Università di Pisa

Dipartimento di Informatica

Technical Report

Incremental bundle methods using upper models

Wim van Ackooij
EDF R&D OSIRIS
7 Boulevard Gaspard Monge, F-91120 Palaiseau Cedex, France, wim.van-ackooij@edf.fr

Antonio Frangioni
Dipartimento di Informatica, Università di Pisa
Largo B.Pontecorvo 3, 56127 Pisa, Italia, frangio@di.unipi.it

August 16, 2016

ADDRESS: Largo B. Pontecorvo 3, 56127 Pisa, Italy. TEL: +39 050 2212700 FAX: +39 050 2212726

Incremental bundle methods using upper models

Wim van Ackooij
EDF R&D OSIRIS
7 Boulevard Gaspard Monge, F-91120 Palaiseau Cedex, France, wim.van-ackooij@edf.fr

Antonio Frangioni
Dipartimento di Informatica, Università di Pisa
Largo B.Pontecorvo 3, 56127 Pisa, Italia, frangio@di.unipi.it

August 16, 2016

Abstract

We propose a family of proximal bundle methods for minimizing sum-structured convex nondifferentiable functions which require two slightly uncommon assumptions, that are satisfied in many relevant applications: Lipschitz continuity of the functions and oracles which also produce *upper estimates* on the function values. In exchange, the methods: i) use *upper models* of the functions that allow to estimate function values at points where the oracle has not been called; ii) provide the oracles with more information about when the function computation can be interrupted, possibly diminishing their cost; iii) allow to skip oracle calls entirely for some of the component functions, not only at "null steps" but also at "serious steps"; iv) provide explicit and reliable a-posteriori estimates of the quality of the obtained solutions; v) work with all possible combinations of different assumptions on the oracles. We also discuss introduction of constraints (or, more generally, of easy components) and use of (partly) aggregated models.

Keywords: *Nonsmooth optimization, bundle methods, incremental approach*

1 Introduction

We are concerned with the following minimization problem

$$\min \left\{ f(x) = \textstyle\sum_{k \in \mathcal{K}} f^k(x) \ : \ x \in X \right\} \ , \tag{1}$$

where $\mathcal{K}$ is a finite index set, $X \subseteq \mathbb{R}^n$ is closed, convex and "easy" in a sense specified later, and each *component* $f^k : \mathbb{R}^n \to \mathbb{R}$ of f is convex but possibly nondifferentiable. Customarily, we assume that each f^k is available through an (approximate) *oracle*, i.e., a procedure which, given x, returns (approximate) information about the value of $f(x)$ and the first-order behavior of f at x; our specific definition is given in (2.2). Our development hinges on a somewhat stronger assumption than usual, though:

$$\text{each } f^k \text{ is } \textit{globally Lipschitz} \text{ on } X \text{ with } \textit{known} \text{ Lipschitz constant } L^k. \tag{2}$$

We are especially motivated by the case of a block-structured problem

$$\sup \left\{ \ \textstyle\sum_{k \in \mathcal{K}} c^k u^k \ : \ \sum_{k \in \mathcal{K}} A^k u^k = b \ , \ \ u^k \in U^k \quad k \in \mathcal{K} \right\} \ , \tag{3}$$

where f is the *Lagrangian function* w.r.t. the "complicating" constraints that link together blocks of variables that would otherwise be independent, i.e.,

$$f(x) = xb + \textstyle\sum_{k \in \mathcal{K}} (\ f^k(x) = \nu(1.3^k_x) \ ) \ , \tag{4}$$

with $\nu(\cdot)$ denoting the optimal value of an optimization problem, and

$$\sup \left\{ \ (c^k - xA^k)u^k \ : \ u^k \in U^k \ \right\} \ . \tag{1.3^k_x}$$

For each $k \in \mathcal{K}$, any optimal solution u^k_* of (1.3^k_x) provides the function value $f^k(x) = (c^k - xA^k)u^k_*$ and the subgradient $z^k = -A^k u^k_* \in \partial f^k(x)$. For such f, (1.2) is often true, for instance because U^k is compact and finite bounds $-\infty < \underline{u}^k \leq u^k \leq \bar{u}^k < \infty$ are known for each $u^k \in U^k$ (very often, $U^k \subseteq \{0,1\}^{n_k}$). Minimizing f solves the *Lagrangian dual* of (1.3), which has countless applications; e.g., [4,5,12,16,17,24,27] among the many others. Typically (1.3) is "difficult", due to either being large-scale, $\mathcal{NP}$-hard, or both. Hence, computing f is cheaper than solving (1.3), if only because it separates into $|\mathcal{K}|$ smaller subproblems. However it may still be costly, as each (1.3^k_x) may still be $\mathcal{NP}$-hard, $|\mathcal{K}|$ may be large, or both. Thus, finding ways to reduce the function evaluation cost may be useful. An attractive strategy is to compute f only *approximately* to within some *error* ε, an issue that has seen substantial interest of late [9–11, 13, 29, 32, 33]. In our sum-function context, this may actually mean two different things: that problems (1.3^k_x) are approximately solved, or that some of them are not solved at all. Approaches doing the latter are called *incremental*. However, in all proposals so far [8, 13, 20], avoiding the solution of some (1.3^k_x) is only possible at "bad" iterations where the f-value does not improve (a.k.a. *Null Steps*, NS), while "good" iterations (*Serious Steps*, SS) require that *all* the f^k are computed. This is basically due to the fact that, in order to prove that any $x \in X$ is approximately optimal to some accuracy ε, one has to compute its function value $f(x)$ with at least the same accuracy [7, Observation 2.7]. Yet, our development will clarify that what is really needed is an *upper bound* on $f(x)$.

Upper bounds on $f(x)$ are not directly mentioned in the literature, except in the recent [31] for a different context. They are indirectly used in the fundamental reference [10]; in particular, the *controllable lower oracle* there explicitly produces lower estimates only, but it has a known maximum error out of which worst-case upper estimates can be derived. This is used to define the *conservative decrease* [10, (5.8)], which is basically what we will use (cf. (3.4)), except that our upper estimates may be tighter than the worst-case ones. Hence, the controllable bundle method [10, Algorithm 5.4] and the asymptotically exact bundle method of [10, §7.1.4] are very close to the methods we analyze here. However, in our analysis upper estimates take center stage: they are explicitly produced by the oracle, which can at some iteration produce *only* them. As a consequence, we don't require the accuracy to be nonincreasing, as in [10, Remark 6.8], nor to be exactly zero when the function value is below a given target, as in [10, §7.1.2] [20]. We also provide the oracle(s) with a full description of the conditions that are needed from the returned information so that the optimization can proceed: *two targets*, an upper and a lower one, and an accuracy. For the Lagrangian case (1.3)/(1.4), this allows to terminate early on the solution of problems (1.3^k_x). Having upper estimates available also allows to complement the usual *lower model*(s of the individual components f^k) of f, that traditionally drive the optimization process, with an *upper model* that provides *upper estimates* of $f(x)$ even if no oracle has ever been called at x. This has already been done in [1], but only on a small subset of the search space: exploiting (1.2) we extend the upper model to all of X. This is the fundamental technical idea that allows us to prove convergence without necessarily requiring that all components have been evaluated at SS.

This work is organized as follows. In §2 we introduce the different ingredients of the algorithm, their rationale, and key notation. The algorithm is given in §3, where we also discuss the convergence analysis in details. In §4 we extend the framework to two different classes of oracles that may not be able to provide information with arbitrary accuracy, and we terminate with conclusions in §5.

2 Ingredients of the algorithm

We start by illustrating the main components that have to be assembled to form a complete algorithm.

2.1 The oracle

The oracle $\mathcal{O}^k$ of f^k, when called at some fixed $x \in X$, has to provide information about both the function value $f^k(x)$ and the first-order behavior of f^k at x. For the latter we require a standard *lower linearization* of f^k; that is, some *lower estimate* $\underline{f}^k \leq f^k(x)$ and a vector $z^k \in \mathbb{R}^n$ such that

$$f^k(\cdot) \geq \underline{f}^k + \langle z^k, \cdot - x \rangle . \tag{5}$$

In order to control the accuracy of $\underline{f}^k$ (and therefore of z^k) we extend the definition of the *inexact, informative, on-demand* oracle of [31]: besides $x \in X$, the oracle inputs *three* parameters $-\infty \leq \underline{\mathtt{tar}}^k \leq$

$\overline{\texttt{tar}}^k \leq \infty$ (the *lower and upper targets*, with $\overline{\texttt{tar}}^k > -\infty$ and $\underline{\texttt{tar}}^k < \infty$) and $0 \leq \varepsilon^k \leq \infty$ (the *accuracy*), and provides

$$\begin{cases} \left[\begin{array}{l} \text{i) } \textit{function value information}\text{: } \textit{two} \text{ values } \underline{f}^k \text{ and } \bar{f}^k \text{ s.t.} \\ \qquad -\infty \leq \underline{f}^k \leq f^k(x) \leq \bar{f}^k \leq \infty \quad \text{and} \quad \bar{f}^k - \underline{f}^k \leq \varepsilon^k; \\ \text{ii) } \textit{first-order information}\text{: if } \underline{f}^k > -\infty, \text{ a } z^k \in \mathbb{R}^n \text{ s.t. (2.1) holds;} \end{array}\right. \\ \text{iii) s.t. } \textit{at least one} \text{ between } \bar{f}^k \leq \overline{\texttt{tar}}^k \text{ and } \underline{f}^k \geq \underline{\texttt{tar}}^k \text{ holds.} \end{cases} \tag{6}$$

It is easy to realize that it is always possible for the oracle to provide a correct answer, possibly at the cost of computing $f^k(x)$ with high accuracy. Our setting is therefore biased towards subproblems (1.3^k_x) that can be solved with arbitrary accuracy; however, the extension to more relaxed assumptions will be provided in §4.

We will denote by $\mathcal{O}^k(\underline{\texttt{tar}}^k, \overline{\texttt{tar}}^k, \varepsilon^k, x)$ a call to the oracle. Three parameters may seem somewhat redundant, especially as all other approaches in the literature only use one or two. For instance, if *both* $\bar{f}^k \leq \overline{\texttt{tar}}^k$ *and* $\underline{f}^k \geq \underline{\texttt{tar}}^k$, then $\bar{f}^k - \underline{f}^k \leq \overline{\texttt{tar}}^k - \underline{\texttt{tar}}^k$, and ε^k might be deemed useless. Setting $\overline{\texttt{tar}}^k = \infty$, $\underline{\texttt{tar}}^k = -\infty$ and $\varepsilon^k < \infty$ gives a standard ε^k-*approximated* oracle: since (2.2.i) holds with $\varepsilon^k < \infty$, *both* $\bar{f}^k$ and $\underline{f}^k$ must be finite, so (2.2.ii) together with (2.1) gives $z^k \in \partial_{\varepsilon^k} f^k(x)$. Hence,

Lemma 1 *Under* (1.2), $\|z^k\| \leq L^k$ *for each* z^k *produced by* (2.2).

Proof. Since z^k is produced whenever $\underline{f}^k > -\infty$, $z^k \in \partial_\varepsilon f^k(x)$ for some *finite* ε: if $\bar{f}^k < \infty$ then $\varepsilon = \bar{f}^k - \underline{f}^k < \infty$ is known, otherwise use the unknown but finite $f(x)$ in place of $\bar{f}^k$. It is well-known that $\|z^k\| \leq L^k$ for any $z^k \in \partial_\varepsilon f^k(x)$: this is immediate to prove by the very definition, or use [22, Proposition 4.1.2]. ∎

However, (2.2) is more flexible than previous definitions of approximate oracles in that it allows $\varepsilon^k = \infty$, whereby it then requires *only one* between $\bar{f}^k$ and $\underline{f}^k$ to be finite. This is particularly relevant if (1.3^k_x) is $\mathcal{NP}$-hard, since then computing $f^k(x)$ typically amounts to *three* different processes:

1. finding a "good enough" feasible solution $\bar{u}^k \in U^k$, providing both the lower bound $\underline{f}^k = c^k\bar{u}^k \leq \nu(1.3^k_x)$ and the first-order information $z^k = -A^k\bar{u}^k$, which requires (arbitrarily complex) *heuristics*;
2. finding a "good enough" upper bound $\bar{f}^k \geq \nu(1.3^k_x)$, which requires the *exact solution* of some appropriate *relaxation* of the problem (itself possibly an arbitrarily complex process, considering that the solution of (1.1) for (1.4) is often nothing but the computation of an upper bound on $\nu(1.3)$);
3. if $\underline{f}^k$ and $\bar{f}^k$ are not "close enough" (cf. (2.2.i), perform an arbitrary amount of branching and/or tightening (say, by valid inequalities) to make them so.

When, say, a general-purpose MILP solver is used to solve (1.3^k_x), these three processes are tightly integrated; however, they nonetheless remain conceptually distinct. Thus, any such solver typically produces candidate values $\underline{f}^k \leq \nu(1.3^k_x) \leq \bar{f}^k$, and gradually reduces the distance between the two. Given the three parameters in (2.2) it is easy to stop the solution process (e.g., via appropriate `callback` functions) as soon as the required conditions are satisfied. The parameters are not redundant to each other: while $\overline{\texttt{tar}}^k$ and $\underline{\texttt{tar}}^k$ set specific targets on $\nu(1.3^k_x)$, but are independent from one another, ε^k requires both bounds, but it makes no assumptions on where $\nu(1.3^k_x)$ lies. For $\varepsilon^k = \infty$, this means that the computation can be interrupted without having one of the two bounds at all. For instance, if $\bar{u}^k \in U^k$ is found such that $c^k\bar{u}^k \geq \underline{\texttt{tar}}^k$ one can entirely avoid to compute any relaxation, since then $\bar{f}^k = \infty$ is allowed. Symmetrically, if a bound $\nu(1.3^k_x) \leq \bar{f}^k \leq \overline{\texttt{tar}}^k$ is obtained, one can stop without producing any $\bar{u}^k \in U^k$, and therefore any z^k, since $\underline{f}^k = -\infty$ is allowed. This is more flexible than all approaches proposed so far in the literature. For instance, [9, 31] for the non-sum case (the former a level method, the latter for a discrete case) produce upper estimates (the former implicitly, the latter explicitly) but only have one target, and must always produce z^k. The recent [8] for the sum case do not set any target to the individual oracles, although a global (upper) target is set for f. Our setting therefore has the potential to reduce the number of iterations in which f^k is computed—i.e., (1.3^k_x) is solved—with high accuracy.

2.2 Lower models

Bundle methods sample the solution space in a sequence $\{x_\ell\}$ of iterates. Assuming for simplicity that all components are evaluated at each x_ℓ, they therefore have at their disposal the corresponding set of triples $\{\,(x_\ell, z_\ell, f_\ell)\,\}$ satisfying (2.1) for the whole of f. These are customarily used to construct the *(lower) bundle* $\mathcal{B} = \{\,(z_i, \alpha_i)\,\}$ and the corresponding (*aggregated*) *cutting plane model*

$$\check{f}_{\mathcal{B}}(x) = \max\left\{\ \langle z_i, x\rangle - \alpha_i\ :\ i \in \mathcal{B}\ \right\}\ , \tag{7}$$

with the useful shorthand "$i \in \mathcal{B}$" for "$(z_i, \alpha_i) \in \mathcal{B}$". Upon first reading one may assume $i = \ell$ and $\alpha_i = \langle z_i, x_i\rangle - f_i$, which, via (2.1), immediately proves that $\check{f}_{\mathcal{B}} \leq f$, i.e., $\check{f}_{\mathcal{B}}$ is a *lower model* of f. In general the pairs in $\mathcal{B}$ may not be directly correlated with the iterates, as we shall see. While $\check{f}_{\mathcal{B}}$ is not the only possible (lower) model of f [2, 28], our development only uses cutting plane models.

The issue with $\check{f}_{\mathcal{B}}$ is that it requires to compute all components at all iterations, which is precisely what we would like to be able to avoid. For this it is useful to employ *individual lower models* for each component, i.e.,

$$\check{f}^k_{\mathcal{B}}(x) = \max\left\{\ \left\langle z^k_i, x\right\rangle - \alpha^k_i\ :\ (z^k_i, \alpha^k_i) \in \mathcal{B}^k\ \right\} \leq f^k(x) \tag{8}$$

depending on *individual (lower) bundles* $\mathcal{B}^k = \{\,(z^k_i, \alpha^k_i = \langle z_i, x_i\rangle - f_i)\,\}$. Whenever possible we will not distinguish between (2.3) and the *disaggregated lower model* $\check{f}_{\mathcal{B}}(x) = \sum_{k\in\mathcal{K}} \check{f}^k_{\mathcal{B}}(x) \leq f(x)$. Note that here we are disregarding possible "simple" terms in f, like the linear one in (1.4) associated to the right-hand-side of the relaxed coupling constraints, or constraints $x \in X$; these will be dealt with in §3.6.

It has to be remarked that the choice between aggregated and disaggregated lower models is by no means inconsequential. In fact, while using (2.4) is well-known to improve, often substantially, the convergence speed of the algorithm, it may also come at a significant cost in terms of solution time of the master problem (cf. §2.4). So, while (2.4) is often preferable (e.g., [19]), there are cases where (2.3) is more efficient. In the recent [34], for instance, *both* models are actually used depending on the type of iteration (NS or SS). While we initially present our approach with the use of disaggregated models, different choices will be discussed in §3.5.

2.3 Upper models

Our oracle definition allows us to define *upper models* of f^k, which is perhaps the main technical innovation of our approach. For each $k \in K$ we define the *upper bundle* $\mathcal{P}^k = \{\,(x_i, \bar{f}^k_i)\,\}$; upon first reading one may again assume that $i = \ell$ and $f^k(x_i) \leq \bar{f}^k_i < \infty$ is the oracle output, but again things will be somewhat more complex. Yet, using the standard shorthand, it is clear that

$$\dot{f}^k_{\mathcal{P}}(x) = \inf\left\{\ \textstyle\sum_{i\in\mathcal{P}^k} \bar{f}^k_i\theta^k_i\ :\ \sum_{i\in\mathcal{P}^k} x_i\theta^k_i = x\ ,\ \theta^k \in \Theta^k\ \right\} \geq f(x) \tag{9}$$

for all $x \in X$, with Θ^k the unitary simplex in appropriate dimension. The issue with $\dot{f}^k_{\mathcal{P}}$ is that (2.5) may admit no solution: $\dot{f}^k_{\mathcal{P}}(x) = \infty$ for $x \notin \bar{X}^k_{\mathcal{P}} = \mathrm{conv}(\{\,x_i\ :\ i \in \mathcal{P}^k\,\})$. Furthermore, since $\bar{f}^k_i \gg f^k(x_i)$ may happen, there is no guarantee that $\dot{f}^k_{\mathcal{P}}$ satisfies (1.2) with L^k. Yet, exploiting (1.2) we can improve $\dot{f}^k_{\mathcal{P}}$ by defining

$$\begin{aligned}\hat{f}^k_{\mathcal{P}}(x) =& \min\{\,\dot{f}^k_{\mathcal{P}}(w) + L^k\|x - w\|_2\ :\ w \in X\,\}\\ =& \min\left\{\ \textstyle\sum_{i\in\mathcal{P}^k} \bar{f}^k_i\theta^k_i + L^k\|s^k\|_2\ :\ \sum_{i\in\mathcal{P}^k} x_i\theta^k_i + s^k = x\ ,\ \theta^k \in \Theta^k\ \right\}\ .\end{aligned} \tag{10}$$

Clearly, $f^k(x) \leq \hat{f}^k_{\mathcal{P}}(x) < \infty$: (1.2) implies that $f^k(x) \leq f^k(w) + L^k\left\|x - w\right\|_2 \leq \dot{f}^k_{\mathcal{P}}(w) + L^k\left\|x - w\right\|_2$ for any $x \in X$ and arbitrary w. This kind of regularization has been studied in [21, §XI.3.4]; Proposition XI.3.4.5 there proves that $\hat{f}^k_{\mathcal{P}}(x) = \dot{f}^k_{\mathcal{P}}(x) \iff \exists z^k \in \partial\dot{f}^k_{\mathcal{P}}(x)$ such that $||z^k|| \leq L^k$. As previously remarked this may not happen, not even for $x \in int\,\bar{X}^k_{\mathcal{P}}$; thus, $\hat{f}^k_{\mathcal{P}}$ both extends and (potentially) improves $\dot{f}^k_{\mathcal{P}}$. As (2.6) shows, $\hat{f}^k_{\mathcal{P}}$ can be computed "cheaply"; thus, the same holds for the global upper model $\hat{f}_{\mathcal{P}}(x) = \sum_{k\in\mathcal{K}} \hat{f}^k_{\mathcal{P}}(x) \geq f(x)$. Clearly, $\hat{f}_{\mathcal{P}} < \infty$ whenever $\mathcal{P}^k \neq \emptyset$ for all $k \in \mathcal{K}$.

Introducing $\hat{f}_{\mathcal{P}}$ is instrumental in avoiding the asymmetry that was present in all previous analyses of incremental bundle approaches: while $\check{f}_{\mathcal{B}} < \infty$ for all $x \in X$, and therefore producing *lower estimates* on $f(x)$ has always been easy, proving that $z \in \partial_\varepsilon f(x)$ requires to actually produce an *upper bound* $\bar{f} \geq f(x)$, i.e., some $\bar{f}^k \geq f^k(x)$ for *all* $k \in \mathcal{K}$. This is why all incremental bundle approaches so far have required computing *all* the components, at least in selected iterations. As we will see, the availability of

$\hat{f}^k_{\mathcal{P}}(x) < \infty$ will allow us to relax this strong requirement.

2.4 The master problem

The main use of the lower model is to drive the search for the next iterate. In the standard cutting-plane approach this would be done by minimizing $\check{f}_{\mathcal{B}}$ over X, but this is well-known to suffer from *instability* issues. This is why bundle methods try to ensure that the next iterate lies in an appropriate (most often, implicitly defined) neighborhood of a suitably chosen point $\bar{x} \in X$, called the *stability center*. In particular, *proximal* bundle methods as those studied here obtain this by solving the *stabilized master problem*

$$x_+ = \operatorname{argmin}\{\, \check{f}_{\mathcal{B}}(x) + \tfrac{1}{2t}||\, x - \bar{x}\,||_2^2 \,\} \quad . \tag{11}$$

The *stabilizing term* $||x-\bar{x}||_2^2/(2t)$, governed by the *stabilization parameter* $t > 0$, ensures that x_+ will be "near" $\bar{x}$, thereby limiting the violent oscillations of the iterates typical of the un-stabilized cutting plane method, and responsible for its slow convergence. Additionally, it ensures that (2.7) is always bounded from below.

The introduction of the stability center $\bar{x}$ suggest to translate the lower models (2.4) using $\bar{x}$ as the origin. In the context of *exact* bundle methods, it is customary to also set the origin of the objective axis to $f(\bar{x})$, i.e., the reference value against which the decrease of f (and of $f_{\mathcal{B}}$) is measured. In *inexact* bundle methods it is necessary to choose a reference value to replace $f(\bar{x})$; in [10] this is the (lower) *level* $l \le f(\bar{x})$, which is used to compute the crucial *model decrease* [10, (3.8)]. That general choice allows to analyze convergence of many different inexact bundle variants, among which those with upper oracles that do not even guarantee (2.1). However, the model decrease using l as the reference can then be compared with an *effective decrease* that uses instead an upper estimate $\bar{f} \ge f(\bar{x}) \ge l$ as reference value [10, (5.8)]. Our simpler setting, where upper estimates are always available (thanks to both the oracle providing them and the upper model) rather suggests to use the same $\bar{f} \ge f(\bar{x})$ as the reference for measuring both decreases. This boils down to redefining the α_i as

$$\alpha_i^k(\bar{x}, \bar{f}^k) = \bar{f}^k - [\, \underline{f}_i^k + z_i^k(\bar{x} - x_i) \,] \quad , \tag{12}$$

called the *linearization errors* of z_i^k w.r.t. $\bar{x}$ (and the given upper estimate $\bar{f}^k$). In an exact bundle approach, where $\bar{f}^k = f^k(\bar{x})$, these are uniquely defined once $\bar{x}$ is set; in our case, instead, they also depend on the value of $\bar{f}^k$ which, as we will see, may change even if $\bar{x}$ doesn't. However, convexity of f^k, $\underline{f}_i^k \le f^k(x_i)$ and $\bar{f}^k \ge f^k(\bar{x})$ give

$$\alpha_i^k(\bar{x}, \bar{f}^k) \ge 0 \quad \Longrightarrow \quad z_i^k \in \partial_{\alpha_i^k(\bar{x},\bar{f}^k)} f^k(\bar{x}) \quad . \tag{13}$$

That is, $\alpha_i^k(\bar{x}, \bar{f}^k)$ measures the accuracy of z_i^k as first-order information of f^k in $\bar{x}$. This measure is easily recomputed if the stability center is updated to any other $\tilde{x}$ (with corresponding $\tilde{f} \ge f(\tilde{x})$) by the well-known *information transport property*

$$\alpha_i^k(\tilde{x}, \tilde{f}^k) = z_i^k(\, \bar{x} - \tilde{x}\,) + \alpha_i^k(\bar{x}, \bar{f}^k) + (\, \tilde{f}^k - \bar{f}^k\,) \ \ (\ge 0\,) \quad . \tag{14}$$

Since $\bar{x}$ is usually clear from the context, to alleviate the notation we will whenever possible use just α_i^k in place of $\alpha_i^k(\bar{x}, \bar{f}^k)$. This hides the fact that the α_i^k depends not only on $\bar{x}$ but also on $\bar{f}^k$, which can independently vary; thus, in the following we will take specific care to comment on this dependency whenever necessary. It is also important to remark that, with our definition, the linearization errors "take into account the *gap* $\bar{f}^k - \check{f}^k_{\mathcal{B}}(\bar{x}) \ge 0$" between the upper and lower estimate on $f^k(\bar{x})$. Indeed, plug $\check{f}^k_{\mathcal{B}}(\bar{x}) \ge \underline{f}_i^k + z_i^k(\bar{x} - x_i)$ into (2.8) to obtain

$$\bar{f}^k - \check{f}^k_{\mathcal{B}}(\bar{x}) = \min\{\, \alpha_i^k \,:\, i \in \mathcal{B}^k \,\} \le \alpha_i^k \quad \forall i \in \mathcal{B}^k \quad . \tag{15}$$

Given $\bar{x}$ and $\bar{f}^k$, the (*disaggregated*, cf. (2.4)) master problem (2.7) can then be written

$$\min\{\ \textstyle\sum_{k\in\mathcal{K}} v^k + \frac{1}{2t}||d||^2 \ :\ v^k \ge z_i^k d - \alpha_i^k \quad i \in \mathcal{B}^k \ \ ,\ \ k \in \mathcal{K}\ \} \tag{16}$$

and its optimal solution d_* gives $x_+ = \bar{x} + d_*$, where f is typically evaluated. Note that $v^k \ge \check{f}^k_{\mathcal{B}}(x_+) - \bar{f}^k$ (cf. (2.8)), and therefore for the optimal value v_*^k

$$v_* = \check{f}_{\mathcal{B}}(x_+) - \bar{f} = \textstyle\sum_{k\in\mathcal{K}} \left(\, v_*^k = \check{f}^k_{\mathcal{B}}(x_+) - \bar{f}^k \,\right) \quad , \tag{17}$$

a relationship that will be crucial later on. The usefulness of defining the linearization errors precisely

via (2.8) lies in the fact that the *dual* of (2.12) is

$$\min \left\{ \tfrac{1}{2} t \left\| \sum_{k\in\mathcal{K}} \sum_{i\in\mathcal{B}^k} z_i^k \theta_i^k \right\|^2 + \sum_{k\in\mathcal{K}} \sum_{i\in\mathcal{B}^k} \alpha_i^k \theta_i^k \;:\; \theta^k \in \Theta^k \;,\; k \in \mathcal{K} \right\} , \tag{18}$$

where Θ^k is the unitary simplex of dimension $|\mathcal{B}^k|$. Standard duality theory guarantees that $\nu(2.12) = -\nu(2.14)$, and the relationships

$$d_* = -t z_* \quad , \quad v_* = -t||z_*||^2 - \alpha_* = \sum_{k\in\mathcal{K}} (d_* z_*^k - \alpha_*^k) = \sum_{k\in\mathcal{K}} v_*^k \tag{19}$$

between the primal and dual optimal solutions, where

$$z_*^k = \sum_{i\in\mathcal{B}^k} z_i^k \theta_{*i}^k, \;\; \alpha_*^k = \sum_{i\in\mathcal{B}^k} \alpha_i^k \theta_{*i}^k, \;\; z_* = \sum_{k\in\mathcal{K}} z_*^k, \;\; \alpha_* = \sum_{k\in\mathcal{K}} \alpha_*^k \tag{20}$$

"translate in the (z,α)-space" the dual optimal solution θ_* of (2.14). These relationships are crucial in the analysis of the method, since their obvious consequence

$$z_*^k \in \partial_{\alpha_*^k} f^k(\bar{x}) \;\; k \in \mathcal{K} \quad \Longrightarrow \quad z_* \in \partial_{\alpha_*} f(\bar{x}) \tag{21}$$

(cf. (2.9)) formally proves that $z_* = 0 \,(\Longrightarrow d_* = 0)$ and $\alpha_* = 0$ imply that $0 \in \partial f(\bar{x})$, i.e., $\bar{x}$ is optimal. In practice one therefore stops when $\|z_*\|$ and α_* are "small". Hence, whenever one does not stop, $v_* < 0$ (cf. (2.15)); in particular it is *not* "small", i.e., d_* is a (significant) *descent direction*. Note that this is *not* true for the individual components, i.e., $v_*^k > 0$ may happen for some (but not all) k. The *predicted descent* (2.13) is crucial in the analysis: its component-wise characterization

$$\check{f}_{\mathcal{B}}^k(x_+) = \bar{f}^k + d_* z_*^k - \alpha_*^k = \bar{f}^k + v_*^k \tag{22}$$

(cf. (2.13) and (2.15)) shows that v_* also "includes the gap $\bar{f} - \check{f}_{\mathcal{B}}(\bar{x}) \geq 0$", since (2.11) implies $\alpha_*^k \geq \bar{f}^k - \check{f}_{\mathcal{B}}^k(\bar{x})$, which in turn gives

$$-v^* \geq \alpha_* = \sum_{k\in\mathcal{K}} \alpha_*^k \geq \sum_{k\in\mathcal{K}} \left(\bar{f}^k - \check{f}_{\mathcal{B}}^k(\bar{x}) \right) = \bar{f} - \check{f}_{\mathcal{B}}(\bar{x}) \;\; . \tag{23}$$

3 The base algorithm

We now describe a first version of the algorithm, that mimics as closely as possible those for the exact case, and discuss its convergence.

3.1 Notation

Extending the convergence arguments to our setting requires a few notational contraptions. Since the aim is to avoid calling all the oracles at x_+, we will have to deal with subsets of components. For instance, $\mathcal{S} \subset \mathcal{K}$ will be the subset of components where the oracle has been called (at least once), producing $\underline{f}_+^k \leq f_+^k = f^k(x_+) \leq \bar{f}_+^k$, and $\mathcal{Z} \subset \mathcal{S}$ that for which a subgradient z^k has been produced as well. We will frequently need to restrict sums to subsets of $\mathcal{K}$, e.g., as in

$$\bar{f}_+^{\mathcal{S}} = \sum_{k\in\mathcal{S}} \bar{f}_+^k \quad , \quad \underline{f}_+^{\mathcal{S}} = \sum_{k\in\mathcal{S}} \underline{f}_+^k \quad , \quad z^{\mathcal{Z}} = \sum_{k\in\mathcal{Z}} z^k \quad , \quad \alpha^{\mathcal{Z}} = \sum_{k\in\mathcal{Z}} \alpha^k \;\; .$$

We will use "$-\mathcal{H}$" to refer to (sums over) the complement of $\mathcal{H}$, i.e., $\mathcal{K} \setminus \mathcal{H}$; taking "$k$" to mean "$\{ k \}$", "$-k$" then has to be intended as $\mathcal{K} \setminus \{ k \}$.

Each time we obtain function value estimates from the oracle, we ensure they are consistent with the currently available models by the obvious

$$\bar{f}_{\mathcal{P}}^k(x_+) = \min\{ \bar{f}_+^k \,,\, \hat{f}_{\mathcal{P}}^k(x_+) \} \quad , \quad \underline{f}_{\mathcal{B}}^k(x_+) = \max\{ \underline{f}_+^k \,,\, \check{f}_{\mathcal{B}}^k(x_+) \} \;\; . \tag{24}$$

Whenever possible we will avoid to distinguish between $\bar{f}_{\mathcal{P}}^k(x_+)$ and $\bar{f}^k(x_+)$, using the same notation $\bar{f}_+$ for both, and similarly for lower estimates. We will also short-hand $\hat{f}_{\mathcal{P}}^k(x_+)$ and $\check{f}_{\mathcal{B}}^k(x_+)$ with $\hat{f}_+^k$ and $\check{f}_+^k$, respectively. Minimal care in the handling of $\mathcal{B}^k$ and $\mathcal{P}^k$ will ensure that

$$-\infty < \check{f}_+^k = \bar{f}^k + v_*^k \leq \underline{f}_+^k \leq f_+^k \leq \bar{f}_+^k \leq \hat{f}_+^k < \infty \;\; , \tag{25}$$

so that (3.1) allows us to always assume $\bar{f}_+^k < \infty$ and $\underline{f}_+^k > -\infty$, even though $\mathcal{O}^k$ has not been called yet (or has produced infinite ones). It is also necessary to estimate the linearization error of z^k w.r.t. $\bar{x}$: the fact that $x_+ = \bar{x} + d_*$ gives, using (2.8)

$$\alpha^k = \bar{f}^k - \underline{f}_+^k + z^k d_* \;\; . \tag{26}$$

3.2 Presentation of the algorithm

The pseudo-code of the algorithm is:

0 (input and initializations) Fix the accuracy parameters $\delta_1 \geq 0$, $\delta_2 \geq 0$ and the algorithmic parameters $0 < m_1 < m_2 < 1$. $\ell := 1$. Choose arbitrarily $\bar{x}_1$ and $t_1 > 0$. For all $k \in \mathcal{K}$, call $\mathcal{O}^k(-\infty, \infty, \varepsilon^k, \bar{x}_1)$ with arbitrary $0 \leq \varepsilon^k < \infty$, collect $-\infty < \underline{f}^k_1 \leq f^k(\bar{x}_1) \leq \bar{f}^k_1 < \infty$ and z^k_1. $\mathcal{B}^k_1 := \{ (z^k_1, \alpha^k_1(\bar{x}_1, \bar{f}^k_1)) \}$. $\mathcal{P}^k_1 := \{ (\bar{x}_1, \bar{f}^k_1) \}$.

1 (master problem) For each $k \in \mathcal{K}$, `if` $(\bar{x}_\ell, \bar{f}^k) \in \mathcal{P}^k$ with $\bar{f}^k > \bar{f}^k_\ell$ `then` update $\mathcal{P}^k$ so that $(\bar{x}_\ell, \bar{f}^k_\ell) \in \mathcal{P}^k$. Solve (2.12)/(2.14) to produce the optimal solutions $d_{*,\ell}$, $v^k_{*,\ell}$, $\theta^k_{*,\ell}$, $z^k_{*,\ell}$ and $\alpha^k_{*,\ell}$ for all $k \in \mathcal{K}$.

2 (stopping condition) `if` $||z_{*,\ell}|| \leq \delta_1$ and $\alpha_{*,\ell} \leq \delta_2$ `then stop.` `else` $\Delta_{*,\ell} := \nu(2.14) = t_\ell ||z_{*,\ell}||^2/2 + \alpha_{*,\ell} = -\nu(2.12) = -v_{*,\ell} - ||d_{*,\ell}||^2/(2t_\ell) \, (> 0)$, $\varepsilon_\ell := m_2(-v_{*,\ell}) - m_1 \Delta_{*,\ell} \, (> 0)$, $x_{\ell+1} := \bar{x}_\ell + d_{*,\ell}$, $\underline{f}^k_{+,\ell} := \check{f}^k_{\mathcal{B}}(x_{\ell+1})$, $\bar{f}^k_{+,\ell} := \hat{f}^k_{\mathcal{P}}(x_{\ell+1})$ (cf. (2.18)/(2.6)) for all $k \in \mathcal{K}$. $\bar{f}^k_{\ell+1} := \bar{f}^k_\ell$ for all $k \in \mathcal{K}$. Define the *global upper target*

$$\overline{\mathtt{tar}}_\ell := \big(\underline{f}_{+,\ell} = \check{f}_{\mathcal{B}}(x_{\ell+1}) = \bar{f}_\ell + v_{*,\ell} \big) - m_2 v_{*,\ell} \, (< \bar{f}_\ell). \tag{27}$$

Select arbitrarily convex combinators $\beta^k_\ell \geq 0$ (such that $\sum_{k \in \mathcal{K}} \beta^k_\ell = 1$), $\Delta^k_{*,\ell} := \beta^k_\ell \Delta_{*,\ell} \, (\geq 0)$. $\mathcal{S} := \mathcal{Z} := \mathcal{S}' := \mathcal{S}'' := \emptyset$.

3 (oracle interaction) Select $k \in -\mathcal{S}''$ arbitrarily.

3.1 (define oracle parameters) Define the upper and lower targets for k

$$\begin{aligned} \overline{\mathtt{tar}}^k_\ell &:= \max \big\{ \, (\, \bar{f}^k_\ell + v^k_{*,\ell} \,) - m_2 \beta^k_\ell v_{*,\ell} \, , \, \overline{\mathtt{tar}}_\ell - \bar{f}^{-k}_{+,\ell} \, \big\} \\ \underline{\mathtt{tar}}^k_\ell &:= \min \left\{ \begin{array}{l} (\, \bar{f}^k_\ell + v^k_{*,\ell} \,) + m_1 \Delta^k_{*,\ell} \ , \\ (\, \bar{f}^{\mathcal{Z}}_\ell + v^{\mathcal{Z}}_{*,\ell} \,) + m_1 \Delta_{*,\ell} - \underline{f}^{\mathcal{Z} \setminus \{k\}}_{+,\ell} \end{array} \right. \end{aligned} \tag{28}$$

`if` $k \in -\mathcal{S}$ `then` $\varepsilon^k_\ell := \infty$, `else if` $k \in -\mathcal{S}'$ `then` $\varepsilon^k_\ell := \overline{\mathtt{tar}}^k_\ell - \underline{\mathtt{tar}}^k_\ell$, `else` $\varepsilon^k_\ell := \beta^k_\ell \varepsilon_\ell$.

3.2 (oracle call) Call $\mathcal{O}^k(\underline{\mathtt{tar}}^k_\ell, \overline{\mathtt{tar}}^k_\ell, \varepsilon^k_\ell, x_{\ell+1})$, collect $\underline{f}^k_{+,\ell}$, $\bar{f}^k_{+,\ell}$ and possibly z^k_ℓ. Update $\underline{f}^k_{+,\ell}$ according to (3.1).

3.3 (upper information management) Add $(x_{\ell+1}, \bar{f}^k_{+,\ell})$ to $\mathcal{P}^k_\ell$. Compute $\hat{f}^k_{\mathcal{P}}(\bar{x}_\ell)$ according to (2.6). $\Delta \bar{f}^k_\ell := \bar{f}^k_{\ell+1} - \hat{f}^k_{\mathcal{P}}(\bar{x}_\ell)$. `if` $\Delta \bar{f}^k_\ell > 0$ `then` $\bar{f}^k_{\ell+1} := \hat{f}^k_{\mathcal{P}}(\bar{x}_\ell)$, $\alpha^k_{i,\ell} := \alpha^k_{i,\ell}(\bar{x}_\ell, \, \bar{f}^k_{\ell+1}) := \alpha^k_{i,\ell} - \Delta \bar{f}^k_\ell$ for all $i \in \mathcal{B}^k_\ell$, $\alpha^k_{*,\ell} := \alpha^k_{*,\ell} - \Delta \bar{f}^k_\ell$, and $\alpha_{*,\ell} := \alpha_{*,\ell} - \Delta \bar{f}^k_\ell$ (note that $v^k_{*,\ell}$, $v_{*,\ell}$ and $\Delta_{*,\ell}$ and $\overline{\mathtt{tar}}_\ell$ are *not* updated accordingly).

3.4 (lower information management) `if` $\underline{f}^k_{+,\ell} > -\infty$ (i.e., z^k_ℓ has been produced) `then` $\mathcal{Z} := \mathcal{Z} \cup \{k\}$ and add $(z^k_\ell, \alpha^k_{\ell,\ell}(\bar{x}_\ell, \bar{f}^k_{\ell+1}))$ to $\mathcal{B}^k_\ell$. Update $\underline{f}^k_{+,\ell}$ according to (3.1).

3.5 (accuracy sets management) $\mathcal{S} := \mathcal{S} \cup \{ \, k \, \}$. `if` $\bar{f}^k_{+,\ell} - \underline{f}^k_{+,\ell} \leq \overline{\mathtt{tar}}^k_\ell - \underline{\mathtt{tar}}^k_\ell$ `then` $\mathcal{S}' := \mathcal{S}' \cup \{ \, k \, \}$. `if` $\bar{f}^k_{+,\ell} - \underline{f}^k_{+,\ell} \leq \beta^k_\ell \varepsilon_\ell$ `then` $\mathcal{S}'' := \mathcal{S}'' \cup \{ \, k \, \}$.

4 (NS/SS test) `if` the condition

$$\bar{f}_{+,\ell} \leq \overline{\mathtt{tar}}_\ell \tag{29}$$

holds `then` possibly perform a Serious Step (SS): $\bar{x}_{\ell+1} := x_{\ell+1}$, $\bar{f}^k_{\ell+1} := \bar{f}^k_{+,\ell}$ for all $k \in \mathcal{K}$ and go to Step 5. `if` instead the condition

$$\begin{aligned} &\Delta f^{\mathcal{Z}} := \textstyle\sum_{k \in \mathcal{Z}} \big(\, \Delta f^k := \underline{f}^k_{+,\ell} - (\, \bar{f}^k_\ell + v^k_{*,\ell} \,) \, \big) \geq m_1 \Delta_{*,\ell} \\ \equiv \ &\underline{f}^{\mathcal{Z}}_{+,\ell} \geq (\, \bar{f}^{\mathcal{Z}}_\ell + v^{\mathcal{Z}}_{*,\ell} \,) + m_1 \Delta_{*,\ell} \end{aligned} \tag{30}$$

holds `then` possibly perform a Null Step (NS): $\bar{x}_{\ell+1} := \bar{x}_\ell$ and go to Step 5. `if` neither a SS nor a NS has been performed `then` go to Step 3.

5 (bookkeeping) For each $k \in \mathcal{K}$ construct $\mathcal{P}^k_{\ell+1}$ and $\mathcal{B}^k_{\ell+1}$ using $\mathcal{P}^k_\ell$ and $\mathcal{B}^k_\ell$ appropriately. Select $t_{\ell+1}$ appropriately. $\ell := \ell + 1$ and go to Step 1.

3.3 Discussion

We now comment on some choices in the algorithm.

1. At Step 0, $\varepsilon_1^k < \infty$ is arbitrary: $\mathcal{O}^k$ have to provide *finite* but *arbitrarily loose* upper and lower estimates, which should be "cheap". It would be possible to call $\mathcal{O}^k$ at a different $\bar{x}_1^k$ for each $k \in K$ and arrange $\mathcal{B}_1^k/\mathcal{P}_1^k$ accordingly—(3.1) still gives finite estimates at $\bar{x}_1$—but this would hardly make sense.
2. The index ℓ counts *major iterations*, i.e., solutions to the master problem. Each major iteration includes one or more *minor iterations*—loops between Step 3 and Step 4—where different $\mathcal{O}^k$ are called (possible multiple times) on $x_{\ell+1}$.
3. If the stopping condition at Step 2 holds, then $\bar{x}_\ell$ is approximately optimal (cf. (2.17)). However, unlike in exact bundle methods, the fact that the stopping condition does not hold does *not* imply $d_{*,\ell} \neq 0$. That is, the algorithm may produce $d_{*,\ell} = 0 \Longrightarrow x_{\ell+1} = \bar{x}_\ell$, but $\alpha_{*,\ell} \gg 0$. This is not an issue: the master problem correctly selects $\bar{x}_\ell$ as the best estimate of the optimum of f, but the gap between $\bar{f}_\ell$ and $\underline{f}_\ell$, which is "comprised" in the α_i^k and therefore in $\alpha_{*,\ell}$ (cf. (2.11)/(2.19)), is too large to stop. As in [7], this means that the function estimates in $x_{\ell+1} = \bar{x}_\ell$ will be improved.
4. It is easy to check (cf. (3.5)) that $\overline{\mathtt{tar}}_\ell^k - \underline{\mathtt{tar}}_\ell^k \geq m_2\beta_\ell^k(-v_{*,\ell}) - m_1\Delta_{*,\ell}^k > 0$, as required by (2.2). Setting $m_1 \approx 0$ and $m_2 \approx 1$ should make the oracle cheaper. Also, $\overline{\mathtt{tar}}_\ell^k \geq \underline{\mathtt{tar}}_\ell^k \geq \check{f}_{\mathcal{B}}^k(x_{\ell+1}) = \bar{f}_\ell^k + v_{*,\ell}^k$, as expected: targets below the value of the lower models would not make sense. In view of the standard analysis of bundle methods, it would have been more natural to set the targets with $v_{*,\ell}^k$ rather than using the somewhat "artificial" combinators β_ℓ^k. This is, however, not possible: $v_{*,\ell}^k > 0$ would then lead to $\overline{\mathtt{tar}}_\ell^k < \underline{\mathtt{tar}}_\ell^k < \check{f}_{\mathcal{B}}^k(x_{\ell+1})$.
5. Besides being necessary for the reason outlined above, the convex combinators β_ℓ^k also make sense. The "desired amount of increase $m_1\Delta_{*,\ell}$/decrease $(1-m_2)v_{*,\ell}$", which are defined for the whole of f, have to be somehow "subdivided" among the components. Since this is used to make targets for the oracles, it has to be done a-priori: hence, some sort of "guess" about the individual values $f^k(x_{\ell+1})$ is unavoidable. The selection of the β_ℓ^k may be significant computationally. Intuitively, it should take into account factors such as "how hard is each component to evaluate" ("easier" ones might get smaller β_ℓ^k, as computing them with high accuracy is less demanding), and that some components may have significantly larger values than others, thereby being more influential on the overall value of f. A simple formula taking into account the latter factor (but not the former) is
$$\gamma_\ell^k = |v_{*,\ell}^k|/|\bar{f}_\ell^k| \quad , \quad \beta_\ell^k = \gamma_\ell^k / \textstyle\sum_{h\in\mathcal{K}} \gamma_\ell^h.$$
6. Due to (2.2.iii), *at least* one of the "partial" SS/NS conditions
$$\text{i)} \quad \bar{f}_{+,\ell}^k \leq \overline{\mathtt{tar}}_\ell^k \qquad , \qquad \text{ii)} \quad \underline{f}_{+,\ell}^k \geq \underline{\mathtt{tar}}_\ell^k \tag{31}$$
must be satisfied for all $k \in \mathcal{K}$, whatever the value of ε_ℓ^k. The targets $\overline{\mathtt{tar}}_\ell^k$ and $\underline{\mathtt{tar}}_\ell^k$ have been chosen so that (3.8.i) for *all* $k \in \mathcal{K} \Longrightarrow$ (3.6), and similarly (3.8.ii) for *all* $k \in \mathcal{K} \Longrightarrow$ (3.7). Hence, initially calling $\mathcal{O}^k$ with $\varepsilon_\ell^k = \infty$ ($k \in \mathcal{S}$) makes sense: if the oracles of *all* components satisfy *the same* relation in (3.8), then either a SS or a NS is done, possibly without having ever produced a single lower/upper estimate. Actually, this may happen even if $\mathcal{S} \subset \mathcal{K}$: due to the use of the upper/lower models, a SS/NS can be performed even (quite) before having computed all components. However, this may not happen; for instance, some oracle may return an upper estimate but no lower one, and some may do the converse. Thus, selecting a finite (but possibly "large") ε_ℓ^k ($k \in \mathcal{S}'$) may be necessary to force all oracles to provide both estimates. This may be enough to decide between SS and NS, or not; thus, a "small" ε_ℓ^k ($k \in \mathcal{S}''$) may ultimately have to be used to ensure that one among (3.6) and (3.7) holds, even if it comes at a larger oracle cost. Note that the algorithm checks if $\mathcal{O}^k$ actually produce information with higher accuracy than the required one, and updates the sets accordingly (for instance, k may "go straight from $-\mathcal{S}$ to $\mathcal{S}''$").
7. Calling the oracle more than once with the same iterate $x_{\ell+1}$ but decreasing ε_ℓ^k (cf. Lemma 3.1 below) also makes computational sense: in many cases the oracle will be able to *reoptimize efficiently*. Typically, state-of-the-art optimization solvers allow one to terminate the search for an optimal solution early by specifying a coarse optimality tolerance, and then resume the computation of

a more accurate solution at little or no extra cost by keeping all their internal data structures updated and just "jumping into the main loop".

8. The upper information management Step 3.3 is specific of our setting. It concerns the fact that computing $\bar{f}^k$ at $x_{\ell+1}$ may, through the upper model, change the upper estimate at the stability center $\bar{x}_\ell$: when $(x_{\ell+1}, \bar{f}^k_{+,\ell})$ is added to $\mathcal{P}^k_\ell$, $\hat{f}^k_{\mathcal{P}}(\bar{x}_\ell)$ may *decrease*, which in turn causes the α^k_i to decrease. It is therefore necessary to distinguish the value of the upper estimate at the *beginning* of a major iteration ℓ ($\bar{f}^k_\ell$) from the one that is dynamically revised during the minor iterations ($\bar{f}^k_{\ell+1}$). However, the algorithm does *not* update $v_{*,\ell}$ and $\Delta_{*,\ell}$, and therefore the targets, that also depend on the α^k_i. This is made clear by consistently using $\bar{f}^k_\ell$—which does not vary during the major iteration—and $v^k_{*,\ell}/v_{*,\ell}$ and $\Delta_{*,\ell}$—that are explicitly excluded from the update in Step 3.3—to define all the crucial rules of the algorithm (3.4), (3.5), (3.6), (3.7).
9. Avoiding to update the main algorithmic thresholds during major iterations has somewhat counter-intuitive consequences that are worth commenting upon: the algorithm may perform "fake" SS or NS. Indeed, assume that in Step 3.3 one has $\Delta\bar{f}^k_\ell > 0$, i.e., $\bar{f}^k_{\ell+1}$ decreases. The algorithm may then perform a SS, which means that (3.6) holds. However, if one were to recompute the target using $\bar{f}^k_{\ell+1} = \bar{f}^k_\ell - \Delta\bar{f}^k_\ell$ and the corresponding $v'_{*,\ell} = v_{*,\ell} + \Delta\bar{f}^k_\ell$, it may well be that $\bar{f}_{+,\ell} > (\bar{f}^k_{\ell+1} + v'_{*,\ell}) - m_2 v'_{*,\ell} = (\bar{f}_\ell + v_{*,\ell}) - m_2(v_{*,\ell} + \Delta\bar{f}^k_\ell) = \overline{\mathtt{tar}}^k_\ell - m_2\Delta\bar{f}^k_\ell$. In other words, the "true" decrease due to moving the stability center to $x_{\ell+1}$ is smaller than what (3.6) requires. Yet, this is not really an issue, as $\bar{f}^k_{\ell+1}$ is indeed "significantly smaller" than $\bar{f}^k_\ell$: that part of the decrease could have also been obtained by keeping $\bar{x}_{\ell+1} = \bar{x}_\ell$ is, in the end, irrelevant. This is tied to point 3: $x_{\ell+1} = \bar{x}_\ell$ is possible, in which case the SS is actually only improving the upper estimate at $\bar{x}_\ell$. This may even happen infinitely many times (say, if $\bar{x}_1$ is optimal and $z^k = 0$ throughout): basically, instead of asking to the $\mathcal{O}^k$ exact information from the start, one is continuously asking them more and more refined one. Analogously, the NS condition (3.7) is designed to ensure that the newly introduced information z^k_ℓ "decreases enough the value of $\nu(2.14)$" (cf. Lemma 3.14). Because $\alpha_{*,\ell} = \alpha_{*,\ell} - \Delta\bar{f}^k_\ell$, $\nu(2.14)$ already decreases even if $\mathcal{B}^k$ is not changed: however, once again the desired reduction of $\nu(2.14)$ is achieved, and this is enough to ensure convergence.
10. One may add specific checks to speed-up the algorithm if $\Delta\bar{f}^k_\ell > 0$ in Step 3.3. For instance, since $\alpha_{*,\ell}$ decreases, one may re-check the stopping condition. Also, one may check if $\bar{f}_{\ell+1} \leq \bar{f}_\ell + (1-m_2)v_{*,\ell}$ (cf. Theorem 3.3), and in case immediately perform a "SS/NS": set $\bar{x}_{\ell+1} = \bar{x}_\ell$ and immediately return to Step 1, but declare this as a SS (which, in particular, means that the $\mathcal{B}^k$ may be entirely cleared, cf. Assumption 3(i)). Similarly, one may check if $\Delta\bar{f}^k_\ell \geq \delta$ for some fixed $\delta > 0$: this means that $\nu(2.14)$ has decreased by at least δ (cf. Theorem 3.5), which allows to declare a NS and return to Step 1.
11. Appropriate rules for managing $\mathcal{B}^k$ and $\mathcal{P}^k$ will be discussed in §3.4 (cf. Assumptions 2 and 3). These can exploit the well-known (for the lower bundle) *aggregation technique*: the dual optimal solutions (2.16) can be used to define the *poorman's lower bundles*

$$\mathcal{B}^k_* = \{\, (z^k_*, \alpha^k_*) \,\} \qquad k \in \mathcal{K} \ , \tag{32}$$

so that $\check{f}^k_{\mathcal{B}_*}(d_*) = \check{f}^k_{\mathcal{B}}(d_*) = v^k_*$ (cf. Lemma 3.14). The same technique also applies to $\mathcal{P}^k$: the optimal solution (θ^k_*, s^k_*) to (2.6) for $\bar{x}_\ell$ immediately provides the *aggregated primal pair* and the corresponding *poorman's upper bundles*

$$\mathcal{P}^k_* = \Big\{\ (\, x^k_* \,,\, f^k_* \,) = \big(\ \textstyle\sum_{i\in\mathcal{P}^k} x_i\theta^k_{*,i} + s^k_* \,,\, \sum_{i\in\mathcal{P}^k} \bar{f}^k_i\theta^k_{*,i} + L^k\|s^k_*\|_2 \ \big) \ \Big\} \tag{33}$$

so that $\hat{f}^k_{\mathcal{P}_*}(\bar{x}_\ell) = \hat{f}^k_{\mathcal{P}}(\bar{x}_\ell)$.

3.4 Convergence analysis

The first step is specific to our development, and concerns the fact that a SS or a NS must occur after finitely many minor iterations.

Lemma 2 *For any $k \in \mathcal{K}$ and ℓ, the three values that ε^k_ℓ can assume during the major iteration ℓ are nonincreasing. As a consequence, $\mathcal{S}'' \subseteq \mathcal{S}' \subseteq \mathcal{Z} \subseteq \mathcal{S}$.*

Proof. Because $\Delta_{*,\ell} = -v_{*,\ell} - ||d_{*,\ell}||^2/(2t_\ell) \leq -v_{*,\ell}$ and $m_1 \leq m_2$, one has

$$0 \leq \beta^k_\ell(\, m_2(-v_{*,\ell}) - m_1\Delta_{*,\ell}\,) = \beta^k_\ell\varepsilon_\ell \ .$$

Now, $\overline{\texttt{tar}}_\ell^k \geq (\,\bar f_\ell^k + v_{*,\ell}^k\,) - m_2\beta_\ell^k v_{*,\ell}$ and $\underline{\texttt{tar}}_\ell^k \leq (\,\bar f_\ell^k + v_{*,\ell}^k\,) + m_1\Delta_{*,\ell}^k$ (cf. (3.5)) gives

$$\beta_\ell^k\varepsilon_\ell = \big(\,(\,\bar f_\ell^k + v_{*,\ell}^k\,) - m_2\beta_\ell^k v_{*,\ell}\,\big) - \big(\,(\,\bar f_\ell^k + v_{*,\ell}^k\,) + m_1\Delta_{*,\ell}^k\,\big) \leq \overline{\texttt{tar}}_\ell^k - \underline{\texttt{tar}}_\ell^k < \infty. \tag{34}$$

The chain inclusion between the sets is obvious; in particular, for $k \in \mathcal{S}'$ or $k \in \mathcal{S}''$ to hold a $\underline{f}_\ell^k > -\infty$ has to have been produced, which means $k \in \mathcal{Z}$ as well. ∎

To proceed, a (pretty obvious) assumption is needed. Indeed, while the algorithm is written in such a way that it is not *necessary* to perform a SS/NS *as soon as* (3.6)/(3.7) is satisfied, ultimately a SS/NS has to be done.

Assumption 1 *In Step 4, if $\mathcal{S}'' = \mathcal{K}$ and at least one between* (3.6) *and* (3.7) *hold true, then either a SS or a NS (whatever appropriate) is indeed made.*

Lemma 3 (sequence of minor iterations) *Under Assumption 1 the algorithm performs at most $3\,|\mathcal{K}|$ minor iterations in any major iteration.*

Proof. In view of Assumption 1 we just have to prove that, after at most $3\,|\mathcal{K}|$ minor iterations, $\mathcal{S}'' = \mathcal{K}$ and at least one among (3.6) or (3.7) holds. At each minor iteration one $k \in \mathcal{K}$ is selected; it is obvious (cf. Lemma 3.1) that the first time k moves (at worst) into $\mathcal{S}$, the second into $\mathcal{S}'$ (and therefore $\mathcal{Z}$), the third into $\mathcal{S}''$. Thus, after at most 3 minor iterations with the same k, Step 3.3 necessarily yields

$$\bar f_{+,\ell}^k - \underline{f}_{+,\ell}^k \leq \beta_\ell^k\varepsilon_\ell \ . \tag{35}$$

Hence, at length (3.12) holds for all $k \in \mathcal{S}'' = \mathcal{Z} = \mathcal{K}$, giving

$$\bar f_{+,\ell} - \underline{f}_{+,\ell} = \textstyle\sum_{k\in\mathcal{K}}(\,\bar f_{+,\ell}^k - \underline{f}_{+,\ell}^k\,) \leq \sum_{k\in\mathcal{K}}\beta_\ell^k\varepsilon_\ell = \varepsilon_\ell \ .$$

If neither (3.6) nor (3.7) holds true, then, using $\underline{f}^{\mathcal{Z}} = \underline{f}^{\mathcal{K}} = \underline{f}$, one has

$$\begin{aligned}\bar f_{+,\ell} - \underline{f}_{+,\ell}^{\mathcal{Z}} &> \overline{\texttt{tar}}_\ell - \underline{f}_{+,\ell}^{\mathcal{Z}} - m_1\Delta_{*,\ell} = (\,\bar f_\ell + v_{*,\ell}\,) - m_2 v_{*,\ell} - (\,\bar f_\ell^{\mathcal{Z}} + v_{*,\ell}^{\mathcal{Z}}\,) - m_1\Delta_{*,\ell}\\ &= -m_2 v_{*,\ell} - m_1\Delta_{*,\ell} = \varepsilon_\ell,\end{aligned}$$

readily yielding a contradiction and thereby concluding the proof. ∎

It may be useful to remark again (cf. §3.3.6) that $\mathcal{S}'' = \mathcal{K}$ is only the "worst case scenario": it is entirely possible that a SS/NS be declared if $\mathcal{Z} = \mathcal{S}' = \emptyset$, and even if $\mathcal{S} \subset \mathcal{K}$, i.e., only a (small) subset of the components have been evaluated.

The analysis now follows well-established guidelines. With $\delta_1 = \delta_2 = 0$, we prove:

1. in an infinite sequence of Serious Steps, $\|z_{*,\ell}\| \to 0$ and $\alpha_{*,\ell} \to 0$, which means that the corresponding sequence $\{\,\bar x_\ell\,\}$ is a minimizing one;
2. in an infinite sequence of consecutive Null Steps, $\|z_{*,\ell}\| \to 0$ and $\alpha_{*,\ell} \to 0$, which means that the (fixed) corresponding stability center $\bar x$ is optimal.

Hence, with $\delta_1 > 0$ and $\delta_2 > 0$, the algorithm finitely terminates.

We start from the first point: let $\mathcal{L}_{\texttt{SS}}$ be the index set of SS, and assume $|\mathcal{L}_{\texttt{SS}}| = \infty$. Rules for "appropriate" handling of $\mathcal{P}^k$ and t at Step 5 are now required.

Assumption 2 (Upper model management) *(i) For all ℓ and for $k \in \mathcal{K}$, let $(\theta_{*,\ell}^k, s_{*,\ell}^k)$ be the optimal solution to* (2.6) *with $x = \bar x_\ell$ having produced the value $\bar f_\ell^k$, which has been computed either in Step 0 or in Step 3.3 (in the latter case, either when computing $\hat f_{\mathcal{P}}^k(x_\ell = \bar x_\ell)$, if a SS has been done at $\ell - 1$, or when computing $\hat f_{\mathcal{P}}^k(\bar x_{\ell-1})$, if a NS has been done at $\ell - 1$). Then, either all the $(x_i, \bar f_i^k) \in \mathcal{P}_{\ell-1}^k$ such that $\theta_{i,*,\ell}^k > 0$ also belong to $\mathcal{P}_\ell^k$, or $(\,x_*^k\,,\,f_*^k\,)$ of* (3.10) *belongs to $\mathcal{P}_\ell^k$.*

(ii) if $|\mathcal{L}_{\texttt{SS}}| = \infty$, then there exists a $\bar t > 0$ such that $t_\ell \leq \bar t < \infty$ for all $\ell \in \mathcal{L}_{\texttt{SS}}$, and $\sum_{\ell\in\mathcal{L}_{\texttt{SS}}} t_\ell = \infty$.

Assumption 2(i) is required to ensure that $\bar f_\ell \geq \bar f_{\ell+1}$ always holds: (3.7) works for a SS, and the assumption ensures that the optimal solution of (2.6) at iteration ℓ always remains feasible (albeit, if (3.10) has been used, in a "surrogate form") at iteration $\ell + 1$, so that the optimal value cannot increase.

As discussed in §3.3.8, $\bar{f}_\ell$ can actually *decrease* when a NS is performed; however, this just means that it is *a fortiori* nonincreasing. Assumption 2(ii) is written in an abstract form, since it is impossible to know beforehand if $|\mathcal{L}_{\mathtt{SS}}| = \infty$; however, it is easy to define mechanisms that ensure that it holds (for instance, ensure $0 < \underline{t} \leq t_\ell \leq \bar{t} < \infty$ for all ℓ).

Theorem 4 (infinite sequence of SS) *Under Assumptions 1 and 2, if* $|\mathcal{L}_{\mathtt{SS}}| = \infty$ *then* $\lim_{\mathcal{L}_{\mathtt{SS}} \ni \ell \to \infty} \bar{f}_\ell = \liminf_{\mathcal{L}_{\mathtt{SS}} \ni \ell \to \infty} f(\bar{x}_\ell) = \nu(1.1)$. *In addition, if* (1.1) *admits any optimal solution then the sequence* $\{\, \bar{x}_\ell \,\}$ *converges to one.*

Proof. Assumption 1 ensures that eventually either a SS or a NS is performed. Assumption 2(i) guarantees that the sequence $\{\, \bar{f}_\ell \,\}$ is nonincreasing, and therefore it has a limit $\bar{f}_\infty$. If $\bar{f}_\infty = -\infty$, then $\nu(1.1) \leq \lim_{\mathcal{L}_{\mathtt{SS}} \ni \ell \to \infty} f(\bar{x}_\ell) \leq \bar{f}_\infty = -\infty$, i.e., $\{\, \bar{x}_\ell \,\}_{\ell \in \mathcal{L}_{\mathtt{SS}}}$ is a minimizing sequence (which cannot converge to an optimal solution to (1.1) since there is none). For the case where $\bar{f}_\infty > -\infty$, since (3.6) holds at SS

$$\begin{aligned} &\bar{f}_{\ell+1} \leq \bar{f}_\ell + (1-m_2)v_{*,\ell} \leq \bar{f}_\ell - (1-m_2)\big(\, \Delta_{*,\ell} = t_\ell ||z_{*,\ell}||^2/2 + \alpha_{*,\ell} \,\big) \quad \Longrightarrow \\ &\quad \textstyle\sum_{\ell \in \mathcal{L}_{\mathtt{SS}}} \big(\, t_\ell ||z_{*,\ell}||^2/2 + \alpha_{*,\ell} \,\big) < \infty \;\; \Longrightarrow \;\; t_\ell \, \|z_{*,\ell}\|^2 \to 0 \;\; \text{ and } \;\; \alpha_{*,\ell} \to 0. \end{aligned} \tag{36}$$

Using (3.13) together with (2.19) and the fact that $\bar{f}_\ell - f(\bar{x}_\ell) \geq \bar{f}_\ell - \check{f}_{\mathcal{B}}(\bar{x}_\ell)$ gives $\lim_{\mathcal{L}_{\mathtt{SS}} \ni \ell \to \infty} \bar{f}_\ell - f(\bar{x}_\ell) = 0$: asymptotically, the upper estimate becomes tight. Hence, $\lim_{\mathcal{L}_{\mathtt{SS}} \ni \ell \to \infty} \bar{f}_\ell = \liminf_{\mathcal{L}_{\mathtt{SS}} \ni \ell \to \infty} f(\bar{x}_\ell)$. The rest follows from [6]: for $\ell \in \mathcal{L}_{\mathtt{SS}}$, one is performing a step of t_ℓ along $z_{*,\ell} \in \partial_{\alpha_{*,\ell}} f(\bar{x}_\ell)$ (cf. (2.17)). Then, $\lim_{\mathcal{L}_{\mathtt{SS}} \ni \ell \to \infty} \bar{f}_\ell = \nu(1.1)$ comes from [6, Proposition 1.2] (in particular, Assumption 2(ii) is [6, (1.4)]). Furthermore, we can reproduce the more detailed analysis of [6, §4], (3.13) being [6, (4.16)]; the argument in [6, Theorem 4.4] gives the second part of the assertion. ■

We now turn to the case $|\mathcal{L}_{\mathtt{SS}}| < \infty$: the "last SS" $\bar{\ell}_{\mathtt{SS}}$ occurs, after which only NS are performed, which means that $\bar{x}$ is fixed. We of course need something akin to Assumption 2 for $\mathcal{B}$:

Assumption 3 (Lower bundle management) *(i) For all* $\ell > \bar{\ell}_{\mathtt{SS}}$ *and* $k \in \mathcal{K}$, *let* $\theta^k_{*,\ell}$ *be the optimal solution to* (2.14) *(computed in Step 2). Then, either all the* $(z^k_i, \alpha^k_i) \in \mathcal{B}^k_\ell$ *such that* $\theta^k_{i,*,\ell} > 0$ *also belong to* $\mathcal{B}^k_{\ell+1}$, *or* $(\, z^k_*\,,\, \alpha^k_* \,) \in \mathcal{B}^k_{\ell+1}$ *(cf.* (3.9)*).*

(ii) There exists $\ell' \geq \bar{\ell}_{\mathtt{SS}}$ *such that* t_ℓ *is nonincreasing for all* $\ell \geq \ell'$.

Assumption 3 is again written in an abstract form, in that it only has to hold *within the last infinite sequence of consecutive NS* (if any). Since it is impossible to say if $\ell > \bar{\ell}_{\mathtt{SS}}$, the conditions have to hold, at length, within *any* sequence of *consecutive* NS. Yet, as soon as a SS is performed, each $\mathcal{B}^k$ can be entirely reset: hence, Assumption 3(i) is weaker than Assumption 2(i). Assumption 3(ii) allows on-line tuning of t_ℓ, which is well-known to be crucial in practice: it is not necessarily true that "the best" $t_{\ell+1}$ after a NS is smaller than t_ℓ (e.g., [1]). Yet, the combined effect of (i) and (ii) is that, during a sequence of consecutive NS, at length the values $\nu(2.14)$ are nonincreasing. Indeed, the previous optimal solution of (2.14) remains feasible (albeit, if (3.9) has been used, in a "surrogate form"), so that $\nu(2.14)$ cannot increase, since t is also nonincreasing. The rest of analysis in fact all hinges on the following technical Lemma, which estimates how much the insertion of the new pairs $(z^k_\ell, \alpha^k_\ell)$ for $k \in \mathcal{Z}$ changes $\nu(2.14)$. To simplify the notation, in the Lemma we drop the iteration index ℓ ($\geq \ell' \geq \bar{\ell}_{\mathtt{SS}}$), and we denote with (2.14_+) and t_+ ($\leq t$), respectively, the master problem and the stabilization parameter at iteration $\ell + 1$.

Lemma 5 *Under Assumption 3,*

$$\nu(2.14) - \nu(2.14_+) \geq \frac{\Delta f^{\mathcal{Z}}}{2} \min \left\{ \, 1 \,,\, \frac{\Delta f^{\mathcal{Z}}}{t_+ ||z^{\mathcal{Z}}_* - z^{\mathcal{Z}}||^2} \, \right\} \;, \tag{37}$$

where $\Delta f^{\mathcal{Z}} = \sum_{k \in \mathcal{Z}} (\, \underline{f}^k_+ - (\bar{f}^k + v^k_*) \,)$ *(cf.* (3.7)*) and we intend* $1/0 = \infty$.

Proof. We define the "minimal, aggregated" form of (2.14_+) as

$$\min \left\{ \, \tfrac{1}{2} t \left\| z^{-\mathcal{Z}}_* + (1-\theta) z^{\mathcal{Z}}_* + \theta z^{\mathcal{Z}} \right\|^2 + \alpha^{-\mathcal{Z}}_* + (1-\theta)\alpha^{\mathcal{Z}}_* + \theta \alpha^{\mathcal{Z}} \;:\; \theta \in [0,1] \, \right\} \;. \tag{38}$$

Clearly, (3.15) is a *restriction* of (2.14_+) even under the minimal assumption that $\mathcal{B}^k = \mathcal{B}^k_* \cup \{(z^k, \alpha^k)\}$ (cf. (3.9)). Indeed, (3.15) is the dual of (2.7) using the aggregated model (2.3) (cf. (3.19))—as opposed to the sum of individual models (2.4) as in (2.12)—with

$$\mathcal{B} = \left\{ \, (z_*, \alpha_*) \,,\, (\bar{z}^{\mathcal{Z}}, \bar{\alpha}^{\mathcal{Z}}) \, = \, \big(\, z^{\mathcal{Z}} + z^{-\mathcal{Z}}_* \,,\, \alpha^{\mathcal{Z}} + \alpha^{-\mathcal{Z}}_* \,\big) \, \right\} \;. \tag{39}$$

In other words, (3.15) is a restriction of (2.14_+)—with "minimal" dual bundle—where $\theta^k = \theta$ for all $k \in \mathcal{K}$. Whichever way it is looked at, it is clear that, under Assumption 3, $\nu(3.15) \geq \nu(2.14_+)$ holds (using $t_+ \leq t$). Hence, we want to estimate

$$\nu(2.14) - \nu(2.14_+) \geq \zeta = \nu(2.14) - \nu(3.15) \geq 0 \ ,$$

where the last inequality comes from the fact that (z_*, α_*) is optimal for (2.14), $\theta = 0$ is feasible in (3.15) and produces the same solution. Estimating ζ requires simple but tedious algebra, starting with

$$\begin{aligned}\zeta =& \tfrac{t}{2}\|z_*\|^2 + \alpha_* - \min_{\theta\in[0,1]} \left\{ \tfrac{t}{2}\left\|z_*^{-\mathcal{Z}} + (1-\theta)z_*^{\mathcal{Z}} + \theta z^{\mathcal{Z}}\right\|^2 + \alpha_*^{-\mathcal{Z}} + (1-\theta)\alpha_*^{\mathcal{Z}} + \theta\alpha^{\mathcal{Z}} \right\} \\ =& \max_{\theta\in[0,1]} \left\{ \tfrac{t}{2}\|z_*\|^2 + \alpha_* - \tfrac{t}{2}\left\|z_*^{-\mathcal{Z}} + (1-\theta)z_*^{\mathcal{Z}} + \theta z^{\mathcal{Z}}\right\|^2 - \alpha_*^{-\mathcal{Z}} - (1-\theta)\alpha_*^{\mathcal{Z}} - \theta\alpha^{\mathcal{Z}} \right\}\end{aligned}$$

and then using $z_* = z_*^{-\mathcal{Z}} + z_*^{\mathcal{Z}}$, $\alpha_* = \alpha_*^{-\mathcal{Z}} + \alpha_*^{\mathcal{Z}}$ to rewrite the objective function as

$$\begin{aligned}&\tfrac{t}{2}\left\|z_*^{-\mathcal{Z}} + z_*^{\mathcal{Z}}\right\|^2 - \tfrac{t}{2}\left\|z_*^{-\mathcal{Z}} + (1-\theta)z_*^{\mathcal{Z}} + \theta z^{\mathcal{Z}}\right\|^2 + \theta(\alpha_*^{\mathcal{Z}} - \alpha^{\mathcal{Z}}) = \\ &\quad = \tfrac{t}{2}\Big[||z_*^{-\mathcal{Z}}||^2 + 2z_*^{-\mathcal{Z}}z_*^{\mathcal{Z}} + ||z_*^{\mathcal{Z}}||^2 - ||z_*^{-\mathcal{Z}}||^2 - (1-\theta)^2||z_*^{\mathcal{Z}}||^2 - \theta^2||z^{\mathcal{Z}}||^2 \\ &\qquad\quad - 2(1-\theta)z_*^{-\mathcal{Z}}z_*^{\mathcal{Z}} - 2\theta z_*^{-\mathcal{Z}}z^{\mathcal{Z}} - 2(1-\theta)\theta z_*^{\mathcal{Z}}z^{\mathcal{Z}} \Big] + \theta(\alpha_*^{\mathcal{Z}} - \alpha^{\mathcal{Z}}) \\ &\quad = -\tfrac{t}{2}\theta^2||z_*^{\mathcal{Z}} - z^{\mathcal{Z}}||^2 + \theta\Big[t(z_*^{\mathcal{Z}} + z_*^{-\mathcal{Z}})z_*^{\mathcal{Z}} - t(z_*^{-\mathcal{Z}} + z_*^{\mathcal{Z}})z^{\mathcal{Z}} + \alpha_*^{\mathcal{Z}} - \alpha^{\mathcal{Z}} \Big] \\ &\quad = -\tfrac{t}{2}\theta^2||z_*^{\mathcal{Z}} - z^{\mathcal{Z}}||^2 + \theta\big(-d_*z_*^{\mathcal{Z}} + d_*z^{\mathcal{Z}} + \alpha_*^{\mathcal{Z}} - \alpha^{\mathcal{Z}} \big),\end{aligned}$$

where in the last step we have used $d_* = -t(z_* = z_*^{\mathcal{Z}} + z_*^{-\mathcal{Z}})$. By summing (3.3) and (2.18) over $k \in \mathcal{Z}$ we obtain, respectively, $\underline{f}_+^{\mathcal{Z}} - \bar{f}^{\mathcal{Z}} = z^{\mathcal{Z}}d_* - \alpha^{\mathcal{Z}}$ and $d_*z_*^{\mathcal{Z}} - \alpha_*^{\mathcal{Z}} = v_*^{\mathcal{Z}}$. Plugging these in the last line of the above derivation we finally conclude

$$\zeta \geq \max\left\{ h(\theta) = \theta\Delta f^{\mathcal{Z}} - \tfrac{1}{2}\theta^2 M^{\mathcal{Z}} \ : \ \theta \in [0,1] \right\} \ , \tag{40}$$

where $M^{\mathcal{Z}} = t||z_*^{\mathcal{Z}} - z^{\mathcal{Z}}||^2 \, (\geq 0)$, and $\Delta f^{\mathcal{Z}} = \underline{f}_+^{\mathcal{Z}} - (\, \bar{f}^{\mathcal{Z}} + v_*^{\mathcal{Z}} \,) (\geq 0)$. Obviously, $\nu(3.17) \geq 0$ ($\theta = 0$ is feasible), showing that $\zeta \geq 0$. Since $h'(\theta) = \Delta f^{\mathcal{Z}} - M^{\mathcal{Z}}\theta$, if $M^{\mathcal{Z}} = 0 \equiv z^{\mathcal{Z}} = z_*^{\mathcal{Z}}$ then h is linear and $h' = \Delta f^{\mathcal{Z}} \geq 0$: hence, the optimal solution to (3.17) is $\theta_* = 1$. Otherwise, the unconstrained maximum of h is $\bar{\theta} = \Delta f^{\mathcal{Z}}/M^{\mathcal{Z}} \geq 0$. If $\bar{\theta} \leq 1$ then $\theta_* = \bar{\theta}$, with $h(\theta_*) = h(\bar{\theta}) = (\Delta f^{\mathcal{Z}})^2/(2M^{\mathcal{Z}})$. If, instead, $\bar{\theta} > 1$, then again $\theta_* = 1$, and therefore $h(\theta_*) = h(1) = \Delta f^{\mathcal{Z}} - M^{\mathcal{Z}}/2$. Hence,

$$\begin{aligned}&\Delta f^{\mathcal{Z}} \leq M^{\mathcal{Z}} \quad \Longrightarrow && && \zeta \geq (\Delta f^{\mathcal{Z}})^2/(2M^{\mathcal{Z}}) \\ &\Delta f^{\mathcal{Z}} > M^{\mathcal{Z}} \quad \Longrightarrow \quad h(1) = \Delta f^{\mathcal{Z}} - \tfrac{1}{2}M^{\mathcal{Z}} > \tfrac{1}{2}\Delta f^{\mathcal{Z}} \quad \Longrightarrow && && \zeta \geq \tfrac{1}{2}\Delta f^{\mathcal{S}}.\end{aligned}$$

Furthermore $\Delta f^{\mathcal{Z}} \leq M^{\mathcal{Z}} \ \equiv \ \Delta f^{\mathcal{Z}}/M^{\mathcal{Z}} \leq 1 \ \equiv \ (\Delta f^{\mathcal{Z}})^2/(2M^{\mathcal{Z}}) \leq \frac{1}{2}\Delta f^{\mathcal{Z}}$, which finally allows us to conclude that

$$\zeta \geq \nu(3.17) \geq \min\left\{ \frac{\Delta f^{\mathcal{Z}}}{2} \ , \ \frac{(\Delta f^{\mathcal{Z}})^2}{2M^{\mathcal{Z}}} \right\} = \frac{\Delta f^{\mathcal{Z}}}{2}\min\left\{ 1 \ , \ \frac{\Delta f^{\mathcal{Z}}}{t||z_*^{\mathcal{Z}} - z^{\mathcal{Z}}||^2} \right\} \ ,$$

i.e., (3.14) holds. The formula also works when $z_*^{\mathcal{Z}} = z^{\mathcal{Z}} \Longrightarrow \theta_* = 1 \Longrightarrow \zeta \geq \Delta f^{\mathcal{Z}}$ by considering $\min\{\, 1 \,, \, \Delta f^{\mathcal{Z}}/0 \,\} = \min\{\, 1 \,, \, \infty \,\} = 1$. ∎

It may be useful to remark again that (3.14) estimates the decrease of $\nu(2.14)$ under the hypothesis $\alpha_{*,\ell} = \alpha_{*,\ell+1}$, which may not be true because the α_i^k may decrease in Step 3.3; this, however, makes the estimate *a fortiori* valid. We are now ready to analyze the case of an infinite sequence of consecutive NS.

Theorem 6 (infinite sequence of consecutive NS) *Under Assumptions 1 and 3, if* $|\mathcal{L}_{\mathtt{SS}}| < \infty$ *then* $\|z_{*,\ell}\| \to 0$ *and* $\alpha_{*,\ell} \to 0$. *Therefore,* $\bar{x}$ *is optimal for* (1.1) *and* $\liminf_{\ell\to\infty} \bar{f}_\ell = \limsup_{\ell\to\infty} \underline{f}_\ell = f(\bar{x}) = \nu(1.1)$.

Proof. We claim that $\Delta_{*,\ell} = \nu(2.14_\ell) \to 0$ as $\ell \to 0$, which implies $\|z_{*,\ell}\| \to 0$ and $\alpha_{*,\ell} \to 0$. Once the claim is proven, the thesis easily follows: (2.17) proves that $0 \in \partial f(\bar{x})$, and $\liminf_{\ell\to\infty} \bar{f}_\ell = \limsup_{\ell\to\infty} \underline{f}_\ell = f(\bar{x}) = \nu(1.1)$—asymptotically, both the upper and the lower estimates become tight—is proven exactly as in Theorem 3.3.

By contradiction, assume $\Delta_{*,\ell} \geq \varepsilon > 0$ for all ℓ. Because NS are always performed, (3.7) always holds: $\Delta f_\ell^{\mathcal{Z}} \geq m_1\Delta_{*,\ell} \geq m_1\varepsilon > 0$. The hypotheses of Lemma 3.4 are satisfied, and therefore (3.14) holds for

all ℓ ($\geq \ell'$). By Lemma 2.1, $||z_i^k|| \leq L^k$; being $z_{*,\ell}^k$ a convex combination, $||z_{*,\ell}^k|| \leq L^k$ as well. Hence, $||z_{*,\ell}^{\mathcal{Z}} - z_\ell^{\mathcal{Z}}||$ is bounded above by some constant, and since $t_{\ell+1} \leq t_\ell$, t_ℓ is bounded above: hence, the denominator of the rightmost term in (3.14) is also bounded above. Therefore, since $\Delta f_\ell^{\mathcal{Z}} \geq m_1 \varepsilon > 0$, the whole rightmost term in (3.14) is bounded away from zero: hence, $\nu(2.14_{\ell+1}) < \nu(2.14_\ell) - \delta$ for some $\delta > 0$. But this means that $\nu(2.14_\ell) \to -\infty$ as $\ell \to \infty$, which contradicts $\nu(2.14) \geq 0$, thereby proving the claim and concluding the proof. ∎

For future reference we remark that if it ever happens that $z_{*,\ell} = 0$, then the algorithm provides a correct *a-posteriori* estimate of the error: $\bar{x}_\ell$ is, then, $\alpha_{*,\ell}$-optimal. This is specific of our setting, and due to the fact that the oracles produce upper estimates, whose gap with the lower ones is "incorporated" in the α_i^k. All in all:

Theorem 7 (Overall Convergence) *Under Assumptions 1, 2 and 3, $\{\bar{x}_\ell\}$ is a minimizing sequence which converges to an optimal solution of problem* (1.1)*, if any.*

3.5 The (partly) aggregated case

Using a *disaggregated approach*—with model (2.4), and therefore master problems (2.12)/(2.14)—is natural in our setting. Besides, this has been reported over and over again (e.g., [3, 19, 23]) to be the most efficient approach, in view of the much faster convergence speed. Yet, the downside is the (much, if $|\mathcal{K}|$ is large) larger size of the master problems and therefore their computational cost, that can easily dominate the overall running time (e.g., [19]). In some cases [34], it might therefore be beneficial to rather use the *aggregated approach*, in which $\mathcal{B} = \{(z_i = \sum_{k\in\mathcal{K}} z_i^k, \alpha_i = \sum_{k\in\mathcal{K}} \alpha_i^k)\}$, and the *aggregated master problems*

$$\min\left\{ v + \tfrac{1}{2t}||d||^2 \ : \ v \geq z_i d - \alpha_i \quad i \in \mathcal{B} \right\} \tag{41}$$

$$\min\ \left\{ \tfrac{1}{2} t \left\|\textstyle\sum_{i\in\mathcal{B}} z_i\theta_i\right\|^2 + \textstyle\sum_{i\in\mathcal{B}} \alpha_i\theta_i \ : \ \theta \in \Theta \right\} \tag{42}$$

are solved instead of (2.12)/(2.14). The fact that one has to sum over *all* $k \in \mathcal{K}$ seems to fly squarely in the face of an incremental method, where at each iteration one is trying to compute as few components as possible.

Yet, in Step 0 sufficient information for computing the aggregated (z_1, α_1) is indeed generated, so at least the algorithm can be started. In order to keep it going, the following strategy can be used: although (3.18)/(3.19) are solved, *the disaggregated representation of each z_i and α_i in terms of their individual components z_i^k and α_i^k are kept.* This allows to reconstruct the individual v_*^k, z_*^k and α_*^k by just using the unique solution θ_* of (2.14) uniformly for all $k \in \mathcal{K}$. On the upper side disaggregated bundles $\mathcal{P}^k$ has to be keept, but this can be inexpensive; for the "poorman's" (3.10) with $x_* = \bar{x}_\ell$, it only amounts to keeping the separate values $\bar{f}^k$ together with the aggregated one $\bar{f}$. Once this is done, the algorithm remains the same: all the necessary information is available. In particular, the SS condition (3.6) only depends on the aggregated v_*; for Theorem 3.3, (2.12) and (3.18) are indistinguishable. The individual $\bar{f}^k$ and v_*^k are required to form the targets (3.5) and therefore the accuracy ε_ℓ^k of each $k \in \mathcal{K}$, but as soon as (3.6) is obtained—possibly with $\mathcal{S} \subsetneq \mathcal{K}$, which is the aim of the whole exercise—a SS can be performed. Forming a unique (z_ℓ, α_ℓ) even if (3.7) is triggered with $\mathcal{Z} \subsetneq \mathcal{K}$—which is the aim of the whole exercise—is also easy: one just has to use $(\bar{z}^{\mathcal{Z}}, \bar{\alpha}^{\mathcal{Z}})$ of (3.16). Indeed, this is all that is needed to form the "approximated" master problem (3.15), which is the crux of Lemma 3.4 providing the crucial estimate (3.14), and therefore Theorem 3.5 still works (with the obvious modification to Assumption 3(i)).

Keeping the disaggregated representation (z_i^k, α_i^k) for all $k \in \mathcal{K}$ of each (z_i, α_i) may come at a considerable memory cost, and the computational cost of forming the individual v_*^k, z_*^k and α_*^k may be non negligible, too. This can actually be avoided by an alternative approach that, to the best of our knowledge, has been never discussed before: insert *partly aggregated cuts/variables*

$$\textstyle\sum_{k\in\mathcal{Z}} v^k \geq z_i^{\mathcal{Z}} d - \alpha_i^{\mathcal{Z}} \tag{43}$$

in the primal/dual disaggregated master problems (2.12)/(2.14). It is easy to see that using (3.20) does not impair the fundamental property of the master problem, i.e., that $v_*^k \leq f^k(\bar{x} + d_*)$: indeed, (3.20) is the *surrogate constraint* of the $|\mathcal{Z}|$ constraints $v^k \geq z_i^k d - \alpha_i^k$, and therefore (2.12) with (3.20) is a relaxation of (2.12) with these. In the dual, the variable $\theta_i^{\mathcal{Z}}$ associated to $(z_i^{\mathcal{Z}}, \alpha_i^{\mathcal{Z}})$ participates to the simplex constraints for all (and only) the $k \in \mathcal{Z}$. Actually, nothing prevents arbitrarily partitioning $\mathcal{Z} =$

$\mathcal{Z}_1 \cup \mathcal{Z}_2 \cup \ldots \cup \mathcal{Z}_p$, and inserting the corresponding p partly aggregated cuts in (2.14): $p = |\mathcal{Z}|$ reproduces the disaggregated approach. Hence, the trade-off between master problem size/cost and convergence speed can be explored, ranging from fully aggregated models, to fully disaggregated ones, to "anything in between". Choosing the best aggregation level can only be done computationally; alternatively, along the lines of [26, 30] one may consider versions wherein the partition automatically adapts. To the best of our knowledge, neither has ever been done before; although these approaches seem interesting, this article is not the appropriate venue to pursue them.

3.6 The constrained case

We now discuss dealing with constraints $x \in X$ in (1.1). Actually, in the spirit of [19] we will treat a more general case: that where f has components $\mathcal{K} = \{ 0 \} \cup \mathcal{K}''$, where f^0 is "easy" in the sense that it can be *effectively written into the master problem*. This covers different situations, such as:

1. $f^0 = \mathbb{1}_X$ with X represented by "few, simple" (say, conic) constraints;
2. $f^0(x) = bx$, with b the RHS of the relaxed constraints in (1.4);
3. $f^0 = \nu(1.3^0_x)$ where U^0 can be represented with "few, simple" constraints [19].

In all these cases (and the combinations thereof), the natural way to deal with f^0 is just to insert it unmodified in the master problem, which then becomes

$$\min \left\{ \; f^0(\bar{x} + d) + \textstyle\sum_{k \in \mathcal{K}} v^k + \frac{1}{2t}||d||^2 \; : \; v^k \geq z_i^k d - \alpha_i^k \quad i \in \mathcal{B}^k \quad , \quad k \in \mathcal{K} \right\} \; . \tag{44}$$

For instance, when $f^0 = \mathbb{1}_X$ this amounts to adding the constraints "$\bar{x} + d \in X$" to (2.12), which ensure that $x_+ = \bar{x} + d_*$ is feasible. The underlying assumption is that (3.21) is still efficiently solvable, which of course depends on the specific form of f^0; however, there are plenty of cases where this happens. The setting obviously extends to the case where $\mathcal{K} = \mathcal{K}' \cup \mathcal{K}''$ with $\mathcal{K}'$ the set of "easy components", but for simplicity of notation we will stick to $\mathcal{K}' = \{ 0 \}$. The *dual* of (3.21) is (see e.g., [18])

$$\begin{cases} \min \;\; \frac{1}{2}t \left\| z^0 + \sum_{k \in \mathcal{K}} \sum_{i \in \mathcal{B}^k} z_i^k \theta_i^k \right\|^2 + \sum_{k \in \mathcal{K}} \sum_{i \in \mathcal{B}^k} \alpha_i^k \theta_i^k - \bar{x} z^0 + (f^0)^*(z^0) \\ \text{s.t.} \quad \theta^k \in \Theta^k \quad k \in \mathcal{K} \end{cases} \tag{45}$$

where $(f^0)^*(z^0) = \sup_x \{ \, xz^0 - f^0(x) \, \}$ is *Fenchel's conjugate* of f^0 (e.g., [25]). It is possible to give a more specific description to (3.22) depending on the exact form of f^0, but this is not necessary for our development. Indeed, it is not difficult to check that, by employing (3.21)/(3.22) in place of (2.12)/(2.14) in Step 1 of the algorithm, "everything just works" with the few modifications necessary to adapt to the fact that the "model" for f^0 is not a cutting-plane one but the original function:

- The function f^0 in (3.21) is not translated in value, because it is in general difficult to translate a generic function—unlike the cutting-plane model(s), where this is simply achieved by changing the linearization errors. Accordingly, the global predicted descent now is $v_* = f^0(\bar{x} + d_*) - f^0(\bar{x}) + \sum_{k \in \mathcal{K}} v_*^k$, where $f^0(\bar{x} + d_*)$ is *exactly* known as a by-product of solving (3.21). There is a minor difficulty for $\ell = 1$ as $f^0(\bar{x}_1)$ is not known, but this is not hard to circumvent: just assume $f^0(\bar{x}_1) = \infty$, forcing a SS at the first iteration.
- At each iteration one initializes $\mathcal{S} := \mathcal{Z} := \mathcal{S}' := \mathcal{S}'' := \{ 0 \}$ $(= \mathcal{K}')$, as there is no need to call any oracle for f^0: $\bar{f}^0_+ = \bar{f}^0 + v^0_*$, with $v^0_* = f^0(\bar{x} + d_*) - (\, \bar{f}^0 = f^0(\bar{x}) \,)$, is easily computed from the solution of (3.21) whatever the exact form of f^0 is. There is no need to ever compute (2.6) for $k = 0$, as the exact value of f^0 is always known; note that for $f^0 = \mathbb{1}_X$ the value is always 0. Analogously, there is no need to compute any z^0 for f^0, as there is no $\mathcal{B}^0$ to be filled: the full description of f^0 is already present in (3.21), although in a different form.
- The individual targets (3.5) need not be defined for f^0, since $k = 0$ will never happen. Yet, f^0 does contribute in setting them for $k \in \mathcal{K}''$ (cf. the rightmost terms in the minimum/maximum of (3.5)). Note, however, that while f^0 may contribute positively to $\overline{\texttt{tar}}^k$ (or not: as for any other f^k, there is no guarantee that $v^0_* < 0$), it surely does not contribute to $\underline{\texttt{tar}}^k$, in that $\bar{f}^0 + v^0_* = \underline{f}^0_+$. In other words, while f^0 may contribute to a SS—it may be so even with $\mathcal{S} = \{ 0 \}$, i.e., on the grounds of the 0-component and the upper models alone, without having called any oracle—it

will never contribute to (3.7) being satisfied. In fact, a NS is aimed at accruing more information to improve the lower models (and therefore d_*), but there is no way—and need—to improve the model of f^0. Note that for $f^0 = \mathbb{1}_X$ one always has $\bar{f}^0 = 0$, hence the 0-th component will not contribute to (3.6) either (but it will enforce feasibility of the iterates).

- Solving the (3.21) means also solving its dual (3.22), which produces a z^0_*, albeit not as a convex combination of z^0_i (typically, as dual optimal solutions of some constraints [18]). Similarly, it is possible to define a correct α^0_*. However, there is no use for (z^0_*, α^0_*): it does not have to be added to $\mathcal{B}^0$, and v^0_* is not computed through it. Yet, one has to be aware that the constant "$+f^0(\bar{x})$" has to be added to $\nu(3.22)$ in order for $\Delta^k = \nu(3.22) + f^0(\bar{x})$ to behave precisely as $\nu(2.14)$ ($\Delta^k = 0 \iff \bar{x}$ optimal).

Ultimately, it is easy to see that the convergence analysis of §3.4 still applies. This also extends, mutatis mutandis, to the case of §3.5: the (partly) aggregated master problems (3.18)/(3.19) can be modified similarly to (3.21)/(3.22) with the explicit term for f^0. That is, the $k \in \mathcal{K}''$ can be aggregated while f^0 remains disaggregated from them. This can have a surprisingly large effect on the convergence speed [19].

4 Uncooperative oracles and noise reduction

The treatment so far has hinged on a "gentleman agreement" between the algorithm and the oracle: while the former always tries to ask as little accuracy as possible, providing the oracle with all possible clues about when it is possible to stop the computation without having attained an exact solution, the latter obliges itself to find a solution as accurate as required. Some oracles may find it very time consuming, or even impossible, to do that. It is therefore important to consider the case of an *uncooperative oracle* that may not be capable, or willing, to satisfy (2.2). Hence, $\overline{\texttt{tar}}^k$, $\underline{\texttt{tar}}^k$ and ε^k cease to be *orders* and become *suggestions*: the oracle can stop as soon as (2.2) is satisfied, but it may elect to stop sooner. We will analyze two types of uncooperative oracles, corresponding to different assumptions on the solution approaches available for (1.3^k_x).

Common to both cases is the fact that bundle methods have a simple way to deal with uncooperative oracles: exploiting the fact that t is, in fact, almost a "free parameter". Hence, if $\mathcal{O}$ is uncooperative at x_+, one may just try to generate a different iterate by changing t, and hope that the oracle will, for whatever reason, be cooperative there. Any such mechanism requires some safeguard to ensure that one is not sampling X forever with the oracles stubbornly refusing to provide any valuable information. As we shall see, the safeguard is that, eventually, the stability parameter grows.

4.1 Faithful uncooperative oracles

The first kind of oracle closely mirrors those encountered in the literature, accounting for our more complex interface: a *faithful boundedly uncooperative* oracle, when called with $\varepsilon^k < \infty$, returns $-\infty < \underline{f}^k \le f^k(x) \le \bar{f}^k < \infty$ and z^k such that for a *fixed*—but possibly *unknown*—$\bar{\varepsilon}^k < \infty$

$$f^k(x) - \underline{f}^k \le \bar{f}^k - \underline{f}^k \le \bar{\varepsilon}^k \quad \equiv \quad z^k \in \partial_{\bar{\varepsilon}^k} f^k(x) \ . \tag{46}$$

An uncooperative oracle may not be able to satisfy (2.2.iii), in particular if $\overline{\texttt{tar}}^k - \underline{\texttt{tar}}^k < \bar{\varepsilon}^k$; yet, it is "faithful" in that, at least, the lower and upper estimates are indeed so. This is the case of solution methods for (1.3^k_x) that can provide upper and lower bounds but cannot (or are not willing to) guarantee that these will be arbitrarily close; examples are approximation methods with worst-case guarantees, or exact methods with a tight bound on the computational resources. Obviously, $\mathcal{O}^k$ for all $k \in \mathcal{K}$ satisfying (4.1) are an $\mathcal{O}$ for f satisfying (4.1) with $\bar{\varepsilon} = \sum_{k\in\mathcal{K}} \bar{\varepsilon}^k = \bar{\varepsilon}^{\mathcal{K}} < \infty$.

The issue with (4.1) is that Lemma 3.2 ceases working: even with $\mathcal{S}'' = \mathcal{K}$, it is not guaranteed that one among (3.6) and (3.7) holds. This is why an "escape clause" is needed if the worst comes to the worst, which is a *Noise Reduction* (NR) step:

Step 3.0 (Noise Reduction) `if` $-\mathcal{S}'' = \emptyset$ `then`, `if` $||z_{*,\ell}||^2 \le \delta_1$ `then stop`, `else` change t_ℓ according to Assumption 4 and `go to` Step 1.

The assumption on how t_ℓ is changed is basically that it grows. More specifically, this only applies to infinite sequences of NR steps, and only if no SS happens between them. That is, with $\mathcal{L}_{\texttt{NR}}$ indicating

the set of NR iterations:

Assumption 4 *If* $|\mathcal{L}_{\texttt{NR}}| = \infty$ *and* $|\mathcal{L}_{\texttt{SS}}| < \infty$*, then* $\liminf_{\ell \ni \mathcal{L}_{\texttt{NR}} \to \infty} t_\ell = \infty$.

The statement of Assumption 4 is again abstract. In practice, some mechanism is required that increases t_ℓ (albeit not necessarily monotonically) when NR are performed, but it is reset as soon as a SS is performed [10, (6.14)]. This avoids conflicts between Assumption 4 and Assumption 2(ii), since t_ℓ needs to be bounded above for $\ell \in \mathcal{L}_{\texttt{SS}}$. A "large" t_ℓ means that (cf. (2.15)) either $v_{*,\ell}$ also grows large (negative), or $||z_{*,\ell}||^2$ goes to 0, which allows to detect approximate optimality.

Lemma 8 *If all oracles* $\mathcal{O}^k$ *are boundedly uncooperative, Assumption 4 holds,* $|\mathcal{L}_{\texttt{SS}}| < \infty$ *and* $|\mathcal{L}_{\texttt{NR}}| = \infty$*, then* $\limsup_{\ell \ni \mathcal{L}_{\texttt{NR}} \to \infty} ||z_{*,\ell}||^2 = 0$.

Proof. Since $\ell \in \mathcal{L}_{\texttt{NR}}$ both (3.6) and (3.7)—with $\mathcal{Z} = \mathcal{K}$—fail: subtracting their opposites and using (4.1), $\Delta_{*,\ell} \leq -v_{*,\ell}$ and $v_{*,\ell} \leq -t_\ell ||z_{*,\ell}||^2$ (cf. (2.15)), yields

$$\infty > \bar{\varepsilon} \geq \bar{f}_{\ell+1} - \underline{f}_{\ell+1} \geq (m_2 - m_1)(-v_{*,\ell}) \geq (m_2 - m_1) t_\ell ||z_{*,\ell}||^2 \ . \tag{47}$$

Then, $||z_{*,\ell}||^2 \geq \delta > 0$ for all $\ell \in \mathcal{L}_{\texttt{NR}}$ would contradict Assumption 4. ∎

The Lemma illustrates the moniker "noise reduction". The oracle frames the "signal" $f(x_{\ell+1})$ between the two measures $\bar{f}_{\ell+1}$ and $\underline{f}_{\ell+1}$, affected by "noise" up to $\bar{\varepsilon}$. If $\underline{\texttt{tar}}^k_\ell$ and $\overline{\texttt{tar}}^k_\ell$ are closer than $\bar{\varepsilon}$, then the noise is too large to distinguish which of the two is satisfied. Increasing t_ℓ either increases $-v_{*,\ell}$, and therefore pulls further apart the targets, or decreases $||z_{*,\ell}||^2$: hence, either the signal becomes significant w.r.t. the noise, or one detects that $\bar{x}_\ell$ is already "as optimal as it gets". Indeed, with $\delta_1 > 0$, $|\mathcal{L}_{\texttt{NR}}| < \infty$: by Lemma 4.1, either ultimately one among (3.6) and (3.7) holds, or the algorithm stops with $||z_{*,\ell}||^2 \leq \delta_1$, proving that $\bar{x}_\ell$ is *approximately* $\alpha_{*,\ell}$-optimal (cf. (2.17)). As in (4.3), $\ell \in \mathcal{L}_{\texttt{NR}}$ implies $\bar{\varepsilon} \geq \bar{f}_{\ell+1} - \underline{f}_{\ell+1} \geq (m_2 - m_1)(-v_{*,\ell})$, i.e.,

$$\alpha_{*,\ell} \leq -v_{*,\ell} \leq \bar{\varepsilon}/(m_2 - m_1) =: \varepsilon' \ . \tag{48}$$

Hence, $\bar{x}$ is (approximately) optimal with (about) the best error that can be obtained given the oracle [7, Observation 2.7]: when $m_2 \approx 1$ and $m_1 \approx 0$, $\varepsilon' \approx \bar{\varepsilon}$. When $\delta_1 = 0$, $|\mathcal{L}_{\texttt{NR}}| = \infty$ can happen. This is not an issue if $|\mathcal{L}_{\texttt{SS}}| = \infty$ as well: Assumption 2(ii) and Assumption 4 have been carefully crafted to be compatible, hence Theorem 3.3 still applies (in fact, it does not depend in any way from what happens between two consecutive SS). Note that this means that $\{\bar{x}_\ell\}_{\ell \in \mathcal{L}_{\texttt{NR}}}$ is an *exact optimizing sequence*, with *no error*; this is unlikely to happen in general, and therefore one should expect that, at length, only NS and NR are done, in any order. Here an issue arises because Assumptions 3(ii) and 4 conflict. The simple solution, however, is to give priority to the latter: over sequences of NR, t_ℓ ultimately has to grow even if NS are made. This in principle only applies if $|\mathcal{L}_{\texttt{NR}}| = \infty$, but since one never knows whether or not this is happening, some mechanism has to ensure that if NR are done and SS aren't, t_ℓ eventually grows. Note that, however, there is no need to completely inhibit decrease of t_ℓ at NS as in [10, (6.14)], provided that it ultimately grows. With this expedient:

Theorem 9 *Under the assumptions of Lemma 4.1,* $\liminf_{\ell \to \infty} \alpha_{*,\ell} = \alpha_{*,\infty} \leq \varepsilon'$ *and the stability center* $\bar{x}$ *is* $\alpha_{*,\infty}$*-optimal.*

Proof. Due to Lemma 4.1, $||z_{*,\ell}||^2 \to 0$; hence, $0 \in \partial_{\alpha_{*,\infty}} f(\bar{x})$ by (2.17). For the bound on $\alpha_{*,\infty}$ just note that $\alpha_{*,\ell} \geq \alpha_{*,\infty}$ for ℓ is large enough and use (4.3). ∎

The last case is that $|\mathcal{L}_{\texttt{SS}}| < \infty$ and $|\mathcal{L}_{\texttt{NR}}| < \infty$, after which only *consecutive* NS are done: here Theorem 3.5 directly applies, which means that $\bar{x}$ is optimal—again, with *no error*. All in all, the algorithm always *asymptotically* attains ε'-optimality. With $\delta_1 > 0$, *approximate* ε'-optimality is *finitely* attained. Interestingly, with a specific twist in the management of $\mathcal{B}$ akin to the "safe β-strategy" of [15, §4.3], *infinite* sequences of NR actually *finitely* attain ε'-optimality.

Assumption 5 *If* $|\mathcal{L}_{\texttt{NR}}| = \infty$*, then at length insertions/removals of pairs in/from all the* $\mathcal{B}^k$ *are inhibited.*

Assumption 5 contrasts with Assumption 3(ii), but the solution is the same as for Assumption 4: give priority to NR, i.e., ensure that during a sequence of NS and NR, eventually the $\mathcal{B}^k$ are kept fixed.

Because NS are based on insertion of new information, this basically means entirely inhibiting NS at length: each time (3.7) is satisfied (and (3.6) is not), a NR is performed instead of the NS. Again, the mechanism can be reset each time a SS is performed.

Corollary 10 *Under Assumptions 4 and 5, if $|\mathcal{L}_{\mathtt{NR}}| = \infty$, then at lenght $x_{\ell+1}$ is a global minimizer of $\check{f}^k_{\mathcal{B}}$: hence, $\bar{x}$ is ε'-optimal and $x_{\ell+1}$ is $(m_2\varepsilon')$-optimal.*

Proof. Due to Assumption 5, at length all the $\mathcal{B}^k$ are fixed and we can refer to the lower model as $\check{f}_{\mathcal{B}}$, independently of ℓ. It is well-known that $z_{*,\ell} \in \partial\check{f}_{\mathcal{B}}(x_{\ell+1})$: [15, (2.3)] or [10, Lemma 4.1]. Since $\check{f}_{\mathcal{B}}$ is a polyhedral function, $S_{\mathcal{B}} = \{\, \partial\check{f}_{\mathcal{B}}(x) \,:\, x \in X \,\}$ is a *finite* set [21, Corollary VI.4.3.2]. Clearly, $S_{\mathcal{B}} = (\, S_0 = \{\, Z \in S_{\mathcal{B}} \,:\, 0 \in Z \,\}) \cup (\, S_+ = \{\, Z \in S_{\mathcal{B}} \,:\, 0 \notin Z \,\})$. Defining $s_Z = \min\{\, \|z\| \,:\, z \in Z \,\}$, one has $s_Z > 0 \iff Z \in S_+$ since subdifferentials of finite-valued functions are closed compact convex sets. Since $S_{\mathcal{B}}$ itself is finite, $s = \min\{\, s_Z \,:\, Z \in S_+ \,\} > 0$. From Lemma 4.1, $||z_{*,\ell}|| < s$ for all large enough ℓ: hence, $\partial\check{f}_{\mathcal{B}}(x_{\ell+1}) \in S_0$, i.e., $0 \in \partial\check{f}_{\mathcal{B}}(x_{\ell+1})$ as desired.

Since $\check{f}_{\mathcal{B}}(x_{\ell+1}) = \bar{f} + v_{*,\ell}$, the linearization error of the null subgradient $0 \in \partial\check{f}_{\mathcal{B}}(x_{\ell+1})$ at $\bar{x}$ is $-v_{*,\ell}$: hence, $\bar{x}$ is ε'-optimal (cf. (2.9) and (4.3)). Similarly, the linearization error of 0 at $x_{\ell+1}$ is $\bar{f}_+ - \check{f}_{\mathcal{B}}(x_+) = \bar{f}_+ - \bar{f} - v_{*,\ell}$ (cf. (2.10)), for which

$$\begin{aligned} & \bar{f}_+ - \check{f}_{\mathcal{B}}(x_{\ell+1}) \le \bar{\varepsilon} + \underline{f}_+ - \check{f}_{\mathcal{B}}(x_{\ell+1}) < \bar{\varepsilon} + \check{f}_{\mathcal{B}}(x_{\ell+1}) + m_1\Delta_{*,\ell} - \check{f}_{\mathcal{B}}(x_{\ell+1}) \\ = \;\; & \bar{\varepsilon} + m_1\Delta_{*,\ell} \le \bar{\varepsilon} + m_1(-v_{*,\ell}) \le \bar{\varepsilon} + m_1\bar{\varepsilon}/(m_2 - m_1) = m_2\bar{\varepsilon}/(m_2 - m_1) \;\; , \end{aligned}$$

where in the first step we have used (4.1), in the second that (3.7) does not hold, in the third $\Delta_{*,\ell} \le -v_{*,\ell}$, and in the fourth (4.3). ∎

Corollary 4.3 suggests the following modification of Step 3.0 which, to the best of our knowledge, has never been discussed before:

Step 3.0′ (Noise Reduction) `if` $-\mathcal{S}'' = \emptyset$ `then`, `if` $0 \in \partial\check{f}_{\mathcal{B}}(x_{\ell+1})$ `then stop`, `else` change t_ℓ according to Assumption 4 and `go to` Step 1.

Checking if $0 \in \partial\check{f}_{\mathcal{B}}(x_{\ell+1})$ can be done e.g., with sensitivity analysis [14, §7], and at worst requires solving a linear feasibility problem surely not more expensive than the master problems. If the algorithm stops, then both $x_{\ell+1}$ and $\bar{x}_\ell$ are guaranteed to be (at least) ε'-optimal (under Assumption 5). Actually, from the proof of Corollary 4.3 we have that $\bar{x}_\ell$ is $(-v_{*,\ell})$-optimal and $x_{\ell+1}$ is $(\bar{f}_{+,\ell} - \bar{f}_\ell - v_{*,\ell})$-optimal: the a-posteriori bounds may be (much) tighter than the a-priori ones. This could, for instance, be used to choose which among $x_{\ell+1}$ and $\bar{x}_\ell$ is a more accurate solution. As already remarked, the availability of explicit a-posteriori error bounds is a characteristic of our development, and it directly descends on the choice of having explicit upper bounds (and using them to define the linearization errors).

4.2 Cheating uncooperative oracles

We now discuss *cheating* oracles, which do satisfy (2.2), but possibly by reporting incorrect data. We keep assuming that $\underline{f}^k$ and z^k satisfy (2.1), i.e., the oracle is *lower* in the parlance of [10]. Using non-lower oracles is also possible, as thoroughly analyzed in [10], but we will limit ourselves to the case where only the upper estimates may be incorrect, i.e., $\bar{f}^k < f^k(x)$ may happen. This, for (1.3^k_x), may be the case of heuristics without any known bound on the accuracy, or of oracles subject to such strict resource limits that do not allow them to produce any upper bound. In our context, this has the difficulty that the only *available correct* upper estimate, ∞, does not satisfy (2.2) whenever $\varepsilon^k < \infty$; in other words, *no significant upper model* $f^k_{\mathcal{P}}$ can be defined if $\bar{f}^k = \infty$. Hence, the oracle has to "cheat" and report finite upper estimates that it does not really have available. An obvious way of doing that is systematically reporting $\bar{f}^k = \underline{f}^k$; as we shall see this is actually, in some sense, the best possible approach, and we define *uniformly cheating* an oracle doing just that. We of course also require (4.1) to hold. We remark that guaranteeing this for (1.3^k_x) is not entirely trivial; a sufficient condition is, besides compactness of all U^k, also compactness of X, which is not common in applications (although, technically, compactness of $\{x_\ell\}$ would suffice). Anyway, once ensured it implies $f^k \le \check{f}^k_{\mathcal{P}} + \bar{\varepsilon}^k$: via (2.8), this gives

$$f^k(\bar{x}) - [\, \underline{f}^k_i + z^k_i(\bar{x} - x_i) \,] \le \bar{f}^k + \bar{\varepsilon}^k - [\, \underline{f}^k_i + z^k_i(\bar{x} - x_i) \,] = \alpha^k_i + \bar{\varepsilon}^k \tag{49}$$

$$z^k_i \in \partial_{(\alpha^k_i + \bar{\varepsilon}^k)} f^k(\bar{x}) \;\; , \;\; \alpha^k_i \ge -\bar{\varepsilon}^k \;\; , \;\; z_{*,\ell} \in \partial_{(\alpha_{*,\ell} + \bar{\varepsilon})} f(\bar{x}) \;\; , \;\; \alpha_{*,\ell} \ge -\bar{\varepsilon} \tag{50}$$

(with $\bar{\varepsilon} = \bar{\varepsilon}^{\mathcal{K}}$). Since the $\bar{\varepsilon}^k$ are not known, the quality of the z_i^k cannot be directly assessed, but at least the error is bounded. Furthermore, the advantage of uniformly cheating oracles $\bar{f}^k = \underline{f}^k$ is that *they obviously never trigger a NR Step 3.0*: whatever the returned value, one of the two targets is surely met, and $\bar{f}^k - \underline{f}^k = 0 \leq \varepsilon^k$. Of course, inexactness has to crop up some other way; in particular, it may happen that $\check{f}^k_{\mathcal{P}}(\bar{x}_\ell) < \hat{f}^k_{\mathcal{P}}(\bar{x}_\ell)$, i.e., $\alpha_{*,\ell} < 0$. This may ultimately lead to $v_{*,\ell} = -t_\ell ||z_{*,\ell}||^2 - \alpha_{*,\ell} > 0$, basically destroying all convergence arguments.

The first, natural reaction would be to "correct" the upper model: if $\bar{f}^k_{\mathcal{P}}(\bar{x}_\ell) < \underline{f}^k_{\mathcal{B}}(\bar{x}_\ell)$, then set $\bar{f}^k_{\mathcal{P}}(\bar{x}_\ell) := \underline{f}^k_{\mathcal{B}}(\bar{x}_\ell)$, and therefore $\alpha_i^k := \alpha_i^k + \underline{f}^k_{\mathcal{B}}(\bar{x}_\ell) - \bar{f}^k_{\mathcal{P}}(\bar{x}_\ell)$ (> 0) for all $i \in \mathcal{B}^k$. This, however, creates problems in the convergence proof, which relies on the fact that $\bar{f}_\ell$ is nonincreasing. Furthermore, the corrected $\bar{f}_\ell$ is still not, in general, a valid upper bound, so little would have been gained. Indeed, because $\bar{\varepsilon}$ is unknown, one is still left with the worst-case bound (4.5), whereas, having increased α_i^k, one could (in theory) reduce the other term (if it were known). Said otherwise, artificially imposing $\alpha_i^k \geq 0$ cancels all traces of the oracles' wrongdoing, preventing the algorithm from taking the appropriate countermeasures. As in §4.1, these amount to ensuring that the descent $v_{*,\ell}$ is "significant" w.r.t. the (unknown) error $\bar{\varepsilon}$ of the problem, to avoid taking decisions about function values that are, basically, undistinguishable given the oracle inaccuracy. To ensure that $v_{*,\ell} = -t_\ell ||z_{*,\ell}||^2 - \alpha_{*,\ell} < 0$, since $-t_\ell ||z_{*,\ell}||^2 < 0$ it is enough to guarantee that $\alpha_{*,\ell}$ is "not too negative":

$$\alpha_{*,\ell} \geq -m_3 t_\ell ||z_{*,\ell}||^2 \tag{51}$$

with $m_3 \in (0,1)$, which immediately implies

$$v_{*,\ell} \leq (1 - m_3)\big(- t_\ell ||z_{*,\ell}||^2 \big) \ \ (< 0) \ . \tag{52}$$

Remarkably, this is similar but different from the condition used in the literature [10, (5.1)] [33], which is stricter because it requires $m_3 \leq 1/2$, whereas we will argue that $m_3 \approx 1$ is preferable. It has been observed in [33] that m_3 influences the definition of predicted decrease; however, our analysis makes it apparent—due to the fact that, in our setting, (3.6) and (3.7) are not mutually exclusive—that the effect is actually on the NS condition. Thus, we add

Step 1.1 (Noise reduction) `if` (4.6) fails `then`, `if` $||z_{*,\ell}||^2 \leq \delta_1$ `then stop`, `else` change t_ℓ according to Assumption 4 and `go to` Step 1.

at the end of Step 1. Clearly, the two forms of NR are quite different. Step 3.0 reacts to the fact that the oracles, in the *current* iterate ℓ, have (already) produced correct but not accurate enough information. Step 1.1 rather reacts to the fact that the oracles, in some iterate *prior to* ℓ, have produced incorrect information which makes $x_{\ell+1}$ not a significant point to even call the oracle upon. With uniformly cheating oracles, Step 3.0 will never be executed. Yet, Step 1.1 allows to prove convergence similarly to §3.4 and §4.1 for the three cases: an infinite sequence of SS, an infinite sequence of NR, an infinite tail of consecutive NS.

Proposition 11 *Under the assumptions of Theorem 3.3,* $\bar{f}_\infty = \lim_{\mathcal{L}_{\mathtt{SS}} \ni \ell \to \infty} \bar{f}_\ell = \limsup_{\mathcal{L}_{\mathtt{SS}} \ni \ell \to \infty} \check{f}_{\mathcal{B}}(\bar{x}_\ell) \leq \nu(1.1) + \bar{\varepsilon}$.

Proof. Reasoning as in Theorem 3.3 and using (3.13) yields $v_{*,\ell} \to 0$. Since $\ell \in \mathcal{L}_{\mathtt{SS}}$ one has that (4.7) holds, which gives (barring the case $\bar{f}_\infty = -\infty$) that $||z_{*,\ell}|| \to 0$; then, (4.6) gives $\alpha_{*,\ell} \to 0$, even if $\alpha_{*,\ell} < 0$. This proves the statement about the limits being equal. Hence, if the sequence $\{ \bar{x}_\ell \}_{\ell \in \mathcal{L}_{\mathtt{SS}}}$ has a cluster point $\bar{x}_\infty$, (4.5) gives $0 \in \partial_{\bar{\varepsilon}} f(\bar{x}_\infty)$, i.e., $\bar{x}_\infty$ is $\bar{\varepsilon}$-optimal. The proof can be generalized without requiring $\{ \bar{x}_\ell \} \to \bar{x}_\infty$ by using e.g. [10, Proposition 6.1], albeit at the cost of requiring t_ℓ to be bounded away from zero (cf. the discussion of [10, Theorem 6.2]). ∎

For $|\mathcal{L}_{\mathtt{NR}}| = \infty$, the fact that (4.6) does *not* hold for $\ell \in \mathcal{L}_{\mathtt{NR}}$ gives

$$v_{*,\ell} = -t_\ell \left\| z_{*,\ell} \right\|^2 - \alpha_{*,\ell} > \tfrac{(1-m_3)}{m_3} \alpha_{*,\ell} \geq \tfrac{(1-m_3)}{m_3} (-\bar{\varepsilon}) \ . \tag{53}$$

If the algorithm stops in Step 1.1, then $\bar{x}$ is approximately $(\alpha_{*,\ell} + \bar{\varepsilon})$-optimal (note that $\alpha_{*,\ell} + \bar{\varepsilon} \geq 0$ due to (4.5)): because $\alpha_{*,\ell} < 0$ for all $\ell \in \mathcal{L}_{\mathtt{NR}}$, it is therefore *a fortiori* $\bar{\varepsilon}$-optimal. The equivalent of Lemma 4.1/Theorem 4.2 also hold:

Proposition 12 *If all oracles* $\mathcal{O}^k$ *are uniformly cheating, Assumption 4 holds,* $|\mathcal{L}_{\mathtt{SS}}| < \infty$ *and* $|\mathcal{L}_{\mathtt{NR}}| = \infty$, *then* $\limsup_{\ell \ni \mathcal{L}_{\mathtt{NR}} \to \infty} ||z_{*,\ell}||^2 = 0$.

Proof. Use (4.5) and (2.15) to obtain $v_{*,\ell} \leq -t_\ell \|z_{*,\ell}\|^2 + \bar{\varepsilon}$, then use the opposite of (4.7) to derive $\bar{\varepsilon} > m_3 t_\ell ||z_{*,\ell}||^2$: $||z_{*,\ell}||^2 \geq \delta > 0$ contradicts $t_\ell \to \infty$. ∎

Proposition 13 *Under the assumptions of Proposition 4.5, $\bar{x}$ is $\bar{\varepsilon}$-optimal.*

Proof. Proposition 4.5 gives $||z_{*,\ell}|| \to 0$: hence $0 \in \partial_{(\alpha_{*,\infty}+\bar{\varepsilon})} f(\bar{x})$. Therefore $\bar{x}$ is $(\alpha_{*,\infty} + \bar{\varepsilon})$-optimal, and since $\alpha_{*,\ell} < 0$ for all $\ell \in \mathcal{L}_{\texttt{NR}}$, $\alpha_{*,\infty} \leq 0$. ∎

Remark 14 *The above proof relies on $\mathcal{O}$ to be uniformly cheating: $\alpha_{*,\ell} < 0$ is true because only Step 1.1 triggers a NR. If $\mathcal{O}$ is not uniformly cheating, Step 3.0 may happen, so $\alpha_{*,\ell} < 0$ might not be true. Using* (4.3)*, which holds when $t_\ell \to \infty$, would give $0 \in \partial_{2\bar{\varepsilon}} f(\bar{x})$, i.e., the error would be twice of that of the oracle. This can actually happen, as illustrated by the following example. Let $f(x) = |x|$ and $\bar{x}_1 = x_1 = -2$. A (uniformly) cheating oracle with $\bar{\varepsilon} = 1$ may return $\underline{f}_1 = \bar{f}_1 = 1 < f(x_1) = 2$ and $z_1 = -1$. If $t = 1$, $x_2 = -1$, $v_{*,1} = -1$, and $\check{f}_{\mathcal{B}_1}(\bar{x}_2) = 0$. A non-uniform oracle may now work "in reverse" on x_2, returning $\underline{f}_2 = 0$, $z_2 = 0$, and $\bar{f}_2 = 1 = f(x_2)$. Neither a SS ($\bar{f}_1 = \bar{f}_2$) nor a NS ($\underline{f}_2 = \check{f}_{\mathcal{B}_1}(\bar{x}_2)$) can be performed, so a NR Step 3.0 is triggered. However, with any $t_\ell > 1$, $x_2 = -1$ is a minimum of $\check{f}_{\mathcal{B}_2}$, hence $x_\ell = x_2$ for all ℓ may happen: the algorithm might never be able to improve $\bar{x}_1$, which has error $f(\bar{x}_1) - \nu(1.1) = 2 - 0 = 2 = 2\bar{\varepsilon}$. Note that if the minimum of f had been < 0, the oracle could have not reported $z_2 = 0$ together with $\underline{f}_2 = 0 = \check{f}_{\mathcal{B}}(\bar{x}_2)$: $z_2 < 0$ would have been required, and therefore increasing t_ℓ would have ultimately triggered a SS.*

Corollary 4.3 can be similarly extended: this requires Assumption 5.

Proposition 15 *Under the assumptions of Proposition 4.5 and additionally Assumption 5, $\bar{x}$ is $(\bar{\varepsilon}/m_3)$-optimal and $x_{\ell+1}$ is $((1 - m_1 + m_1/m_3)\bar{\varepsilon})$-optimal.*

Proof. The first part of Corollary 4.3, unchanged, yields that, eventually, $0 \in \partial \check{f}_{\mathcal{B}}(x_{\ell+1})$, with $\check{f}_{\mathcal{B}}(x_{\ell+1}) = \bar{f}_\ell + v_{*,\ell} \leq \nu(1.1)$. Hence, $0 \in \partial \check{f}_{\bar{\varepsilon}-v_{*,\ell}}(\bar{x}_\ell)$ which, using (4.8), gives that $\bar{x}_\ell$ is $(\bar{\varepsilon}/m_3)$-optimal. For $x_{\ell+1}$,

$$\begin{aligned} & f(x_{\ell+1}) - \check{f}_{\mathcal{B}}(x_{\ell+1}) \leq \bar{\varepsilon} + \underline{f}_+ - \check{f}_{\mathcal{B}}(x_{\ell+1}) < \bar{\varepsilon} + \check{f}_{\mathcal{B}}(x_{\ell+1}) + m_1\Delta_{*,\ell} - \check{f}_{\mathcal{B}}(x_{\ell+1}) \\ = \quad & \bar{\varepsilon} + m_1\Delta_{*,\ell} \leq \bar{\varepsilon} + m_1(-v_{*,\ell}) \leq \bar{\varepsilon} + m_1(1-m_3)/m_3\bar{\varepsilon} = (1 - m_1 + m_1/m_3)\bar{\varepsilon} \ , \end{aligned}$$

where the second inequality is that (3.7) does not hold, and the last one (4.8). Note that $(1-m_1)m_3+m_1 < 1$, hence $x_{\ell+1}$ has a better a-priori error bound than $\bar{x}$. ∎

Therefore, once again $|\mathcal{L}_{\texttt{NR}}| = \infty$ asymptotically proves $\bar{\varepsilon}$-optimality of $\bar{x}$: Step 1.1 finitely prove approximate optimality with $\delta_2 > 0$, and with $\delta_2 = 0$ one can use

Step 1.1′ (Noise reduction) `if` (4.6) fails `then`, `if` $0 \in \partial \check{f}_{\mathcal{B}}(x_{\ell+1})$ `then stop`, `else` change t_ℓ according to Assumption 4 and `go to` Step 1.

to finitely attain $(\bar{\varepsilon}/m_3)$-optimality. The case of the infinite tail of consecutive NS is slightly more complex, because a modification to the algorithm is needed: one has to *replace $\Delta_{*,\ell}$ with $-v_{*,\ell}$* in the definition of ε_ℓ, $\underline{\texttt{tar}}^k_\ell$ and in (3.7), i.e.,

$$\varepsilon_\ell := (m_2 - m_1)(-v_{*,\ell}) \tag{54}$$

$$\underline{\texttt{tar}}^k_\ell := \min\left\{ (\bar{f}^k_\ell + v^k_{*,\ell}) - m_1\beta^k_\ell v_{*,\ell} \ , \ (\bar{f}^{\mathcal{Z}}_\ell + v^{\mathcal{Z}}_{*,\ell}) - m_1 v_{*,\ell} - \underline{f}^{\mathcal{Z}\setminus\{k\}}_{+,\ell} \right\} \tag{55}$$

$$\Delta f^{\mathcal{Z}} := \textstyle\sum_{k\in\mathcal{Z}} \Delta f^k \geq -m_1 v_{*,\ell} \qquad \equiv \qquad \underline{f}^{\mathcal{Z}}_{+,\ell} \geq (\bar{f}^{\mathcal{Z}}_\ell + v^{\mathcal{Z}}_{*,\ell}) - m_1 v_{*,\ell} \ . \tag{56}$$

The reason is that while (4.6) ensures that $-v_{*,\ell} > 0$ (cf. (4.7)), $\Delta_{*,\ell} < 0$ still can happen. This would lead to $\underline{\texttt{tar}}^k_\ell < \check{f}_{\mathcal{B}}(x_{\ell+1})$, and to (3.7) be always satisfied, which would break all the development. Since $-v_{*,\ell} \geq \Delta_{*,\ell}$, (4.11) is harder to satisfy than (3.7), which is why—in the tradition of the standard analysis of bundle methods—we prefer to keep the weaker condition if possible. However, cheating oracles require (4.11), at least with "large" m_3; restricting to $m_3 < 1/2$ as in [10, (5.1)] would allow using (3.7) instead. When (4.10)–(4.11) are used, it is necessary to define ε_ℓ by (4.9), for otherwise Lemma 3.1 and 3.2 would break, rendering all the rest useless. Then:

Proposition 16 *Using* (4.10)–(4.11)*, and under the Assumptions of Theorem 3.5 plus $|\mathcal{L}_{\texttt{NR}}| < \infty$, $\|z_{*,\ell}\| \to 0$ and $\alpha_{*,\ell} \to 0$: therefore, $\bar{x}$ is $\bar{\varepsilon}$-optimal.*

Proof. The crucial point is showing that $\limsup_{\ell\to\infty} v_{*,\ell} = 0$, so assume that $v_{*,\ell} \geq \varepsilon > 0$: then (4.11) would imply that $\Delta f_\ell^{\mathcal{Z}} \geq m_1 v_{*,\ell} \geq m_1\varepsilon > 0$, and hence $\nu(2.14_\ell) \to -\infty$ as $\ell \to \infty$, contradicting $\nu(2.14) \geq -\bar{\varepsilon} > -\infty$ (the latter a consequence of (4.5)). As in the SS case, $v_{*,\ell} \to 0$ gives $||z_{*,\ell}|| \to 0$ via (4.7), and then (4.6) implies $\alpha_{*,\ell} \to 0$: hence, $0 \in \partial_{\bar{\varepsilon}} f(\bar{x})$, i.e., $\bar{x}$ is $\bar{\varepsilon}$-optimal. ∎

While the above derivation closely mirrors that of §3.4 and §4.1, all the results only involve the a-priori (unknown) error $\bar{\varepsilon}$: no a-posteriori bounds are available. It could not be otherwise, since no valid explicit upper estimates for the f-values are available.

4.3 Arbitrary mixture of oracles

The previous section may have conveyed the idea that faithful and cheating oracles cannot be used together: fortunately, this is not true. Indeed, let $\mathcal{K} = \mathcal{F} \cup \mathcal{C}$ ($\mathcal{F} \cap \mathcal{C} = \emptyset$), with $\mathcal{F}$ the components having faithful oracles (cooperative oracles are faithful with $\bar{\varepsilon}^k = 0$), and $\mathcal{C}$ these having (uniformly) cheating ones. From (4.4) we have $z_{*,\ell} \in \partial_{(\alpha_{*,\ell}+\bar{\varepsilon}^{\mathcal{C}})} f(\bar{x})$ and $\nu(2.14) \geq \alpha_{*,\ell} \geq -\bar{\varepsilon}^{\mathcal{C}}$, whereas from the fact that all the cheating oracles are uniformly so we have $\bar{f}_{\ell+1} - \underline{f}_{\ell+1} \leq \bar{\varepsilon}^{\mathcal{F}}$. Because we now have both possible types of NR steps, $\mathcal{L}_{\mathtt{NR}} = \mathcal{L}_{\mathtt{NR}}^{\mathcal{F}} \cup \mathcal{L}_{\mathtt{NR}}^{\mathcal{C}}$, indicating respectively the NR of Step 3.0 ("for faithful oracles") and of Step 1.1 ("for cheating ones"); also, "NR" in Assumption 4 is now intended to mean "either kind of NR". Hence we can copy Lemma 4.1/Proposition 4.5:

Proposition 17 *Under Assumption 4, if $|\mathcal{L}_{\mathtt{SS}}| < \infty$ and $|\mathcal{L}_{\mathtt{NR}}| = \infty$, then $\limsup_{\ell\ni\mathcal{L}_{\mathtt{NR}}\to\infty} ||z_{*,\ell}||^2 = 0$.*

Proof. If $\ell \in \mathcal{L}_{\mathtt{NR}}^{\mathcal{F}}$ one can use $\bar{\varepsilon}^{\mathcal{F}} \geq \bar{f}_{\ell+1} - \underline{f}_{\ell+1}$ to get $\bar{\varepsilon}^{\mathcal{C}} > (m_2 - m_1)t_\ell ||z_{*,\ell}||^2$ as in Lemma 4.1, while if $\ell \in \mathcal{L}_{\mathtt{NR}}^{\mathcal{C}}$ one can use $\alpha_{*,\ell} \geq -\bar{\varepsilon}^{\mathcal{C}}$ to get $\bar{\varepsilon}^{\mathcal{C}} > m_3 t_\ell ||z_{*,\ell}||^2$ as in Proposition (4.1), if. Hence, in both cases, $||z_{*,\ell}||^2 \geq \delta > 0$ contradicts $t_\ell \to \infty$. ∎

Taking for simplicity $m_3 = m_2 - m_1$ and $\delta_2 = \bar{\varepsilon}(1 - m_3)/m_3$, we can now prove:

Theorem 18 (Convergence for arbitrary mixture of oracles) *Under Assumptions 1–5 and given any $\mathcal{F}$ and $\mathcal{C}$, the algorithm with Step 1.1 and using* (4.9)–(4.11) *(asymptotically) finds a(n approximated) $(\bar{\varepsilon}/m_3)$-optimal solution.*

Proof. If the algorithm finitely stops, this happens either in Step 1.1, or in Step 2, or in Step 3.0. In all cases, $\bar{x}$ is approximately $(\alpha_{*,\ell} + \bar{\varepsilon}^{\mathcal{C}})$-optimal, in the sense that $||z_{*,\ell}|| \leq \delta_1$. In the first case $\alpha_{*,\ell} < 0$ because $\ell \in \mathcal{L}_{\mathtt{NR}}^{\mathcal{C}}$; in the second, $\alpha_{*,\ell} \leq \delta_2$; in the third, $\alpha_{*,\ell} \leq \bar{\varepsilon}^{\mathcal{F}}$ because $\ell \in \mathcal{L}_{\mathtt{NR}}^{\mathcal{F}}$ (cf. (4.3)). Hence, $\bar{x}_\ell$ is always (approximately) $(\bar{\varepsilon}/m_3)$-optimal. Using Step 1.1′ and Step 3.0′, with Assumption 5 in force, Proposition 4.10 ensures $||z_{*,\ell}|| \to 0$: reasoning as in Corollary 4.3 gives that at length $0 \in \partial_{(\bar{\varepsilon}^{\mathcal{C}} - v_{*,\ell})} f(\bar{x}_\ell)$, which yields the same bound (and a slightly better one for $x_{\ell+1}$) by using (4.3)—with $\bar{\varepsilon}^{\mathcal{F}}$—if $\ell \in \mathcal{L}_{\mathtt{NR}}^{\mathcal{F}}$, and (4.8)—with $\bar{\varepsilon}^{\mathcal{C}}$—if $\ell \in \mathcal{L}_{\mathtt{NR}}^{\mathcal{C}}$ (cf. Proposition 4.8).

If $|\mathcal{L}_{\mathtt{SS}}| = \infty$, reasoning as in Proposition 4.4 proves that $\bar{f}_\infty \leq \nu(1.1) + \bar{\varepsilon}^{\mathcal{C}}$. Hence, we can assume the last SS to happen, after which only NR (of either type) or NS are performed: as in Proposition 4.10 $\bar{x}$ is $(\alpha_{*,\infty} + \bar{\varepsilon}^{\mathcal{C}})$-optimal. Now, if $\left|\mathcal{L}_{\mathtt{NR}}^{\mathcal{C}}\right| = \infty$ then $\alpha_{*,\infty} \leq 0$, otherwise $\alpha_{*,\infty} \leq \bar{\varepsilon}^{\mathcal{F}}/m_3$ as in Theorem 4.2; all in all, $\alpha_{*,\infty} + \bar{\varepsilon}^{\mathcal{C}} \leq \bar{\varepsilon}/m_3$. Finally, for the infinite tail of consecutive NS we can reason as in Proposition 4.9 to obtain that $\bar{x}$ is $\bar{\varepsilon}^{\mathcal{C}}$-optimal. ∎

If the cheating oracles are not uniformly so, we can no longer use the fact that $\bar{f}_{\ell+1} - \underline{f}_{\ell+1} \leq \bar{\varepsilon}^{\mathcal{F}}$; the bound is $\bar{f}_{\ell+1} - \underline{f}_{\ell+1} \leq \bar{\varepsilon}$. This bumps the worst-case error to $\bar{\varepsilon} + \bar{\varepsilon}^{\mathcal{F}} = \bar{\varepsilon}^{\mathcal{F}} + 2\bar{\varepsilon}^{\mathcal{C}}$: non-uniformly cheating oracles do "double damage". This is not entirely surprising, as they actually have "double the range" of both faithful and uniformly cheating ones: they can report $\bar{f}^k = f^k(x) + \bar{\varepsilon}^k$ (and, then, $\underline{f}^k = f^k(x)$) as well as $\underline{f}^k = f^k(x) - \bar{\varepsilon}^k$ (and, then, $\bar{f}^k = f^k(x)$), hence the range between the worst-case upper and lower estimate is indeed $2\bar{\varepsilon}^k$. An "adversarial" oracle can use it, as in Remark 4.7, to enforce an error of that magnitude. This allows us to conclude with a "morale": "better always cheating than being faithful only at times, at least the partner (algorithm) knows what to expect and can react accordingly".

5 Conclusions

In this paper we have analyzed a novel class of bundle methods for minimizing sum-structured convex nondifferentiable objective. Our starting point is to change the oracle definition so as to make apparent a feature that oracles often have in applications, but that has so far not been exploited: the fact that, besides lower estimates on the function value (and valid lower linearizations of the function epigraph), the oracles also provide *upper estimates*. Fully exploiting the latter requires a Lipschitz continuity assumption—even more, that the Lipschitz constant is actually *known*—which, however, is often satisfied in many important applications. The availability of this extra information has several notable impacts on the method:

– it allows to define *upper models* of the functions, which in turn can be used to derive upper estimates of function values even at points where some (or even all) of the oracles have not been called;

– it allows to provide both *upper and lower targets* to the oracles which, together with the required accuracy, can allow them to stop their computation just as soon as information that is "accurate enough" to allow the computation to proceed is obtained, possibly diminishing their cost;

– it it allows to skip oracle calls entirely for some of the component functions, not only at NS as in previous proposals in the literature, but also at SS;

– it provides explicit and reliable a-posteriori estimates of the quality of the obtained solutions, provided that the upper estimates are themselves reliable.

The method works with oracles that cannot attain arbitrary precision, as well as with those that do not provide reliable upper estimates, clearly losing some of the properties in the process. For oracle with nonzero (but bounded) maximum accuracy, one gets a solution with (about) the same error, which is the best that can be expected [7, Observation 2.7] and consistent with other methods in the literature. For oracles that do not provide reliable upper estimates, one loses the a-posteriori estimates of the quality of the obtained solutions. It is worth remarking that, in fact, reliable upper estimates provide something more: the fact that $\bar{f}_\ell$ is always a *reliable upper estimate on* $\nu(1.1)$, and *therefore on* $\nu(1.3)$. Computing tight upper estimates on $\nu(1.3)$ is very often the reason for solving (1.1) in the first place. For instance, in practical, large-scale industrial applications rarely problems are solved to optimality, and the decision maker can benefit from knowing whether or not it may be worthwhile to invest more to get better solutions. In general, if (1.1) is solved as a step of a more complex (e.g., implicitly enumerative) approach to (1.3), valid upper estimates are often crucial. All this provide a compelling argument against "erasing" the provided upper estimate and setting $\bar{f}^k = \underline{f}^k$, as (implicitly) advocated in the literature so far, thus making oracles systematically (but, at least, uniformly) cheat, although this ultimately yields the same bound. Whether or not exploiting upper estimates works better computationally can only be determined by an in-depth computational study, which is therefore the logical next step for this line of research.

Acknowledgements

The authors would like to acknowledge networking support by the COST Action TD1207, and financial support from the Gaspard-Monge program for Optimization and Operations Research (PGMO) project "Consistent Dual Signals and Optimal Primal Solutions".

References

[1] A. Astorino, A. Frangioni, A. Fuduli, and E. Gorgone. A nonmonotone proximal bundle method with (potentially) continuous step decisions. *SIAM Journal on Optimization*, 23(3):1784–1809, 2013.

[2] A. Astorino, A. Frangioni, M. Gaudioso, and E. Gorgone. Piecewise Quadratic Approximations in Convex Numerical Optimization. *SIAM Journal on Optimization*, 21(4):1418–1438, 2011.

[3] L. Bacaud, C. Lemaréchal, A. Renaud, and C. Sagastizábal. Bundle methods in stochastic optimal power management: a disaggregate approach using preconditionners. *Computation Optimization and Applications*, 20(3):227–244, 2001.

[4] O. Briant, C. Lemaréchal, Ph. Meurdesoif, S. Michel, N. Perrot, and F. Vanderbeck. Comparison of bundle and classical column generation. *Mathematical Programming*, 113(2):299–344, 2008.

[5] N. Chatzipanagiotis, D. Dentcheva, and M. M. Zavlanos. An augmented lagrangian method for distributed optimization. *Mathematical Programming*, 152(1):405–434, 2015.

[6] R. Correa and C. Lemaréchal. Convergence of some algorithms for convex minimization. *Mathematical Programming*, 62(2):261–275, 1993.

[7] G. d'Antonio and A. Frangioni. Convergence Analysis of Deflected Conditional Approximate Subgradient Methods. *SIAM Journal on Optimization*, 20(1):357–386, 2009.

[8] W. de Oliveira and J. Eckstein. A bundle method for exploiting additive structure in difficult optimization problems. *Optimization Online*, 2015.

[9] W. de Oliveira and C. Sagastizábal. Level bundle methods for oracles with on demand accuracy. *Optimization Methods and Software*, 29(6):1180–1209, 2014.

[10] W. de Oliveira, C. Sagastizábal, and C. Lemaréchal. Convex proximal bundle methods in depth: a unified analysis for inexact oracles. *Math. Prog. Series B*, 148:241–277, 2014.

[11] W. de Oliveira, C.A. Sagastizábal, and S. Scheimberg. Inexact bundle methods for two-stage stochastic programming. *SIAM Journal on Optimization*, 21(2):517–544, 2011.

[12] L. Dubost, R. Gonzalez, and C. Lemaréchal. A primal-proximal heuristic applied to french unitcommitment problem. *Mathematical Programming*, 104(1):129–151, 2005.

[13] G. Emiel and C. Sagastizábal. Incremental like bundle methods with applications to energy planning. *Computational Optimization and Applications*, 46(2):305–332, 2009.

[14] A. Frangioni. Solving Semidefinite Quadratic Problems Within Nonsmooth Optimization Algorithms. *Computers & Operations Research*, 21:1099–1118, 1996.

[15] A. Frangioni. Generalized bundle methods. *SIAM Journal on Optimization*, 13(1):117–156, 2002.

[16] A. Frangioni. About Lagrangian Methods in Integer Optimization. *Annals of Operations Research*, 139(1):163–193, 2005.

[17] A. Frangioni, C. Gentile, and F. Lacalandra. Solving Unit Commitment Problems with General Ramp Contraints. *International Journal of Electrical Power and Energy Systems*, 30:316–326, 2008.

[18] A. Frangioni and E. Gorgone. Generalized bundle methods for sum-functions with "easy" components: Applications to multicommodity network design. Technical Report 12-12, Dipartimento di Informatica, Università di Pisa, 2012.

[19] A. Frangioni and E. Gorgone. Generalized bundle methods for sum-functions with "easy" components: Applications to multicommodity network design. *Mathematical Programming*, 145(1):133–161, 2014.

[20] M. Gaudioso, G. Giallombardo, and G. Miglionico. An incremental method for solving convex finite min-max problems. *Math. of Oper. Res.*, 31, 2006.

[21] J.B. Hiriart-Urruty and C. Lemaréchal. *Convex Analysis and Minimization Algorithms I*. Number 305 in Grundlehren der mathematischen Wissenschaften. Springer-Verlag Berlin Heidelberg, 2nd edition, 1996.

[22] J.B. Hiriart-Urruty and C. Lemaréchal. *Convex Analysis and Minimization Algorithms II*. Number 306 in Grundlehren der mathematischen Wissenschaften. Springer-Verlag Berlin Heidelberg, 2nd edition, 1996.

[23] K.L. Jones, I.J. Lustig, J.M. Farwolden, and W.B. Powell. Multicommodity Network Flows: The Impact of Formulation on Decomposition. *Mathematical Programming*, 62:95–117, 1993.

[24] C. Lemaréchal. Lagrangian relaxation. In M. Jünger and D. Naddef, editors, *Computational Combinatorial Optimization: Optimal or Provably Near-Optimal Solutions*, volume 9 of *Lecture Notes in Computer Science*, pages 112–156. Springer-Verlag, 2001.

[25] C. Lemaréchal and A. Renaud. A geometric study of duality gaps, with applications. *Math. Programming*, 90:399–427, 2001.

[26] Y. Song and J. Luedtke. An adaptive partition-based approach for solving two-stage stochastic programs with fixed recourse. *SIAM Journal on Optimization*, 25(3):1344–1367, 2015.

[27] M. Tahanan, W. van Ackooij, A. Frangioni, and F. Lacalandra. Large-scale unit commitment under uncertainty: a literature survey. *4OR*, 13(2):115–171, 2015.

[28] W. van Ackooij, V. Berge, W. de Oliveira, and C. Sagastizábal. Probabilistic optimization via approximate p-efficient points and bundle methods. *Optimization Online*, 2015.

[29] W. van Ackooij and W. de Oliveira. Level bundle methods for constrained convex optimization with various oracles. *Computation Optimization and Applications*, 57(3):555–597, 2014.

[30] W. van Ackooij, W. de Oliveira, and Y. Song. An adaptive partition-based level decomposition for solving two-stage stochastic programs with fixed recourse. *Optimization Online*, 2016.

[31] W. van Ackooij, A. Frangioni, and W. de Oliveira. Inexact stabilized benders' decomposition approaches: with application to chance-constrained problems with finite support. *Computational Optimization And Applications*, To appear:1–24, 2016.

[32] W. van Ackooij and J. Malick. Decomposition algorithm for large-scale two-stage unit-commitment. *Annals of Operations Research*, 238(1):587–613, 2016.

[33] W. van Ackooij and C. Sagastizábal. Constrained bundle methods for upper inexact oracles with application to joint chance constrained energy problems. *SIAM Journal on Optimization*, 24(2):733–765, 2014.

[34] C. Wolf, C. I. Fábián, A. Koberstein, and L. Stuhl. Applying oracles of on-demand accuracy in two-stage stochastic programming a computational study. *European Journal of Operational Research*, 239(2):437–448, 2014.