Incremental-like Bundle Methods with Application to Energy Planning*

This report corrects and expands Sections 5 and 6 in DOI 10.1007/s10589-009-9288-8.

Grégory Emiel†

Claudia Sagastizábal‡

October 1, 2009

Abstract

An important field of application of non-smooth optimization refers to decomposition of large-scale or complex problems by Lagrangian duality. In this setting, the dual problem consists in maximizing a concave non-smooth function that is defined as the sum of sub-functions. The evaluation of each sub-function requires solving a specific optimization sub-problem, with specific computational complexity. Typically, some sub-functions are hard to evaluate, while others are practically straightforward. When applying a bundle method to maximize this type of dual functions, the computational burden of solving sub-problems is preponderant in the whole iterative process. We propose to take full advantage of such separable structure by making a dual bundle iteration after having evaluated only a subset of the dual sub-functions, instead of all of them. This type of incremental approach has already been applied for subgradient algorithms. In this work we use instead a specialized variant of bundle methods and show that such an approach is related to bundle methods with inexact linearizations. We analyze the convergence properties of two incremental-like bundle methods. We apply the incremental approach to a generation planning problem over an horizon of one to three years. This is a large scale stochastic program, unsolvable by a direct frontal approach. For a real-life application on the French power mix, we obtain encouraging numerical results, achieving a significant improvement in speed without

*The first author research was supported by a PhD grant from Electricité de France R&D, France. The work of the second author was partially supported by a research contract with EDF, CNPq Grant No. 303540-03/6, PRONEX-Optimization, and FAPERJ.

†Grégory Emiel: Instituto Nacional de Matemática Pura e Aplicada, Estrada Dona Castorina 110, Rio de Janeiro, 22460-320 – Brazil. Email: gemiel@impa.br.

‡Claudia Sagastizábal: CEPEL, Electric Energy Research Center, Eletrobrás Group. On leave from INRIA Rocquencourt, France. Email: sagastiz@impa.br.

losing accuracy.

Keywords: Non-smooth optimization, Generation planning, Incremental methods, Large scale stochastic programming

1 Introduction

We are interested in solving convex problems with separable objective functions in an efficient and accurate manner, by bundle methods. To ease the presentation, and without loss of generality, we consider the simpler case of minimizing the sum of only two functions:

$$\min_{x\in\mathbb{R}^N} (f^1 + f^2)(x)\,, \tag{1}$$

where f^1 and f^2 are lower semicontinuous convex functions. Suppose f^1 is easy to evaluate, while f^2 is very hard and time consuming. In this setting, it is desirable to design an algorithmic procedure that can "skip" the evaluation of sub-function f^2 at some iterations. To replace the lacking information, we consider using instead an approximate value, obtained from a cutting-planes model of f^2.

This type of incremental approach has already been applied for subgradient algorithms, [1]; see also [2]. For a function $f = \sum f^i$, these methods perform subgradient iterations sequentially, along the subgradients g^i of each sub-function f^i. Promising numerical results are reported in [2] and these methods are further studied from a theoretical point of view in [3]. For bundle methods, [4] studies the case of a non-smooth objective function defined as the pointwise maximum of a large number of non-smooth sub-functions, but to our knowledge there is so far no study of incremental bundle methods for the *sum* of sub-functions. A possible explanation could be that the bundle methodology does not seem to be straightforwardly customizable to the "sum" setting. However, recent results on inexact bundle methods from [5] (extending previous works [6], [7], [8], [9] and completed recently for constrained optimization by [10]) shed a new light on how to proceed to devise *incremental-like bundle methods* in our setting of interest.

More precisely, with respect to problem (1), inexactness appears when skipping the evaluation of the "difficult" sub-function, f^2. Indeed, replacing this sub-function by the corresponding current cutting-planes model can be interpreted as an inexact evaluation. Recent developments of bundle methods have shown that such algorithms may handle inexact information to achieve approximate optimality for non-smooth optimization programs. Inaccuracies may affect the evaluation of the objective function, or of the subgradient returned by the available oracle.

In [6, 7], the author proposes an adapted bundle method to deal with an oracle providing inexact linearizations with a controllable inaccuracy that can be driven to 0. Unlike [6, 7], the work [8] proposes a methodology to deal

with inexact subgradient evaluations without knowledge of the approximation quality. The method, however, requires the exact evaluation of the objective function. This case is generalized in [5] to unknown inaccuracies affecting both the function value and the subgradient. This framework is further extended for constrained optimization in [10]. A different point of view is adopted in [9], that studies the precision required by the oracle to achieve optimality within a given tolerance using an almost standard bundle method. Our incremental approach can be interpreted as an inexact bundle method along the lines of [5], which introduces a specific mechanism for detecting and attenuating noise when the inaccuracy becomes unduly large. However, unlike [5], inaccuracy is no longer considered constant, but varies with the evaluation point.

As a by-product of the incremental bundle method, we study the behaviour of an alternative method, working with function evaluations of *unknown* and *vanishing* inaccuracy, but without noise attenuation step. We refer to this method as *inexact classical bundle method*. In [7] the author considers varying accuracies that can be arbitrarily modified along the process to be driven to 0. Under this assumption, the method finds asymptotically an optimal solution to (1). In [8], the case of unknown and vanishing inaccuracy is handled, but it requires the exact evaluation of the function value while subgradients may be computed with errors. In [9], the author gives a bound on the inaccuracy acceptable to achieve a given tolerance. The algorithm involved is almost identical to a classical proximal bundle method and as a consequence shares similarities with our approach. However, it assumes that inaccuracy of the oracle is *known* and possibly controllable and this information is used to build up the descent test.

Our paper is organized as follows. We give in Section 2 a real-life stochastic program that motivated our research, on mid-term generation planning. Section 3 reviews disaggregate bundle methods, tailored for structured objective functions like (1). Section 4 considers how to replace the difficult-to-compute information, corresponding to evaluating f^2. The incremental bundle method with varying inaccuracy and the inexact classical bundle method with vanishing accuracy are given in Sections 5 and 6, respectively. Finally, encouraging numerical results on a real-life application on the French power mix are reported in Section 7.

2 Motivation: price decomposition on a scenario tree

Electricité de France's (EDF) electric generation system is nuclear–dominated: about 75% of the total company power is produced by nuclear plants. We focus here on mid-term generation planning, that considers a time horizon of 1 to 3 years. In this framework, operational constraints are usually modeled with far less details than for short term scheduling. For example, hydro-plants are often aggregated into energy equivalent reservoirs, also, thermal plants starting costs

and ramp constraints are disregarded.

However, because nuclear plants can be seen as "equivalent energy reservoirs", it is important to keep some level of detail, specifically to reflect the dynamics resulting from using nuclear fuel.

Indeed, unlike classical thermal plants, a nuclear plant alternates predetermined **outage periods** of 1 to 3 months, with **generation campaigns** of about 18 months. During each outage, one third of the reactor fuel is renewed. The length of the next generation campaign is given by the months it should take to consume one third of the reservoir capacity. In addition, in order to allow for some flexibility, the total campaign generation is constrained by given lower and upper bounds.

The nuclear dynamics should also reflect how the ability of the plant to generate any desired power level depends on the amount of fuel in the reactor, as follows.

- The power generation is always above an ideal lower bound.
- However, some *"modulation"* is allowed: the lower bound may be violated, as long as the modulation along the whole optimization horizon does not exceed a certain upper bound, limiting the total amount of violation.
- Once all the refueled combustible of the last outage is consumed, the plant can only generate power according to a predetermined decreasing production profile.

Finally, when the end of a generation campaign coincides with a period of high demand, it is desirable to benefit from full flexibility of the nuclear plant for a period as long as possible.

Due to the predominance of nuclear generation over the global French production mix, taking into account all these constraints is of crucial interest to design sound strategies. The modeling of such constraints not only requires many additional variables, but also introduces a level of temporal coupling considerably larger than when dealing with hydro or classical thermal plants.

As a result, considering the 58 nuclear reservoirs and the 2 or 3 aggregate hydraulic reservoirs usually modeled at EDF, at least 60 reservoir dynamics need to be jointly optimized, over a time horizon with more than 1000 time steps. The corresponding multistage stochastic linear programming problem is large scale. It has mixed-integer variables when interruptible contracts are taken into account. These contracts give the possibility to cut-off the supply to some clients, if warned in advance. Thus, interruptible contracts are alternative energy reservoirs, that further increase the state dimension of the problem. In our runs, we do not consider these contracts.

For mid-term horizon, generation units are modeled by linear cost functions $\mathcal{C}^i$ and technical constraints, defining polyhedral sets $\mathcal{P}^i$, where $i = 1, \ldots, Nu$ denotes the index of the unit. Generation units correspond not only to power plants, but also to some financial market, or even to load shedding. For example, the spot market is represented by means of two additional classical thermal

plants, with bounded generation (positive or negative), and linear costs for each time period. The whole generation mix is coupled by the satisfaction of demand constraint.

In addition, in the mid-term horizon many elements of the problem are no longer deterministic. In our case, the level of demand is the main source of uncertainty, particularly during winter periods. Other parameters that introduce randomness are water inflows, the thermal plants availability, and future prices on the spot market. Uncertainty is modeled with a scenario tree obtained from a sample of historical scenarios aggregated according to scenario reduction techniques ([11]).

For practical applications, a time horizon of 2 years is usually chosen, with a daily time discretization. Each day is further divided into three periods, representing peak hours of high demand, base demand, and off–peak hours. To provide a reasonable vision of future uncertainties, the scenario tree typically has more than 50.000 nodes, yielding an optimization problem with more that 10^7 variables and 10^8 constraints. As a result, even with a rather simplistic description of the power mix, we still have to deal with a large-scale linear optimization problem, that needs to be solved by some decomposition technique. The approach currently used at EDF involves applying price–decomposition on the scenario tree [12], and solving the dual problem by the variable metric proximal bundle method [13].

We now explain how, after decomposition, computing the, difficult, nuclear sub-function boils down to solving at each iteration a huge linear program.

We denote with a subscript $n \leq N$ each node of the scenario tree, d_n the demand at that node and π_n the probability of reaching this node from the initial step. Generation units are referred to with a subscript $i \leq Nu$, hence $p^i := (p^i_1, \ldots, p^i_N)$ stands for the generation vector of unit i over all the nodes in the tree and $\mathcal{C}^i(p^i_n)$ denotes the generation cost of unit i at node n.

With this notation, our optimization problem can be schematically written as:

$$\left\{ \begin{array}{rll} \min_{\{p^i\}_{i \leq Nu}} & \sum_{i=1}^{Nu} \sum_{n=1}^{N} \pi_n \mathcal{C}^i(p^i_n) & \\ \text{s.t.} & g(p) := \sum_{i=1}^{Nu} p^i - d = 0\,, & \textit{(coupling constraints of demand)} \\ & p^i \in \mathcal{P}^i & \textit{(technical constraints for plant } i\textit{)}\,. \end{array} \right. \tag{2}$$

As mentioned, for our application functions $\mathcal{C}^i$ are linear and sets $\mathcal{P}^i$ are polyhedrals. We assume furthermore that problem (2) satisfies a Slater type condition, so there is no duality gap (see e.g. [14, Ch. 8]), and primal and dual optimal values are the same. More precisely, applying price-decomposition on problem (2) means relaxing the demand constraint by introducing multipliers $x \in \mathbb{R}^N$. The corresponding (separable) Lagrangian

$$L(p, x) := \sum_{i=1}^{Nu} \Big(\sum_{n=1}^{N} \pi_n \mathcal{C}^i(p^i_n) - \langle x, p^i - \frac{d}{Nu} \rangle \Big) =: \sum_{i=1}^{Nu} L^i(p^i, x)\,,$$

gives the dual problem

$$\max_x \Big(\sum_{i=1}^{Nu} \min_{p^i \in \mathcal{P}^i} L^i(p^i, x) \Big) ,$$

that is solved instead of (2). For convenience, we consider the negative of the corresponding dual function, decomposable by generation units:

$$\begin{aligned} f(x) \quad &:= \quad -\min_p L(p,x) = -\sum_{i=1}^{Nu} \min_{p^i} L^i(p^i, x) \qquad (3) \\ &=: \quad \sum_{i=1}^{Nu} f^i(x) \, . \end{aligned}$$

Each term f^i is convex, possibly non-smooth, and is referred to as a *dual sub-function*. Solving the dual problem is equivalent to minimizing f over $\mathbb{R}^N$ with adapted algorithms, for example the bundle method [13] we employ in our application.

When evaluating each sub-function f^i at some point x, the corresponding primal point $\bar{p}^i_x$, minimizing $L^i(\cdot, x)$ from (3), may not be unique, but it always gives a subgradient for f^i. Hence,

$$g(\bar{p}^i_x) = \sum_{i=1}^{Nu} \bar{p}^i_x - d \in \partial f(x) \, .$$

For notational convenience, we sometimes write

$$g(\bar{p}^i_x) = \sum_{i=1}^{Nu} g^i(\bar{p}^i_x) \, , \text{ with } g^i(\bar{p}^i_x) = \bar{p}^i_x - \frac{d}{Nu} \in \partial f^i(x)$$

for $i = 1, \ldots, Nu$.

The advantage of price decomposition is that the separable dual function is easy to compute, because the dual evaluation can be performed separately for each generation unit, as illustrated by Figure 1.

In our setting, computing the pairs $(f^i(x), g^i(\bar{p}^i_x))$ is very costly for some units i. In particular, for a scenario tree with 50.000 nodes, evaluating the corresponding sub-function for nuclear plants amounts to solving a large linear program with 100.000 variables (generation and reservoir levels) and about 300.000 constraints. By contrast, solving a classical thermal sub-functions requires 50.000 variables and constraints and it may be performed for each node successively since there is no temporal coupling. As mentioned, an incremental strategy may exploit the separable structure by performing dual steps without having exact values for all the sub-functions.

Before introducing the incremental bundle method, we recall briefly the main features of a bundle methodology when applied to a separable function, also called a disaggregate bundle method.

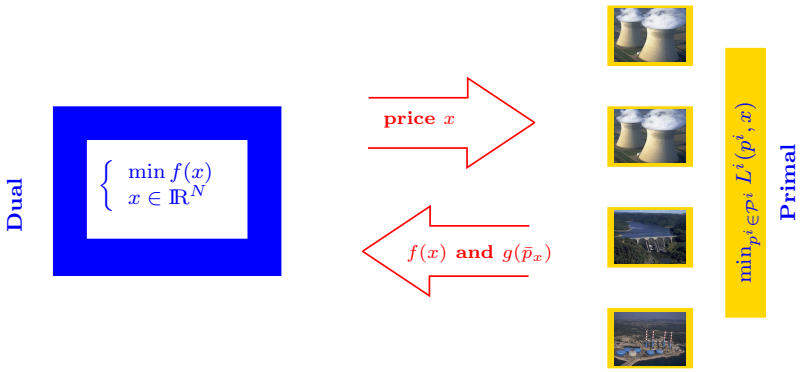

Figure 1: Price decomposition

3 Disaggregate bundle methods

Bundle methods appear as one of the most effective approaches to solve non-smooth optimization problems. For minimizing a function $f = \sum_i f^i$ as in (3), we employ a *disaggregate* proximal bundle method. In this section, we recall briefly some basic elements from bundle methods, for a detailed description we refer to [15], [14], or [16].

Disaggregate bundle methods keep memory of past iterations to build up individual cutting-planes models approximating each sub-function f^i. With our notation, at iteration ℓ, for $i = 1, \ldots, Nu$, the bundle $\mathcal{B}^i_\ell$ for sub-function i is defined as

$$\mathcal{B}^i_\ell := \left\{ \left(x^j, f^i(x^j), g^{i,j} \in \partial f^i(x^j)\right), j = 1, \ldots, \ell \right\} . \tag{4}$$

where each couple $(f^i, g^{i,j})$ defines a hyperplane supporting f^i at x^i. For comparison, the more standard aggregate bundle for f is defined by

$$\mathcal{B}_\ell := \left\{ \left(x^j, f(x^j) = \sum_{i=1}^{Nu} f^i(x^j), g^j = \sum_{i=1}^{Nu} g^{i,j} \in \partial f(x^j) \right), j = 1, \ldots, \ell \right\} .$$

Along iterations a sequence of points is generated. Iterates can be of two types: either *candidate* points, used essentially to increase the model's accuracy; or *serious* points, that significantly decrease the objective function (and also improve the model's accuracy). The corresponding iterations of the algorithmic scheme are respectively called *null* and *serious* steps. Serious points, sometimes referred to as prox-center or stability center in the literature, are denoted by $\hat{x}^{k(\ell)}$ and they form a sub-sequence of the candidates sequence $\{x^\ell\}$. Each candidate is found by employing a stabilized version of cutting-planes methods, via a quadratic term centered around the last serious point, as in (5) below.

In order to allow for a partial evaluation of sub-functions, we need to work with a *disaggregate* variant of bundle methods as proposed in [17]; see also

[12]. Indeed, when evaluating the dual function f at some point x, we have at hand sub-function values $f^i(x)$ and corresponding subgradients $g^i \in \partial f^i(x)$, for all $i = 1, \ldots, Nu$. Hence, we build up individual cutting-planes models $\check{f}^i_\ell(\cdot)$ for each sub-function. A cutting-planes model is defined as the pointwise maximum of affine functions. Since the sum of maxima is bigger than the maximum of the sum ($\sum_{i=1}^{Nu} \check{f}^i \geq \check{f}$), when summing up individual models we should obtain a better model for f. Moreover, the disaggregate variant provides individual models for each sub-function, a useful feature for our incremental proposal. However, it is important to keep in mind that disaggregation increases significantly the amount of bundle information (instead of one aggregate bundle, there are Nu individual ones). For this reason, partial aggregation, using some partition of the set of sub-functions, can sometimes offer a good compromise, we refer to [12] for more details.

For a disaggregate bundle method, the new iterate is given by

$$x^{\ell+1} := \arg\min_{x \in \mathbb{R}^N} \left(\sum_{i=1}^{Nu} \check{f}^i_\ell(x) + \frac{1}{2}\mu_k |x - \hat{x}^{k(\ell)}|^2 \right) . \tag{5}$$

To measure the quality of the candidate, the nominal decrease $\delta_{\ell+1}$ is used, representing the decrease predicted by the cutting-planes model

$$\delta_{\ell+1} := f(\hat{x}^{k(\ell)}) - \sum_i \check{f}^i(x^{\ell+1}) .$$

More precisely, after computing $f(x^{\ell+1})$ and $g^{\ell+1} \in \partial f(x^{\ell+1})$, a descent test compares the effective descent to the predicted value $\delta_{\ell+1}$. If the test is satisfied, then $x^{\ell+1}$ can be considered as a new serious point: we set $k(\ell+1) := k(\ell)+1$, $\hat{x}^{k(\ell+1)} := x^{\ell+1}$ and both the model and the bundle are updated. If we declare a null step, the model is updated without changing the serious point, so $k(\ell+1) := k(\ell)$.

Bundle data in (4) can be represented alternatively, by referring the information to the current serious point, using the *linearization errors*

$$e^i_j = f^i(\hat{x}^{k(\ell)}) - f^i(x^j) - \langle g^{i,j}, \hat{x}^{k(\ell)} - x^j \rangle .$$

Then, instead of keeping $(x^j, f^i(x^j), g^{i,j})$, the bundle is composed by triplets of the form $(e^i_j, f^i(x^j), g^{i,j})$. This is useful when there is bundle compression, that is, when only a subset $j \in J_\ell \subset \{1, \ldots, \ell\}$ is kept in the bundle, the pairs $(f^i_j, g^{i,j})$ may no longer correspond to some x^j. In fact, it is possible to keep in the bundle just one element without impairing convergence, as long as the element kept is the so-called *aggregate* couple:

$$(\hat{\varepsilon}_{\ell+1}, \hat{g}^{\ell+1}) = \sum_{i=1}^{Nu} (\hat{\varepsilon}^i_{\ell+1}, \hat{g}^{i,\ell+1}) , \text{ with } (\hat{\varepsilon}^i_{\ell+1}, \hat{g}^{i,\ell+1}) := \sum_{j \in J_\ell} \alpha^\ell_j (e^i_j, g^{i,j}).$$

In this expression, the simplicial multipliers α^ℓ are a by-product of solving the quadratic program of (5). As a result, after compression, bundle elements

correspond either to past dual evaluations or to past compressions; see [14, Ch. 10]. The aggregate element $\hat{g}^{\ell+1} = \sum_{i=1}^{Nu} \hat{g}^{i,\ell+1}$ is a subgradient at $x^{\ell+1}$ for the current cutting-planes model, and an approximate subgradient for the real function f.

Bundle methods stop when both aggregate values, $\hat{\varepsilon}_{\ell+1}$ and $\hat{g}^{\ell+1}$, are small enough or, keeping in mind the relation (e.g. [14, Lemma 10.8]):

$$\delta_{\ell+1} \quad = \quad \frac{|\hat{g}^{\ell+1}|^2}{\mu_\ell} + \hat{\varepsilon}_{\ell+1}\,, \tag{6}$$

when the predicted descent is small enough.

4 Replacing the missing information

When skipping the evaluation of some sub-functions in a disaggregate bundle method, the lacking information needs to be replaced by some approximation. In this case, a natural substitute for f^i would be to use the individual cutting-planes model, $\check{f}^i_\ell$. Such replacement is closely related to the inexact bundle methods studied by various authors and especially to the framework developed in [5], as explained below.

To ease the presentation, we refer to the general problem (1) corresponding to $Nu = 2$ in the previous setting, with f^1 easy to evaluate while f^2 is time consuming. Sub-functions f^i are general dual functions of the form:

$$f^i(\cdot) := \max_{p\in\mathcal{P}^i}\{C^i(p) + \langle g^i(p), \cdot\rangle\}\,.$$

Hence, for notational convenience, C^i corresponds here to $-\sum_n \pi_n \mathcal{C}^i(p^i_n)$ in the generation planning problem (2), f^1 to the, easy-to-evaluate, classical thermal sub-function and f^2 to the, difficult, nuclear sub-function.

The inexact setting of [5] supposes that for any $x \in \mathbb{R}^N$ the oracle returns $\left(f^2_x, g^2_x\right)$ satisfying:

$$f^2_x \quad \geq \quad f^2(x) - E^f \text{ and} \tag{7}$$

$$f^2(\cdot) \quad \geq \quad f^2_x + \langle g^2_x, \cdot - x\rangle - E^g\,, \tag{8}$$

where E^f and E^g are fixed but unknown function and subgradient inaccuracies.

Now, suppose that instead of computing $f^2(x)$ by finding $\bar{p}^2_x$ such that

$$\bar{p}^2_x \in \operatorname*{Argmax}_{p\in\mathcal{P}^2}\{C^2(p) + \langle g^2(p), \cdot\rangle\}\,,$$

we use any $p^2 \in \mathcal{P}^2$ to extrapolate inexact values: $f^2_x := C^2(p^2) + \langle g^2(p^2), x\rangle$, and $g^2_x := g^2(p^2)$, for some $p^2 \in \mathcal{P}^2$. Then

$$\begin{aligned} f^2(y) \quad &= \quad \max_{p\in\mathcal{P}^2}\{C^2(p) + \langle g^2(p), y\rangle\} \\ &\geq \quad C^2(p^2) + \langle g^2(p^2), y\rangle \\ &= \quad f^2_x + \langle g^2(p^2), y - x\rangle\,. \end{aligned}$$

Consequently, such oracle is compatible with the inexact setting of [5], with function inaccuracy

$$E_x^f := C^2(\bar{p}_x^2) - C^2(p^2) + \langle g^2(\bar{p}_x^2) - g^2(p^2), x\rangle$$

and a null subgradient error $E^g = 0$. We notice that the function inaccuracy is not fixed but varying; it is nevertheless bounded, because the set $\mathcal{P}^2$ is compact.

Clearly, the choice of p^2 will determine the quality of the approximation. A first, straightforward, possibility is to perform the maximization process yielding $f^2(x)$ only approximately, stopping the optimization after a fixed amount of CPU time. But this is not the only possibility. Another natural choice for p^2 would be to use some past responses from the oracle. All available past information is contained in the bundle $\mathcal{B}_\ell^2$ and this bundle gives the cutting-planes model $\check{f}_\ell^2$. Hence, the best choice for $f_{x^{\ell+1}}^2$ when using only the information of the bundle, would be to evaluate the current cutting-planes model. Indeed, since $\hat{g}^{2,\ell} \in \partial \check{f}_\ell^2(x^{\ell+1})$ and $\check{f}_\ell^2 \leq f$, when we use for the new iterate the inexact values $\left(f_{x^{\ell+1}}^2, g_{x^{\ell+1}}^2\right) := \left(\check{f}_\ell^2(x^{\ell+1}), \hat{g}^{2,\ell}\right)$, we have

$$\begin{aligned} f^2(y) &\geq \check{f}_\ell^2(y) \\ &\geq \check{f}_\ell^2(x^{\ell+1}) + \langle \hat{g}^{2,\ell}, y - x^{\ell+1}\rangle \\ &= f_{x^{\ell+1}}^2 + \langle g_{x^{\ell+1}}^2, y - x^{\ell+1}\rangle\,, \end{aligned} \tag{9}$$

as desired. Therefore, we are still in the setting of [5] with $E^g = 0$, and a function inaccuracy E_x^f that varies along iterations and satisfies

$$E_{x^{\ell+1}}^f \leq \min_{j\in\mathcal{B}_\ell} \left\{C(\bar{p}_{x^{\ell+1}}^2) - C(\bar{p}_{x^j}^2) + \langle g^2(\bar{p}_{x^{\ell+1}}^2) - g^2(\bar{p}_{x^j}^2), x^{\ell+1}\rangle\right\}\,.$$

5 Incremental bundle method with varying inaccuracy

We now address the problem of solving (1) by an incremental approach. We first consider the convergence properties of a proximal bundle method with inexact linearizations when inaccuracies vary along iterations. In the next section we will give an alternative method, with vanishing inaccuracy, more close to classical bundle methods.

5.1 Bundle methods with inexact linearizations

Inexact bundle methods are devised to handle inaccuracies both in the evaluation of the objective function, or in the subgradient returned by the available oracle. We follow here a setting similar to [5], for inaccuracies that are no longer constant (as in [5]), but vary with the evaluation point. In our analysis we suppose that f in (1) is coercive and, hence, has a finite optimal value f^{opt} (this holds for example when (1) comes from Lagrangian relaxation of a primal problem satisfying a Slater–type condition). The analysis could be easily

generalized to $Nu \geq 2$ sub-functions, but for simplicity we consider (1), with f^1 easy to evaluate and f^2 hard to compute and time consuming. Hence, an exact oracle is available for f^1, providing for any $y \in \mathbb{R}^N$ the values $f^1(y)$ and $g^1(y) \in \partial f^1(y)$. However, for f^2, an inexact oracle only provides (f_y^2, g_y^2) with *unknown* but *bounded* inaccuracy:

$$\begin{aligned} f_y^2 &\geq f^2(y) - E_y^f \text{ and} && (10)\\ f^2(\cdot) &\geq f_y^2 + \langle g_y^2, \cdot - y\rangle - E_y^g\,, && (11)\end{aligned}$$

such that:

$$E_x^f \leq E_{max}^f \text{ and } E_x^g \leq E_{max}^g\,. \qquad (12)$$

As we saw in Section 4, the incremental setting can be interpreted as an instance of this framework. Inaccuracy is unknown, varying, but bounded by (12).

We notice that

$$\begin{aligned} g_y^2 &\in \partial_{E_y^f + E_y^g} f^2(y) \quad \text{and}\\ f_y^2 &\in [f^2(y) - E_y^f; f^2(y) + E_y^g]\,. \end{aligned}$$

Our framework is slightly different from [5], where it is assumed that, for any $y \in \mathbb{R}^N$:

$$\begin{aligned} f_y^2 &\geq f^2(y) - E^f \quad \text{and}\\ f^2(\cdot) &\geq f_y^2 + \langle g_y^2, \cdot - y\rangle - E^g\,. \end{aligned}$$

For this reason, in Section 5.2 below convergence proofs from [5] are reviewed in the new framework.

As in the exact setting of Section 3, inexact bundle methods generate a sequence of trial points x^ℓ at which functions values and subgradients are evaluated. The sub-sequence of serious points $\{\hat{x}^{k(\ell)}\}$ is chosen so that the corresponding objective functions are significantly decreasing. Trial points are provided by the solution of the quadratic program (13):

$$x^{\ell+1} := \arg\min_{y \in \mathbb{R}^N} \{\Phi_\ell(y) + \frac{1}{2}\mu_\ell |y - \hat{x}^{k(\ell)}|^2\}\,, \qquad (13)$$

where $\mu_\ell > 0$ is the prox-parameter, $\hat{x}^{k(\ell)}$ is the current serious point, and Φ_ℓ is the (inexact) disaggregate cutting-planes model approximating f. More precisely, keeping in mind the disaggregate model defined in Section 3,

$$\begin{aligned} \Phi_\ell(\cdot) &:= (\check{f}_\ell^1 + \varphi_\ell^2)(\cdot)\,, \text{ with}\\ \check{f}_\ell^1(\cdot) &:= f^1(\hat{x}^{k(\ell)}) + \max_{j \in \mathcal{B}_\ell^1} \left\{-e_{j,\ell}^1 + \langle g^1(x^j), \cdot - \hat{x}^{k(\ell)}\rangle\right\} \text{ and} && (14)\\ \varphi_\ell^2(\cdot) &:= f_{\hat{x}^{k(\ell)}}^2 + \max_{j \in \mathcal{B}_\ell^2} \left\{-e_{j,\ell}^2 + \langle g_{x^j}^2, \cdot - \hat{x}^{k(\ell)}\rangle\right\}\,, && (15)\end{aligned}$$

where $\mathcal{B}_\ell^1$ and $\mathcal{B}_\ell^2$ denote the current bundles of cutting planes. In the expression above, $e_{j,\ell}^1$ and $e_{j,\ell}^2$ are linearization errors, often denoted in the compact form

$e^{1,2}_{j,\ell}$. Linearization errors $e^{1,2}_{j,\ell}$ measure the difference between cutting planes contained in $\mathcal{B}^{1,2}_\ell$ and the function value returned by the oracle for the current serious point:

$$\begin{aligned} e^1_{j,\ell} &= f^1(\hat{x}^{k(\ell)}) - f^1(x^j) - \langle g^1(x^j), \hat{x}^{k(\ell)} - x^j \rangle \,, \\ e^2_{j,\ell} &= f^2_{\hat{x}^{k(\ell)}} - f^2_{x^j} - \langle g^2_{x^j}, \hat{x}^{k(\ell)} - x^j \rangle \,. \end{aligned} \tag{16}$$

When using an exact oracle, i.e. for f^1, the cutting-planes model $\check{f}^1_\ell$ always underestimates the real function and is exact for previous candidates if the corresponding elements have not been deleted from the bundle. As a consequence, linearization errors $e^1_{j,\ell}$ are always nonnegative. By contrast, in the inexact context the relation $\varphi^2_\ell \leq f^2$ does not necessarily hold, and $e^2_{j,\ell}$ may be negative. Indeed, by (10), (11) and (16), $e^2_{j,\ell}$ is only known to satisfy

$$e^2_{j,\ell} \geq -(E^f_{\hat{x}^{k(\ell)}} + E^g_{x^j}) \,. \tag{17}$$

Furthermore, model φ^2_ℓ may overestimate the real function f^2 at some points, since by (11) only inequality (18) holds:

$$\varphi^2_\ell(\cdot) \leq f^2(\cdot) + \max_{j \in \mathcal{B}^2_\ell} E^g_{x^j} \,. \tag{18}$$

Writing the optimality conditions for (13), there are simplicial multipliers $\{\alpha^1_j\}_{j\in\mathcal{B}^1_\ell}$ and $\{\alpha^2_j\}_{j\in\mathcal{B}^2_\ell}$ such that

$$\begin{aligned} x^{\ell+1} &= \hat{x}^{k(\ell)} - \frac{1}{\mu_\ell}\hat{g}^{\ell+1} \,, \text{ where} \\ \hat{g}^{\ell+1} &= \sum_{j\in\mathcal{B}^1_\ell} \alpha^1_j g^1(x^j) + \sum_{j\in\mathcal{B}^2_\ell} \alpha^2_j g^2_{x^j} \text{ is the aggregate subgradient.} \end{aligned}$$

The aggregate vector $\hat{g}^{\ell+1}$ is a convex combination of inexact subgradients of f. From the optimality conditions of (13), we know that $\hat{g}^{\ell+1} \in \partial\Phi_\ell(x^{\ell+1})$. We define the corresponding aggregate linearization for the model as

$$\Phi^{lin}_\ell(\cdot) = \Phi_\ell(x^{\ell+1}) + \langle \hat{g}^{\ell+1}, \cdot - x^{\ell+1} \rangle \,.$$

The aggregate linearization error $\hat{\varepsilon}_{\ell+1} := \sum_{j\in\mathcal{B}^1_\ell} \alpha^1_j e^1_{j,\ell} + \sum_{j\in\mathcal{B}^2_\ell} \alpha^2_j e^2_{j,\ell}$ corresponds to the difference between the value of the oracle at the last serious point and the value of the aggregate linearization at that point:

$$\hat{\varepsilon}_{\ell+1} = (f^1(\hat{x}^{k(\ell)}) + f^2_{\hat{x}^{k(\ell)}}) - \Phi^{lin}_\ell(\hat{x}^{k(\ell)}) \,.$$

In addition, being a convex combination of linearization errors $e^{1,2}_{j,\ell}$, the aggregate linearization error satisfies:

$$\hat{\varepsilon}_{\ell+1} \geq -(E^f_{\hat{x}^{k(\ell)}} + \sum_{j\in\mathcal{B}^2_\ell} \alpha^2_j E^g_{x^j}) \geq -(E^f_{max} + E^g_{max}) \,. \tag{19}$$

As for the predicted descent, it takes as reference the value returned by the oracle for the current serious point and is defined as $\delta_{\ell+1} = \delta^1_{\ell+1} + \delta^2_{\ell+1}$, where

$$\begin{aligned} \delta^1_{\ell+1} &= f^1(\hat{x}^{k(\ell)}) - \check{f}^1_\ell(x^{\ell+1})\,, \text{ and} \\ \delta^2_{\ell+1} &= f^2_{\hat{x}^{k(\ell)}} - \varphi^2_\ell(x^{\ell+1})\,. \end{aligned}$$

A serious step is declared when the following inequality is satisfied:

$$\left(f^1(\hat{x}^{k(\ell)}) + f^2_{\hat{x}^{k(\ell)}}\right) - \left(f^1(x^{\ell+1}) + f^2_{x^{\ell+1}}\right) \quad \geq \quad m\delta_{\ell+1} \tag{20}$$

where $m \in (0,1)$ is an Armijo-type parameter.

For later use, we recall that the relation (6) still holds for the inexact setting.

Using the fact that $x^{\ell+1}$ solves the quadratic program (13), we also have that

$$\begin{aligned} \delta_{\ell+1} &\geq \frac{1}{2}\mu_\ell |x^{\ell+1} - \hat{x}^{k(\ell)}|^2 - \left(\check{f}^1_\ell(\hat{x}^{k(\ell)}) + \varphi^2_\ell(\hat{x}^{k(\ell)}) - f^1(\hat{x}^{k(\ell)}) - f^2_{\hat{x}^{k(\ell)}}\right) \\ &\geq \frac{1}{2}\mu_\ell |x^{\ell+1} - \hat{x}^{k(\ell)}|^2 - \left(\varphi^2_\ell(\hat{x}^{k(\ell)}) - f^2_{\hat{x}^{k(\ell)}}\right), \end{aligned}$$

where we used (14) and the nonnegativity of $e^1_{j,\ell}$ for all j and ℓ.

Since by (15), $\varphi^2_\ell(\hat{x}^{k(\ell)}) - f^2_{\hat{x}^{k(\ell)}} = \max_j\{-e^2_{j,\ell}\}$, as soon as a negative linearization error occurs, the predicted descent may be negative too. In this case, inequality (21) below can be used as an accuracy test and should be satisfied if only "small errors" have been introduced in the model φ^2

$$\delta_{\ell+1} \quad > \quad \frac{1}{2}\mu_\ell |x^{\ell+1} - \hat{x}^{k(\ell)}|^2\,. \tag{21}$$

By (6), we have the following equivalent inequalities for (21):

$$\text{(21) if and only if } \frac{|\hat{g}^{\ell+1}|^2}{2\mu_\ell} > -\hat{\varepsilon}_{\ell+1} \tag{22a}$$

$$\text{if and only if } \delta_{\ell+1} > \frac{|\hat{g}^{\ell+1}|^2}{2\mu_\ell}\,. \tag{22b}$$

Finally, as in [5, Eq.2.16], we define the optimality measure

$$V_\ell := \max\{|\hat{g}^{\ell+1}|, \hat{\varepsilon}_{\ell+1}\}\,. \tag{23}$$

Indeed, by linearity of Φ^{lin}_ℓ we know that:

$$\begin{aligned} \Phi^{lin}_\ell(\cdot) &= \Phi^{lin}_\ell(\hat{x}^{k(\ell)}) + \langle \hat{g}^{\ell+1}, \cdot - \hat{x}^{k(\ell)}\rangle \\ &= (f^1(\hat{x}^{k(\ell)}) + f^2_{\hat{x}^{k(\ell)}}) + \langle \hat{g}^{\ell+1}, \cdot - \hat{x}^{k(\ell)}\rangle - \hat{\varepsilon}_{\ell+1}\,. \end{aligned}$$

By the subgradient inequality, $\Phi^{lin}_\ell(\cdot) \leq \Phi_\ell(\cdot)$, so for any $y \in \mathbb{R}^N$,

$$(f^1(\hat{x}^{k(\ell)}) + f^2_{\hat{x}^{k(\ell)}}) \quad \leq \quad \Phi_\ell(y) - \langle \hat{g}^{\ell+1}, y - \hat{x}^{k(\ell)}\rangle + \hat{\varepsilon}_{\ell+1}\,.$$

By (18), this implies that for any $y \in \mathbb{R}^N$,

$$(f^1(\hat{x}^{k(\ell)}) + f^2_{\hat{x}^{k(\ell)}}) \quad \leq \quad f(y) + \max_{j \in \mathcal{B}^2_\ell} E^g_{x^j} + \langle \hat{g}^{\ell+1}, y - \hat{x}^{k(\ell)} \rangle + \hat{\varepsilon}_{\ell+1} \,. \quad (24)$$

Using again (6), we deduce the following inequalities regarding the optimality measure:

$$V_\ell \quad \leq \quad \max\{\sqrt{2\mu_\ell \delta_{\ell+1}}, \delta_{\ell+1}\}\,, \text{if (21) holds, and} \quad (25)$$

$$V_\ell \quad \leq \quad \sqrt{-2\mu_\ell \hat{\varepsilon}_{\ell+1}} \leq \sqrt{2\mu_\ell (E^f_{max} + E^g_{max})}\,, \text{ otherwise.} \quad (26)$$

The following result justifies the choice of V_ℓ as optimality measure.

Lemma 1 ([5], Lemma 2.3)**.** *Suppose that for an infinite subset of iterations $\mathcal{L} \subset \{1, 2, \ldots\}$ the sub-sequence $\{V_\ell\}_{\ell \in \mathcal{L}} \to 0$ as $\mathcal{L} \ni \ell \to \infty$. Suppose furthermore that the corresponding sub-sequence of serious points $\{\hat{x}^{k(\ell)}\}_{\ell \in \mathcal{L}}$ is bounded, and let $\hat{x}^{acc}$ denote an accumulation point. Then $\hat{x}^{acc}$ is an approximate solution of* (1)*, with*

$$f(\hat{x}^{acc}) \leq f^{opt} + \limsup_{\ell \in \mathcal{L}} E^f_{\hat{x}^{k(\ell)}} + \limsup_{\ell \in \mathcal{L}} \left(\max_{j \in \mathcal{B}^2_\ell} E^g_{x^j} \right) . \quad (27)$$

Proof. Passing to the limit in inequality (24) above we have that

$$\lim_{\mathcal{L} \ni \ell \to \infty} (f^1(\hat{x}^{k(\ell)}) + f^2_{\hat{x}^{k(\ell)}}) \quad \leq \quad f^{opt} + \limsup_{\ell \in \mathcal{L}} \left(\max_{j \in \mathcal{B}^2_\ell} E^g_{x^j} \right) .$$

Moreover, for any cluster point $\hat{x}^{acc}$ of $\{\hat{x}^{k(\ell)}\}_{\ell \in \mathcal{L}}$, passing to the limit in (10),

$$f(\hat{x}^{acc}) - \limsup_{\ell \in \mathcal{L}} E^f_{\hat{x}^{k(\ell)}} \quad \leq \quad \lim_{\mathcal{L} \ni \ell \to \infty} (f^1(\hat{x}^{k(\ell)}) + f^2_{\hat{x}^{k(\ell)}})\,.$$

□

As a direct corollary of Lemma 1, if for some iteration ℓ we have that $V_\ell = 0$, the serious point $\hat{x}^{k(\ell)}$ is an approximate solution of (1) with

$$f(\hat{x}^{k(\ell)}) \leq f^{opt} + E^f_{\hat{x}^{k(\ell)}} + \max_{j \in \mathcal{B}^2_\ell} E^g_{x^j} \,. \quad (28)$$

Similarly, if the serious point sequence stabilizes, i.e. if there is some L such that $\hat{x}^{k(\ell)} = \hat{x}^{k(L)}$ for all $\ell \geq L$, $\hat{x}^{k(L)}$ is an approximate solution of (1) with

$$f(\hat{x}^{k(L)}) \leq f^{opt} + E^f_{\hat{x}^{k(L)}} + \limsup_{\ell \in \mathcal{L}} \left(\max_{j \in \mathcal{B}^2_\ell} E^g_{x^j} \right) . \quad (29)$$

To handle inexact linearizations, the adapted bundle method for inexact linearizations proposed in [5] introduces a prox-parameter management step based on inequality (21) and on a noise attenuation parameter $NAP \geq 0$. Schematically, when (21) does not hold, μ_ℓ is reduced in order to get $\delta_{\ell+2} > \delta_{\ell+1}$ and help ensuring (21). This additional step is called here of Noise Management.

Algorithm 1. *(inexact bundle method with varying inaccuracy)*

1. ***Initialization:*** *Choose* $x^1 \in \mathbb{R}^N$, $\kappa, m \in (0,1)$ *and* $\mu_{\max} > 0$. *Set* $NAP = 0$, *compute* $(f^1(x^1), g^1(x^1))$, $(f^2_{x^1}, g^2_{x^1})$, *and define* $\check{f}^1_1$ *and* φ^2_1. *Set* $\ell = k = 1$, $k(1) = 1$, *and* $\hat{x}^1 = x^1$.

2. ***New candidate:*** *Solve* (13) *to get* $x^{\ell+1}$, $\delta_{\ell+1}$, $\hat{g}^{\ell+1}$, *and* $\hat{\varepsilon}_{\ell+1}$.

3. ***Stopping test:*** *If* $V_\ell = 0$, *stop.*

4. ***Noise management:*** *If* (21) *does not hold, set* $NAP = 1$, $\mu_{\ell+1} = \kappa\mu_\ell$, $\ell = \ell + 1$, *go to 2.*

5. ***Oracle:*** *compute* $f^1(x^{\ell+1}), g^1(x^{\ell+1})$ *and* $f^2_{x^{\ell+1}}, g^2_{x^{\ell+1}}$

6. ***Descent test:*** *If* $f^1(x^{\ell+1}) + f^2_{x^{\ell+1}} \leq f^1(\hat{x}^{k(\ell)}) + f^2_{\hat{x}^{k(\ell)}} - m\delta_{\ell+1}$, *declare a Serious Step, otherwise declare a Null Step.*

   If $x^{\ell+1}$ *gave a serious step, set* $\hat{x}^{k+1} = x^{\ell+1}$, $NAP = 0$, $f^2_{\hat{x}^{k+1}} = f^2_{x^{\ell+1}}$, $k(\ell+1) = k+1$, $k = k+1$, *choose* $\mu_{\ell+1} \leq \mu_{\max}$.

   If $x^{\ell+1}$ *gave a null step,* $k(\ell+1) = k$, *and: if* $NAP = 0$ *choose* $\mu_{\ell+1} \in [\mu_\ell, \mu_{\max}]$, *else if* $NAP = 1$ *take* $\mu_{\ell+1} = \mu_\ell$.

7. ***Update:*** *Define* $\check{f}^1_{\ell+1}$ *and* $\varphi^2_{\ell+1}$ *according to the compression/selection rules of [14, Lemma 10.10],* $\ell = \ell + 1$, *loop to 2.*

If the algorithmic scheme above does not stop, three situations may occur:

- an infinite loop of Noise Management between steps 2 and 4, driving μ_ℓ to 0;
- a finite number of serious steps, followed by an infinite number of null steps;
- an infinite number of serious steps.

In the following, we study separately these three cases to show that they all yield an approximate minimizer in (1).

5.2 Convergence analysis

The inexact setting presented in §5.1 is quite similar to the one proposed by [5], but we allow the inaccuracy (E^f_x, E^g_x) to vary along iterations. We will see here that the convergence results for the inexact bundle method of [5] remain almost unchanged.

5.2.1 Infinite loop of noise management

Proposition 1. *If an infinite loop between steps 2 and 4 occurs in Algorithm 1, then* $V_\ell \to 0$.

Proof. Suppose that at some iteration L, an infinite loop between steps 2 and 4 begins, so that for $\ell \geq L$ neither the stability center $\hat{x}^{k(\ell)} = \hat{x}^{k(L)}$ nor the model $\Phi_\ell = \Phi_L$ change. Hence, when solving sequentially the quadratic program (13), only μ_ℓ is updated. Since $\mu_\ell \to 0$, using (26), we have that

$$0 \leq V_\ell < \sqrt{2\mu_\ell(E^f_{max} + E^g_{max})} \to 0 \text{ as } \ell \to \infty \ .$$

□

Consequently, applying (29) we have that if an infinite loop of Noise Management begins at iteration L in Algorithm 1, $\hat{x}^{k(L)}$ is an approximate solution to (1) satisfying

$$f(\hat{x}^{k(L)}) \leq f^{opt} + \limsup_{\ell \in \mathcal{L}} \left(\max_{j \in \mathcal{B}^2_\ell} E^g_{x^j} \right) + E^f_{\hat{x}^{k(L)}} \ .$$

For comparison, when inaccuracies are fixed and an infinite loop of Noise Management occurs, we recall the analogous result of [5]:

$$f(\hat{x}) \leq f^{opt} + E^g + E^f \ .$$

5.2.2 Finite number of serious steps

We suppose now that the algorithm never enters an infinite loop of noise management.

Proposition 2. *Suppose that, after some iteration $\ell last$, no serious step is declared in Algorithm 1. Then there is a sub-sequence $\mathcal{L} \subset \{1, 2, \ldots\}$ such that $V_\ell \to 0$ as $\mathcal{L} \ni \ell \to \infty$.*

Proof. Here, after some iteration $\ell last$, no serious step is declared. Hence, either noise management steps or null steps are done for $\ell \geq \ell last$. The serious point does not move: for $\ell \geq last$, $\hat{x}^{k(\ell)} = x^{k(last)} =: \hat{x}$.

If the number of noise management steps is infinite, i.e., if (21) does not hold for an infinite sub-sequence of iterates $\{x^{\ell+1}\}_{\ell \in \mathcal{L}}$, we have again that $\mu_\ell \to 0$ as $\mathcal{L} \ni \ell \to \infty$. The previous developments hold for that sub-sequence and $V_\ell \to 0$ as $\mathcal{L} \ni \ell \to \infty$.

Suppose now that only a finite number of noise management steps are done. There is some iteration $L \geq \ell last$ such that (21) holds. Consequently, μ_ℓ is a non-decreasing sequence since $\mu_{\ell+1} \in [\mu_\ell, \mu_{\max}]$ for $\ell > L$, and $\mu_\ell \to \bar{\mu} \leq \mu_{max}$ as $\ell \to \infty$. In the following, we derive standard arguments for bundle methods to show that $\delta_\ell \to 0$. We define the partial linearization of the objective of (13) by:

$$L_\ell(x) := \Phi^{lin}_\ell(x) + \frac{1}{2}\mu_\ell |x - \hat{x}|^2 \ .$$

First, using Lemma 10.10 in [14], we know that the rules to apply compression/selection on the bundle guarantee that

$$\Phi^{lin}_\ell(\cdot) \leq \Phi_{\ell+1}(\cdot).$$

By (18),

$$\begin{aligned} L_\ell(\hat{x}) &= \Phi_\ell^{lin}(\hat{x}) \\ &\leq \Phi_{\ell+1}(\hat{x}) \\ &\leq f(\hat{x}^{k(\ell)}) + E_{max}^g \end{aligned} \tag{30}$$

Similarly, evaluating L_ℓ at $x^{\ell+1}$ and using the fact that $\mu_{\ell+1} \geq \mu_\ell$ gives

$$\begin{aligned} L_\ell(x^{\ell+2}) &\leq \Phi_{\ell+1}(x^{\ell+2}) + \frac{1}{2}\mu_{\ell+1}|x^{\ell+2} - \hat{x}|^2 \\ &= \Phi_{\ell+1}^{lin}(x^{\ell+2}) + \frac{1}{2}\mu_{\ell+1}|x^{\ell+2} - \hat{x}|^2 \\ &= L_{\ell+1}(x^{\ell+2})\,. \end{aligned}$$

Furthermore, $x^{\ell+1}$ being the solution to (13), $\nabla L_\ell(x^{\ell+1}) = 0$ and by Taylor's expansion,

$$L_\ell(\cdot) = L_\ell(x^{\ell+1}) + \frac{1}{2}\mu_\ell|\cdot - x^{\ell+1}|^2\,.$$

Hence,

$$\begin{aligned} L_\ell(x^{\ell+2}) &= L_\ell(x^{\ell+1}) + \frac{1}{2}\mu_\ell|x^{\ell+2} - x^{\ell+1}|^2\,, \text{ and} \\ L_\ell(\hat{x}) &= L_\ell(x^{\ell+1}) + \frac{1}{2}\mu_\ell|\hat{x} - x^{\ell+1}|^2\,. \end{aligned}$$

Using the developments above, the fact that $\mu_\ell \geq \mu_L$, and (30), we obtain the following inequalities:

$$L_\ell(x^{\ell+1}) + \frac{1}{2}\mu_L|x^{\ell+2} - x^{\ell+1}|^2 \leq L_{\ell+1}(x^{\ell+2})\,, \text{ and} \tag{31}$$

$$L_\ell(x^{\ell+1}) + \frac{1}{2}\mu_L|x^{\ell+1} - \hat{x}|^2 = L_\ell(\hat{x}) \leq f(\hat{x}) + E_{max}^g\,. \tag{32}$$

From (32) and (31), we deduce that the sequence $\left\{L_\ell(x^{\ell+1})\right\}_{\ell \geq L}$ is non-decreasing and bounded. Hence,

$$\exists L^\infty < \infty : L_\ell(x^{\ell+1}) \to L^\infty \text{ and } |x^{\ell+1} - x^\ell| \to 0 \text{ as } \ell \to \infty. \tag{33}$$

The sequence of null steps $\{x^\ell\}$ is bounded, by (32). Since the approximate subdifferential $\partial_{E_f^{max}+E_g^{max}} f(\cdot)$ is locally bounded (e.g. [15] Prop 6.2.2), $\{g^\ell\}$ is also bounded.

Since only null steps occur for $\ell > L$, the descent test is not satisfied and we have that $\left(f^1(x^{\ell+1}) + f^2_{x^{\ell+1}}\right) - \Phi_\ell(x^{\ell+1}) > (1-m)\delta_{\ell+1}$. However, since the model $\Phi_{\ell+1}$ includes the information returned by the oracle for the last iterate $x^{\ell+1}$,

$$\left(f^1(x^{\ell+1}) + f^2_{x^{\ell+1}}\right) \leq \Phi_{\ell+1}(x^{\ell+2}) + \langle \hat{g}^{\ell+1}, x^{\ell+1} - x^{\ell+2}\rangle\,,$$

so, together with the definition of partial linearization, we obtain that

$$\begin{aligned}\left(f^1(x^{\ell+1}) + f^2_{x^{\ell+1}}\right) - \Phi_\ell(x^{\ell+1}) &\leq \Phi_{\ell+1}(x^{\ell+2}) - \Phi_\ell(x^{\ell+1}) + |\hat{g}^{\ell+1}||x^{\ell+1} - x^{\ell+2}| \\ &= \Phi^{lin}_{\ell+1}(x^{\ell+2}) - \Phi^{lin}_{\ell}(x^{\ell+1}) + |\hat{g}^{\ell+1}||x^{\ell+1} - x^{\ell+2}| \\ &= L_{\ell+1}(x^{\ell+2}) - L_\ell(x^{\ell+1}) + |\hat{g}^{\ell+1}||x^{\ell+1} - x^{\ell+2}| \\ &\quad -\frac{1}{2}\mu_{\ell+1}|x^{\ell+2} - \hat{x}|^2 + \frac{1}{2}\mu_\ell|x^{\ell+1} - \hat{x}|^2\end{aligned}$$

Using (33) and $\mu_\ell \to \bar{\mu} \leq \mu_{max}$ we obtain that all terms on the right hand side vanish as $\ell \to \infty$. As a consequence, $\left(f^1(x^{\ell+1}) + f^2_{x^{\ell+1}}\right) - \Phi_\ell(x^{\ell+1}) \to 0$ as $\ell \to \infty$. Hence,

$$0 \leq (1-m)\delta_{\ell+1} < \left(f^1(x^{\ell+1}) + f^2_{x^{\ell+1}}\right) - \Phi_\ell(x^{\ell+1}) \to 0\,.$$

By (25), for $\ell > L$ we have that $V_\ell \leq \max\{\sqrt{2\mu_\ell\delta_{\ell+1}}, \delta_{\ell+1}\}$ since (21) holds. Hence, $V_\ell \to 0$ as $\ell \to \infty$. □

As a consequence, by (29), Proposition 2 implies that if a finite number of serious steps occurs, $\hat{x}$ is an approximate solution to (1) with

$$f(\hat{x}) \leq f^{opt} + \limsup_{\ell\to\infty}\left(\max_{j\in\mathcal{B}^2_\ell} E^g_{x^j}\right) + E^f_{\hat{x}}\,.$$

For comparison, when inaccuracies are fixed and a finite number of serious steps occurs, we recall the analogous result of [5]:

$$f(\hat{x}) \leq f^{opt} + E^g + E^f\,.$$

5.2.3 Infinite number of serious steps

When the sequence of serious steps is infinite, we denote by $\mathcal{L}_s$ the subset of iterations yielding serious steps: $\mathcal{L}_s = \{\ell \geq 1 : \hat{x}^{k(\ell+1)} = x^{\ell+1}\}$.

Proposition 3. *Suppose that Algorithm 1 generates an infinite sequence of serious steps. Then, if f is coercive, the sequence $\{\hat{x}^{k(\ell)}\}$ is bounded and $V_\ell \to 0$ as $\mathcal{L}_s \ni \ell \to \infty$.*

Proof. Recall that the coercivity assumption implies that f in (1) has bounded level sets and a finite optimal value, f^{opt}. By (11) and (12), for all $\ell \in \mathcal{L}_s$

$$f(\hat{x}^{k(\ell)}) - E^f_{max} \leq f^1(\hat{x}^{k(\ell)}) + f^2_{\hat{x}^{k(\ell)}} \leq f^1(\hat{x}^{k(1)}) + f^2_{\hat{x}^{k(1)}} \leq f(\hat{x}^{k(1)}) + E^g_{max}$$

and, hence, the serious point sequence is bounded, because $\{\hat{x}^{k(\ell)}\} \subset \{x : f(x) \leq f(\hat{x}^{k(1)}) + E^f_{max} + E^g_{max}\}$. Furthermore, the sequence $\left\{f^1(\hat{x}^{k(\ell)}) + f^2_{\hat{x}^{k(\ell)}}\right\}_{\ell\in\mathcal{L}_s}$ of approximate functional values has a finite limit $f^\infty \geq f^{opt} - E^f_{max}$. As a

result, summing for all $\ell \in \mathcal{L}_s$ the descent inequality for two successive serious points:

$$f^1(\hat{x}^{k(\ell+1)}) + f^2_{\hat{x}^{k(\ell+1)}} - \left(f^1(\hat{x}^{k(\ell)}) + f^2_{\hat{x}^{k(\ell)}} \right) \geq m\delta_{\ell+1} \,,$$

we see that the series $\sum_{\ell \in \mathcal{L}_s} \delta_{\ell+1}$ is finite with the corresponding terms $\delta_{\ell+1} \to 0$. Since (21) holds, we have that the (nonnegative) optimality measure satisfies $V_\ell \leq \max\{\sqrt{2\mu_{max}\delta_{\ell+1}}, \delta_{\ell+1}\}$ and $V_\ell \to 0$ for $\mathcal{L}_s \ni \ell \to \infty$, as desired. □

Hence, by (27), when Algorithm 1 declares Serious Steps an infinite number of iterations in a set $\mathcal{L}_s$, if f is coercive, then any accumulation point x^{acc} of the bounded sequence $\{\hat{x}^{k(\ell)}\}_{\ell \in \mathcal{L}_s}$ satisfies the relation

$$f(\hat{x}^{acc}) \leq f^{opt} + \limsup_{\ell \in \mathcal{L}_s} \left(\max_{j \in \mathcal{B}^2_\ell} E^g_{x^j} \right) + \limsup_{\ell \in \mathcal{L}_s} E^f_{\hat{x}^{k(\ell)}} \,.$$

For comparison, when inaccuracies are fixed and an infinite number of serious steps occurs, we recall the analogous result of [5]:

$$f(\hat{x}^{acc}) \leq f^{opt} + E^g + E^f \,.$$

6 Incremental classical bundle method

In the setting of an incremental bundle method, the error term $E^g_{x^\ell}$ is null and, consequently, all cutting planes stay below the real objective function (cf. (11)). Other examples of inexact oracles also correspond to this framework, for example in the context of Lagrangian relaxation where the subproblem involved in the evaluation of f is only solved approximately.

When the error vanishes along the iterative process, i.e., when $E^f_{x^\ell} \to 0$ as $\ell \to \infty$, we could expect to be able to find (asymptotically) an optimal solution, without any error.

If we apply the results presented in Section 5 to the particular setting $E^f_{x^\ell} \to 0$ but $E^g_{x^\ell} = 0$ for all ℓ, the following can be said for Algorithm 1:

- if a finite number of serious steps occurs, the last serious step $\hat{x}^{k(last)}$ provides an approximate solution with error on the objective lower than $E^f_{\hat{x}^{k(last)}}$;
- if an infinite number of serious steps occurs, any cluster point of the sequence of serious steps is solution to (1).

However, in this framework, as soon as the inexact oracle returns a value $f^1(x^{\ell+1}) + f^2_{x^{\ell+1}}$ below the infimum value of the cutting-planes model Φ_ℓ, the descent test is necessarily satisfied and $x^{\ell+1}$ qualifies as new serious point. In Algorithm 1, when the noise management mechanism is activated, the bundle does not change, so neither does the infimum value of the cutting-planes model. But then (recalling from (6) that the predicted descent is the difference between

the value returned by the oracle at the current serious point and the value of the model at the new point), we see that the predicted decrease will be negative for all subsequent iterations. As a consequence, the algorithm enters into an infinite noise management loop. In these circumstances, the last serious step $\hat{x}^{k(last)} = x^{\ell+1}$ would be an approximate solution to (1), with objective error smaller than $E^f_{\hat{x}^{k(last)}}$, by (29).

In summary, the inexact setting considered in Algorithm 1 is such that when the oracle returns underestimating hyperplanes, if the new candidate has a lower functional value than the one predicted by the model, then the descent test will never be satisfied, and there will be no more improvement. The variant described below addresses this issue.

6.1 Inexact classical bundle method

As a by-product of the analysis in Section 5.2, we studied the behaviour of an *inexact classical* bundle method, working with function evaluations of *unknown* and *vanishing* inaccuracy, but without noise attenuation step.

Our analysis starts from the following observation: when adding a new cutting plane to the bundle after a null step, the value $f^2_{\hat{x}^{k(\ell)}}$ may not be the best estimate available for $f^2(\hat{x}^{k(\ell)})$. Indeed, we have that

$$f(\hat{x}^{k(\ell)}) \geq \Phi_\ell(\hat{x}^{k(\ell)}) \geq f^1(\hat{x}^{k(\ell)}) + f^2_{\hat{x}^{k(\ell)}} \,.$$

The rightmost inequality is satisfied if the bundle $\mathcal{B}^{1,2}_\ell$ is forced to include the cut returned for the current serious point $\hat{x}^{k(\ell)}$. Denoting by $\ell(k)$ the index of the iterate that yielded the k-th serious step, this corresponds to the additional condition $\ell(k) \in \mathcal{B}^{1,2}_\ell$ imposed when updating the bundle in Algorithm 2 below. Hence, one could consider replacing the inexact value $\left(f^1(\hat{x}^{k(\ell)}) + f^2_{\hat{x}^{k(\ell)}}\right)$ by the, possibly better, value $\Phi_\ell(\hat{x}^{k(\ell)})$, since $\Phi_\ell(\hat{x}^{k(\ell)}) = \left(f^1(\hat{x}^{k(\ell)}) + f^2_{\hat{x}^{k(\ell)}}\right) + \max_{j\in\mathcal{B}^2_\ell}\{-e^2_{j,\ell}\}$. This results in an alternative definition for the predicted descent. More precisely,

$$\begin{aligned} \text{either } \delta_{\ell+1} &:= \left(f^1(\hat{x}^{k(\ell)}) + f^2_{\hat{x}^{k(\ell)}}\right) - \Phi_\ell(x^{\ell+1}) \text{ as in last section,} \\ \text{or } \hat{\delta}_{\ell+1} &:= \Phi_\ell(\hat{x}^{k(\ell)}) - \Phi_\ell(x^{\ell+1}) \text{ in this section.} \end{aligned} \tag{34}$$

As a consequence, in Step 6 of Algorithm 1 page 15, two different descent tests may be performed: $x^{\ell+1}$ qualifies as new serious point either if (20) holds, i.e., if

$$\left(f^1(\hat{x}^k) + f^2_{\hat{x}^k}\right) - \left(f^1(x^{\ell+1}) + f^2_{x^{\ell+1}}\right) \geq m\left(f^1(\hat{x}^{k(\ell)}) + f^2_{\hat{x}^{k(\ell)}} - \Phi_\ell(x^{\ell+1})\right)$$

or if

$$\Phi_\ell(\hat{x}^{k(\ell)}) - \left(f^1(x^{\ell+1}) + f^2_{x^{\ell+1}}\right) \geq m\left(\Phi_\ell(\hat{x}^{k(\ell)}) - \Phi_\ell(x^{\ell+1})\right) . \tag{35}$$

Actually, the two alternative definitions for predicted descent are related through the linearization errors $e^{1,2}_{j,\ell}$ used in the definition of the cutting-planes models

(14) and (15). Indeed, we have

$$\hat{\delta}_{\ell+1} \quad = \quad \delta_{\ell+1} + \max_{j \in \mathcal{B}_\ell^2} \{-e_{j,\ell}^2\} \, .$$

Similarly to (6), the predicted descent $\hat{\delta}_{\ell+1}$ can be expressed in terms of the aggregate subgradient, the prox-parameter and the linearization errors:

$$\hat{\delta}_{\ell+1} \quad = \quad \frac{1}{\mu_\ell} |\hat{g}^{\ell+1}|^2 + \hat{\varepsilon}_{\ell+1} + \max_{j \in \mathcal{B}_\ell^2} \{-e_{j,\ell}^2\} \, .$$

Furthermore, $\hat{\varepsilon}_{\ell+1}$ may be negative but, since $\hat{\varepsilon}_{\ell+1}$ is a convex combination of linearization errors by definition,

$$\hat{\varepsilon}_{\ell+1} + \max_{j \in \mathcal{B}_\ell^2} \{-e_{j,\ell}^2\} \geq 0.$$

As a consequence, if $\hat{\delta}_{\ell+1} \to 0$ and using the fact that $\mu_\ell \leq \mu_{max}$, we have that:

$$\begin{aligned} \hat{\varepsilon}_{\ell+1} + \max_{j \in \mathcal{B}_\ell^2} \{-e_{j,\ell}^2\} \quad &\to \quad 0 \text{ and} \\ |\hat{g}^{\ell+1}| \quad &\to \quad 0 \, , \end{aligned}$$

so the optimality measure becomes $\hat{V}_\ell := \max\{|\hat{g}^{\ell+1}|, \hat{\varepsilon}_{\ell+1} + \max_{j \in \mathcal{B}_\ell^2}\{-e_{j,\ell}^2\}\}$. For future use, write the subgradient inequality for $\hat{g}^{\ell+1} \in \partial \Phi_\ell(x^{\ell+1})$ (resulting from (13)), use that $\Phi_\ell \leq f$ (resulting from (18) with $E_x^g = 0$), and add $\pm f(\hat{x}^{k(\ell)}) \pm \langle \hat{g}^{\ell+1}, \hat{x}^{k(\ell)} \rangle$ to obtain for all $y \in \mathbb{R}^N$

$$f(y) \geq f(\hat{x}^{k(\ell)}) + \langle \hat{g}^{\ell+1}, y - \hat{x}^{k(\ell)} \rangle - \Big(f(\hat{x}^{k(\ell)}) - \Phi_\ell(x^{\ell+1}) - \frac{1}{\mu_\ell} |\hat{g}^{\ell+1}|^2 \Big) \, . \quad (36)$$

It is important to notice that, unlike $\delta_{\ell+1}$, the new predicted descent $\hat{\delta}_{\ell+1}$ is always nonnegative. Hence, the additional information used in the new definition avoids having to perform noise attenuation steps. Indeed, recall that for Algorithm 1, when $\delta_{\ell+1}$ is not considered large enough to cope with the inaccuracy, the prox-parameter μ_ℓ is decreased and a new trial point is generated, before creating any new cutting plane. Here, we consider instead a more (inexact) classical bundle method, where the usual expected decrease $\delta_{\ell+1}$ is replaced by $\hat{\delta}_{\ell+1}$.

Algorithm 2. *("inexact classical" bundle method)*

1. ***Initialization:*** *Choose* $x^1 \in \mathbb{R}^N$, $m \in (0,1)$ *and* $\mu_{\max} > 0$. *Compute* $(f^1(x^1), g^1(x^1))$, $(f^2_{x^1}, g^2_{x^1})$, *and define* $\check{f}^1_1$ *and* φ^2_1. *Set* $\ell = k = 1$, $k(\ell) = \ell(k) = 1$, *and* $\hat{x}^1 = x^1$.

2. ***New candidate:*** *solve* (13) *to get* $x^{\ell+1}$, $\hat{\delta}_{\ell+1}$, $\hat{g}^{\ell+1}$, *and* $\hat{\varepsilon}_{\ell+1}$.

3. ***Stopping test:*** *If* $\hat{V}_\ell = 0$, *stop.*

4. ***Oracle:*** *compute* $f^1(x^{\ell+1}), g^1(x^{\ell+1})$ *and* $f^2_{x^{\ell+1}}, g^2_{x^{\ell+1}}$

5. ***Descent test:*** *If* $f^1(x^{\ell+1}) + f^2_{x^{\ell+1}} \leq \Phi_\ell(\hat{x}^{k(\ell)}) - m\hat{\delta}_{\ell+1}$, *declare a Serious Step, otherwise declare a Null Step.*

   If $x^{\ell+1}$ *gave a serious step, set* $\hat{x}^{k+1} = x^{\ell+1}$, $f^2_{\hat{x}^{k+1}} = f^2_{x^{\ell+1}}$, $\ell(k+1) = \ell+1$, $k(\ell+1) = k+1$, $k = k+1$, *choose* $\mu_{\ell+1} \leq \mu_{\max}$.

   If $x^{\ell+1}$ *gave a null step,* $k(\ell+1) = k$, *choose* $\mu_{\ell+1} \in [\mu_\ell, \mu_{\max}]$.

6. ***Update:*** *Define* $\check{f}^1_{\ell+1}$ *and* $\varphi^2_{\ell+1}$ *according to the compression/selection rules of [14, Lemma 10.10] with* $\ell(k) \in \mathcal{B}^{1,2}_{\ell+1}$, *set* $\ell = \ell+1$, *loop to 2.*

6.2 Convergence analysis

In the following, we suppose Algorithm 2 does not stop and study separately the two possible asymptotic behaviours: infinite or finite number of serious steps.

6.2.1 Infinite number of serious steps

First, recall that in the case of an infinite sub-sequence of iterations for which $\hat{\delta}_{\ell+1} \to 0$, since

$$0 \leq \max_{j \in \mathcal{B}^2_\ell} \{-e^2_{j,\ell}\} \leq E^f_{\hat{x}^{k(\ell)}} \to 0\,,$$

we have that $\hat{\varepsilon}_{\ell+1} \to 0$ and $|\hat{g}^{\ell+1}| \to 0$ as $\ell \to \infty$. We now show that for the infinite sub-sequence of iterations yielding serious steps, denoted as before by $\mathcal{L}_s$, the corresponding predicted decrease $\hat{\delta}_{\ell+1}$ converges to 0.

In the new setting, when performing a serious step, we cannot guarantee that we obtain a better point: the relation $f(\hat{x}^{k(\ell)}) > f(\hat{x}^{k(\ell+1)})$ does not necessarily hold. Moreover, and contrary to the case developed in §5, neither can we say that the inexact values returned by the oracle are improving: we may have $\big(f^1(\hat{x}^{k(\ell)}) + f^2_{\hat{x}^{k(\ell)}}\big) \leq \big(f^1(\hat{x}^{k(\ell+1)}) + f^2_{\hat{x}^{k(\ell+1)}}\big)$. This explains the weaker result obtained for the new algorithm, when compared to Proposition 3, see §§ 6.2.3 below.

Proposition 4. *Assume that* $E^g_x = 0$, $E^f_{x^\ell} \to 0$ *as* $\ell \to \infty$, *and suppose that Algorithm 2 declares serious steps at infinitely many iterations in the set* $\mathcal{L}_s$.

If f has a finite optimal value f^{opt} and the corresponding sequence $\{\hat{x}^{k(\ell)}\}_{\mathcal{L}_s}$ is bounded, then at least one cluster point of the sequence solves (1).

Proof. Consider $\ell \in \mathcal{L}_s$. By (34) and (35), only the following inequality holds:

$$\Phi_\ell(\hat{x}^{k(\ell)}) - \left(f^1(\hat{x}^{k(\ell+1)}) + f^2_{\hat{x}^{k(\ell+1)}}\right) \quad \geq \quad m\hat{\delta}_{\ell+1}\,.$$

By (15) and the fact that $e^1_{j,\ell} = 0$ for $j = \ell(k)$, $\Phi_\ell(\hat{x}^{k(\ell)}) = f^1(\hat{x}^{k(\ell)}) + f^2_{\hat{x}^{k(\ell)}} + \max_{j\in\mathcal{B}^2_\ell}\{-(e^2_{j,\ell})\}$. Since, in addition, by (17) written with $E^g_x \equiv 0$, $\max_{j\in\mathcal{B}^2_\ell}\{-(e^2_{j,\ell})\} \leq E^f_{\hat{x}^{k(\ell)}}$, it follows that

$$\left(f^1(\hat{x}^{k(\ell)}) + f^2_{\hat{x}^{k(\ell)}}\right) - \left(f^1(\hat{x}^{k(\ell+1)}) + f^2_{\hat{x}^{k(\ell+1)}}\right) \geq m\hat{\delta}_{\ell+1} - E^f_{\hat{x}^{k(\ell)}}\,. \tag{37}$$

We show that $\liminf_{\ell\in\mathcal{L}_s} \hat{\delta}_{\ell+1} = 0$. Suppose, for contradiction purposes, that $\liminf_{\ell\in\mathcal{L}_s} \hat{\delta}_{\ell+1} > 0$. Then, since $E^f_{\hat{x}^{k(\ell)}} \to 0$, there exists $\eta_0 > 0$ and an iteration $\ell_0 \in \mathcal{L}_s$ such that $\left(m\hat{\delta}_{\ell+1} - E^f_{\hat{x}^{k(\ell)}}\right) \geq \eta_0$ for all $\mathcal{L}_s \ni \ell \geq \ell_0$. However, summing up inequality (37) for $\ell \geq \ell_0$ would result in having

$$\liminf_{\ell\in\mathcal{L}_s} \left(f^1(\hat{x}^{k(\ell)}) + f^2_{\hat{x}^{k(\ell)}}\right) = -\infty,$$

a contradiction, because f is bounded from below by assumption. Hence, $\liminf \hat{\delta}_{\ell+1} = 0$. Let us denote by $\mathcal{J}$ a subset of iterates in $\mathcal{L}_s$ such that $\hat{\delta}_{\ell+1} \to 0$ as $\mathcal{J} \ni \ell \to \infty$. Let x^{acc} be a cluster point of $\{\hat{x}^{k(\ell)}\}_{\mathcal{J}}$ and $\mathcal{H} \subset \mathcal{J}$ the subset of iterates such that $\hat{x}^{k(\ell)} \to x^{acc}$ as $\mathcal{H} \ni \ell \to \infty$. Since $\hat{\delta}_{\ell+1} \to 0$ as $\mathcal{J} \supset \mathcal{H} \ni \ell \to \infty$, both $|\hat{g}^{\ell+1}|$ and $\hat{\varepsilon}_{\ell+1} \to 0$ as $\mathcal{H} \ni \ell \to \infty$. Now rewrite the rightmost term in (36) using (6):

$$f(\hat{x}^{k(\ell)}) - \Phi_\ell(x^{\ell+1}) - \frac{1}{\mu_\ell}|\hat{g}^{\ell+1}|^2 = f(\hat{x}^{k(\ell)}) - (f^1(\hat{x}^{k(\ell)}) + f^2_{\hat{x}^{k(\ell)}}) + \hat{\varepsilon}_{\ell+1},$$

and pass to the limit in the reformulated inequality (36). The result follows, because $E^f_{x^\ell} \to 0$. □

Summing up, if an infinite number of serious steps occurs in Algorithm 2 and inaccuracy is vanishing, then if the sequence of serious points is bounded, it has at least one cluster point x^{acc} that solves (1).

6.2.2 Finite number of serious steps

Proposition 5. *Assume that $E^g_x = 0$, $E^f_{x^\ell} \to 0$ as $\ell \to \infty$ and suppose that, after some iteration $\ell last$, no serious step is declared in Algorithm 2. Then, the last serious point $\hat{x} = \hat{x}^{k(\ell last)}$ is a solution to* (1) *with $x^{\ell+1} \to \hat{x}$ as $\ell \to \infty$.*

Proof. Suppose there exists an iteration $\ell last$ such that for all $\ell > \ell last$, only null steps occur with $\mu_{max} \geq \mu_{\ell+1} \geq \mu_\ell$ and the serious point does not change, i.e. $\hat{x}^{k(\ell)} =: \hat{x}$. Let us show that $\hat{\delta}_\ell \to 0$ as $\ell \to \infty$ and that the last serious point is a solution to (1). As in the previous section, we use the partial linearization of the objective of the quadratic program (13): $L_\ell(x) := \Phi_\ell^{lin}(x) + \frac{1}{2}\mu_\ell |x - \hat{x}^{k(\ell)}|^2$. Again, by (31) and (32), we have that

$$L_\ell(x^{\ell+1}) \uparrow L_\infty < \infty \quad \text{and} \quad |x^{\ell+1} - x^\ell| \to 0\,,$$

with $\{x^\ell\}$ and $\{g^\ell\}$ bounded, and $\limsup f^1(x^{\ell+1}) + f^2_{x^{\ell+1}} - \Phi_\ell(x^{\ell+1}) \leq 0$. Furthermore, since the descent test does not hold for $\ell \geq \ell last$,

$$0 \leq (1-m)\hat{\delta}_{\ell+1} < f^1(x^{\ell+1}) + f^2_{x^{\ell+1}} - \Phi_\ell(x^{\ell+1})\,.$$

As a consequence,

$$\hat{\delta}_{\ell+1} = \hat{\varepsilon}_{\ell+1} + \frac{1}{\mu_\ell}|\hat{g}^{\ell+1}|^2 + \max_{j \in \mathcal{B}^2_\ell}\{-e^2_{j,\ell}\} \to 0\,.$$

Since $\hat{\varepsilon}_{\ell+1} + \max_{j\in\mathcal{B}^2_\ell}\{-e^2_{j,\ell}\} \geq 0$ and $\mu_\ell \leq \mu_{max}$, we have that $|\hat{g}^{\ell+1}|^2 \to 0$. Furthermore, $x^{\ell+1} = \hat{x} - \frac{1}{\mu_\ell}\hat{g}^{\ell+1}$, so $x^{\ell+1} \to \hat{x}$. Since the model is below the function and it always includes oracle information for the last iterate,

$$\begin{array}{rcl} f(\hat{x}) & \geq & \Phi_{\ell+1}(\hat{x}) \geq f^1(x^{\ell+1}) + f^2_{x^{\ell+1}} + \langle g^{\ell+1}, \hat{x} - x^{\ell+1}\rangle \\ & \geq & f(x^{\ell+1}) - E^f_{x^{\ell+1}} + \langle g^{\ell+1}, \hat{x} - x^{\ell+1}\rangle\,, \end{array}$$

with $\{g^{\ell+1}\}$ bounded. When passing to the limit as $\ell \to \infty$, the last inequality converges to $f(\hat{x})$, so $\Phi_{\ell+1}(\hat{x}) \to f(\hat{x})$ too. But $\Phi_\ell(\hat{x}) \to f(\hat{x})$ implies that $f(\hat{x}) - \Phi_\ell(x^{\ell+1}) \to 0$ (using that $\hat{\delta}_{\ell+1} \to 0$ in (34)). Therefore, by (36), the last serious step $\hat{x}$ is a solution to (1). □

Summing up, if an finite number of serious steps occurs in Algorithm 2 and inaccuracy is vanishing, then the last stability center $\hat{x}$ is a solution of (1).

Since inaccuracy vanishes only asymptotically, if at some iteration ℓ the algorithm stops, the last serious iterate is an approximate solution, in the sense of (28), written with $E^g_{x^j} = 0$ for all j.

We have shown that an almost classical bundle method may achieve optimality when linearizations are computed with inexact but vanishing inaccuracies. Our setting does not involve any noise management step, but rather incorporates different incoming information for the predicted descent. However, our analysis needs the serious iterates sequence to remain bounded. We comment now on such assumption and discuss how to address this issue, that is not uncommon for nonmonotone methods and introduces an additional "unboundedness management" step in the algorithm.

6.2.3 Ensuring boundedness of serious iterates

In exact bundle methods, boundedness of the sequence of serious iterates follows from coercivity, because the objective function has bounded level sets and the sequence $\{f(\hat{x}^k)\}$ is monotonically decreasing. The issue is more involved and subtle when dealing with inexact oracles, even for vanishing inaccuracies, as assumed in this section. We are grateful to Krzysztof C. Kiwiel for pointing out this fact, because his remark allowed us to correct a wrong statement in Proposition 4, as well as various imprecise or confusing statements Sections 5 and 6.

To force serious points to form a bounded sequence, a "boundedness control" step should be incorporated into Algorithm 2 before calling the oracle. Such step manages unboundedness in a manner similar to how Step 4 in Algorithm 1 manages noise:

> **Unboundedness detection:** If for some positive parameter η, set at the Initialization step,
>
> $$\Phi_\ell(x^{\ell+1}) > f^1(x^1) + f^2_{x^1} + E^f_{max} + \eta\,, \tag{38}$$
>
> the candidate point is deemed to have an unduly large inexact functional value. Then, we set $UD = 1$, $\mu_{\ell+1} = \kappa\mu_\ell$, $\ell = \ell + 1$, and go back to Step 2, to find a new candidate point.
> Like the noise atenuation parameter in Algorithm 1, if $UD = 1$ then the prox-parameter cannot increase at null steps, and UD is reset to 0 at serious steps.

Note that since candidates $x^{\ell+1}$ solve (13), it holds that

$$\begin{aligned}\Phi_\ell(x^{\ell+1}) + \frac{\mu_\ell}{2}|x^{\ell+1} - \hat{x}^{k(\ell)}|^2 &\leq \Phi_\ell(x^1) + \frac{\mu_\ell}{2}|x^1 - \hat{x}^{k(\ell)}|^2 \\ &\leq f^1(x^1) + f^2_{x^1} + E^f_{max} + \frac{\mu_\ell}{2}|x^1 - \hat{x}^{k(\ell)}|^2\,,\end{aligned}$$

because $\Phi_\ell(x^1) \leq f^1(x^1) + f^2_{x^1} + E^f_{max}$ by (10) and (18). This implies that (38) cannot hold if $\mu_\ell|x^1 - \hat{x}^{k(\ell)}|^2 \leq 2\eta$. As a consequence, if after some iteration $\ell last$ there is a final serious step $\hat{x}$, there cannot be an infinite number of iterations $\ell > \ell last$ such that (38) holds, otherwise μ_ℓ would be driven to 0, which whould yield a contradiction.

Hence, we see that when the modified Algorithm 2 generates a last serious iterate, since only a finite number of steps controlling unboundedness may follow, eventually prox-parameters at null steps remain bounded away from 0 and form a nondecreasing sequence, so Proposition 5 applies.

As for serious steps, the following lemma shows that nonsatisfaction of (38), together with the descent test, ensures that the sequence $\{f^1(\hat{x}^{k(\ell)}) + f^2_{\hat{x}^{k(\ell)}}\}$, of serious iterates inexact functional values, is bounded above, as shown in Lemma 2 below. As a result, if f is coercive, the sequence $\{\hat{x}^{k(\ell)}\}$ is bounded and the assumption in Proposition 4 holds.

Remark: The test (38) uses the value E^f_{max}, possibly unknown in our setting. If this value is unknown, one may force an exact evaluation for the first call to the oracle. In that case, E^f_{max} could be suppressed in (38).

Lemma 2. *Suppose that Algorithm 2 with additional step of* ***Unboundedness detection*** *generates an infinite sequence of serious steps. If the function f is coercive, then the corresponding sequence $\{\hat{x}^{k(\ell)}\}$ is bounded.*

Proof. For convenience, let F denote the right hand side value in (38) and let $\hat{x}^k = \hat{x}^{k(\ell)}$ denote each serious point. We will prove by induction that for $k \geq 2$,

$$f_{\hat{x}^k} \leq F + E^f_{max} \sum_{j=1}^{k-1} (1-m)^j \,.$$

Recall that, by assumption, $E^f_{x^\ell}$ is bounded by E^f_{max}. By construction, the starting point is the first serious point and satisfies $\Phi_1(\hat{x}^1) = f_{\hat{x}^1} \leq F$. Now, denote by ℓ_2 the first iterate such that the descent test (35) is satisfied: $\hat{x}^2 = x^{\ell_2+1}$. Both inequalities below hold:

$$f_{\hat{x}^2} \leq \Phi_{\ell_2}(\hat{x}^1) - m\hat{\delta}_{\ell_2+1} \text{ and } \Phi_{\ell_2}(\hat{x}^2) \leq F \,.$$

Since $\hat{\delta}_{\ell_2+1} = \Phi_{\ell_2}(\hat{x}^1) - \Phi_{\ell_2}(\hat{x}^2)$, we have that $f_{\hat{x}^2} \leq (1-m)\Phi_{\ell_2}(\hat{x}^1) + mF$; and, since the model is below the function,

$$f_{\hat{x}^2} \leq (1-m) f(\hat{x}^1) + mF \leq F \,.$$

The inductive process starts with $f_{\hat{x}^2} \leq F + (1-m)E_{\hat{x}^1}$.

Suppose now that the inequality holds for some arbitrary $k \geq 2$. Let ℓ denote the iterate such that candidate $x^{\ell+1}$ is chosen as new serious step: $\hat{x}^{k+1} := x^{\ell+1}$. Again, by (35) and recalling that the model is below the function,

$$f_{\hat{x}^{k+1}} \leq \Phi_\ell(\hat{x}^k) - m\hat{\delta}_{\ell+1} \text{ and } \Phi_\ell(\hat{x}^{k+1}) \leq F \,.$$

Since $\hat{\delta}_{\ell+1} = \Phi_\ell(\hat{x}^k) - \Phi_\ell(\hat{x}^{k+1})$, we have that $f_{\hat{x}^{k+1}} \leq (1-m)\Phi_\ell(\hat{x}^k) + mF$, with $\Phi_\ell(\hat{x}^k) \leq f(\hat{x}^k) \leq f_{\hat{x}^k} + E^f_{max}$. By the induction hypothesis,

$$f_{\hat{x}^{k+1}} \leq (1-m)(F + E^f_{max} \sum_{j=1}^{k-1} (1-m)^j + E^f_{max}) + mF$$

and, hence, $f_{\hat{x}^{k+1}} \leq F + E^f_{max} \sum_{j=1}^{k} (1-m)^j$. As a consequence, for any $k \geq 1$, $f(\hat{x}^{k+1}) \leq F + (1 + \frac{1}{m}) E^f_{max}$. The level set of f being bounded by coercivity, the corresponding sequence $\{\hat{x}^k\}$ is bounded. □

7 Application to mid-term energy planning

We assess the incremental approach by implementing it in a mid-term generation planning tool used by EDF for optimizing nuclear generation and described

in [18]. The methodology of this computational tool involves applying price decomposition on a scenario tree, as presented in section 2. Our test problem is a real dataset with 58 nuclear plants, 85 classical thermal plants, and an aggregate representation for the spot market. The optimization horizon covers one year with three time steps per day, resulting in a scenario tree with about 20.000 nodes, computed out of 484 historical scenarios of demand and spot prices. The tests were made on a SUN Sparc, Sun-Fire-V490.

As mentioned, classical thermal sub-functions correspond to subproblems that are easy to solve: at each node, the optimal generation is either the maximal power of the plant or it is null. Similarly for the spot market, modeled as a set of thermal plants with stochastic costs. For nuclear plants however, computations are much more involved. Recall that maximal and minimal power at a given node depend on the current level of combustible in the reactor. The amount of *modulation* (power not at its maximum) between two outage periods is bounded above, as in a ramp constraint. The corresponding optimization problem is a large scale linear program solved by a commercial solver. The CPU-time needed for solving such linear program, and hence, computing the nuclear sub-function, is comparable to one iteration of the bundle method. A more sophisticated model considers, instead of ramp constraints, the use of integer variables and constraints to limit the number of modulation periods over a generation campaign. In order to allow for such future evolution of the computational tool, we solved the nuclear subproblems by using dynamic programming techniques on a discretized grid. For large-scale instances of nuclear subproblems, Dynamic Programming has shown to be competitive with linear programming solvers in term of computing times.

When applying a classical bundle method over the mid-term generation problem, hence evaluating all sub-functions after each dual iteration, 95% of the total CPU-time is spent in solving the nuclear subproblems. Therefore, there is a potential gain in applying the incremental approach.

The incremental bundle method from Algorithm 1 was coded by adding an additional noise management step over a classical bundle method, in a disaggregate variant. More precisely, the classical bundle method used for comparisons is a disaggregate implementation of the proximal bundle method with poor man quasi-Newton update from [13] that can exploit sparsity and was made by the authors for EdF.

Parameters for the algorithm are: $m = 0.1\,, \kappa = 10\,, \mu_{max} = 10^9$.The starting point x_0 is chosen as the vector of marginal costs of the production mix on the scenario tree with a simplified optimal strategy where stock constraints are disregarded. The initial prox-parameter μ_0 is an approximation of the curvature of the dual function at x_0, computed as in [19, Section 6]. For the incremental setting, we split the 58 nuclear plants into 3 subsets with 20/20/18 plants, and skip the computation of sub-functions for one of such subsets at each iteration. We noticed that performing dual steps with partial information at the very beginning of the procedure yielded rather poor results. Hence, we decided to start the incremental approach only after 20 dual iterations (with exact evaluation) have been done.

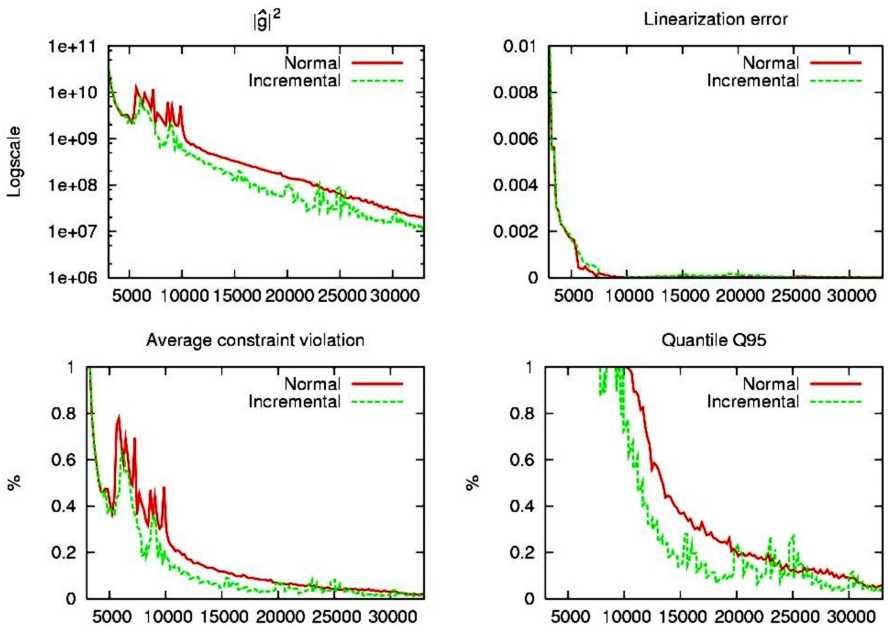

Figure 2: Incremental bundle method

Figure 2 reports the results on various indicators. In the graphs, *plain red lines* refer to the classical bundle method, and *dashed green lines* to the incremental approach. For all indicators, CPU-times given on the horizontal axis are clock ticks of the processor. In the upper graph, we plotted $|\hat{g}|^2$ and $\hat{\varepsilon}$. In view of (6) and (23), these two parameters are usually used in the stopping test for both exact and inexact bundle methods. For an exact bundle method, approximate convergence is achieved when both values reach the corresponding tolerance chosen by the user. We see on these plots that, as usual for bundle methods, the parameter that decreases less easily is the L^2-norm of the aggregate subgradient. For our application, a tolerance of 10^8 for $|\hat{g}|^2$ is usually chosen. This corresponds to a tolerance on the dualized constraints of merely 20MW on each node of the scenario tree, which is negligible regarding the average level of power load ($\simeq 50.000MW$). The same precision is obtained with the incremental approach, but using 25% less CPU-time.

In the bottom section of Figure 2, we plotted two indicators of the overall quality of the approximate primal solution $\hat{p}$. Namely, the average violation of demand constraint and the 95%-th quantile of the distribution of violations over the scenario tree. These values are related to $\hat{g}$ so, not surprisingly, we see that the incremental approach performs better than the classical bundle method. However, we notice that the distribution of demand violations is more volatile.

These preliminary results seem to indicate advantages of the incremental

method. The new algorithmic scheme achieves a given tolerance in less computing time for a realistic mid-term power problem. However, more tests are currently being done on other data sets, with e.g. bigger scenario trees. Furthermore, we believe that significant improvements may be obtained with a tighter disaggregation of the bundle, and perhaps with a more clever strategy to select which subset of subproblems to evaluate and which to skip in the incremental approach.

Concluding Remarks

In this work, we introduced a general incremental bundle method for structured non-smooth optimization problems, with objective function written as the sum of sub-functions. With the incremental approach, iterations can be done without evaluating all of the sub-functions. This is a particularly interesting feature when the time required for a bundle iteration is negligible when compared to the CPU-time needed for the sub-functions evaluation. Such is the case in many complex problems solved by decomposition methods, yielding a dual master program that requires minimizing a structured non-smooth convex function. Our first numerical experiments on a real-life problem of mid-term power planning show that, when compared to a standard bundle method, the incremental variant can achieve faster convergence without losing precision.

Convergence properties of the incremental variant were derived by interpreting the method as a special instance of inexact bundle methods. We also analyzed an alternative algorithm, which does not need to manage noise, the inexact classical bundle method. Convergence properties of this algorithm are based on the capacity of the inexact oracle to drive inaccuracy to 0. For an inexact oracle that substitutes lacking information by the evaluation of the cutting-planes model, it would be interesting to find conditions under which the algorithm itself can force such inaccuracies to vanish asymptotically. At this stage, we can only conjecture that such could be the case for a polyhedral function f, like a dual function, as in our application.

References

[1] Solodov, M., Zavriev, S.: Error stability properties of generalized gradient-type algorithms. J. Optim. Theory Appl. **98**(663 - 680) (1998)

[2] Nedič, A., Bertsekas, D.: Incremental subgradient methods for nondifferentiable optimization. SIAM J. on Optimization **12**(1), 109–138 (2001)

[3] Kiwiel, K.: Convergence of approximate and incremental subgradient methods for convex optimization. SIAM J. on Optimization **14**(3), 807 – 840 (2003)

[4] Gaudioso, M., Giallombardo, G., Miglionico, G.: An incremental method for solving convex finite min-max problems. Math. Oper. Res. **31**(1), 173–187 (2006)

[5] Kiwiel, K.: A proximal bundle method with approximate subgradient linearizations. SIAM J. on Optimization **16**(4), 1007–1023 (2006)

[6] Kiwiel, K.: An algorithm for nonsmooth convex minimization with errors. Mathematics of computation **45**(171) (1985)

[7] Kiwiel, K.: Approximations in proximal bundle methods and decomposition of convex programs. J. Optim. Theory Appl. **84**(3), 529–548 (1995)

[8] Hintermüller, M.: A proximal bundle method based on approximate subgradients. Comput. Optim. Appl. **20**(3), 245–266 (2001)

[9] Solodov, M.: On approximations with finite precision in bundle methods for nonsmooth optimization. J. Optim. Theory Appl. **119**(1), 151 – 165 (2003)

[10] Kiwiel, K., Lemaréchal, C.: An inexact bundle variant suited to column generation. Math. Program. Ser. A. (2007)

[11] Heitsch, H., Römisch, W., Strugarek, C.: Stability of multistage stochastic programs. SIAM Journal on Optimization **17**(2), 511–525 (2006)

[12] Bacaud, L., Lemaréchal, C., Renaud, A., Sagastizábal, C.: Bundle methods in stochastic optimal power management: a disaggregate approach using preconditioners. Comp. Opt. and App. **20**(3), 227 – 244 (2001)

[13] Lemaréchal, C., Sagastizábal, C.: Variable metric bundle methods: from conceptual to implementable forms. Math. Program. **76**(3), 393–410 (1997)

[14] Bonnans, J., Gilbert, J., Lemaréchal, C., Sagastizábal, C.: Numerical Optimization: Theoretical and Practical Aspects. Springer (2006)

[15] Hiriart-Urruty, J., Lemaréchal, C.: Convex Analysis and Minimization Algorithms 2. Springer (1996)

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

[17] Feltenmark, S., Kiwiel, K.: Dual applications of proximal bundle methods, including Lagrangian relaxation of nonconvex problems. SIAM J. on Optimization **10**(3), 697–721 (2000)

[18] Avellà–Fluvià, M., Boukir, K., Martinetto, P.: Handling a CO2 reservoir in mid term generation scheduling. Proccedings 15th Power System Computation Conference (2005)

[19] Mifflin, R., Sagastizábal, C.: A $\mathcal{VU}$-algorithm for convex minimization. Math. Program. **104**(2-3), 583–608 (2005)