^* = -\nabla f(\mathbf{x}) / \|\nabla f(\mathbf{x})\|$. As for the latter, it holds that $\mathbf{d}^*$ is the solution of the following minmax optimization problem

$$\min \Big\{ \max \left\{ \mathbf{g}^\top \mathbf{d} : \mathbf{g} \in \partial f(\mathbf{x}) \right\} : \|\mathbf{d}\| \le 1,\, \mathbf{d} \in \mathbb{R}^n \Big\}$$

By applying the minmax-maxmin theorem it easily follows that

$$\mathbf{d}^* = -\arg\min \left\{ \|\mathbf{g}\| : \mathbf{g} \in \partial f(\mathbf{x}) \right\}.$$

Hence, in the nonsmooth case, the steepest descent direction can only be determined, if the complete knowledge of the subdifferential is available, by finding the minimum norm point in a compact convex set.

As already mentioned, our review of the main classes of iterative algorithms for NSO is based on the distinction between *single-point* and *multi-point* models. In the rest of the article, we will denote by $\mathbf{x}_k$ the estimate of a minimizer of f at the (current) iteration k. Methods based on single-point models look for the new iterate $\mathbf{x}_{k+1}$ by only exploiting the available information on the differential properties of the function at $\mathbf{x}_k$. Such information consist either of a single subgradient or of a larger subset of the subdifferential, possibly coinciding with the entire subdifferential. Sometimes, an appropriate metric is also associated to $\mathbf{x}_k$. The aim is to define a *local* approximation of f around $\mathbf{x}_k$ to suggest a move towards $\mathbf{x}_{k+1}$, possibly obtained via a univariate minimization (line search).

Methods based on *multi-point* models exploit similar local information about $\mathbf{x}_k$, which are enriched by data coming from several other points (typically the iterates $\mathbf{x}_1, \dots, \mathbf{x}_{k-1}$), no matter how far from $\mathbf{x}_k$ they are. Here the aim is no longer to obtain a local approximation of f, but to construct an (outer) approximation of the *entire* level set of f at $\mathbf{x}_k$, that is, of the set

$$S_k \triangleq \big\{\mathbf{x} : f(\mathbf{x}) \leq f(\mathbf{x}_k)\big\}.$$

In the next two sections we will survey the two classes of methods. We wish, however, to remark that the intrinsic difficulty in calculating a descent direction for a nonsmooth function, suggests to look for iterative methods that do not require at each iteration *decrease* of the objective function. In other words, monotonicity is not necessarily a “pole star” for designing NSO algorithms (note, in passing, that also in smooth optimization the monotonicity of objective function values is not a *must* [13,69]).

4 Methods based on single-point models

The focus of this section in on the celebrated subgradient method, introduced by N.Z. Shor in the early 60s of the last century, see [136]. In particular, we aim to review the convergence properties of the classic versions of the method, next giving some hints on recent improvements. In its simplest configuration the subgradient method works according to the following iteration scheme

$$\mathbf{x}_{k+1} = \mathbf{x}_k - t\frac{\mathbf{g}_k}{\|\mathbf{g}_k\|}, \tag{10}$$

where $\mathbf{g}_k \in \partial f(\mathbf{x}_k)$ and $t > 0$ is a constant *stepsize*. In order to develop a convergence theory it is crucial to introduce the concept of *minimum approaching* direction at any point $\mathbf{x}$, as a direction along which there exist points which are closer to a minimizer than $\mathbf{x}$. More specifically, a direction $\mathbf{d}$ is defined as a minimum approaching direction at $\mathbf{x}$, if there exists a positive threshold $\bar{t}$ such that

$$\|\mathbf{x} + t\mathbf{d} - \mathbf{x}^*\| < \|\mathbf{x} - \mathbf{x}^*\|, \quad \forall t \in (0, \bar{t}).$$

As previously pointed out, see (6)–(8), taking an anti–subgradient direction $\mathbf{d} = -\mathbf{g}$, for any $\mathbf{g} \in \partial f(\mathbf{x})$, thus satisfying the condition $\mathbf{g}^\top \mathbf{d} < 0$, does not guarantee that $\mathbf{d}$ is a descent direction at $\mathbf{x}$. On the other hand, it can be easily proved that such $\mathbf{d}$ is a minimum approaching direction at $\mathbf{x}$. In fact, for any $\mathbf{x} \notin M^*$, the convexity of f implies that

$$\mathbf{g}^\top(\mathbf{x}^* - \mathbf{x}) < 0. \tag{11}$$

As a consequence, by letting

$$\bar{t} = \frac{2\mathbf{g}^\top(\mathbf{x} - \mathbf{x}^*)}{\|\mathbf{g}\|^2},$$

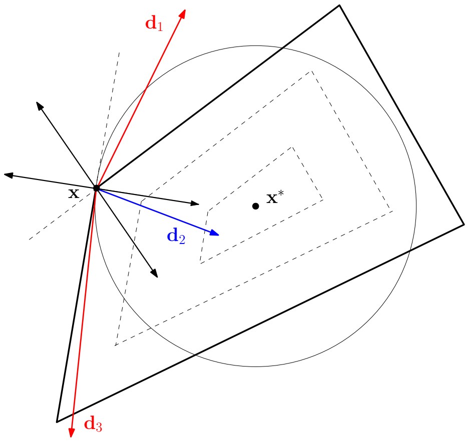

Fig. 1 Descent and/or minimum approaching directions

from inequality (11) it follows that

$$\begin{aligned}\|\mathbf{x}+t\mathbf{d}-\mathbf{x}^*\|^2 &= \|\mathbf{x}-t\mathbf{g}-\mathbf{x}^*\|^2\\ &= \|\mathbf{x}-\mathbf{x}^*\|^2 + t(t\|\mathbf{g}\|^2 + 2\mathbf{g}^\top(\mathbf{x}^*-\mathbf{x}))\\ &< \|\mathbf{x}-\mathbf{x}^*\|^2 \qquad (12)\end{aligned}$$

for every $t \in (0,\bar{t})$, namely, that $\mathbf{d} = -\mathbf{g}$ is a minimum approaching direction at $\mathbf{x}$.

Different types of directions are depicted in Figure 1, where the contour lines of a convex, piecewise affine function with minimum at $\mathbf{x}^*$ are represented. Note in fact that, at point $\mathbf{x}$, direction $\mathbf{d}_2$ is both a descent and a minimum approaching one, since it points inside both the contour at $\mathbf{x}$ and the sphere of radius $\|\mathbf{x}-\mathbf{x}^*\|$ centered at $\mathbf{x}^*$. Note also that $\mathbf{d}_1$ is minimum approaching but not descent, while $\mathbf{d}_3$ is descent but not minimum approaching.

In the following, taking any subgradient $\mathbf{g} \in \partial f(\mathbf{x})$, we will indicate by $\mathbf{d} = -\mathbf{g}$ an anti–subgradient direction, possibly normalized by setting $\mathbf{d} = -\frac{\mathbf{g}}{\|\mathbf{g}\|}$.

The property of the anti–subgradient directions of being minimum approaching ones is crucial for ensuring convergence of the constant stepsize method based on the iteration scheme (10), as we show in the following theorem [137].

Theorem 1 *Let f be convex and assume that M^*, the set of minima of f, is nonempty. Then, for every $\epsilon > 0$ and $\mathbf{x}^* \in M^*$ there exist a point $\bar{\mathbf{x}}$ and an index $\bar{k}$ such that*

$$\|\bar{\mathbf{x}}-\mathbf{x}^*\| < \frac{t}{2}(1+\epsilon)$$

and

$$f(\mathbf{x}_{\bar{k}}) = f(\bar{\mathbf{x}}).$$

Proof Let $U_k = \{\mathbf{x} \mid f(\mathbf{x}) = f(\mathbf{x}_k)\}$ and $L_k = \{\mathbf{x} \mid \mathbf{g}_k^\top(\mathbf{x}-\mathbf{x}_k) = 0\}$ be, respectively, the contour line passing through $\mathbf{x}_k$ and the supporting hyperplane at $\mathbf{x}_k$ to the level set $S_k = \{\mathbf{x} \mid f(\mathbf{x}) \le f(\mathbf{x}_k)\}$, with normal $\mathbf{g}_k$.

Consider now, see Figure 2, $a_k(\mathbf{x}^*) = \|\mathbf{x}^* - \mathbf{x}_P^*\|$, the distance of any point $\mathbf{x}^* \in M$ from its projection $\mathbf{x}_P^*$ onto L_k, and observe that $a_k(\mathbf{x}^*) \geq b_k(\mathbf{x}^*) = \|\mathbf{x}^* - \mathbf{x}_L^*\|$. Note also that $b_k(\mathbf{x}^*)$ is an upper bound on $dist(\mathbf{x}^*, U_k)$, the distance of $\mathbf{x}^*$ from contour line U_k. It is easy to verify that

$$a_k(\mathbf{x}^*) = \frac{\mathbf{g}_k^\top(\mathbf{x}_k - \mathbf{x}^*)}{\|\mathbf{g}_k\|},$$

and, as a consequence, that

$$dist(\mathbf{x}^*, L_k) \leq b_k(\mathbf{x}^*) \leq a_k(\mathbf{x}^*) = \frac{\mathbf{g}_k^\top(\mathbf{x}_k - \mathbf{x}^*)}{\|\mathbf{g}_k\|}. \tag{13}$$

Now, observe that from (10) and (13) it follows that

$$\begin{aligned}
\|\mathbf{x}_{k+1} - \mathbf{x}^*\|^2 &= \|\mathbf{x}_k - \mathbf{x}^* - t\frac{\mathbf{g}_k}{\|\mathbf{g}_k\|}\|^2 \\
&= \|\mathbf{x}_k - \mathbf{x}^*\|^2 + t^2 - 2t\,\frac{\mathbf{g}_k^\top(\mathbf{x}_k - \mathbf{x}^*)}{\|\mathbf{g}_k\|} \\
&= \|\mathbf{x}_k - \mathbf{x}^*\|^2 + t^2 - 2ta_k(\mathbf{x}^*) \\
&\leq \|\mathbf{x}_k - \mathbf{x}^*\|^2 + t^2 - 2tb_k(\mathbf{x}^*).
\end{aligned} \tag{14}$$

Next, suppose for a contradiction that $b_k(\mathbf{x}^*) \geq t(1+\epsilon)/2$ for every k. Denoting by $\mathbf{x}_1$ the starting point of the algorithm, and repeatedly applying the inequality (14), we have that for every k it holds

$$\begin{aligned}
\|\mathbf{x}_{k+1} - \mathbf{x}^*\|^2 &\leq \|\mathbf{x}_k - \mathbf{x}^*\|^2 - \epsilon t^2 \\
&\;\;\vdots \\
&\leq \|\mathbf{x}_1 - \mathbf{x}^*\|^2 - \epsilon k t^2,
\end{aligned}$$

which contradicts $\|\mathbf{x}_{k+1} - \mathbf{x}^*\|^2 \geq 0$ for all k. □

Remark 2 Note that the above theorem does not ensure that the method generates a point arbitrarily close to a minimum. In fact, it only allows to guarantee that a contour line is reached whose distance from any minimizer is arbitrarily small.

The constant stepsize subgradient method is interesting from the historical point of view but its numerical performance is strongly affected by the choice of t. Classic subgradient method (SM in the following), instead, is based on adjustable stepsize and works according to the iterative scheme

$$\mathbf{x}_{k+1} = \mathbf{x}_k - t_k\frac{\mathbf{g}_k}{\|\mathbf{g}_k\|}, \tag{15}$$

where $\mathbf{g}_k \in \partial f(\mathbf{x}_k)$ and $t_k > 0$. The following theorem guarantees standard convergence of SM under appropriate conditions on the stepsize t_k [137].

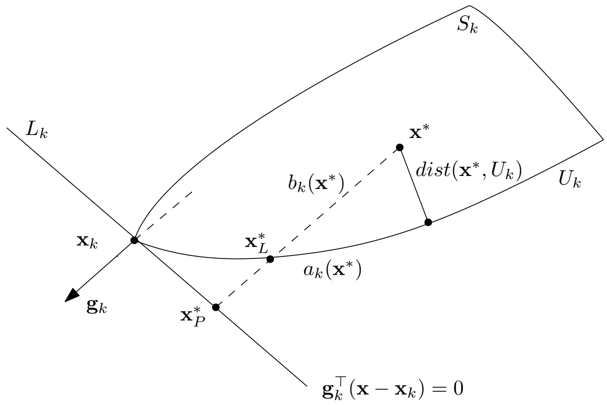

Fig. 2 Convergence of the subgradient method

Theorem 2 *Let f be convex and assume that M^* is bounded and nonempty, with $f^* = f(\mathbf{x}^*)$. If the stepsize sequence $\{t_k\}$ in (15) satisfies the conditions*

$$\lim_{k\to\infty} t_k = 0 \qquad \text{and} \qquad \sum_{k=1}^{\infty} t_k = \infty, \tag{16}$$

then either there exists an index k^* such that $\mathbf{x}_{k^*} \in M^*$ or $\lim_{k\to\infty} f(\mathbf{x}_k) = f^*$.

Remark 3 A possible choice of t_k satisfying (16) is $t_k = c/k$, where c is any positive constant. The choice

$$t_k = \frac{f(\mathbf{x}_k) - f^*}{\|\mathbf{g}_k\|}, \tag{17}$$

known as Polyak stepsize (see [126] for an alternative proof of convergence) is particularly popular in the area of application of nonsmooth convex optimization to solution of Integer Linear Programming (ILP) problems via Lagrangian relaxation [65]. In the fairly common case when f^* is unknown, it is usually replaced in (17) by any lower bound on the optimal objective function value.

The following proposition provides an evaluation of the convergence speed of the subgradient method and an estimate of the number of iterations, under some simplifying assumptions. Detailed discussions can be found in [66,137].

Proposition 1 *Assume that*

i) f admits a sharp minimum, i.e., there exists $\mu > 0$ such that $f(\mathbf{x}) \geq f^* + \mu\|\mathbf{x} - \mathbf{x}^*\|$, for every $\mathbf{x} \in \mathbb{R}^n$, and
ii) the minimum value f^* is known, so that the Polyak stepsize (17) can be calculated.

Then, the subgradient method has linear convergence rate $q = \sqrt{1 - \frac{\mu^2}{c^2}}$, where c is any upper bound on the norm of $\mathbf{g}_k$. Moreover an ϵ-approximate solution is achieved in $O(\frac{1}{\epsilon^2})$ iterations.

Proof Note first that convexity of f implies

$$0 \geq f^* - f(\mathbf{x}_k) \geq \mathbf{g}_k^\top (\mathbf{x}^* - \mathbf{x}_k) \quad \forall k. \tag{18}$$

From (15), by adopting the Polyak stepsize (17) and taking into account (18), it follows that

$$\begin{aligned}\|\mathbf{x}_{k+1} - \mathbf{x}^*\|^2 &= \|\mathbf{x}_k - \mathbf{x}^* - t_k \frac{\mathbf{g}_k}{\|\mathbf{g}_k\|}\|^2 \\ &= \|\mathbf{x}_k - \mathbf{x}^*\|^2 + t_k^2 - 2t_k \frac{\mathbf{g}_k^\top (\mathbf{x}_k - \mathbf{x}^*)}{\|\mathbf{g}_k\|} \\ &\leq \|\mathbf{x}_k - \mathbf{x}^*\|^2 - \frac{(f(\mathbf{x}_k) - f^*)^2}{\|\mathbf{g}_k\|^2}, \end{aligned} \tag{19}$$

hence that

$$\|\mathbf{x}_{k+1} - \mathbf{x}^*\| \leq \|\mathbf{x}_1 - \mathbf{x}^*\| \ \forall k.$$

The latter inequality implies boundedness of the sequence $\{\mathbf{x}_k\}$, which in turn implies boundedness of the corresponding sequence of subgradients $\{\mathbf{g}_k\}$, say $\|\mathbf{g}_k\| \leq c$, for every k, for some positive constant c. Taking into account assumption *i)* we rewrite (19) as

$$\frac{\|\mathbf{x}_{k+1} - \mathbf{x}^*\|^2}{\|\mathbf{x}_k - \mathbf{x}^*\|^2} \leq 1 - \frac{\mu^2}{c^2} = q^2,$$

which proves first part of the Proposition. Next, let $f_k^* = \min_{1 \leq i \leq k} f(\mathbf{x}_i)$, the best objective function value obtained up to iteration k, let $R = \|\mathbf{x}_1 - \mathbf{x}^*\|$, and observe that

$$0 \leq f_k^* - f^* \leq f(\mathbf{x}_i) - f^*.$$

From iterated application of inequality (19), for $i = 1, \ldots, k$, since $\|\mathbf{x}_{k+1} - \mathbf{x}^*\| \geq 0$ and $\|\mathbf{g}_i\| \leq c$, we obtain that

$$\begin{aligned} 0 &\leq \|\mathbf{x}_1 - \mathbf{x}^*\|^2 - \sum_{i=1}^{k} \frac{(f(\mathbf{x}_i) - f^*)^2}{\|\mathbf{g}_i\|^2} \\ &\leq R^2 - \sum_{i=1}^{k} \frac{(f(\mathbf{x}_i) - f^*)^2}{c^2} \\ &\leq R^2 - \frac{k(f_k^* - f^*)^2}{c^2} \end{aligned}$$

hence that

$$f_k^* - f^* \leq \frac{Rc}{\sqrt{k}}.$$

The latter inequality implies that an ϵ-optimal solution is obtained in a number of iterations $k \geq \frac{R^2c^2}{\epsilon^2}$ and the proof is complete. □

Remark 4 We observe that monotonicity of the sequence $\{f(\mathbf{x}_k)\}$ is not ensured, while the *minimum approaching* nature of the anti-subgradient direction is apparent from (19). On the other hand, we note that the convergence rate q can be arbitrarily close to 1.

Slow convergence of the subgradient method has stimulated several improvement attempts in more recent years. Starting from observation that the method is a *black box* one, as no problem structure is exploited, the newly introduced approaches have been designed for classes of *weakly* structured problems, still covering most of convex nonsmooth optimization programs of practical interest.

Here, we recall Nesterov's smoothing method [118], where the bound on the number of iterations improves from $O(\frac{1}{\epsilon^2})$ to $O(\frac{1}{\epsilon})$. Denoting by S_1 and S_2 two convex and compact subsets of $\mathbb{R}^n$ and $\mathbb{R}^m$, respectively, the problem addressed in [118] is of type

$$\min\left\{f(\mathbf{x}) : \mathbf{x} \in S_1 \subset \mathbb{R}^n\right\}, \tag{20}$$

with

$$f(\mathbf{x}) = \max\left\{\mathbf{x}^\top A^\top \mathbf{u} - \phi(\mathbf{u}) : \mathbf{u} \in S_2 \subset \mathbb{R}^m\right\}, \tag{21}$$

where A is a matrix of appropriate dimension, and $\phi : \mathbb{R}^m \to \mathbb{R}$ is a convex function. Note that f is convex, being the pointwise maximum of (an infinite number of) convex functions, and nonsmoothness of f occurs at those point $\mathbf{x}$ where the maximum is not unique. In fact, smoothing of f is pursued in [118] by forcing such maximum to be unique. In particular, the following perturbation f_μ of f is introduced

$$f_\mu(\mathbf{x}) = \max\left\{\mathbf{x}^\top A^\top \mathbf{u} - \phi(\mathbf{u}) - \mu\omega(\mathbf{u}) : \mathbf{u} \in S_2 \subset \mathbb{R}^m\right\},$$

where $\mu > 0$ is the perturbation parameter, and $\omega : \mathbb{R}^m \to \mathbb{R}$ is a strongly convex continuously differentiable function, i.e., for every $\mathbf{v} \in \mathbb{R}^m$ it satisfies the condition

$$\omega(\mathbf{u}) \geq \omega(\mathbf{v}) + \nabla\omega(\mathbf{v})^\top(\mathbf{u} - \mathbf{x}) + \sigma\|\mathbf{u} - \mathbf{v}\|^2 \qquad \forall \mathbf{u} \in \mathbb{R}^m,$$

for some $\sigma > 0$. Minimization of the smooth function $f_\mu(\mathbf{x})$ is then pursued via a gradient-type method (see also [49] for a discussion on tuning of the smoothing parameter μ.)

The *Mirror Descent Algorithm* (MDA) [116] is yet another method inspired by SM. We give here its basic elements, following the presentation of [14], and confine ourselves to treatment of the unconstrained problem (1). Consider the following iteration scheme for passing from $\mathbf{x}_k$ to $\mathbf{x}_{k+1}$, once an *oracle* has provided both $f(\mathbf{x}_k)$ and $\mathbf{g}_k \in \partial f(\mathbf{x}_k)$,

$$\mathbf{x}_{k+1} = \arg\min\left\{f(\mathbf{x}_k) + \mathbf{g}_k^\top(\mathbf{x} - \mathbf{x}_k) + \frac{1}{2\gamma_k}\|\mathbf{x} - \mathbf{x}_k\|^2 : \mathbf{x} \in \mathbb{R}^n\right\}, \tag{22}$$

where $\gamma_k > 0$. Simple calculation provides

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \gamma_k \mathbf{g}_k,$$

which coincides with (15) when $\gamma_k = \frac{t_k}{\|\mathbf{g}_k\|}$. Note that $\mathbf{x}_{k+1}$ is obtained as the minimizer of the linearization of f rooted at $\mathbf{x}_k$, augmented by a *proximity* term which penalizes long steps away from $\mathbf{x}_k$, on the basis of the proximity parameter γ_k. Now consider, as in previous Nesterov's method, any strongly convex continuously differentiable function $\omega : \mathbb{R}^n \to \mathbb{R}$ and let

$$\alpha_k(\mathbf{x}) = \omega(\mathbf{x}) - \omega(\mathbf{x}_k) - \nabla\omega(\mathbf{x}_k)^\top(\mathbf{x} - \mathbf{x}_k). \tag{23}$$

Function $\alpha_k(\mathbf{x})$ measures the error at $\mathbf{x}$ associated to the linearization of ω rooted at $\mathbf{x}_k$, and resumes information about the curvature of ω along the direction $(\mathbf{x}-\mathbf{x}_k)$. Moreover α_k can be considered as a *distance-like* function, since strong convexity of ω implies $\alpha_k(\mathbf{x})>0$ for $\mathbf{x}\neq\mathbf{x}_k$. On the basis of the definition of α_k the iterative scheme (22) is generalized by setting:

$$\mathbf{x}_{k+1}=\arg\min\left\{f(\mathbf{x}_k)+\mathbf{g}_k^\top(\mathbf{x}-\mathbf{x}_k)+\frac{1}{\gamma_k}\alpha_k(\mathbf{x}):\mathbf{x}\in\mathbb{R}^n\right\}. \tag{24}$$

Note that, letting $\omega(\mathbf{x})=\frac{1}{2}\|\mathbf{x}\|^2$, it is easy to verify that

$$\alpha_k(\mathbf{x})=\frac{1}{2\gamma_k}\|\mathbf{x}-\mathbf{x}_k\|^2,$$

and the two iterative schemes (22) and (24) coincide. Hence, we conclude that the SM iteration scheme (22) is a special case of (24) which is, in fact, MDA (see [14]). The function α_k is usually referred to as a Bregman-like distance generated by function ω.

5 Methods based on multi-point models

As previously mentioned, here we deal with those iterative methods for NSO where the next iterate $\mathbf{x}_{k+1}$ is calculated on the basis of information related to both the current iterate $\mathbf{x}_k$ and to several other points (e.g., previous estimates of an optimal solution).

The fundamental leverage in constructing such class of methods is that a convex function is the pointwise supremum of affine ones, namely, for any convex function $f:\mathbb{R}^n\to\mathbb{R}$ and every $\mathbf{x}\in\mathbb{R}^n$ it holds that

$$f(\mathbf{x})=\sup\left\{f(\mathbf{y})+\mathbf{g}(\mathbf{y})^\top(\mathbf{x}-\mathbf{y}):\mathbf{y}\in\mathbb{R}^n\right\},$$

where $\mathbf{g}(\mathbf{y})\in\partial f(\mathbf{y})$, see [76, Th. 1.3.8]. The latter formula has some relevant consequences. In fact, taking any finite set of points $\mathbf{x}_1,\mathbf{x}_2,\dots,\mathbf{x}_k\in\mathbb{R}^n$, letting

$$\ell_i(\mathbf{x})\triangleq f(\mathbf{x}_i)+\mathbf{g}_i^\top(\mathbf{x}-\mathbf{x}_i)$$

for every $i\in\{1,\dots,k\}$, with $\mathbf{g}_i\in\partial f(\mathbf{x}_i)$, and defining

$$f_k(\mathbf{x})\triangleq\max\left\{\ell_i(\mathbf{x}):i\in\{1,\dots,k\}\right\} \tag{25}$$

it holds that

$$f_k(\mathbf{x})\le f(\mathbf{x})\qquad\forall\mathbf{x}\in\mathbb{R}^n, \tag{26}$$

$$f_k(\mathbf{x})=f(\mathbf{x}_i)\qquad\forall i\in\{1,\dots,k\}, \tag{27}$$

$$\mathbf{g}_i\in\partial f_k(\mathbf{x}_i)\qquad\forall i\in\{1,\dots,k\}. \tag{28}$$

Thus, function f_k is a *global* approximation of f, as it minorizes f everywhere, while interpolating it at points $\mathbf{x}_1,\dots,\mathbf{x}_k$. Note, in addition, that f_k is convex and piecewise affine, being the pointwise maximum of the affine functions $\ell_i(\mathbf{x})$, the linearizations of f rooted at $\mathbf{x}_1,\dots,\mathbf{x}_k$. We observe in passing that, even for the same set of points $\mathbf{x}_1, \dots, \mathbf{x}_k$, the model function f_k can be not unique, since the subdifferential $\partial f(\cdot)$ is a multifunction. In the following, we will refer to f_k as to the *cutting plane* function, a term which deserves some explanation.

Consider the epigraph of f, namely, the subset of $\mathbb{R}^{n+1}$ defined as

$$epif \triangleq \{(\mathbf{x}, v) : \mathbf{x} \in \mathbb{R}^n, \, v \in \mathbb{R}, \, v \geq f(\mathbf{x})\},$$

and define the set of halfspaces $H_i \subset \mathbb{R}^{n+1}$, for every $i \in \{1, \dots, k\}$:

$$H_i \triangleq \big\{(\mathbf{x}, v) : \mathbf{x} \in \mathbb{R}^n, \, v \in \mathbb{R}, \, v \geq \ell_i(\mathbf{x})\big\}.$$

Observing that for every $\mathbf{x} \in \mathbb{R}^n$ there holds

$$f(\mathbf{x}) \geq \ell_i(\mathbf{x}) \;\; \forall i \in \{1, \dots, k\},$$

and that

$$epif_k \triangleq \big\{(\mathbf{x}, v) : \mathbf{x} \in \mathbb{R}^n, \, v \in \mathbb{R}, \, v \geq f_k(\mathbf{x})\big\} = \bigcap_{i=1}^{k} H_i,$$

it is easy to see that $\big(\mathbf{x}, f(\mathbf{x})\big) \in \bigcap_{i=1}^{k} H_i$ and, consequently, that

$$epif \subseteq epif_k.$$

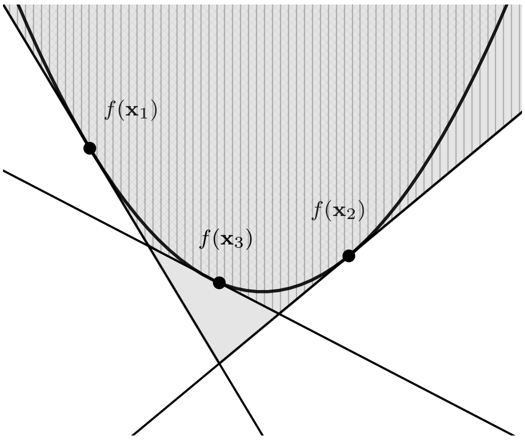

Fig. 3 Adding cuts

Now take any point $\mathbf{x}_{k+1}$ such that $f(\mathbf{x}_{k+1}) > f_k(\mathbf{x}_{k+1})$, define the corresponding halfspace

$$H_{k+1} = \left\{(\mathbf{x}, v) : \mathbf{x} \in \mathbb{R}^n, \, v \in \mathbb{R}, \, v \geq \ell_{k+1}(\mathbf{x})\right\},$$

and the new approximation

$$f_{k+1}(\mathbf{x}) = \max\big\{\ell_i(\mathbf{x}) : \, i \in \{1, \dots, k+1\}\big\}.$$

Observe that, while $\big(\mathbf{x}_{k+1}, f_k(\mathbf{x}_{k+1})\big) \in epif_k$, we have $\big(\mathbf{x}_{k+1}, f_k(\mathbf{x}_{k+1})\big) \notin epif_{k+1}$ and

$$epif \subseteq epif_{k+1} \subset epif_k. \tag{29}$$

In other words, the hyperplane

$$L_{k+1} = \big\{(\mathbf{x}, v) : \mathbf{x} \in \mathbb{R}^n,\, v \in \mathbb{R},\, v = \ell_{k+1}(\mathbf{x})\big\}$$

separates point $\big(\mathbf{x}_{k+1}, f_k(\mathbf{x}_{k+1})\big)$ from $epif_{k+1}$, thus it represents a *cut* for $epif_k$. Note that the bigger is the difference $f(\mathbf{x}_{k+1}) - f_k(\mathbf{x}_{k+1})$, the *deeper* is the cut defined by L_{k+1}, see Figure 3.

The definition of the cutting plane function f_k provides a natural way to select the next trial point by setting

$$\mathbf{x}_{k+1} = \arg\min\big\{f_k(\mathbf{x}) : \mathbf{x} \in \mathbb{R}^n\big\}, \tag{30}$$

which is exactly the iteration scheme of the classic *cutting plane* method, see [28, 79]. Problem (30) is still nondifferentiable, but it is equivalent to the following linear program, defined in $\mathbb{R}^{n+1}$, thanks to the introduction of the additional scalar variable w

$$\min\Big\{w : w \geq f(\mathbf{x}_i) + \mathbf{g}_i^\top(\mathbf{x} - \mathbf{x}_i)\ \forall i \in \{1, \dots, k\},\, \mathbf{x} \in \mathbb{R}^n,\, w \in \mathbb{R}\Big\}, \tag{31}$$

whose optimal solution is denoted by $(\mathbf{x}_{k+1}, w_k)$, with $w_k = f_k(\mathbf{x}_{k+1})$. Note that, since the feasible region is the nonempty set $epif_k$, boundedness of (31) requires feasibility of its dual which, denoting by $\boldsymbol{\lambda} \in \mathbb{R}^k$ the vector of dual variables, can be formulated as the following program

$$\max\left\{\sum_{i=1}^k \lambda_i(f(\mathbf{x}_i) - \mathbf{g}_i^\top \mathbf{x}_i) : \sum_{i=1}^k \lambda_i = 1,\, \sum_{i=1}^k \lambda_i \mathbf{g}_i = 0,\, \boldsymbol{\lambda} \geq \mathbf{0}\right\}. \tag{32}$$

We observe that feasibility of (32) is equivalent to the condition

$$\mathbf{0} \in \operatorname{conv}\big\{\mathbf{g}_i : i \in \{1, \dots, k\}\big\}. \tag{33}$$

Hence, the boundedness of problem (31), which allows to calculate $\mathbf{x}_{k+1}$, requires a kind of hard-to-test *qualification* of points $\mathbf{x}_1, \dots, \mathbf{x}_k$, expressed in terms of the corresponding subgradients. We show in Figure 4 an example where the cutting plane function is unbounded since the derivatives $f'(\mathbf{x}_1)$ and $f'(\mathbf{x}_2)$ are both negative, thus $0 \notin [f'(\mathbf{x}_1), f'(\mathbf{x}_2)])$.

To avoid the difficulties related to possible unboundedness of the cutting plane function, we put the original problem (1) in an (artificial) constrained optimization setting. In fact, we consider the problem

$$\min\big\{f(\mathbf{x}) : \mathbf{x} \in Q \subset \mathbb{R}^n\big\}, \tag{34}$$

where Q is a nonempty compact convex subset of $\mathbb{R}^n$. One would think of Q as a set defined by *simple* constraints (e.g., box constraints) and sufficiently large to contain M^*. We further assume that for each $\mathbf{x} \in Q$ both the objective function value $f(\mathbf{x})$ and a subgradient $g \in \partial f(\mathbf{x})$ can be computed. We also let L_Q denote the Lipschitz constant of f on Q. Thus, the cutting-plane iteration becomes

$$\mathbf{x}_{k+1} = \arg\min\big\{f_k(\mathbf{x}) : \mathbf{x} \in Q\big\}, \tag{35}$$

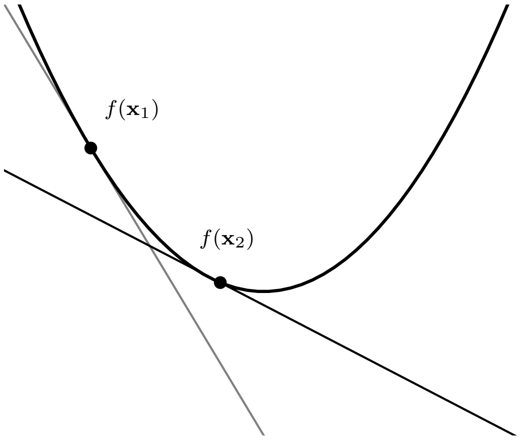

Fig. 4 Unbounded cutting plane function

whose well-posedness is guaranteed by the continuity of f_k, together with compactness of Q. Moreover, we note that, since by convexity $f_k(\mathbf{x}) \leq f(\mathbf{x})$ for every $\mathbf{x} \in Q$, the optimal value f_k^* of $f_k(\mathbf{x})$ provides a lower bound on f^*, the optimal value of f. In addition, since

$$f_{k+1}(\mathbf{x}) = \max\left\{f_k(\mathbf{x}), f(\mathbf{x}_{k+1}) + \mathbf{g}_{k+1}^\top(\mathbf{x} - \mathbf{x}_{k+1})\right\} \geq f_k(\mathbf{x}) \qquad \forall \mathbf{x} \in Q,$$

the sequence $\{f_k^*\}$ is monotonically nondecreasing and thus the lower bound becomes increasingly better.

We state now the convergence of a slightly more general cutting plane-like method, presented in Algorithm 1, which comprises the classic version where the iteration scheme (35) is adopted, see [126].

Algorithm 1 General cutting plane method (GCPM)

```
Input: a starting point $\mathbf{x}_1 \in \mathbb{R}^n$, a stopping tolerance parameter $\epsilon > 0$
Output: an $\epsilon$-optimal solution $\mathbf{x}^* \in \mathbb{R}^n$
1: Calculate $\mathbf{g}_1 \in \partial f(\mathbf{x}_1)$, build $f_1(\mathbf{x})$, and set $k = 1$      ▷ Initialization
2: Calculate $\mathbf{x}_{k+1} \in S_k^* \triangleq \{\mathbf{x} : \mathbf{x} \in Q,\ f_k(\mathbf{x}) \leq f^*\} \neq \emptyset$      ▷ Select the new iterate point
3: if $f(\mathbf{x}_{k+1}) - f_k(\mathbf{x}_{k+1}) < \epsilon$ then      ▷ Stopping test
4:     set $\mathbf{x}^* = \mathbf{x}_{k+1}$ and exit      ▷ Return $\mathbf{x}^*$
5: else
6:     Calculate $\mathbf{g}_{k+1} \in \partial f(\mathbf{x}_{k+1})$ and build $f_{k+1}(\mathbf{x})$      ▷ Improve the cutting-plane function
7:     set $k = k + 1$ and go to 2      ▷ Make a new iteration
8: end if
```

We note that $f_k(\mathbf{x}) \leq f(\mathbf{x})$, for every $\mathbf{x} \in \mathbb{R}^n$, ensures that selection of $\mathbf{x}_{k+1}$ as in (35) perfectly fits with the condition $\mathbf{x}_{k+1} \in S_k^*$ at Step 2 of GCPM. The rationale of the definition of S_k^* is to take $\mathbf{x}_{k+1}$ *well inside* into the level set of f_k. This allows to accommodate, at least in principle, possible *inexact* solution of the program

$$\min\left\{w : w \geq f(\mathbf{x}_i) + \mathbf{g}_i^\top(\mathbf{x} - \mathbf{x}_i)\ \forall i \in \{1, \ldots, k\},\ \mathbf{x} \in Q,\ w \in \mathbb{R}\right\}, \qquad (36)$$

which is still linear provided Q has a polyhedral structure. GCPM with $\mathbf{x}_{k+1}$ selected as in (35) will be simply referred in the following as Cutting Plane Method (CPM).

Remark 5 GCPM is an intrinsically nonmonotone method as no objective function decrease is guaranteed at each iteration

The proof of the convergence of Algorithm 1 is rather simple and relies on convexity of f and on compactness of Q.

Theorem 3 *GCPM terminates at an ϵ-optimal point.*

Proof We observe first that, since $f_k(\mathbf{x}_{k+1)} \leq f^*$, satisfaction of the stopping condition at Step 3 of GCPM implies that

$$f(\mathbf{x}_{k+1}) - f^* < \epsilon,$$

i.e., that the point $\mathbf{x}_{k+1}$ is ϵ-optimal. Now, assume for a contradiction that the stopping condition is not satisfied for infinitely many iterations and, consequently, that

$$f(\mathbf{x}_{k+1}) - f_k(\mathbf{x}_{k+1}) \geq \epsilon \tag{37}$$

holds for every k. Convexity of f, along with (37) and (25), ensure that the following inequalities hold for every $i \in \{1, \dots, k\}$

$$\begin{aligned} f(\mathbf{x}_i) &\geq f(\mathbf{x}_{k+1}) + \mathbf{g}_{k+1}^\top(\mathbf{x}_i - \mathbf{x}_{k+1}) \\ &\geq f_k(\mathbf{x}_{k+1}) + \epsilon + \mathbf{g}_{k+1}^\top(\mathbf{x}_i - \mathbf{x}_{k+1}) \\ &\geq f(\mathbf{x}_i) + \mathbf{g}_i^\top(\mathbf{x}_{k+1} - \mathbf{x}_i) + \epsilon + \mathbf{g}_{k+1}^\top(\mathbf{x}_i - \mathbf{x}_{k+1}), \end{aligned}$$

which imply

$$0 \geq \epsilon - 2L_Q\|\mathbf{x}_{k+1} - \mathbf{x}_i\| \qquad \forall i \in \{1, \dots, k\}. \tag{38}$$

A consequence of (38) is that the sequence of points generated by the algorithm does not have an accumulation point, which contradicts compactness of Q. □

While cutting plane method represents an elegant way to handle convex optimization, it exhibits some major drawbacks. We observe first that the convergence proof is based on the hypothesis of infinite storage capacity. In fact, the size of the linear program to be solved increases at each iteration as consequence of the introduction of a new constraint. A second drawback of the method is related to its numerical instability. In fact, not only monotonicity of the sequence $\{f(\mathbf{x}_k)\}$ is not ensured (this being a fairly acceptable feature of the method, though) but it may happen that after the iterate sequence gets to points very close to the minimizer, some successive iterate points might roll very far away from it, as we show in the following simple example.

Example 1 Consider the one-dimensional quadratic program $\min\{\frac{1}{2}x^2 : x \in \mathbb{R}\}$, whose minimizer is $x^* = 0$. Assume that $k = 2$, let $x_1 = -1$ and $x_2 = 0.01$, with point x_2 being *rather close* to the minimizer. It is easy to verify that $x_3 = \arg\min\{f_2(x) : x \in \mathbb{R}\} = -0.495$, with the algorithm jumping to a point whose distance from the minimizer is much bigger than 0.01. Illustrative examples of such poor behavior of the method can be found in [76, Chapter XV.1].

5.1 Bundle methods (BM)

Bundle methods are a family of algorithms originating from the pioneering work by Lemarechal [92]. They can be considered as a natural evolution of CPM which provides an effective answer to the previously mentioned drawbacks. The term *bundle* is meant to recall that, similarly to CPM, at each iteration a certain amount of cumulated information about points scattered throughout the function domain is necessary to create a model-function, whose minimization delivers the new iterate. In particular, we denote by B_k the *bundle* of the cumulative information available at iteration k, where B_k is the following set of point/function/subgradient triplets

$$B_k \triangleq \left\{ \big(\mathbf{x}_i, f(\mathbf{x}_i), \mathbf{g}_i\big) : \mathbf{g}_i \in \partial f(\mathbf{x}_i),\, i \in \{1, \dots, k\} \right\}.$$

In bundle methods, however, one among points $\mathbf{x}_i$ is assigned the special role of *stability center*. One may think of such point as the *best* in terms of objective function value, but this is not strictly necessary. In the following, we will denote by $\overline{\mathbf{x}}_k$ the current stability center, singled out from the set of iterates $\{\mathbf{x}_1 \dots \mathbf{x}_k\}$. Adopting a term commonly used in discrete optimization, it will be referred to as the *incumbent*, $f(\overline{\mathbf{x}}_k)$ being the *incumbent value*.

Once the stability center $\overline{\mathbf{x}}_k$ has been fixed, the change of variables $\mathbf{x} = \overline{\mathbf{x}}_k + \mathbf{d}$ is introduced. It expresses every point of function domain in terms of its displacement $\mathbf{d} \in \mathbb{R}^n$ with respect to the stability center, and allows to rewrite the cutting plane function $f_k(\mathbf{x})$ in the form of *difference function* as

$$f_k(\overline{\mathbf{x}}_k + \mathbf{d}) - f(\overline{\mathbf{x}}_k) = \max \left\{ \mathbf{g}_i^\top \mathbf{d} - \alpha_i : i \in \{1, \dots, k\} \right\} \tag{39}$$

where α_i, for every $i \in \{1, \dots, k\}$, is the *linearization error*, see (23), associated to the affine expansion $\ell_i(\mathbf{x})$ at $\overline{\mathbf{x}}_k$, and is defined as

$$\alpha_i \triangleq f(\overline{\mathbf{x}}_k) - \big(f(\mathbf{x}_i) + \mathbf{g}_i^\top (\overline{\mathbf{x}}_k - \mathbf{x}_i)\big). \tag{40}$$

Note that convexity of f guarantees nonnegativity of the linearization error. Moreover, for every $\mathbf{x} \in \mathbb{R}^n$ and $i \in \{1, \dots, k\}$, since $\mathbf{g}_i \in \partial f(\mathbf{x}_i)$, it holds that

$$\begin{aligned} f(\mathbf{x}) &\geq f(\mathbf{x}_i) + \mathbf{g}_i^\top (\mathbf{x} - \mathbf{x}_i) \\ &= f(\overline{\mathbf{x}}_k) - f(\overline{\mathbf{x}}_k) + f(\mathbf{x}_i) + \mathbf{g}_i^\top (\mathbf{x} - \overline{\mathbf{x}}_k + \overline{\mathbf{x}}_k - \mathbf{x}_i) \\ &= f(\overline{\mathbf{x}}_k) + \mathbf{g}_i^\top (\mathbf{x} - \overline{\mathbf{x}}_k) - \alpha_i, \end{aligned} \tag{41}$$

i.e.,

$$\mathbf{g}_i \in \partial f(\mathbf{x}_i) \quad \Rightarrow \quad \mathbf{g}_i \in \partial_{\alpha_i} f(\overline{\mathbf{x}}_k). \tag{42}$$

The latter property, often referred to as *subgradient transport*, is both conceptually and practically important; it indicates that even points which are far from the stability center provide approximate information on its differential properties.

Note also that points $\mathbf{x}_i$ do not play any role in the difference function (39), thus the bundle B_k, instead of *triplets*, can be considered as made up of *couples* as follows

$$B_k \triangleq \left\{ \big(\mathbf{g}_i, \alpha_i\big) : \mathbf{g}_i \in \partial_{\alpha_i} f(\overline{\mathbf{x}}_k),\, i \in \{1, \dots, k\} \right\}.$$

Note that the definition of the linearization errors is related to the current stability center. In case a new one is selected, say $\overline{\mathbf{x}}_{k+1}$, the α_i need to be updated. In fact, denoting by α_i^+, for each $i \in \{1, \ldots, k\}$, the new linearization errors updated with respect to $\overline{\mathbf{x}}_{k+1}$, it is easy to obtain the following update formula

$$\begin{aligned} \alpha_i^+ &= f(\overline{\mathbf{x}}_{k+1}) - f(\mathbf{x}_i) - g_i^\top(\overline{\mathbf{x}}_{k+1} - \mathbf{x}_i) \\ &= f(\overline{\mathbf{x}}_{k+1}) + f(\overline{\mathbf{x}}_k) - f(\overline{\mathbf{x}}_{k+}) - f(\mathbf{x}_i) - g_i^\top(\overline{\mathbf{x}}_{k+1} - \overline{\mathbf{x}}_k + \overline{\mathbf{x}}_k - \mathbf{x}_i) \\ &= \alpha_i + f(\overline{\mathbf{x}}_{k+1}) - f(\overline{\mathbf{x}}_k) + g_i^\top(\overline{\mathbf{x}}_{k+1} - \overline{\mathbf{x}}_k) \end{aligned} \tag{43}$$

which is independent of the explicit knowledge of points $\mathbf{x}_i$.

Under the transformation of variables introduced above, problem (31) becomes

$$\min \left\{ v : v \geq \mathbf{g}_i^\top \mathbf{d} - \alpha_i \ \forall i \in \{1, \ldots, k\},\, \mathbf{d} \in \mathbb{R}^n,\, v \in \mathbb{R} \right\} \tag{44}$$

whose optimal solution $(\mathbf{d}_k, v_k)$ is related to the optimal solution $(\mathbf{x}_{k+1}, w_k)$ of (31) by the relations:

$$\mathbf{d}_k = \mathbf{x}_{k+1} - \overline{\mathbf{x}}_k, \quad \text{and} \quad v_k = w_k - f(\overline{\mathbf{x}}_k) = f_k(\mathbf{x}_{k+1}) - f(\overline{\mathbf{x}}_k).$$

Note that from nonnegativity of the linearization errors it follows that the point $(\mathbf{d}, v) = (\mathbf{0}, 0)$ is feasible in (44), hence $v_k \leq 0$ represents the *predicted reduction* returned by the model at point $\mathbf{x}_{k+1} = \overline{\mathbf{x}}_k + \mathbf{d}_k$.

Bundle methods elaborate on CPM as they ensure:

i) Well-posedness of the optimization subproblem to be solved at each iteration;
ii) Stabilization of the next iterate.

A conceptual and very general scheme of a bundle method is now given, aiming at highlighting the main differences with CPM.

The schema of Algorithm 2 provides just the backbone of most bundle methods, as it leaves open a number of algorithmic decisions that can lead to fairly different methods. In fact, the body of literature devoted to BM is huge, as a vast number of variants have been proposed over time by many scientists, in order to implement such decisions. We postpone to Section 9 some bibliographic notes.

We give in the following a general classification of bundle methods, mainly based on the different variants of the subproblem (44) to be solved at Step 3, in order to satisfactorily deal with the aforementioned issues of well-posedness and stabilization. The approaches are substantially three:

- Proximal BM;
- Trust region BM;
- Level BM.

They share the same rationale of inducing some limitation on the distance between two successive iterates $\mathbf{x}_{k+1}$ and $\mathbf{x}_k$ gathering, at the same time, well-posedness and stability with respect to CPM. As it will be clarified in the following, the actual magnitude of such limitation is controlled by an approach-specific parameter (to be possibly updated at each iteration). Its appropriate tuning is the real crucial point affecting numerical performance of all BM variants.

For a better understanding of the impact on the performance of the stabilization strategies, we report the results of an instructive experiment described in [50]. For a given nonsmooth optimization problem, the Lagrangian dual of a Linear

Algorithm 2 Conceptual BM

```
Input: a starting point $\mathbf{x}_1 \in \mathbb{R}^n$
Output: an approximate $\epsilon$-optimal solution $\mathbf{x}^* \in \mathbb{R}^n$
 1: Calculate $\mathbf{g}_1 \in \partial f(\mathbf{x}_1)$, set $\overline{\mathbf{x}}_1 = \mathbf{x}_1$, and $\alpha_1 = 0$
 2: Set $B_1 = \{(\mathbf{g}_1, \alpha_1)\}$ and $k = 1$
 3: Solve an appropriate variant of subproblem (44)
 4: if solution of (44) certifies approximate optimality of point $\overline{\mathbf{x}}_k$ then
 5:     Set $\mathbf{x}^* = \overline{\mathbf{x}}_k$ and exit
 6: else
 7:     Adopt $\mathbf{d}_k$ as a tentative displacement from the current stability center $\overline{\mathbf{x}}_k$
 8:     Test the quality of the current cutting plane model (39) by comparing expected and
        actual reduction in the objective function at a testing point $\mathbf{x}_{k+1} = \overline{\mathbf{x}}_k + t\mathbf{d}_k$ for $t = 1$, or
        possibly for $t \in (0, 1]$
 9:     if a sufficient decrease in the objective function is achieved at $\mathbf{x}_{k+1}$ then
10:         Update the stability center $\overline{\mathbf{x}}_{k+1} = \mathbf{x}_{k+1}$
11:         Calculate $\mathbf{g}_{k+1} \in \partial f(\mathbf{x}_{k+1})$, and set $\alpha_{k+1} = 0$
12:         Update the linearization errors according to (43)
13:         Update the bundle $B_{k+1} = B_k \cup \{(\mathbf{g}_{k+1}, \alpha_{k+1})\}$, set $k = k + 1$ and go to 3
14:     else
15:         Leave the stability center unchanged $\overline{\mathbf{x}}_{k+1} = \overline{\mathbf{x}}_k$
16:         Calculate $\mathbf{g}_{k+1} \in \partial f(\mathbf{x}_{k+1})$
17:         Set $\alpha_{k+1} = f(\overline{\mathbf{x}}_{k+1}) - \big(f(\mathbf{x}_{k+1}) + \mathbf{g}_{k+1}^\top(\overline{\mathbf{x}}_{k+1} - \mathbf{x}_{k+1})\big)$
18:         Update the bundle $B_{k+1} = B_k \cup \{(\mathbf{g}_{k+1}, \alpha_{k+1})\}$, set $k = k + 1$ and go to 3
19:     end if
20: end if
```

Program, the minimizer $\mathbf{x}^*$ has been first calculated by standard CPM. Then, such an optimal point has been given as the starting point both to standard CPM and to a variant of CPM equipped with a constraint of the type $\|\overline{\mathbf{x}}_k + \mathbf{d} - \mathbf{x}^*\|_\infty \leq \varDelta$, for different values of $\varDelta$. In Table 1, for decreasing values of $\varDelta$, the ratio between the number of iterations upon termination of the modified CPM, $N_{it}^{\varDelta}$, and that of the standard CPM, N_{it}, is reported. The impressive effect of making more and more stringent the constraint on distance of two successive iterates is apparent.

Table 1 Instability of the standard CPM

$\varDelta$	1e+4	1e+2	1e+0	1e−2	1e−4	1e−5	1e−6
$\frac{N_{it}^{\varDelta}}{N_{it}}$	1.07	1.12	0.86	0.77	0.56	0.19	0.04

5.1.1 Proximal BM (PBM)

The proximal point variant of BM is probably the one that attracted most of the research efforts. It has solid theoretical roots in both the properties of the *Moreau-Yosida Regularization* [76] and Rockafellar's *Proximal Point Algorithm* [134]. In such class of methods the variant of subproblem (44), to be solved at Step 3 of Algorithm 2, is

$$\min\left\{v + \frac{1}{2}\gamma_k\|\mathbf{d}\|^2 : v \geq \mathbf{g}_i^\top\mathbf{d} - \alpha_i \ \forall i \in \{1, \ldots, k\}, \mathbf{d} \in \mathbb{R}^n, v \in \mathbb{R}\right\} \tag{45}$$

where $\gamma_k > 0$ is the adjustable *proximity* parameter. The latter problem can be rewritten, taking into account (39), in an equivalent unconstrained form as

$$\min\left\{f_k(\overline{\mathbf{x}}_k + \mathbf{d}) + \frac{1}{2}\gamma_k\|\mathbf{d}\|^2 - f_k(\overline{\mathbf{x}}_k) : \mathbf{d} \in \mathbb{R}^n\right\}, \tag{46}$$

hence it has a unique minimizer as a consequence of strict convexity of the objective function.

It is worth observing that in PBM the subproblem (45) is a quadratic program (QP), whose solution can be found either by applying any QP algorithm in $\mathbb{R}^n$, or by working in the dual space $\mathbb{R}^k$. In fact, the standard definition of Wolfe's dual for problem (45) is

$$\max\left\{v + \frac{1}{2}\gamma_k\|\mathbf{d}\|^2 - \sum_{i=1}^{k}\lambda_i\big(v - \mathbf{g}_i^\top\mathbf{d} + \alpha_i\big) : \gamma_k\mathbf{d} + \sum_{i=1}^{k}\lambda_i\mathbf{g}_i = 0,\, \mathbf{e}^\top\boldsymbol{\lambda} = 1,\right.$$
$$\left.\boldsymbol{\lambda} \geq \mathbf{0},\, \mathbf{d} \in \mathbb{R}^n,\, v \in \mathbb{R},\, \boldsymbol{\lambda} \in \mathbb{R}^k\right\} \tag{47}$$

where $\boldsymbol{\lambda} = (\lambda_1, \dots, \lambda_k)^\top$ is the vector of dual variables (or multipliers), and $\mathbf{e}$ is a vector of ones of appropriate size. Taking into account that

$$\mathbf{d} = -\frac{1}{\gamma_k}\sum_{i=1}^{k}\lambda_i\mathbf{g}_i,$$

it is possible to eliminate $\mathbf{d}$ and to restate the dual of problem (45) as follows:

$$\min\left\{\frac{1}{2\gamma_k}\|\sum_{i=1}^{k}\lambda_i\mathbf{g}_i\|^2 + \sum_{i=1}^{k}\lambda_i\alpha_i : \mathbf{e}^\top\boldsymbol{\lambda} = 1,\, \boldsymbol{\lambda} \geq \mathbf{0},\, \boldsymbol{\lambda} \in \mathbb{R}^k\right\} \tag{48}$$

Letting $(\mathbf{d}_k, v_k)$ and $\boldsymbol{\lambda}^{(k)}$ be, respectively, optimal solutions to (45) and (48), the following relations hold:

$$\mathbf{d}_k = -\frac{1}{\gamma_k}\sum_{i=1}^{k}\lambda_i^{(k)}\mathbf{g}_i \tag{49}$$

and

$$v_k = -\frac{1}{\gamma_k}\|\sum_{i=1}^{k}\lambda_i^{(k)}\mathbf{g}_i\|^2 - \sum_{i=1}^{k}\lambda_i^{(k)}\alpha_i, \tag{50}$$

which allow to equivalently solve either problem (45) or problem (48) at Step 3 of the Conceptual BM. Note that $\mathbf{d}_k$ is the opposite of a (scaled) convex combination of the $\mathbf{g}_i$s, and it reduces to the anti-subgradient with stepsize $\frac{1}{\gamma_k}$ in case the bundle is the singleton $\{(\mathbf{g}_k, 0)\}$, where $\mathbf{g}_k \in \partial f(\overline{\mathbf{x}}_k)$.

Working in the dual space is, in general, preferred for both practical and theoretical reasons. In fact, problem (48) has a nice structure, being the minimization of a convex quadratic function over the unit simplex, for which powerful ad hoc algorithms are available in the literature (e.g., [82,112,47]). On the other hand, relations (49)-(50) provide the theoretical basis for possibly certifying (approximate) optimality of the current stability center at Step 4 of Conceptual BM. Let $\mathbf{g}(\boldsymbol{\lambda}) = \sum_{i=1}^{k} \lambda_i^{(k)} \mathbf{g}_i$ and suppose the following holds

$$v_k \geq -\epsilon,$$

for some *small* $\epsilon > 0$. Thus, from (49)-(50) it follows that

$$\|\mathbf{g}(\boldsymbol{\lambda})\| \leq \sqrt{\gamma_k \epsilon} \tag{51}$$

and

$$\sum_{i=1}^{k} \lambda_i^{(k)} \alpha_i \leq \epsilon. \tag{52}$$

Moreover, condition (52), taking into account (42), implies that

$$\mathbf{g}(\boldsymbol{\lambda}) \in \partial_\epsilon f(\overline{\mathbf{x}}_k),$$

i.e., $\mathbf{g}(\boldsymbol{\lambda})$ is in the ϵ-subdifferential of f at $\overline{\mathbf{x}}_k$. On the other hand, condition (51) provides an upper bound on the norm of $\mathbf{g}(\boldsymbol{\lambda})$ and hence, taking into account the inequality (4) and letting $\delta = \sqrt{\gamma_k \epsilon}$ we obtain

$$f(\mathbf{x}) \geq f(\overline{\mathbf{x}}_k) + \mathbf{g}(\boldsymbol{\lambda})^\top (\mathbf{x} - \overline{\mathbf{x}}_k) - \epsilon \geq f(\overline{\mathbf{x}}_k) - \delta \|\mathbf{x} - \overline{\mathbf{x}}_k\| - \epsilon, \quad \forall \mathbf{x} \in \mathbb{R}^n, \tag{53}$$

which can be interpreted as an approximate optimality condition at point $\overline{\mathbf{x}}_k$, provided that δ is not *too big*. Note, however, that the magnitude of δ, once ϵ has been fixed, depends on the adjustable proximity parameter γ_k and, consequently, (53) is a sound approximate optimality condition only if the sequence $\{\gamma_k\}$ is bounded from above. Such condition is intuitively aimed at avoiding *shrinking* of the model around $\overline{\mathbf{x}}_k$, which would lead to both a very small $\mathbf{d}_k$ and to an *artificial* satisfaction of the condition $v_k \geq -\epsilon$. A complementary reasoning suggests to keep the sequence $\{\gamma_k\}$ bounded away from zero, in order for the algorithm to avoid behaving in a way very similar to standard CPM.

We have described, so far, the two possible outcomes from solving at Step 3 of Algorithm 2 problem (45) or, better, problem (48). In fact, in the PBM approach a significant displacement $\mathbf{d}_k$ is obtained if $v_k < -\epsilon$, while termination occurs in the opposite case when $v_k \geq -\epsilon$.

Now suppose that Step 6 has been reached, the point $\overline{\mathbf{x}}_k + \mathbf{d}_k$ being available as a possible candidate to become the new stability center. The predicted reduction at such point is v_k, which is to be compared with the actual reduction $f(\overline{\mathbf{x}}_k + \mathbf{d}_k) - f(\overline{\mathbf{x}}_k)$. Reasonable agreement between the two values indicates that the current cutting plane model is of good quality. Since at this stage $v_k < -\epsilon$, the agreement test at Step 8 is generally aimed at verifying that the actual reduction is just a fixed fraction of the predicted one, as shown in the following inequality

$$f(\overline{\mathbf{x}}_k + \mathbf{d}_k) - f(\overline{\mathbf{x}}_k) \leq m v_k, \tag{54}$$

where $m \in (0,1)$ is the *sufficient decrease* parameter. Hence, (54) also plays the role of the sufficient decrease condition to be checked at Step 9. In fact, if condition (54) is fulfilled, the algorithm can proceed through Steps 10 to 13, where the stability center is updated by setting $\overline{\mathbf{x}}_{k+1} = \overline{\mathbf{x}}_k + \mathbf{d}_k$ and a new iteration starts after updating the bundle. Such an exit is usually referred to as *Serious Step*. If, instead, there is poor agreement between actual and predicted reduction (i.e., a sufficient decrease has not been attained), it holds

$$f(\mathbf{x}_k + d_k) - f(\mathbf{x}_k) > mv_k,$$

and two possible implementations of the Conceptual BM are available, depending on whether or not a *line search* strategy is adopted.

In case no line-search approach is embedded in the algorithm, Conceptual BM proceeds to Steps 15 to 18, as the attempt to find a better stability center failed, and the stability center remains unchanged (i.e., a *Null Step* has occurred). Letting $\mathbf{x}_{k+1} = \overline{\mathbf{x}}_k + \mathbf{d}_k$, the new couple $(\mathbf{g}_{k+1}, \alpha_{k+1})$ is joined to the bundle, where $\mathbf{g}_{k+1} \in \partial f(\mathbf{x}_{k+1})$, and $\alpha_{k+1} = f(\overline{\mathbf{x}}_k) - f(\mathbf{x}_{k+1}) + \mathbf{g}_{k+1}^\top \mathbf{d}_k$.

In case a line-search strategy is adopted, the algorithm remains at Step 8, $\mathbf{d}_k$ is taken as a *search direction* and a line search (LS) is executed by checking at points $\overline{\mathbf{x}}_k + t\mathbf{d}_k$, with $t \in (0, 1]$, the objective function sufficient decrease condition

$$f(\overline{\mathbf{x}}_k + t\mathbf{d}_k) - f(\overline{\mathbf{x}}_k) \leq mtv_k, \tag{55}$$

skipping to Step 15 as soon as t falls below a given threshold $\eta \in (0, 1)$. Checks are performed for decreasing values of t, starting from $t = 1$, according to classic Armijo's rule [3].

Detailed presentation of nonsmooth LS algorithms (that is, the minimization of a nonsmooth function of one variable) is beyond the scope for this paper. We wish, however, to point out the fundamental difference between the smooth and the nonsmooth case. In the former case, once at any point $\mathbf{x}$ a search direction $\mathbf{d}$ is given within a descent algorithm, a *trusted* model, constituted by the negative directional derivative along $\mathbf{d}$ is available. It ensures that there exists a positive threshold $\bar{t}$ such that $f(\mathbf{x} + t\mathbf{d}) < f(\mathbf{x})$, for every $t \in (0, \bar{t})$. In the nonsmooth framework, instead, the cutting plane model is "untrusted", to recall the evocative term used in [50]. In fact, in the Conceptual BM the directional derivative at $\overline{\mathbf{x}}_k$ along $\mathbf{d}_k$ is not necessarily known, since v_k is just an approximation. Thus, the possibility that $\mathbf{d}_k$ is not a descent direction has to be accommodated by the algorithm.

We have now completed the discussion on the two possible implementations of Step 8 within the proximal version of Conceptual BM. We observe that null step is a result which can occur in both cases. It corresponds to the fact that the cutting plane has revealed a poor approximation of the objective function. Consequently, whenever a null step occurs, the stability center remains unchanged, and a new couple subgradient/linearization-error is added to the bundle, with the aim of improving the model. As for the latter, some explanations are in order. Consider, for an example, the null-step occurring when

$$f(\overline{\mathbf{x}}_k + \mathbf{d}_k) - f(\mathbf{x}_k) > mv_k, \tag{56}$$

with no line search performed. In such a case, after generating the new iterate $\mathbf{x}_{k+1} = \overline{\mathbf{x}}_k + \mathbf{d}_k$, the couple $\big(\mathbf{g}_{k+1}, \alpha_{k+1}\big)$ is appended to the bundle, where $\mathbf{g}_{k+1} \in \partial f(\mathbf{x}_{k+1})$ and $\alpha_{k+1} = f(\overline{\mathbf{x}}_k) - f(\mathbf{x}_{k+1}) + \mathbf{g}_{k+1}^\top \mathbf{d}_k$. The updated model in terms of difference function, see (39), becomes

$$\begin{aligned} f_{k+1}(\overline{\mathbf{x}}_k + \mathbf{d}) - f(\overline{\mathbf{x}}_k) &= \max\left\{\mathbf{g}_i^\top \mathbf{d} - \alpha_i : i \in \{1, \dots, k+1\}\right\} \\ &= \max\left\{f_k(\overline{\mathbf{x}}_k + \mathbf{d}) - f(\overline{\mathbf{x}}_k),\, \mathbf{g}_{k+1}^\top \mathbf{d} - \alpha_{k+1}\right\} \end{aligned} \tag{57}$$

Observe that, for $\mathbf{d} = \mathbf{d}_k$ there hold

$$\begin{aligned} \mathbf{g}_{k+1}^\top \mathbf{d}_k - \alpha_{k+1} &= \mathbf{g}_{k+1}^\top \mathbf{d}_k - f(\mathbf{x}_{k+1}) + f(\overline{\mathbf{x}}_k) - \mathbf{g}_{k+1}^\top \mathbf{d}_k \\ &= f(\mathbf{x}_{k+1}) - f(\overline{\mathbf{x}}_k) > m v_k > v_k \end{aligned} \tag{58}$$

and

$$f_k(\overline{\mathbf{x}}_k + \mathbf{d}_k) - f(\overline{\mathbf{x}}_k) = v_k, \tag{59}$$

which combined means that the updated model provides a more accurate estimate of the objective function f, at least around point $\mathbf{x}_{k+1}$. Perfectly analogous considerations can be made in case a line search scheme is adopted at Step 8.

We have presented, so far, some general ideas on how the Conceptual BM works in case the proximal approach is adopted. We do not enter into the details of convergence proofs, which depend on the different strategies adopted at various steps. We only wish to sketch how a typical convergence proof works, under the assumptions that f has a finite minimum, that the proximity parameter γ_k stays within a range $0 < \gamma_{min} \leq \gamma_k \leq \gamma_{max}$, possibly being adjusted upon modification of the stability center only. As already mentioned, such tuning is a crucial issue in view of granting numerical efficiency to the method.

The proof is based on the following three facts:

a) The objective function reduction, every time the stability center is updated, is bounded away from zero. This is a consequence of $v_k < -\epsilon$ and of the sufficient decrease condition (54), in case no line search strategy is adopted. Whenever, instead, a line search is performed, objective function reduction is still bounded away from zero as a consequence, again, of $v_k < -\epsilon$, of condition (55), and of the lower bound η on the stepsize length t.

b) Since it has been assumed that the function has finite minimum, from a) it follows that only a finite number of stability center updates may take place.

c) The Conceptual BM cannot loop infinitely many times through Step 18, that is an infinite sequence of null steps cannot occur. To prove this fact it is necessary to observe that, being the proximity parameter constant by assumption, the sequence $\{v_k\}$ is monotonically increasing, see (58), and bounded from above by zero, hence it is convergent. The core of the proof consists in showing that $\{v_k\} \to 0$ and, consequently, the stopping test $v_k \geq -\epsilon$ is satisfied after a finite number of null steps.

5.1.2 Trust region BM (TBM)

The approach consists in solving at Step 3 of the Conceptual BM the following variant of problem (44), obtained through the addition of a trust region constraint

$$\min\left\{v : v \geq \mathbf{g}_i^\top \mathbf{d} - \alpha_i \ \forall i \in \{1,\ldots,k\}, \|\mathbf{d}\| \leq \mathit{\Delta}_k,\, \mathbf{d} \in \mathbb{R}^n,\, v \in \mathbb{R}\right\}, \tag{60}$$

where $\mathit{\Delta}_k > 0$. Well-posedness is a consequence of continuity of the objective function, problem (60) being in fact a finite min-max, and compactness of the feasible region.

A first issue about the statement of problem (60) is the choice of the norm in the trust region constraint. It is in general preferred to adopt the ℓ_1 or the ℓ_∞ norm, so that (60) is still a Linear Program. A second relevant point is the setting of the trust region parameter. Intuitively, $\mathit{\Delta}_k$ must not be *too small*, which would result in slow convergence due to shrinking of the next iterate close to the stability center. On the other hand, a *too large* $\mathit{\Delta}_k$ would kill the stabilizing effect of the trust region. A simple approach is to provide two thresholds $\mathit{\Delta}_{min}$ and $\mathit{\Delta}_{max}$, letting $\mathit{\Delta}_k \in [\mathit{\Delta}_{min}, \mathit{\Delta}_{max}]$. Such choice is necessary to guarantee convergence of the algorithm, but the type of heuristics adopted for tuning $\mathit{\Delta}_k$ within the prescribed interval strongly affects both convergence and the overall performance of the algorithm (see the discussion about the effect of the proximity parameter γ_k in PBM).

Also for the trust region approach the two classes of variants, with or without line search, can be devised. Moreover, the interplay serious-step/null-step is still embedded into the conceptual scheme.

5.1.3 Proximal Level BM (PLBM)

The level set approach to BM stems from the general setting of CPM we gave earlier in this section, where point $\mathbf{x}_{k+1}$ calculated at Step 2 of GCPM was not necessarily a minimizer of f_k, convergence being ensured provided it was sufficiently inside the level set of function f_k at point $\mathbf{x}_k$.

The approach consists in finding the closest point to the current stability center $\overline{\mathbf{x}}_k$ where the difference function (39) takes a sufficiently negative value. In fact, problem (44) is modified as follows

$$\min\left\{\frac{1}{2}\|\mathbf{d}\|^2 : \mathbf{g}_i^\top \mathbf{d} - \alpha_i \leq -\theta_k \ \forall i \in \{1,\ldots,k\},\, \mathbf{d} \in \mathbb{R}^n\right\} \tag{61}$$

where the adjustable parameter $\theta_k > 0$ indicates the desired reduction in the cutting plane function. In fact, letting, as usual, the stability center $\overline{\mathbf{x}}_k$ be the incumbent, and denoting by $\mathbf{d}_k$ the optimal solution of (61), the point $\mathbf{x}_{k+1} = \overline{\mathbf{x}}_k + \mathbf{d}_k$ belongs to the following level set

$$S_k(\theta_k) = \{\mathbf{x} : f_k(\mathbf{x}) \leq f_k(\overline{\mathbf{x}}_k) - \theta_k\}.$$

of the cutting plane function f_k. Note that an appropriate choice of θ_k provides the required stabilization effect, as a small value of θ_k results in small $\|\mathbf{d}_k\|$.

The approach is known as Proximal Level Bundle Method (PLBM) and indeed the setting of θ_k is the key issue to address. To this aim, the optimal value of the model function f_k, say f_k^*, is required. Consequently, we stay in the same constrained context (34) adopted in stating CPM, so that problem

$$f_k^* = \min \left\{ f_k(\mathbf{x}) : \mathbf{x} \in Q \subset \mathbb{R}^n \right\}$$

is well posed, being the convex set Q nonempty and compact. Since the incumbent value $f_k(\overline{\mathbf{x}}_k)$ and f_k^* are, respectively, an upper and a lower bound on f^*, it is quite natural to set θ_k on the basis of the gap

$$\Gamma(k) = f_k(\overline{\mathbf{x}}_k) - f_k^*, \tag{62}$$

which is a nonincreasing function of the bundle size k. A possible choice is to set $\theta_k = \mu\Gamma(k)$, for some $\mu \in (0,1)$, but modifications of such criterion are to be accommodated on the basis of comparison with the previous value of the gap. Note that $\Gamma(k) \le \epsilon$ provides an obvious stopping criterion for PLBM, since from $f_k(\overline{\mathbf{x}}_k) = f(\overline{\mathbf{x}}_k)$ it follows that $f(\overline{\mathbf{x}}_k) - f^* \le \Gamma(k)$. In terms of the Conceptual BM, Step 8 is neglected, and the test at Step 9, for possibly updating the stability center, is based on the simple reduction of the incumbent value. As for method implementation, further observations are in order.

- Compared to PBM and TBM, setting of θ_k appears definitely more amenable than choosing γ_k and Δ_k, respectively, as it simply refers to function values, while γ_k and Δ_k are meant to capture some kind of second order behavior of f, an ambitious and fairly hard objective.
- Unlike PBM and BTM, two distinct optimization subproblems are to be solved at each iteration: the quadratic problem (61), which consists in projecting $\overline{\mathbf{x}}_k$ onto $S_k(\theta_k)$, and (62), which is a linear program, in case Q has a *simple* structure (e.g., it is a hyper-interval).

The following theoretical result, see [99, Th. 2.2.2], provides a bound on the number of iterations needed to get an ϵ-approximate solution.

Theorem 4 *Let L_Q be the Lipschitz constant of f on Q, denote by D the diameter of Q, and by c a constant depending on parameter μ. For any given $\epsilon > 0$ it holds that*

$$k > c\left(\frac{L_Q D}{\epsilon}\right)^2 \qquad \Rightarrow \qquad f(\overline{\mathbf{x}}_k) - f^* \le \epsilon.$$

5.2 Making BM implementable

The algorithms we have described in this Section suffer from a major drawback. They are all based on unlimited accumulation of information, in terms of number of generated linearization or, equivalently, of bundle size. Convergence properties we have discussed are in fact valid under such hypothesis. This makes such methods, at least in theory, not implementable. In the sequel, focusing in particular on PBM, we briefly review two strategies to overcome such difficulty, introduced in [80,81], named subgradient *selection* and *aggregation*, respectively.

The strategies are both based on thorough analysis of the dual formulation (48) of the problem to be solved at Step 3 of Conceptual BM. Observe, in fact, that strict convexity of problem (45) ensures that the optimal solution $\mathbf{d}_k$ is unique and it is a (scaled) convex combination of the $\mathbf{g}_i$s, see (49). Note also that the optimal solution of the dual (48) is not necessarily unique, but there exists (by Carathéodory's Theorem) a set of at most $n+1$ optimal dual multipliers $\lambda_i^{(k)} > 0$, $i \in I_k$, $|I_k| \leq n+1$ such that

$$\mathbf{d}_k = -\frac{1}{\gamma_k} \sum_{i \in I_k} \lambda_i^{(k)} \mathbf{g}_i.$$

They can be calculated, in fact, by finding an optimal basic solution of the following linear program

$$\min \left\{ \sum_{i=1}^{k} \lambda_i \alpha_i : \sum_{i=1}^{k} \lambda_i \mathbf{g}_i = -\gamma_k \mathbf{d}_k,\, \mathbf{e}^\top \boldsymbol{\lambda} = 1,\, \boldsymbol{\lambda} \geq \mathbf{0},\, \boldsymbol{\lambda} \in \mathbb{R}^k \right\}, \tag{63}$$

which is characterized by $(n+1)$ constraints.

On the basis of previous observation there is an obvious possibility, once such set of subgradients has been detected, to *select* the corresponding bundle couples and to cancel the remaining ones, while the solutions of (48) and (45) remain unchanged. In this way the bundle size can be kept finite, without impairing overall convergence. It is worth noting that ad hoc algorithms for solving (48) are designed to automatically satisfy the condition that no more than $(n+1)$ subgradients are "active" in the definition of d_k, so that solution of problem (63) is not necessary for subgradient selection purposes.

In many practical cases, however, $n+1$ is still too large in view of the need of solving at each iteration the quadratic program (48) of corresponding size. In such a case, a very strong reduction in bundle size can be obtained by means of the subgradient *aggregation* mechanism. Once the optimal solution $\boldsymbol{\lambda}^{(k)}$ to (48) has been found, the aggregate couple $(\mathbf{g}_a, \alpha_a)$ is obtained by letting

$$\mathbf{g}_a = \sum_{i=1}^{k} \lambda_i^{(k)} \mathbf{g}_i \qquad \text{and} \qquad \alpha_a = \sum_{i=1}^{k} \lambda_i \alpha_i^{(k)}.$$

In addition, define the single-constraint aggregate quadratic program

$$\min \left\{ v + \frac{1}{2}\gamma_k \|\mathbf{d}\|^2 : v \geq \mathbf{g}_a^\top \mathbf{d} - \alpha_a,\, \mathbf{d} \in \mathbb{R}^n,\, v \in \mathbb{R} \right\}, \tag{64}$$

and observe that it is equivalent to the simple unconstrained quadratic problem

$$\min \left\{ \frac{1}{2}\gamma_k \|\mathbf{d}\|^2 + \mathbf{g}_a^\top \mathbf{d} - \alpha_a : \mathbf{d} \in \mathbb{R}^n \right\}.$$

Hence, the optimal solution $(\mathbf{d}_a, v_a)$ to (64) coincides with the solution to (48) since it can be obtained in closed form as

$$\mathbf{d}_a = -\frac{1}{\gamma_k} \mathbf{g}_a = -\frac{1}{\gamma_k} \sum_{i=1}^{k} \lambda_i^{(k)} \mathbf{g}_i = \mathbf{d}_k$$

and

$$v_a = \mathbf{g}_a^\top \mathbf{d}_a - \alpha_a = -\frac{1}{\gamma_k} \| \sum_{i=1}^{k} \lambda_i^{(k)} \mathbf{g}_i \|^2 - \sum_{i=1}^{k} \lambda_i^{(k)} \alpha_i = v_k.$$

Summing up, the aggregate problem (64) retains the fundamental properties of (48), so that, when point $\mathbf{x}_{k+1}$ is generated, all past bundle couples $(\mathbf{g}_i, \alpha_i)$ can be replaced by the unique $(\mathbf{g}_a, \alpha_a)$, and the new couple $(\mathbf{g}_{k+1}, \alpha_{k+1})$, with $\mathbf{g}_{k+1} \in \partial f(\mathbf{x}_{k+1})$ is added to the bundle. Under such aggregation scheme, with the bundle containing just two elements, it is possible to show convergence. Such version of proximal BM is sometimes referred to as the "poorman" bundle. Of course many other selection-aggregation schemes have been discussed in the literature. Their treatment is, however, beyond the scope of this paper.

6 Miscellaneous algorithms

In Section 5 we have mostly discussed about bundle methods, a family of NSO algorithms related to cutting plane, which is a model function grounded on information coming from many points spread throughout the objective function domain. Such a feature keeps bundle methods somehow apart from the smooth optimization mainstream, where most of the popular iterative methods (Gradient type, Conjugate Gradient, Newton, quasi–Newton etc.) are based on information on the objective function related to the current iterate or, sometimes, also to the previous one. Several scientists have thus tried to convey to NSO, and in particular to cutting-plane based area, some ideas coming from smooth optimization, upon appropriate modifications to cope with nonsmoothness. In this section we briefly survey some of such attempts.

6.1 Variable metric

In discussing the proximal BM we have already observed that tuning of the proximity parameter γ_k in problem (45) has a strong impact on the numerical performance of such class of algorithms. The problem has been addressed by many authors (see, e.g., [83]) and several heuristic techniques are available. More in general, setting of γ_k is related to the attempt of capturing some kind of second order approximation of the objective function. After all, the quadratic term $\frac{1}{2}\gamma_k \|\mathbf{d}\|^2$, in case f is twice differentiable, would be seen as a single–parameter positive definite approximation $\gamma_k I$ of the Hessian at point $\overline{\mathbf{x}}_k$, I being the identity matrix[1].

Thus, the simplest idea, see [94], is to replace (45) with the following problem

$$\min \left\{ v + \frac{1}{2}\mathbf{d}^\top B_k \mathbf{d} : \, v \geq \mathbf{g}_i^\top \mathbf{d} - \alpha_i \ \forall i \in \{1, \dots, k\}, \, \mathbf{d} \in \mathbb{R}^n, \, v \in \mathbb{R} \right\}, \tag{65}$$

where B_k is a positive definite matrix in $\mathbb{R}^{n \times n}$ to be updated any time the stability center changes, according to some rule inspired by the Quasi–Newton updates for smooth minimization. We recall that in all Quasi–Newton methods the Hessian (or its inverse) approximation is updated, in correspondence to the iterate $\overline{\mathbf{x}}_k$, on the basis of the following differences in *points* and *gradients* between two successive iterates

$$\mathbf{s}_k = \overline{\mathbf{x}}_k - \overline{\mathbf{x}}_{k-1} \quad \text{and} \quad \mathbf{q}_k = \nabla f(\overline{\mathbf{x}}_k) - \nabla f(\overline{\mathbf{x}}_{k-1}).$$

[1] Given the features of the adopted machinery, we keep on denoting the current iterate (i.e., the estimate of a minimizer) by $\overline{\mathbf{x}}_k$, although the methods involved in this class are not necessarily of the bundle type.

A straightforward and practical way to adopt a Quasi-Newton approach in the nonsmooth environment would be to use any classic variable metric algorithm based on updating formulae (e.g., DFP, BFGS, etc.), with $\mathbf{q}_k$ defined as difference of subgradients instead of gradients. Note, in passing, that due to possible discontinuities in derivatives, *large* $\mathbf{q}_k$ may correspond to *small* $\mathbf{s}_k$. This, however, is not a reportedly serious drawback in terms of practical applications (see classic [97] and [144] for an accurate analysis).

As a consequence of previous observation, research has focused on the definition of some *differentiable object*, related to f, thus suitable for application of Quasi-Newton methods. Such an object, the Moreau-Yosida regularization of f, is the function $\phi_\rho : \mathbb{R}^n \to \mathbb{R}$, defined as

$$\phi_\rho(\mathbf{x}) \triangleq \min\left\{f(\mathbf{y}) + \frac{\rho}{2}\|\mathbf{y}-\mathbf{x}\|^2 \,:\, \mathbf{y} \in \mathbb{R}^n\right\}, \tag{66}$$

for some $\rho > 0$, whose minimizer is denoted by

$$p_\rho(\mathbf{x}) = \arg\min\left\{f(\mathbf{y}) + \frac{\rho}{2}\|\mathbf{y}-\mathbf{x}\|^2 \,:\, \mathbf{y} \in \mathbb{R}^n\right\}$$

and referred to as the *proximal point* of $\mathbf{x}$, see [134]. Function ϕ_ρ enjoys the following properties:

- The sets of minima of f and ϕ_ρ coincide;
- ϕ_ρ is differentiable (see [76]);
- $\nabla\phi_\rho(\mathbf{x}) = \rho(\mathbf{x} - p_\rho(\mathbf{x})) \in \partial f(p_\rho(\mathbf{x}))$, since at $p_\rho(\mathbf{x})$ it is $0 \in \partial h(\mathbf{y})$, where $h(\mathbf{y}) = f(\mathbf{y}) + \frac{\rho}{2}\|\mathbf{y}-\mathbf{x}\|^2$ is a strictly convex function.

The latter properties allow to find a minimum of f by solving the following (smooth) problem

$$\min\left\{\phi_\rho(\mathbf{x}) \,:\, \mathbf{x} \in \mathbb{R}^n\right\}. \tag{67}$$

Here, we note that smoothness is not gathered for free, as calculation of the new objective function ϕ_ρ requires solution of a convex (nonsmooth) optimization problem.

Straightforward application of any Quasi-Newton paradigm (equipped with a line search) to minimize ϕ_ρ leads to the following iteration scheme:

$$\overline{\mathbf{x}}_{k+1} = \overline{\mathbf{x}}_k - t_k B_k^{-1}\nabla\phi_\rho(\overline{\mathbf{x}}_k), \tag{68}$$

where B_k is the classic approximation of the Hessian, and a line search is accommodated into the iteration scheme to fix the stepsize $t_k > 0$ along the Quasi–Newton direction $\mathbf{d}_k = -B_k^{-1}\nabla\phi_\rho(\overline{\mathbf{x}}_k)$

Matrix B_k comes from updating of B_{k-1} (usually choosing $B_1 = I$), by means of one of the effective Quasi–Newton formulae. A popular one is BFGS, according to which it is

$$B_k = B_{k-1} - \frac{B_{k-1}\mathbf{s}_k\mathbf{s}_k^\top B_{k-1}}{\mathbf{s}_k^\top B_{k-1}\mathbf{s}_k} + \frac{\mathbf{q}_k\mathbf{q}_k^\top}{\mathbf{q}_k^\top\mathbf{s}_k}$$

with $\mathbf{s}_k = \overline{\mathbf{x}}_k - \overline{\mathbf{x}}_{k-1}$ and $\mathbf{q}_k = \nabla\phi_\rho(\overline{\mathbf{x}}_k) - \nabla\phi_\rho(\overline{\mathbf{x}}_{k-1})$. In fact, matrix B_k satisfies the secant equation

$$B_k\mathbf{s}_k = \mathbf{q}_k.$$

A (simplified) algorithmic scheme is reported in Algorithm 3.

Algorithm 3 Quasi–Newton Scheme (QN)

```
At point $\overline{\mathbf{x}}_k$, let $\phi_\rho(\overline{\mathbf{x}}_k)$, $\nabla\phi_\rho(\overline{\mathbf{x}}_k)$, and a p.d. matrix $B_k$ be available.   ▷ Outer loop
Calculate $\overline{\mathbf{x}}_{k+1}$ as in (68).
Calculate $\phi_\rho(\overline{\mathbf{x}}_{k+1})$ by solving (66)   ▷ Inner loop
Check a stopping condition.   ▷ Termination test
Calculate $B_{k+1}$ as a Quasi–Newton update of $B_k$.
Set $k = k + 1$ and return to the outer loop.
```

The QN scheme of Algorithm 3 leaves open several relevant issues. We note first that the inner loop deals with minimization of a (strictly) convex nonsmooth function. Thus, it is quite natural to apply in such framework the machinery we have discussed in previous sections (e.g., any bundle-type algorithm would be in order). On the other hand, the idea of exactly solving at each iteration a problem of the same difficulty as the original one does not appear viable in terms of computation costs. In fact, it is appropriate to settle for an *approximate* solution of problem (66) in the inner loop, which results in *inexact* calculation of $\overline{\mathbf{x}}_{k+1}$ as, instead of the exact optimality condition $\rho(\overline{\mathbf{x}}_{k+1} - p_\rho(\overline{\mathbf{x}}_{k+1})) \in \partial f(p_\rho(\overline{\mathbf{x}}_{k+1})$, the approximate one $\rho(\overline{\mathbf{x}}_{k+1} - p_\rho(\overline{\mathbf{x}}_{k+1})) \in \partial_\epsilon f(p_\rho(\overline{\mathbf{x}}_{k+1})$, for some $\epsilon > 0$, is enforced. We note in passing that the Quasi–Newton framework is one of the areas that have solicited the development of a convergence theory for NSO algorithms with inexact calculation of function and/or subgradient (see Section 7).

The need of accommodating for inexact calculation of the Moreau-Yosida regularization ϕ_ρ (consider that also tuning of ρ is a significant issue), has also an impact on the implementation of the choice of $\overline{\mathbf{x}}_{k+1}$ in the outer loop, irrespective of whether a line search is executed, as evoked by formula (68), or the constant stepsize $t_k = 1$ is adopted. We do not enter into the technicalities of the above mentioned issues. Possible choices are relevant in establishing the theoretical convergence rate of QN type algorithms. Discussion on such topics can be found in [20,100,27].

6.2 Methods of centers (MoC)

We have already seen how fecund was the cutting plane idea of using *many* linearizations, generated all over the function domain, in order to obtain a *global*, not just *local*, model of the objective function. Yet another approach deriving from cutting plane is a class of methods known as Methods of Centres, whose connection with interior methods for Linear Programming is apparent. To explain the basic ideas it is convenient to assume a *function-oriented* viewpoint instead of a *set-oriented* one.

In solving the (constrained) problem (34), the same framework as CP (or BM) is adopted. Given the cutting plane function f_k, available at iteration k, we denote by $F_k(z_k)$ the following subset of $\mathbb{R}^{n+1}$

$$F_k(z_k) = \{(\mathbf{x}, v) : \mathbf{x} \in Q,\ f_k(\mathbf{x}) \le v \le z_k\}\,,$$

where z_k is any upper bound on the optimal value of f_k (e.g., the value of f calculated at any feasible point). The set $F_k(z_k)$, next referred to as the *localization set*, is contained in *epi*f_k, being obtained by horizontally cutting *epi*f_k, and it contains the point $(\mathbf{x}^*, f^*)$.

The basic idea of MoC is to construct a nested sequence of sets $F_k(z_k)$ shrinking as fast as possible around the point $(\mathbf{x}^*, f^*)$, by introducing a *cut* at each iteration. To obtain substantial volume reduction in passing from $F_k(z_k)$ to $F_{k+1}(z_{k+1})$, one looks for a *central* cut, i.e., a cut generated on the basis of some notion of center of $F_k(z_k)$. Several proposals in this context can be found in the literature, stemming from Levin's "Center of Gravity" method [101], which is based on the property that for a given convex set C with nonempty interior, any hyperplane passing through the center of gravity generates a cut which reduces the volume of C by a factor of at least $(1 - e^{-1})$. However, such substantial reduction in the volume of F_k can only be obtained by solving the hard problem of locating the center of gravity.

Next we particularly focus on a more practical proposal, the *Analytic Center Cutting Plane Method* (ACCPM), see [67,68,123], which is based on the notion of "analytic center" introduced in [139] as a point that maximizes the product of distances to all faces of $F_k(z_k)$.

Thus, in the ACCPM the required central point of the localization set is calculated as the unique maximizer of the potential function

$$\psi_k(\mathbf{x}, v) = \log(z_k - v) + \sum_{i=1}^{k} \log[v - f(\mathbf{x}_i) - \mathbf{g}_i^\top(\mathbf{x} - \mathbf{x}_i)].$$

Once the analytic center

$$(\mathbf{x}_{k+1}, v_{k+1}) = \arg\max \Big\{ \psi_k(\mathbf{x}, v) : (\mathbf{x}, v) \in F_k(z_k) \Big\}$$

has been obtained, function f_k is updated thanks to the new cut generated at $\mathbf{x}_{k+1}$, and the value z_k is possibly updated. A stopping condition is tested, which is based on the difference between the upper bound and a lower bound obtained by minimizing f_{k+1} over Q, and the procedure possibly iterated. Calculation of the analytic center can be performed by adapting interior point algorithms for Linear Programming based on the use of potential functions (see, e.g., [31]). Complexity estimates of the method, with possible embedding of a proximal term in calculating the analytic center, are presented in [117]

Yet another possibility is to adopt, instead of the analytic center, the *Chebyshëv center*, defined as the center of the largest sphere contained in $F_k(z_k)$. The approach, originally proposed in [42], has been equipped with a quadratic stabilizing term in [123].

An original approach somehow related to this area can be finally found in [19].

6.3 Gradient sampling

The fundamental fact behind most of NSO method is that satisfaction of an angle condition, that of forming an obtuse angle with a subgradient, is not enough for a direction to be a descent one. The angle condition, in fact, must be *robust*, that is the direction has to make an obtuse angle with *many* subgradients around the point. Based on this observation, and considering that for most practical problems the objective function is differentiable almost everywhere, *gradient sampling* algorithms have been introduced, whose key feature is the evaluation of subgradient (i.e., gradient with probability 1) on a set of random points close to the current iterate. All such gradients are then used to obtain a search direction.

A sketch of an iteration of gradient sampling algorithm is reported in Algorithm 4, see [22, 24]. We do not report, for simplicity of notation, the iteration counter and thus we indicate by $\mathbf{x}$ the current iterate. The algorithm works on the basis of two couples of *stationarity/sampling-radius* tolerances, the overall (η, ϵ) and the iteration–dependent (θ, δ), respectively.

Algorithm 4 Gradient Sampling Scheme (GS)

```
1: Let $\mathbf{x}$ be the current iterate, where function $f$ is differentiable;
   Compute the gradient $\mathbf{g}_0 = \nabla f(\mathbf{x})$;
   Sample independently $m \geq n+1$ points $\mathbf{y}_1, \dots, \mathbf{y}_m$ uniformly random      ▷ Sampling
   in the ball of radius $\delta$ centered at $\mathbf{x}$;
   Obtain at each of such points a gradient, say $\mathbf{g}_i = \nabla f(\mathbf{y}_i)$, $i \in \{1, \dots, m\}$;
2: Obtain a direction $\mathbf{d}$, if any, that forms an obtuse angle with all $m+1$ gradients;
   It can be obtained (see (45) or (48)-(49) for $\gamma_k = 1$ and $\alpha_i = 0$ for every $i$)
   as $\mathbf{d} = -\mathbf{g}^* = -\arg\min\{\frac{1}{2}\|\mathbf{g}\|^2 : \mathbf{g} \in \mathrm{conv}\{\mathbf{g}_0, \dots, \mathbf{g}_m\}\}$;      ▷ Direction finding
3: Stop in case $\|\mathbf{d}\| < \eta$ and $\delta < \epsilon$ (overall tolerances met);      ▷ Termination test
   In case $\|\mathbf{d}\| < \theta$, reduce by constant reduction factors both $\theta$ and $\delta$;  ▷ Parameter update
4: Perform an Armijo-type line search along $\mathbf{d}$ and calculate a sufficient      ▷ Line search
   decrease stepsize $t$;
   Move to the new point $\mathbf{x} + t\mathbf{d}$ if at such point $f$ is differentiable or, if this
   is not the case, to a point *close* to $\mathbf{x} + t\mathbf{d}$ where sufficient decrease is still
   achieved and $f$ is differentiable.
```

It can be proved that an algorithm based on the above iteration scheme provides a sequence of points $\{\mathbf{x}_k\}$ converging to a Clarke stationary point with probability 1, unless $f(\mathbf{x}_k) \to -\infty$. A necessary assumption is that the set of points where f is *continuously* differentiable is open, dense and full measure in $\mathbb{R}^n$, while no convexity assumption is required for ensuring convergence.

7 Inexact calculation of function and/or subgradient

We have already seen a case where it is advisable to dispose of a method for minimizing a convex function without requiring its *exact* calculation, see §6.1. This is a typical case in the wide application field of Lagrangian relaxation for hard ILP problems.

Next we briefly recall some basic facts, see [65]. Suppose the following ILP problem is to be solved

$$z_I = \max\left\{\mathbf{c}^\top \mathbf{x} : A\mathbf{x} = \mathbf{b},\ B\mathbf{x} = \mathbf{d}, \mathbf{x} \geq \mathbf{0}, \mathbf{x} \in \mathbb{Z}^n\right\} \tag{69}$$

with $\mathbf{c} \in \mathbb{R}^n$, $A \in \mathbb{R}^{m \times n}$, $B \in \mathbb{R}^{p \times n}$, $\mathbf{b} \in \mathbb{R}^m$, $\mathbf{d} \in \mathbb{R}^p$, and $\mathbb{Z}^n$ denoting the set of n-dimensional vectors. We assume that the problem is feasible and that the set

$$X = \left\{\mathbf{x} \in \mathbb{Z}^n : B\mathbf{x} = \mathbf{d}, \mathbf{x} \geq 0\right\}$$

is finite, that is $X = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_K\}$ and $\mathcal{K} = \{1, 2, \dots, K\}$ is the corresponding index set.

Assume also that constraints are partitioned into two families, those defined through $A\mathbf{x} = \mathbf{b}$ being the *complicating* ones. A Lagrangian relaxation of (69) is obtained by relaxing complicating constraints as follows

$$z(\boldsymbol{\lambda}) = \max \left\{ \mathbf{c}^\top \mathbf{x} + \boldsymbol{\lambda}^\top (\mathbf{b} - A\mathbf{x}) : \mathbf{x} \in X \right\}, \tag{70}$$

where $\boldsymbol{\lambda} \in \mathbb{R}^m$. Problem (70), which is still an ILP, provides an *upper bound* for problem (69), namely,

$$z(\boldsymbol{\lambda}) \geq z_I.$$

Moreover, denoting by $\mathbf{x}(\boldsymbol{\lambda}) \in \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_K\}$ the optimal solution of (70), it holds that

$$\begin{aligned} z(\boldsymbol{\lambda}) &= \mathbf{c}^\top \mathbf{x}(\boldsymbol{\lambda}) + \boldsymbol{\lambda}^\top \big(\mathbf{b} - A\mathbf{x}(\boldsymbol{\lambda})\big) \\ &= \max \left\{ \mathbf{c}^\top \mathbf{x}_k + \boldsymbol{\lambda}^\top (\mathbf{b} - A\mathbf{x}_k) : k \in \{1, \dots, K\} \right\}, \end{aligned} \tag{71}$$

$z(\boldsymbol{\lambda})$ being often referred to as the *dual* function. We note that, in case $\mathbf{x}(\boldsymbol{\lambda})$ is feasible (i.e., $A\mathbf{x}(\boldsymbol{\lambda}) = \mathbf{b}$), then it is also optimal for (69).

Aiming for the *best* among the upper bounds (i.e., the one closest to z_I), we define the Lagrangian dual problem as

$$\begin{aligned} z_{LD} &= \min \left\{ z(\boldsymbol{\lambda}) : \boldsymbol{\lambda} \in \mathbb{R}^m \right\} \\ &= \min \left\{ \max \left\{ \mathbf{c}^\top \mathbf{x}_k + \boldsymbol{\lambda}^\top (\mathbf{b} - A\mathbf{x}_k) : k \in \{1, \dots, K\} \right\} : \boldsymbol{\lambda} \in \mathbb{R}^m \right\}, \end{aligned} \tag{72}$$

z_{LD} being the best upper bound obtainable through Lagrangian relaxation.

Problem (72) consists in the minimization of a convex function defined as the pointwise maximum of K affine functions of $\boldsymbol{\lambda}$, one for each feasible point in X. In fact, it is a convex nonsmooth optimization problems which can be tackled by means of any of the methods described in previous sections.

Very often, once the complicating constraints have been removed, the Lagrangian relaxation is easy to solve. If this is not the case, however, any iterative NSO method which requires at each iteration its exact solution may lead to prohibitive computation time. Now suppose we are able to solve approximately the Lagrangian relaxation (70), that is, we are able to obtain for any given $\bar{\boldsymbol{\lambda}}$ an approximation of $z(\bar{\boldsymbol{\lambda}})$, say $\tilde{z}(\bar{\boldsymbol{\lambda}}) = z(\bar{\boldsymbol{\lambda}}) - \epsilon$, for some $\epsilon \geq 0$. Suppose, in particular, that

$$\tilde{z}(\bar{\boldsymbol{\lambda}}) = \mathbf{c}^\top \widetilde{\mathbf{x}}(\bar{\boldsymbol{\lambda}}) + \bar{\boldsymbol{\lambda}}^\top \big(\mathbf{b} - A\widetilde{\mathbf{x}}(\bar{\boldsymbol{\lambda}})\big)$$

for some $\widetilde{\mathbf{x}}(\bar{\boldsymbol{\lambda}}) \in \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_K\}$. Hence, for every $\boldsymbol{\lambda} \in \mathbb{R}^m$ the following inequality holds

$$\begin{aligned} z(\boldsymbol{\lambda}) &\geq \mathbf{c}^\top \widetilde{\mathbf{x}}(\bar{\boldsymbol{\lambda}}) + \boldsymbol{\lambda}^\top \big(\mathbf{b} - A\widetilde{\mathbf{x}}(\bar{\boldsymbol{\lambda}})\big) \\ &= \mathbf{c}^\top \widetilde{\mathbf{x}}(\bar{\boldsymbol{\lambda}}) + \boldsymbol{\lambda}^\top \big(\mathbf{b} - A\widetilde{\mathbf{x}}(\bar{\boldsymbol{\lambda}})\big) + \bar{\boldsymbol{\lambda}}^\top \big(\mathbf{b} - A\widetilde{\mathbf{x}}(\bar{\boldsymbol{\lambda}})\big) - \bar{\boldsymbol{\lambda}}^\top \big(\mathbf{b} - A\widetilde{\mathbf{x}}(\bar{\boldsymbol{\lambda}})\big) \\ &= z(\bar{\boldsymbol{\lambda}}) - \epsilon + (\boldsymbol{\lambda} - \bar{\boldsymbol{\lambda}})^\top \big(\mathbf{b} - A\widetilde{\mathbf{x}}(\bar{\boldsymbol{\lambda}})\big), \end{aligned} \tag{73}$$

which indicates that $\big(\mathbf{b} - A\widetilde{\mathbf{x}}(\bar{\boldsymbol{\lambda}})\big) \in \partial_\epsilon z(\bar{\boldsymbol{\lambda}})$.

Lagrangian relaxation and corresponding solution of the (convex and nonsmooth) Lagrangian dual problem is a very common example of the general case where, in minimizing a convex function f, at any point $\mathbf{x}$ we have at hand both an approximate value of the function $\tilde{f}(\mathbf{x}) = f(\mathbf{x}) + \epsilon_f$, and an approximate subgradient $\tilde{g}(\mathbf{x}) \in \partial_{\epsilon_g} f(\mathbf{x})$, for some positive ϵ_f and ϵ_g. Convergence analysis of algorithms based on such an approximation has been extensively used both in subgradient (see [87,30,7]) and in bundle methods (see [74,88,37,143]). In particular, in [37] a taxonomy of possible kinds of inexactness in function and/or subgradient evaluation is provided, together with a classification of the methods. It is relevant, in fact, the distinction between cases where ϵ_f and ϵ_g are completely unknown and those where such errors can be *estimated* or, sometimes, even *controlled.*

8 Nonconvex NSO: a bundle view

The extension of the cutting plane idea and, consequently, of bundle methods to (local) minimization of nonconvex functions is not straightforward. In fact, in such a case it is still possible to define the convex piecewise affine function f_k, exactly as in (25), provided that vectors $\mathbf{g}_i$ are now elements of Clarke's subdifferential $\partial_C f(\mathbf{x})$. Nevertheless, two fundamental properties valid in the convex framework get lost:

- it is no longer ensured that f_k is a lower approximation of f;
- f_k does not necessarily interpolates f at points $\mathbf{x}_i$, $i \in \{1, \ldots, k\}$.

If we adopt the stability center viewpoint and rewrite f_k, see (39), as

$$f_k(\overline{\mathbf{x}}_k + \mathbf{d}) = f(\overline{\mathbf{x}}_k) + \max\left\{\mathbf{g}_i^\top \mathbf{d} - \alpha_i \,:\, i \in \{1, \ldots, k\}\right\}$$

it may happen that f_k does not even interpolate f at point $\overline{\mathbf{x}}_k$, in case some α_i takes a negative value, which is likely to occur since f is nonconvex. Note that such drawback is independent of the nonsmoothness assumption. Several authors, see [85,104,135], have handled it by embedding into a standard bundle scheme possible downward shifting of one or more of the affine pieces which give rise to the cutting plane function. This can be obtained by replacing the definition (40) of the linearization error α_i with

$$\alpha_i = \max\left\{f(\overline{\mathbf{x}}_k) - f(\mathbf{x}_i) - \mathbf{g}_i^\top(\overline{\mathbf{x}}_k - \mathbf{x}_i),\, \sigma\|\overline{\mathbf{x}}_k - \mathbf{x}_i\|^2\right\} \geq 0,$$

for some $\sigma > 0$. Such modification, although somehow arbitrary, ensures the interpolation $f_k(\overline{\mathbf{x}}_k) = f(\overline{\mathbf{x}}_k)$.

An alternative way to handle possibly negative linearization errors is based on the idea of *bundle splitting*, see [51,60]. It is based on the distinction between affine pieces that exhibit a kind of *convex* or *nonconvex* behavior relative to the stability center. The approach requires a slightly different definition of the elements of the bundle, which is now

$$B_k \triangleq \left\{\big(\mathbf{x}_i, f(\mathbf{x}_i), \mathbf{g}_i, \alpha_i, a_i\big) \,:\, \mathbf{g}_i \in \partial_C f(\mathbf{x}_i),\, a_i = \|\overline{\mathbf{x}}_k - \mathbf{x}_i\|,\, i \in \{1, \ldots, k\}\right\}.$$

Letting $I = \{1, \dots, k\}$ be the index set of B_k, we introduce the partition $I = I^+ \cup I^-$ with I_+ and I_- defined as follows

$$I_+ = \{i \in I : \alpha_i \geq 0\} \qquad \text{and} \qquad I_- = \{i \in I : \alpha_i < 0\}. \tag{74}$$

The bundles defined by the index sets I_+ and I_- are related to points that somehow exhibit, respectively, a "convex behavior" and a "concave behavior" with respect to $\overline{\mathbf{x}}_k$. We observe that I_+ is never empty as at least the element $(\overline{\mathbf{x}}_k, f(\overline{\mathbf{x}}_k), \mathbf{g}_k, 0, 0)$ belongs to the bundle.

The basic idea is to treat differently the two bundles in the construction of a piecewise affine model. The following two piecewise affine functions are thus defined

$$\Delta^+(\mathbf{d}) \triangleq \max \left\{ \mathbf{g}_i^\top \mathbf{d} - \alpha_i : i \in I_+ \right\}$$

and

$$\Delta^-(\mathbf{d}) \triangleq \min \left\{ \mathbf{g}_i^\top \mathbf{d} - \alpha_i : i \in I_- \right\}.$$

Function $\Delta^+(\mathbf{d})$ is intended as an approximation to the difference function $f(\overline{\mathbf{x}}_k + \mathbf{d}) - f(\overline{\mathbf{x}}_k)$, and interpolates it at $\mathbf{d} = \mathbf{0}$ as it is $\Delta^+(\mathbf{0}) = 0$, being $k \in I_+$. On the other hand, $\Delta^-(\mathbf{d})$ is a locally *pessimistic* approximation of the same difference function, since at $\mathbf{d} = \mathbf{0}$ it is $\Delta^-(\mathbf{0}) = \min\{-\alpha_i : i \in I_-\} > 0$. Summing up, around $\mathbf{d} = \mathbf{0}$ (i.e., around the stability center $\overline{\mathbf{x}}_k$) it is

$$\Delta^+(\mathbf{0}) < \Delta^-(\mathbf{0}). \tag{75}$$

Consequently, it appears reasonable to consider significant the difference function approximation $\Delta^+(\mathbf{d})$ as far as condition (75) is fulfilled. Thus, we come out with a kind of trust region model $\mathcal{S}_k$ defined as

$$\mathcal{S}_k = \left\{ \mathbf{d} \in \mathbb{R}^n : \Delta^+(\mathbf{d}) \leq \Delta^-(\mathbf{d}) \right\}.$$

As in all bundle methods, the building block of the double–bundle approach is the subproblem to be solved in order to find a (tentative) displacement $\mathbf{d}_k$ from the stability center $\overline{\mathbf{x}}_k$. Under the *trust region* constraint $\mathbf{d} \in S_k$, the choice in [51] is to solve

$$\min \left\{ \Delta^+(\mathbf{d}) : \mathbf{d} \in \mathcal{S}_k \right\}$$

which, by introducing also in this case the classic proximity term, can be put in the form

$$\min \left\{ v + \gamma_k \frac{1}{2} \|\mathbf{d}\|^2 : v \geq \mathbf{g}_i^\top \mathbf{d} - \alpha_i \ \forall i \in I_+, \ v \leq \mathbf{g}_i^\top \mathbf{d} - \alpha_i \ \forall i \in I_-, \ \mathbf{d} \in \mathbb{R}^n, \ v \in \mathbb{R} \right\}.$$

We do not enter into the (rather technical) details on how subproblem above can be cast into a working bundle scheme. Implementations of the algorithm described in [51] have been fruitfully used in many nonconvex optimization applications.

9 Bibliography, complements, and reading suggestions

We discuss, without the ambition of being exhaustive, a number of bibliographic references, some already cited throughout the paper, on various topics touched in this survey. We also open some windows on certain research sub-areas, that it has been impossible to treat for the sake of brevity. From time to time we draw the reader's attention to some contributions we feel of particular interest.

Mathematical background Convex analysis is the well-grounded theoretical basis of numerical NSO. Cornerstone references are the (unpublished but well known) 1951 Lecture notes by W. Fenchel at Princeton [46], the Moreau paper [114], Rockafellar's "Convex Analysis" [133], the book by Hiriart-Urruty and Lemaréchal [76], which covers both theoretical and algorithmic aspects, and the books by Bertsekas [17,18] and Mordukhovich [113].

Historical Contributions Some books provide a complete view of the well advanced state of the art of numerical NSO, mainly in former Soviet Union, during the 70s of last century. Most of the successive developments have their roots there. We cite Demyanov and Malozemov book on minmax problems [32], the book by Pshenichnyi and Danilin [128] which covers both smooth and nonsmooth optimization, Shor's book [137] on subgradient method and its variants, Polyak's complete presentation [126], both in deterministic and in stochastic setting, and Nemirovski and Yudin book [116], where the complexity and efficiency issues are treated in depth. A real milestone in the development of numerical NSO was the workshop held in spring 1977 at IIASA, in Laxenburg, near to Wien, were for the first time scientists from both sides of what, at that time, was named the *iron curtain* had the opportunity of a long and fruitful debate. In particular, the meeting represented the starting point of a rapid development of the NSO area in western countries. The Proceedings of the workshop [93] contain a number of fine contributions. To our knowledge, the term *bundle method* was coined by Lemaréchal in that occasion [94] and it is very interesting to note that similar ideas, independently developed, were present in other contributions, see [129].

Comprehensive books and surveys Among books which give a complete overview of both theory and practice of NSO, apart the already mentioned [76], we recall here [81,104,138,10]. We suggest, in particular, [9] for its admirable clarity. Excellent surveys are [98,105,50]. We also suggest the reading of [15] for the original approach to convex optimization.

Subgradient methods The methods discussed in Section 4 were, to our knowledge, introduced in a note by N.Z. Shor [136]. From the very beginning several other scientists gave their contributions [44,43,125]. As far as the classic approach is concerned, reference books, whose reading is strongly suggested, are [137,126]. In more recent years, the interest in subgradient–type methods was renewed, thanks to the Mirror Descent Algorithm introduced by Nemirowski and Yudin (see also [14]), and to some papers by Nesterov [118,119,120] (see also the variant [49]). Very recent developments are in [41]. Apart from subgradient methods, we recall that also the concept of ϵ-subdifferential has been at the basis of some early algorithms (see, e.g., [16,122]).

Cutting plane and bundle methods The cutting plane method stems, as already mentioned, from the seminal papers by Kelley [79] and Cheney and Goldstein [28], where the reader finds much more than just the description of the algorithm. A similar approach was independently devised by Levitin and Polyak [102]. As for bundle method, fundamental references are the papers by Lemaréchal [92] and by Wolfe [145]. The approach known as Method of Linearisations also embedding the proximity concept was independently proposed at about the same time by Pshenichnyi, see [127] and [128, Chapter 3, §5]. Since the beginning of the 80s the interest towards bundle methods has flourished within the mathematical programming community, and a large number of papers has appeared in outstanding journals. It is impossible to provide a complete list. We just mention the early papers [96,106,53]. As examples of the use of the three stabilizing strategies described in Section 5.1 we recall Kiwiel's paper [83] for a deep view on the proximal point BM; trust region BM is analysed in Schramm and Zowe [135], with possible application also to nonconvex functions and, finally, the level bundle variant of BM, somehow already evoked in [129], is presented in [99,21]. Apart from the three main classes of BM described in Section 5.1, we wish to mention some other proposals.

– Methods based on possible decomposition of function domain into a subspace where the function is smooth, while nonsmoothness is confined into the orthogonal subspace, see [110]. Such approach is usually referred to as *VU decomposition.* A fine historical note about it (and much more) is in [111].
– Methods which adopt different stabilization strategies. We cite, in particular, the Generalized BM [48], the use of Bregman distance [86], and the doubly stabilized BM [38].
– Methods where the condition that the model function f_k is a lower approximation of f is removed, by replacing the α_is in (45) with adjustable (non negative) parameters,see [55,57,6].
– Methods where bundle update takes place every time a new stability center $\overline{\mathbf{x}}_{k+1}$ is found, through *simultaneous* moves of all points $\mathbf{x}_i$s towards $\overline{\mathbf{x}}_{k+1}$, see [35].
– Methods based on piecewise quadratic approximations of the objective function, see [56,4].
– Spectral BM for dealing with eigenvalue optimization and semidefinite relaxations of combinatorial problems, see [73].
– The Volume Algorithm which is midway between subgradient and simplified bundle methods, thus appearing suitable for large scale applications, see [12, 11].

Line searches Line searches tailored on nonsmooth (not necessarily convex) functions constitute an important chapter of NSO. A line search algorithm embedded into any BM method must accommodate for possible null-step. We have already mentioned in Section 5.1 the Armijo's rule [3]. In the literature, specific line searches have been designed, and we recall here the method due to Wolfe [145], the Lemarechal's survey [95], and the Mifflin's paper [107], where a method with superlinear convergence rate for locally Lipschitz functions is discussed.

Solving the quadratic subproblem In bundle methods a quadratic subproblem is to be solved at each iteration and, consequently, the overall performance is strongly affected by the quality of the correspondent quadratic solver. In particular, in proximal BM either problem (45) or (48) are to be tackled to provide the direction $\mathbf{d}_k$. The special structure of the latter has suggested the design of ad hoc algorithms. Efficient methods are described in [82,112,47]. We also mention the historical paper [146], where the quadratic problem (48) is treated for the case when α_is are all equal to zero, in the framework of classic Wolfe's conjugate subgradient method [145].

Variable metric methods As for the extension to NSO of Quasi-Newton formulae, we have already cited [97] and [144], the latter being also able to deal with nonconvex objective functions. A different way to embed QN ideas in the bundle framework is presented in [103]. References for QN methods based on Moreau-Yosida regularization and bundle-QN methods are [130,20,100,54,108,109,132,27]. An interesting area where QN ideas have been fruitfully employed, mainly to deal with large scale NSO, is the *Limited memory* BM [70,64] where ideas coming from [103,144] have been employed in the framework of the limited memory QN for smooth problems [25]. The method has been extended to very large scale problems, also nonconvex, by adopting a sparse (diagonal, in fact) form for the QN matrix [78]. We wish to mention, finally, that celebrated Shor's subgradient with space dilatation algorithm can be viewed as a QN method with symmetric rank-one update formula, see [141,23].

Minmax problems A large part of NSO problems arising in practical applications are of the *minmax* type, mainly in consideration that the *worst case analysis*, which naturally leads to minmax (or maxmin) model, is an increasingly popular paradigm in decision making. We recall here the already cited fundamental book [32] and the papers [39,40] where minmax problems are dealt with by transformation into smooth problems. Some basic references are [71,124,115]. Minmaxmin optimization is revisited in [34] (see also [61]). Inexact calculation of the max function has been considered in both cases of finite and semi-infinite convex minmax in [58] and [52], respectively; an application to a minmax problem in a Lagrangian relaxation setting is presented in [59].

Nonconvex NSO and DC programming There exist numerous bundle type algorithms applicable to nonconvex functions. We recall here [122,135,131,85,121, 72,1]. Papers [8,90,45] are examples of derivative free NSO methods capable to cope with nonconvexity. In recent years the class of DC (Difference of Convex) functions [75,140,142] has received considerable attention. A DC function $f(\mathbf{x})$ is expressed in the form:

$$f(\mathbf{x}) = f^{(1)}(\mathbf{x}) - f^{(2)}(\mathbf{x}),$$

where both $f^{(1)}$ and $f^{(2)}$ are convex. The well established algorithm DCA [2] works as follows. Letting $\mathbf{x}_k$ be the current iterate, point $\mathbf{x}_{k+1}$ is obtained as

$$\mathbf{x}_{k+1} = \arg\min \left\{ f^{(1)}(\mathbf{x}) - f^{(2)}(\mathbf{x}_k) - \mathbf{g}^{(2)}(\mathbf{x}_k)^\top (\mathbf{x} - \mathbf{x}_k) \right\},$$

where $\mathbf{g}^{(2)}(\mathbf{x}_k) \in \partial f^{(2)}(\mathbf{x}_k)$. In other words, the linearization of function $f^{(2)}$ gives rise, at each iteration, to a convex program to be solved in order to obtain the next iterate. The bundle philosophy has been extensively used in handling DC optimization, introducing the cutting plane model for $f^{(1)}$ and/or $f^{(2)}$. Some recent references are [5,36,62,63,77].

Compliance with Ethical Standards

Conflict of Interest:

The authors declare that they have no conflict of interest.

References

1. Akbari, Z., R. Yousefpour and M. Reza Peyghami, "A New Nonsmooth Trust Region Algorithm for Locally Lipschitz Unconstrained Optimization Problems". *Journal of Optimization Theory and Applications* 164, 2014, pp. 733-754.
2. An, L.T.H. and P.D. Tao, "The DC (difference of convex functions) programming and DCA revisited with DC models of real world nonconvex optimization problems". *Journal of Global Optimization* 133, 2005, pp.23–46.
3. Armijo, L., " Minimization of functions having Lipschitz continuous first partial derivatives". *Pacific Journal of Mathematics* 16, 1966, pp. 1–3.
4. Astorino, A., A. Frangioni, M. Gaudioso and E. Gorgone, "Piecewise quadratic approximations in convex numerical optimization". *SIAM Journal on Optimization* 21, 2011, pp. 1418–1438.
5. Astorino, A. and G. Miglionico, "Optimizing sensor cover energy via DC programming". *Optimization Letter*, 10, 2016, pp. 355–368.
6. Astorino, A., M. Gaudioso and E. Gorgone, "A method for convex minimization based on translated first-order approximations". *Numerical Algorithms* 76, 2017, pp. 745760.
7. Astorino, A., A. Fuduli and M. Gaudioso, "A Lagrangian relaxation approach for binary Multiple Instance Classification". *IEEE Transactions on Neural Networks and Learning Systems* 30, 2019, pp. 2662-2671.
8. Bagirov, A.M., B. Karasözen and M. Sezer, "Discrete gradient method: derivative-free method for nonsmooth optimization". *Journal of Optimization Theory and Applications* 137, 2008, pp. 317–334.
9. Bagirov, A.M., N. Karmitsa and M.M. Mäkelä, "Introduction to Nonsmooth Optimization: Theory, Practice and Software". Springer, 2014.
10. Bagirov, A.M., M. Gaudioso, N. Karmitsa, M.M. Mäkelä and S. Taheri eds., "Numerical nonsmooth optimization–State of the art algorithms" Springer, to appear.
11. Bahiense, L., N. Maculan and C. Sagastizábal, "The Volume Algorithm Revisited: Relation with Bundle Methods". *Mathematical Programming* 94, 2002, pp. 41–69.
12. Barahona, F. and R. Anbil,"The volume algorithm: producing primal solutions with a subgradient method". *Mathematical Programming* 87, 2000, pp. 385–399.
13. Barzilai, J. and J.M. Borwein, "Two-point step size gradient methods". *IMA Journal of Numerical Analysis* 8, 1988, pp. 141–148.
14. Beck, A. and M. Teboulle, "Mirror descent and nonlinear projected subgradient methods for convex optimization". *Operations Research Letters* 31, 2003, pp. 167–175.
15. Ben-Tal, A., A. Nemirovski, Lectures on modern optimization, *MPS/SIAM Series on Optimization*, SIAM, Philadelphia, 2001.
16. Bertsekas, D.P. and S. K. Mitter, "A descent numerical method for optimization problems with nondifferentiable cost functionals". *SIAM Journal on Control* 11, 1973, pp. 637–652.
17. Bertsekas, D. P., "Nonlinear Programming". Athena Scientific, Belmont, Mass., 1995.
18. Bertsekas, D. P., "Convex Optimization Theory". Athena Scientific, 2009, Belmont. USA.
19. Bertsimas, D. and S. Vempala, "Solving convex programs by random walks". *Journal of the ACM* 51, 2004, pp. 540556.
20. Bonnans, J., J. Gilbert, C. Lemaréchal and C. Sagastizábal, C., "A family of variable metric proximal methods". *Mathematical Programming* 68, 1995, pp. 15–47.
21. Brännlund, U., K. C. Kiwiel and P. O. Lindberg, "A descent proximal level bundle method for convex nondifferentiable optimization". *Operations Research Letters* 17, 1995, pp.121–126.
22. Burke, J.V., A.S. Lewis and M.L. Overton. "A robust gradient sampling algorithm for nonsmooth, nonconvex optimization". *SIAM Journal on Optimization* 15, 2005, pp. 751–779.

23. Burke, J.V., A.S. Lewis and M.L. Overton. "The speed of Shor's R-algorithm". *IMA Journal of Numerical Analysis* 28, 2008, pp. 711–720.
24. Burke, J.V., F.E. Curtis, A.S. Lewis, M.L. Overton and L.E.A. Simões, "Gradient Sampling Methods for Nonsmooth Optimization". In *Numerical nonsmooth optimization-State of the art algorithms*, Bagirov, A.M, M. Gaudioso, N. Karmitsa, M. Mäkelä and S. Taheri, eds., Springer, to appear.
25. Byrd, R.H., J. Nocedal and R. B. Schnabel, "Representations of quasi-Newton matrices and their use in limited memory methods". *Mathematical Programming* 63, 1994, pp. 129–156.
26. Chebyshëv, P. L., "Sur les questions de minima qui se rattachent a la représentation approximative des fonctions", 1859, in *Oeuvres de P. L. Tchebychef*, Volume 1, pp. 273-378, Chelsea, New York, 1961.
27. Chen, X., and M. Fukushima. "Proximal quasi–Newton methods for nondifferentiable convex optimization". *Mathematical Programming* 85, 1999, pp. 313–334.
28. Cheney, E. W., and A. A. Goldstein, "Newton's method for convex programming and Tchebycheff approximation". *Numerische Mathematik* 1, 1959, pp. 253–268.
29. Clarke, F.H., "Optimization and nonsmooth analysis", 1983, John Wiley and Sons, pp. 357–386.
30. D'Antonio, G. and A. Frangioni, "Convergence analysis of deflected conditional approximate subgradient methods". *SIAM Journal on Optimization* 20, 2009, pp. 357–386.
31. de Ghellinck, G. and J.-P. Vial, "A polynomial Newton method for linear programming". *Algorithmica* 1, 1986, pp. 425-453.
32. Demyanov, V. F., and V.N. Malozemov, "Introduction to Minimax". Wiley, New York, 1974.
33. Demyanov, V. F and A. M. Rubinov, "Constructive nonsmooth analysis". Verlag Peter Lang, Berlin, 1995.
34. Demyanov, A.V., V.F. Demyanov and V.N. Malozemov, "Minmaxmin problems revisited". *Optimization Methods and Software* 17, 2002, pp. 783–804.
35. Demyanov, A.V., A. Fuduli and G. Miglionico, "A bundle modification strategy for convex minimization". *European Journal of Operational Research* 180, 2007, pp. 38-47.
36. de Oliveira, W., "Proximal bundle methods for nonsmooth DC programming". *Journal of Global Optimization* 75, 2019, pp. 523–563.
37. de Oliveira, W., C. Sagastizábal and C. Lemaréchal, "Convex proximal bundle methods in depth: a unified analysis for inexact oracles". *Mathematical Programming*, 148, 2014, pp. 241–277.
38. de Oliveira, W. and M. Solodov, " A doubly stabilized bundle method for nonsmooth convex optimization". *Mathematical Programming* 156, 2016, pp. 125–159.
39. Di Pillo, G., L. Grippo, and S. Lucidi. "A smooth method for the finite minimax problem". *Mathematical Programming* 60, 1993, pp. 187–214.
40. Di Pillo, G., L. Grippo, and S. Lucidi, "Smooth transformation of the generalized minimax problem". *Journal of Optimization Theory and Applications* 95, 1997, pp. 1–24.
41. Dvurechensky, P.E., A.V. Gasnikov, E.A. Nurminski and F. S. Stonyakin, "Advances in Low-Memory Subgradient Optimization". In *Numerical nonsmooth optimization–State of the art algorithms*, Bagirov, A., M. Gaudioso, N. Karmitsa, M. Mäkelä and S. Taheri, eds., Springer, to appear.
42. Elzinga, J., and T.G. Moore, "A central cutting plane algorithm for the convex programming problem". *Mathematical Programming* 8, 1975, pp. 134–145.
43. Eremin, I.I., "The method of penalties in convex programming". *Dokladi Academii Nauk USSR* 173, 1967, pp. 748–751.
44. Ermoliev, Yu. M., "Methods of solution of nonlinear extremal problems". *Cybernetics* 2, 1966, pp. 1–16.
45. Fasano, G., G. Liuzzi, S. Lucidi, and F. Rinaldi, "A Linesearch-Based Derivative-Free Approach for Nonsmooth Constrained Optimization". *SIAM Journal on Optimization* 24, 2014, pp. 959–992.
46. Fenchel, W., "Convex cones, sets and functions". *Lectures at Princeton University*, Princeton, 1951.
47. Frangioni, A. "Solving semidefinite quadratic problems within nonsmooth optimization algorithms". *Computers and Operations Research* 23, 1996, pp.1099–1118.
48. Frangioni, A., "Generalized Bundle Methods". *SIAM Journal on Optimization* 13, 2002, pp.117–156.

49. Frangioni, A., B. Gendron and E. Gorgone,“Dynamic smoothness parameter for fast gradient methods”. *Optimization Letters* 12, 2018, pp. 43–53.
50. Frangioni, A., “Standard bundle methods: untrusted models and duality ”. In *Numerical nonsmooth optimization–State of the art algorithms*, Bagirov, A.M., M. Gaudioso, N. Karmitsa, M. Mäkelä and S. Taheri, eds., Springer, to appear.
51. Fuduli, A., M. Gaudioso and G. Giallombardo, “Minimizing nonconvex nonsmooth functions via cutting planes and proximity control”. *SIAM Journal on Optimization* 14, 2004, pp. 743-756.
52. Fuduli A., M. Gaudioso, G. Giallombardo and G. Miglionico, “A partially inexact bundle method for convex semi-infinite minmax problems”. *Communications in Nonlinear Science and Numerical Simulation* 21, 2014, pp.172–180.
53. Fukushima, M. “A descent algorithm for nonsmooth convex optimization”. *Mathematical Programming* 30, 1984, pp. 163–175.
54. Fukushima, M., and L. Qi. “A globally and superlinearly convergent algorithm for nonsmooth convex minimization”. *SIAM Journal on Optimization* 6, 1996, pp. 1106–1120.
55. Gaudioso, M., and M. F. Monaco, “A bundle type approach to the unconstrained minimization of convex nonsmooth functions”. *Mathematical Programming* 23, 1982, pp. 216–223.
56. Gaudioso, M., and M. F. Monaco. “Quadratic approximations in convex nondifferentiable optimization”. *SIAM Journal on Control and Optimization* 29, 1991, pp. 1–10.
57. Gaudioso, M. and M. F. Monaco, “Variants to the cutting plane approach for convex nondifferentiable optimization. *Optimization* 25, 1992, pp. 65–75.
58. Gaudioso, M., G. Giallombardo and G. Miglionico, “An incremental method for solving convex finite min-max problems”. *Mathematics of Operations Research* 31, 2006, pp. 173–187.
59. Gaudioso, M., G. Giallombardo and G. Miglionico, “On solving the Lagrangian dual of integer programs via an incremental approach”. *Computational Optimization and Applications* 44, 2009, pp. 117-138.
60. Gaudioso, M. and E. Gorgone, “Gradient set splitting in nonconvex nonsmooth numerical optimization”. *Optimization methods and software* 25, 2010, pp.59-74.
61. Gaudioso, M., G. Giallombardo and G. Miglionico, “Minimizing piecewise concave functions over polyhedra”. *Mathematics of Operations Research* 43, 2018, pp. 580–597.
62. Gaudioso, M., G. Giallombardo, G. Miglionico and A.M. Bagirov, “Minimizing nonsmooth DC functions via successive DC piecewise-affine approximations”. *Journal of Global Optimization* 71, 2018, pp.37–55.
63. Gaudioso, M., G. Giallombardo, G. Miglionico and E. Vocaturo, “Classification in the multiple instance learning framework via spherical separation”. *Soft Computing*, 2019, doi:10.1007/s00500-019-04255-1.
64. Gaudioso, M., G. Giallombardo and M. Mukhametzhanov, “Numerical infinitesimals in a variable metric method for convex nonsmooth optimization”. *Applied Mathematics and Computation* 318, 2018, pp. 312-320.
65. Gaudioso, M., “A view of Lagrangian relaxation and its applications”. In *Numerical nonsmooth optimization–State of the art algorithms*, Bagirov, A.M, M. Gaudioso, N. Karmitsa, M. Mäkelä and S. Taheri, eds., Springer, to appear.
66. Goffin, J.-L., “On convergence rates of subgradients optimization methods”. *Mathematical Programming* 13, 1977, pp. 329–347.
67. Goffin, J.-L., A. Haurie, and J.-P. Vial, “Decomposition and nondifferentiable optimization with the projective algorithm”. *Management Science* 38, 1992, pp. 284–302.
68. Goffin, J.-L., J. Gondzio, R. Sarkissian and J.-P. Vial.“Solving nonlinear multicommodity flow problems by the analytic center cutting plane method”. *Mathematical Programming* 76B, 1997, pp.131–154.
69. Grippo, L., F. Lampariello and S. Lucidi, “A class of nonmonotone stabilization methods in unconstrained optimization”. *Numerische Mathematik* 59, 1991, pp. 779–805.
70. Haarala, N., K. Miettinen and M.M. Mäkelä, “Globally convergent limited memory bundle method for large-scale nonsmooth optimization”. *Mathematical Programming* 109, 2007, pp. 181–205.
71. Hald, J., and K. Madsen, “Combined LP and Quasi–Newton methods for minimax optimization”. *Mathematical Programming* 20, 1981, pp. 49–62.
72. Hare, W. and C. Sagastizábal, “A redistributed proximal bundle method for nonconvex optimization”. *SIAM Journal on Optimization* 20, 2010, pp. 2242–2473.

73. Helmberg C. and F. Rendl, “A Spectral Bundle Method for Semidefinite Programming”. *SIAM Journal on Optimization* 10, 2000, pp. 673–696.
74. Hintermüller, M., “A proximal bundle method based on approximate subgradients”. *Computational Optimization and Applications*, 20, 2001, pp. 245266.
75. Hiriart-Urruty, J.-B., “Generalized differentiability/ duality and optimization for problems dealing with differences of convex functions”. *Lecture Notes in Economic and Mathematical Systems* 256, Springer Verlag, 1986, pp. 37–70.
76. Hiriart-Urruty, J. B., and C. Lemaréchal, “Convex Analysis and Minimization Algorithms”. Vol. I and II, Springer-Verlag, Berlin Heidelberg, 1993.
77. Joki K., A. M. Bagirov, N. Karmitsa, M. M. Mäkelä and S. Taheri, “Double Bundle Method for finding Clarke Stationary Points in Nonsmooth DC Programming”. *SIAM Journal on Optimization* 28, 2018, pp. 1892–1919.
78. Karmitsa, N., “Diagonal bundle method for nonsmooth sparse optimization”. *Journal of Optimization Theory and Applications* 166, 2015, pp. 889–905.
79. Kelley, J. E., “The cutting plane method for solving convex programs”. *Journal of SIAM* 8, 1960, pp. 703-712.
80. Kiwiel, K. C., “An aggregate subgradient method for nonsmooth convex minimization”. *Mathematical Programming* 27, 1983, pp. 320–341.
81. Kiwiel, K. C., “Methods of descent for nondifferentiable optimization”. *Lecture Notes in Mathematics* 1133, 1985, Springer-Verlag, Berlin Heidelberg.
82. Kiwiel, K.C.,“ A method for solving certain quadratic programming problems arising in nonsmooth optimization”. *IMA Journal of Numerical Analysis* 6, pp. 137–152 (1986).
83. Kiwiel, K. C., “Proximity control in bundle methods for convex nondifferentiable minimization”. *Mathematical Programming* 46, 1990, pp. 105–122.
84. Kiwiel, K. C., “A Cholesky dual method for proximal piecewise linear programming”. *Numerische Mathematik* 68, 1994, pp. 325–340.
85. Kiwiel, K.C., “Restricted step and Levenberg-Marquardt techniques in proximal bundle methods for nonconvex nondifferentiable optimization”. *SIAM Journal on Optimization* 6, 1996, pp. 227–249.
86. Kiwiel, K. C., “A bundle Bregman proximal method for convex nondifferentiable minimization”. *Mathematical Programming* 85, 1999, pp. 241–258.
87. Kiwiel, K.C., “Convergence of approximate and incremental subgradient methods for convex optimization”. *SIAM Journal on Optimization* 14, 2004, pp. 807–840.
88. Kiwiel, K.C.,“A proximal bundle method with approximate subgradient linearizations”. *SIAM Journal on Optimization* 16, 2006, pp.1007–1023.
89. Kiwiel, K.C., “Convergence of the Gradient Sampling Algorithm for Nonsmooth Nonconvex Optimization”. *SIAM Journal on Optimization* 18, 2007, pp.379–388.
90. Kiwiel, K.C.,“A Nonderivative Version of the Gradient Sampling Algorithm for Nonsmooth Nonconvex Optimization”. *SIAM Journal on Optimization* 20, 2010, pp. 1983-1994.
91. Lemaréchal, C., ”An algorithm for minimizing convex functions”, in J.L. Rosenfeld, ed., *Proceedings IFIP ’74 Congress*, 1974, pp. 20–25, North-Holland, Amsterdam.
92. Lemaréchal, C., ”An extension of Davidon methods to nondifferentiable problems”. *Mathematical Programming Study* 3, 1975, pp. 95–109.
93. Lemaréchal, C. and R. Mifflin eds., ”Nonsmooth Optimization”, 1978, Pergamon Press, Oxford.
94. Lemaréchal, C., “Nonsmooth optimization and descent methods”. Report RR-78-4, 1978, IIASA, Laxenburg, Austria.
95. Lemaréchal, C., “A view of line-searches”, in Auslender A., W., Oettli and J., Stoer, eds, *Optimization and Optimal Control*. Lecture Notes in Control and Information Sciences, 30, 1981, Springer, Berlin.
96. Lemaréchal, C., J.-J. Strodiot and A. Bihain, “On a bundle algorithm for nonsmooth optimization”, in Mangasarian, O.L., R.R. Meyer and S.M. Robinson, eds., *Nonlinear Programming 4*, Academic Press, New York, 1981, pp. 245-282.
97. Lemaréchal, C., “Numerical experiments in nonsmooth optimization”, in E. A. Nurminski, ed., *Progress in Nondifferentiable Optimization* CP82S8, IIASA, Laxenburg, Austria, 1982, pp. 6184.
98. Lemaréchal, C. “Nondifferentiable Optimization”, in Nemhauser G.L. et al. eds., Handbooks in OR & MS, Vol 1, North–Holland, 1989.
99. Lemaréchal, C, A. Nemirovskii and Y. Nesterov, “New variants of bundle methods”. *Mathematical Programming* 69, 1995, pp. 111–147.

100. Lemaréchal, C., and C. Sagastizábal, “Variable metrics bundle methods: From conceptual to implementable forms”. *Mathematical Programming* 76, 1997, pp. 393–410.
101. Levin, A. Yu., “On an algorithm for minimization of convex functions”. *Soviet Mathematical Doklady* 6, 1965, pp. 286–290.
102. Levitin, E.C. and B.T. Polyak, “Constrained minimization methods”. *Journal of Computational Mathematics and Mathematical Physics* 6, 1966, pp.787-823 (in Russian).
103. Luksǎn, L.and J. Vlček,“A bundle-Newton method for nonsmooth unconstrained minimization”. *Mathematical Programming* 83, 1998, pp. 373–391.
104. Mäkelä, M.M. and P. Neittaanmäki, “Nonsmooth optimization”, World Scientific, 1992.
105. Mäkelä, M. M., “Survey of Bundle Methods for Nonsmooth Optimization”. *Optimization Methods and Software* 17, 2002, pp.1-29.
106. Mifflin, R., “A modification and an extension of Lemaréchal’s algorithm for nonsmooth minimization”. *Mathematical Programming Study* 17, 1982, pp. 77-90.
107. Mifflin, R., “Stationarity and superlinear convergence of an algorithm for univariate locally Lipschitz constrained minimization”. *Mathematical Programming* 28, 1984, pp. 50–71.
108. Mifflin, R., “A quasi–second order proximal bundle algorithm”. *Mathematical Programming* 73, 1996, pp. 51–72.
109. Mifflin, R., D., Sun and L., Qi, “Quasi-Newton bundle-type methods for nondifferentiable convex optimizations”. *SIAM Journal on Optimization* 8, 1998, pp. 583–603.
110. Mifflin, R. and C. Sagastizábal. “A VU-algorithm for convex minimization”. *Mathematical Programming* 104, 2005, pp. 583–608.
111. Mifflin, R. and C. Sagastizábal. “A Science Fiction Story in Nonsmooth Optimization Originating at IIASA”. *Documenta Mathematica* Extra Volume: Optimization Stories, 2012, pp. 291–300.
112. Monaco, M.F., “An algorithm for the minimization of a convex quadratic function over a simplex”. *Technical Report, Dipartimento di Sistemi, Universitá della Calabria* 56, 1987.
113. Mordukhovich, B. S., “Variational Analysis and Generalized Differentiation”, 2006, Springer, Berlin.
114. Moreau, J.-J., “Proximité et dualité dans un espace hilbertien”. *Bulletin de la Société Mathématique de France* 93, 1965, pp. 272–299.
115. Nedić, A. and D.P. Bertsekas, “Incremental subgradient methods for nondifferentiable optimization”. *SIAM Journal on Optimization* 12, 2001, pp. 109–138.
116. Nemirovski, A. and D. Yudin, “Problem complexity and method efficiency in Optimization”, 1983, Wiley, New York.
117. Nesterov, Yu.,“Complexity estimates of some cutting plane methods based on the analytic barrier”. *Mathematical Programmming* 69, 1995, pp. 149–176.
118. Nesterov, Yu.,“Smooth minimization of non-smooth functions”. *Mathematical Programmming* 103, 2005, pp. 127–152.
119. Nesterov, Yu.,“Primal-dual subgradient methods for convex problems”. *Mathematical Programmming* 120, 2009, pp. 221–259.
120. Nesterov, Yu.,“Universal gradient methods for convex optimization problems”. *Mathematical Programmming* 152, 2009, pp. 381–404.
121. Noll, D. and P. Apkarian, “Spectral bundle methods for non-convex maximum eigenvalue functions: first-order methods”. *Mathematical programming* 104, 2005, pp. 701–727.
122. Nurminski, E.A., “Subgradient method for minimizing weakly convex functions and ϵ-subgradient methods of convex optimization”, in Nurminski, E. A., ed., *Progress in Nondifferentiable Optimization* CP82S8, IIASA, Laxenburg, Austria, 1982, pp. 97123.
123. Ouorou, A.,“A proximal cutting plane method using Chebychev center for nonsmooth convex optimization”. *Mathematical Programmming* 119, 2009, pp. 239–271.
124. Polak, E., D. Q. Mayne, and J. E. Higgins, “Superlinearly convergent algorithm for min–max problems”. *Journal of Optimization Theory and Applications* 69, 1991, pp. 407–439.
125. Polyak, B.T., “Subgradient methods: A survey of Soviet research”, in C. Lemaréchal and R. Mifflin, eds., *Nonsmooth Optimization*, Pergamon Press, Oxford, UK, 1978, pp. 5–29.
126. Polyak, B. T., “Introduction to Optimization.” Optimization Software Inc., New York, 1987.
127. Pshenichnyi, B.N., “An algorithm for general problems of mathematical programming”. *Kybernetika* 5, 1970, pp. 120–125 (in Russian).
128. Pshenichnyi, B.N. and Yu. M. Danilin, “Numerical methods for extremum problems”, Nauka, Moscow, 1975.

129. Pshenichnyi, B.N., “Nonsmooth optimization and nonlinear programming”, in C. Lemaréchal and R. Mifflin, eds., *Nonsmooth Optimization*, Pergamon Press, Oxford, UK, 1978, pp. 71–78.
130. Qi, L. and J. Sun, “A nonsmooth version of Newtons method”. *Mathematical Programming* 58, 1993, pp.353–368.
131. Qi, L. and J., Sun, “A trust region algorithm for minimization of locally Lipschitzian functions”. *Mathematical Programming* 66, 1994, pp. 25–43.
132. Rauf, A.I. and M. Fukushima, “Globally convergent BFGS method for nonsmooth convex optimization”. *Journal of Optimization Theory and Applications* 104, 1998, pp. 539–558.
133. Rockafellar, R. T., “Convex Analysis”, Princeton University Press, Princeton, 1970.
134. Rockafellar, R. T., “Monotone operators and the proximal point algorithm”. *SIAM Journal on Control and Optimization* 14, 1976, pp. 877–898.
135. Schramm, H., and J. Zowe, “A version of the bundle idea for minimizing a nonsmooth function: conceptual idea, convergence analysis, numerical results”. *SIAM Journal on Optimization* 2, 1992, pp. 121–152.
136. Shor, N.Z., “Application of the gradient method for the solution of network transportation problems”. *Notes, Scientific Seminar on Theory and Application of Cybernetics and Operations Research*, Academy of Science, Kiev, 1962 (in Russian).
137. Shor, N.Z., “Minimization methods for nondifferentiable functions”, Springer-Verlag, Berlin, 1985.
138. Shor, N. Z. “Nondifferentiable Optimization and Polynomial Problems”, Kluwer Academic Publishers, Boston, 1998.
139. Sonnevend, G. “An analytic center for polyhedrons and new classes of global algorithms for linear (smooth, convex) programming”, in A. Prekopa, ed., Lecture notes in Control and Information Sciences 84, 1985, pp. 866–876, Springer, New York.
140. Strekalovsky, A.S., “Global optimality conditions for nonconvex optimization”. *Journal of Global Optimization* 12, 1998, pp. 415–434.
141. Todd, M. J., “The symmetric rank-one quasi-Newton algorithm is a space-dilation subgradient algorithm”. *Operations Research Letters* 5, 1986, 217–219.
142. Tuy, H., “Convex Analysis and Global Optimization”, Springer, Berlin, 2016.
143. van Ackooij, W. and C. Sagastizábal, “Constrained Bundle Methods for Upper Inexact Oracles with Application to Joint Chance Constrained Energy Problems”. *SIAM Journal on Optimization* 24, 2014, pp.733–765.
144. Vlček, J. and L. Lukšan, “Globally convergent variable metric method for nonconvex nondifferentiable unconstrained minimization”. *Journal of Optimization Theory and Applications* 111, 2001, pp. 407–430.
145. Wolfe, P., “A method of conjugate subgradients for minimizing nondifferentiable functions”. *Mathematical programming Study* 3, 1975, pp. 143–173.
146. Wolfe, P., “Finding the nearest point in a polytope”. *Mathematical programming* 11, 1976, pp. 128–149.