PIECEWISE QUADRATIC APPROXIMATIONS IN CONVEX NUMERICAL OPTIMIZATION

A. ASTORINO*, A. FRANGIONI‡, M. GAUDIOSO†, AND E. GORGONE†

Abstract. We present a bundle method for convex nondifferentiable minimization where the model is a piecewise quadratic convex approximation of the objective function. Unlike standard bundle approaches, the model only needs to support the objective function from below at a properly chosen (small) subset of points, as opposed to everywhere. We provide the convergence analysis for the algorithm, with a general form of master problem which combines features of trust-region stabilization and proximal stabilization, taking care of all the important practical aspects such as proper handling of the proximity parameters and of the bundle of information. Numerical results are also reported.

Key words. NonDifferentiable Optimization, Bundle methods, Quadratic model

AMS subject classifications. 90C26, 65K05

1. Introduction.

We are interested in the numerical solution of the problem

$$f^* \;=\; \inf \{\; f(x) \;:\; x \in \mathbb{R}^n \;\}$$

where $f : \mathbb{R}^n \to \mathbb{R}$ is convex, not necessary differentiable, and only known through an "oracle" which, given any $\bar{x} \in \mathbb{R}^n$, returns the value $f(\bar{x})$ and one subgradient $g \in \partial f(\bar{x})$. The method we will develop can be easily adapted to the case where constraints $x \in X$ are added to the problem for a known and "relatively easy" convex set X, or, alternatively, f is an extended-value function and the oracle can provide tight defining inequalities for its effective domain X; there are several ways to perform the necessary modifications (e.g. [18, 7, 20, 12]) that will not be discussed here for the sake of notational simplicity. Also, techniques developed to cope with inexact computation of the objective function [19] and/or of the constraints [21] can be adapted to the new algorithm; again, we refrain from doing this in order to focus on the fundamental differences with standard approaches of the same class.

All bundle methods are based on the idea of sampling the space in a sequence of *tentative points* x_i, collecting the corresponding set of triples $(x_i, f(x_i), g_i)$ with $g_i \in \partial f(x_i)$. We will denote by $\mathcal{B}$ the currently available set of triples or equivalently, with a slight abuse of notation, the set of their indices; upon first reading one may assume that $\mathcal{B} = \{\; 0, 1, \ldots, k \;\}$, where k is the current iteration of the algorithm,

*Istituto di Calcolo e Reti ad Alte Prestazioni C.N.R., c/o Dipartimento di Elettronica Informatica e Sistemistica, Università della Calabria, 87036 Rende (CS), Italia. E-mail: astorino@icar.cnr.it

†Dipartimento di Elettronica Informatica e Sistemistica, Università della Calabria, 87036 Rende (CS), Italia. E-mail: gaudioso@deis.unical.it, egorgone@deis.unical.it

‡Dipartimento di Informatica, Università di Pisa, Polo Universitario della Spezia, Via dei Colli 90, 19121 La Spezia, Italy. E-mail: frangio@di.unipi.it

although in practice things are more complex, as discussed below. The standard use of the "bundle" $\mathcal{B}$ is that of constructing the *cutting plane model*

$$\hat{f}_{\mathcal{B}}(x) \;=\; \max \{\; f(x_i) + g_i(x - x_i) \;:\; i \in \mathcal{B} \;\}$$

which estimates the objective function from below, i.e., $\hat{f}_{\mathcal{B}} \leq f$. This is used to drive the choice of the next iterate, clearly in the region where $\hat{f}_{\mathcal{B}}$ improves over the best value found so far. It is well-known that some form of *stabilization* is needed for this process, if only because $\hat{f}_{\mathcal{B}}$ may well be bounded below. In the simplest form (that of *proximal* bundle methods), this takes the shape of a distinguished point $y \in \mathbb{R}^n$, e.g. the best iterate found so far, which leads to the definition of the translated model

$$\hat{f}_{\mathcal{B}}(d) \;=\; \hat{f}_{\mathcal{B}}(y + d) - f(y) \;=\; \max \{\; g_i d - \alpha_i \;:\; i \in \mathcal{B} \;\}$$

where $\alpha_i = f(y) - f(x_i) - g_i(y - x_i) \geq 0$ is the *linearization error* of g_i w.r.t. the *stability center* y. Note that in so doing one may avoid to store the iterate x_i in $\mathcal{B}$, as the linearization errors can be easily updated with the well-known *information transport property* when y changes. Then, for an appropriately chosen *proximity parameter* $\rho > 0$, one finds the optimal solution d^* of the *master problem*

$$\begin{array}{l} \min_d \;\{\; \hat{f}_{\mathcal{B}}(d) + \rho\|d\|^2/2 \;\} \\ = \min_{v,d} \;\{\; v + \rho\|d\|^2/2 \;:\; v \geq\; g_i d - \alpha_i \quad i \in \mathcal{B} \;\} \end{array} \tag{1.1}$$

and probes $y + d^*$ as the next iterate. The dual of (1.1)

$$\min_\lambda \;\left\{\; \tfrac{1}{2\rho} \left\|\textstyle\sum_{i\in\mathcal{B}} g_i\lambda_i\right\|^2 + \textstyle\sum_{i\in\mathcal{B}} \alpha_i\lambda_i \;:\; \textstyle\sum_{i\in\mathcal{B}} \lambda_i = 1 \;,\; \lambda \geq 0 \;\right\} \tag{1.2}$$

is also relevant. From the algorithmic viewpoint, the optimal solution λ^* of (1.2) reveals the *aggregated subgradient and linearization error*

$$z^* = \textstyle\sum_{i\in\mathcal{B}} g_i\lambda_i^* \qquad , \qquad \sigma^* = \textstyle\sum_{i\in\mathcal{B}} \alpha_i\lambda_i^* \tag{1.3}$$

which also provide $d^* = -(1/\rho)z^*$ and $v^* = -\|z^*\|^2/\rho - \sigma^*$; thus, dual approaches to (1.1) are possible, and are in fact often preferred, especially if n is "large" w.r.t. $|\mathcal{B}|$ [5]. From the analytic viewpoint, since (as it is easy to verify) $g_i \in \partial_{\alpha_i} f(y)$, one has that $z^* \in \partial_{\sigma^*} f(y)$; thus, whenever both $\|z^*\|$ and σ^* are "small", an approximate optimality condition is reached.

Several variants of this approach are possible. For instance, different forms of stabilization ([25, 7] and many others) can be used with only slight modifications to the master problems, and next to none to the convergence theory [7]. In particular, we mention here that the *trust region* version of the "proximal" master problem (1.1)

$$\begin{array}{l} \min_d \;\{\; \hat{f}_{\mathcal{B}}(d) \;:\; \gamma\|d\|^2 \leq 2 \;\} \\ = \min_{v,d} \;\{\; v \;:\; v \geq\; g_i d - \alpha_i \quad i \in \mathcal{B} \;,\; \gamma\|d\|^2 \leq 2 \;\} \end{array} \tag{1.4}$$

leads to the similar dual master problem

$$\min_{\lambda,\mu} \left\{ \tfrac{1}{2\mu} \left\| \sum_{i\in\mathcal{B}} g_i\lambda_i \right\|^2 + \sum_{i\in\mathcal{B}} \alpha_i\lambda_i + \mu/\gamma \ : \ \lambda\in\Lambda \ , \ \mu\geq 0 \right\} \tag{1.5}$$

(where $\Lambda = \{ \ \lambda \geq 0 \ : \ \sum_{i\in\mathcal{B}} \lambda_i = 1 \ \}$ is the unitary simplex of appropriate dimension), with analogous primal-dual relationships $d^* = -(1/\mu^*)z^*$ and $v^* = -\|z^*\|^2/2\mu^* - \sigma^* - \gamma\mu^*$. This particular approach has not received much attention in the past, most likely due to the fact that (1.4) is a quadratically constrained linear problem, hence potentially more difficult to solve in practice than the linearly constrained quadratic problem (1.1); analogously, (1.5) presents a fractional term in the objective function which makes it more difficult to deal with than (1.2). Yet, it is worth remarking that (1.5) can be rewritten as

$$\min_{\lambda,\mu,t} \left\{ t + \sum_{i\in\mathcal{B}} \alpha_i\lambda_i + \mu/\gamma \ : \ t\mu \geq \left\| \sum_{i\in\mathcal{B}} g_i\lambda_i \right\|^2 \ , \ \lambda\in\Lambda \ , \ \mu\geq 0 \right\}$$

which, using the well-known trick $t\mu = (t+\mu)^2/4 - (t-\mu)^2/4$, can in turn be cast as a Second-Order Cone Program (SOCP). Thus, the trust-region version of the bundle algorithm is well within the realm of practical implementability.

All these approaches employ the cutting plane model, mainly because it provides a lower approximation of f *everywhere*. This feature is not, however, necessary to construct a convergent algorithm. For instance, the recent [31] uses a different model $\psi_y(x)$ which is not in general a lower approximation to f but which "conserves the sign of $f(x) - f(y)$", in the sense that if $f(x) \leq f(y)$ then $\psi_y(x) \leq 0$, whereas if $f(x) > f(y)$ then $\psi_y(x) > 0$. The possibility of employing different models for f is potentially relevant in view of the fact that, despite being useful in several applications ([2, 4, 9] among the many others), these algorithms can be painfully slow both in theory and in practice. This is not surprising, as the piecewise-linear representation of the curvature of f contained in the model $\hat{f}_{\mathcal{B}}$ is clearly far less efficient, especially around a local optima, than that of the second-order model of Newton-type approaches. Whence the push towards second-order bundle-type algorithms [30, 33] which, however, are hindered by the complexity of second-order objects in the nondifferentiable case. It can be shown that, locally to each point, $\mathbb{R}^n$ can be partitioned into the subspace where f is essentially smooth, and therefore second-order approaches converge rapidly, and the subspace where f is essentially "kinky", and therefore accumulation of linear inequalities is efficient. This $\mathcal{VU}$-theory [28] allows to develop, under appropriate assumptions, second-order-type approaches that are rapidly convergent both in theory and in practice [29]. However, these approaches are not easy to analyze and implement.

Here we aim at a conceptually simpler approach which may ultimately lead to rapidly convergent algorithms. Since second-order objects are "piecewise in nature" in the nondifferentiable case [3, 17] one may want to develop a piecewise-smooth model of f. The most natural form is that of a piecewise-quadratic (convex) model [15];

however, such a model will by necessity loose the property employed in most bundle approaches, that is, being a lower approximation of the objective function. We will show that this is indeed doable, provided that the model, while actually overestimating f somewhere, never does so *knowingly* at least on a (potentially very small) set of points. In other words, we keep the property that the model underestimates f in a properly selected subset of the iterates x_i where the f-value is actually known. Doing so we retain global convergence of the approach under mostly the same technical conditions as ordinary bundle methods, with similar algorithmic options in the important aspects such as management of the parameters governing the stabilization and of $\mathcal{B}$. While the quadratic models we employ here are the simplest possible ones, this paves the way to algorithms using richer second-order information.

The structure of the paper is the following. In Section 2 we present the new model and discuss the properties of the corresponding master problems. In Section 3 we present the algorithm and discuss its convergence properties. In Section 4 we discuss the implementation issues of the approach and present our numerical results. Finally, in Section 5 we draw some conclusions and directions for future research.

The following standard notations are adopted throughout the paper. We denote by $\|\cdot\|$ the Euclidean norm in $\mathbb{R}^n$, by ab the standard inner product of the vectors a and b, by $\mathrm{dist}(x, A)$ the Euclidian distance of point x from the set A, and by $S_\delta(f) = \{\ x\ :\ f(x) \leq \delta\ \}$ the level set corresponding to the f-value δ.

2. The Piecewise-Qudratic Model. For every (ordered) pair $(i, j) \in \mathcal{B} \times \mathcal{B}$, the "mutual linearization error" computed in x_j for the i-th element of the bundle is

$$\alpha_{ij} = f(x_j) - f(x_i) - g_i(x_j - x_i)\ (\geq 0)\ ;$$

obviously, $\alpha_{ii} = 0$. For the quadratic expansion of f generated at point x_i

$$q_i(x) = f(x_i) + g_i(x - x_i) + \epsilon_i \|x - x_i\|^2/2$$

one has that

$$q_i(x_j) \leq f(x_j) \quad \Longleftrightarrow \quad \epsilon_i \leq \epsilon_{ij} = \frac{2\alpha_{ij}}{\|x_j - x_i\|^2}\ . \tag{2.1}$$

Consequently, let $\mathcal{I} \subseteq \mathcal{B}$ be an arbitrarily selected subset of the bundle containing the "important" (or "interpolating") points; by requiring that

$$0 \leq \epsilon_i \leq \min\{\ \epsilon_{ij}\ :\ j \in \mathcal{I}\ \} \qquad \forall\ i \in \mathcal{B} \tag{2.2}$$

we can rest assured that

$$f(x_j) \geq q_i(x_j) \qquad \forall\ (i, j) \in \mathcal{B} \times \mathcal{I}\ , \tag{2.3}$$

i.e., that no q_i *knowingly overestimates* f on the points in $\mathcal{I}$. It is convenient to take $\epsilon_{ii} = +\infty$ in (2.1) (as the limit suggests), which reveals that—obviously—in the case

of a singleton bundle $\mathcal{B} = \{i\}$ (2.3) holds for an arbitrary scaling factor ϵ_i. Of course, if each q_i individually satisfies (2.3), then so does the "natural" piecewise-quadratic model of f

$$\breve{f}_{\mathcal{B}}(x) = \max\{\ q_i(x)\ :\ i \in \mathcal{B}\ \}\ , \tag{2.4}$$

which, since $q_i(x_i) = f(x_i)$ by definition, therefore shares with the ordinary cutting-plane model $\hat{f}_{\mathcal{B}}$ the property

$$\breve{f}_{\mathcal{B}}(x_i) = f(x_i) \qquad \forall\ i \in \mathcal{I}$$

which justifies the moniker "set of interpolating points" for $\mathcal{I}$. As with $\hat{f}_{\mathcal{B}}$, it is convenient to express each q_i with respect to the displacement $d = x - y$

$$q_i(d) = f(y) + \hat{g}_i d - \hat{\alpha}_i + \epsilon_i \|d\|^2/2\ ,$$

where

$$\hat{\alpha}_i = \alpha_i - \epsilon_i \|y - x_i\|^2/2 \qquad \text{and} \qquad \hat{g}_i = g_i + \epsilon_i (y - x_i)\ . \tag{2.5}$$

Note that, unlike for the cutting-plane model, it is now necessary to explicitly keep track of the current iterates x_i, as they are needed to recompute $\hat{g}_i$ and $\hat{\alpha}_i$ each time that y and/or the ϵ_is change. For future reference, let us also remark here that the translation obviously does not change the fact that each q_i lies above the corresponding standard linear approximation of f (with $\epsilon_i = 0$); that is,

$$\epsilon_i \|d\|^2/2 + \hat{g}_i d - \hat{\alpha}_i \geq g_i d - \alpha_i \qquad \forall\ i \in \mathcal{B} \text{ and } \forall\ d \in \mathbb{R}^n\ . \tag{2.6}$$

It is immediate to verify (owing to the fact that $\alpha_i = \alpha_{ih}$ for the index h such that $y = x_h$) that

$$y \in \mathcal{I} \qquad \Rightarrow \qquad \hat{\alpha}_i \geq 0\ \ \forall i \in \mathcal{B}\ . \tag{2.7}$$

Tis property is essential, i.e., $\mathcal{I} = \{y\}$ is the minimal possible set of interpolating points for our analysis to work. In fact, the corresponding translated model $\breve{f}_{\mathcal{B}}(d) = \breve{f}_{\mathcal{B}}(y + d) - f(y)$ then leads to master problem

$$\begin{aligned} &\min_d\ \left\{\ \breve{f}_{\mathcal{B}}(d) + \rho\|d\|^2/2\ \right\} \\ &= \min_{v,d}\ \left\{\ v + \rho\|d\|^2/2\ :\ v \geq\ \epsilon_i\|d\|^2/2 + \hat{g}_i d - \hat{\alpha}_i \quad i \in \mathcal{B}\ \right\}\ . \end{aligned} \tag{2.8}$$

The dual of (2.8) can be obtained by direct application of the strict converse duality theorem [27, p. 117], which yields

$$\min_{\lambda}\ \left\{\ \frac{\left\|\sum_{i\in\mathcal{B}} \lambda_i \hat{g}_i\right\|^2}{2\left(\rho\ +\ \sum_{i\in\mathcal{B}} \lambda_i \epsilon_i\right)} + \sum_{i\in\mathcal{B}} \lambda_i \hat{\alpha}_i\ :\ \lambda \in \Lambda\ \right\}\ . \tag{2.9}$$

Under (2.7) one has that the optimal value of (2.9) is nonnegative, and therefore the optimal value of (2.8) is nonpositive. It is easy to check that this means that $v^* = \breve{f}_{\mathcal{B}}(d^*) \leq 0$, i.e., that the optimal solution d^* is a descent direction for the model $\breve{f}_{\mathcal{B}}$, a property that is crucial in the analysis of the approach. What is interesting here is that (2.8) is unavoidably a quadratically constrained problem; this means that its trust-region variant is no longer significantly different from the proximal version as far as practical solvability is concerned. Indeed, one can as well consider a *hybrid proximal/trust region* master problem

$$\begin{aligned} &\min_d \ \left\{ \ \breve{f}_{\mathcal{B}}(d) \ + \ \rho\|d\|^2/2 \ : \ \gamma\|d\|^2 \leq 2 \ \right\} \\ &= \min_{v,d} \ \left\{ \ v \ + \ \rho\|d\|^2/2 \ : \ v \geq \ \epsilon_i\|d\|^2/2 + \hat{g}_i d - \hat{\alpha}_i \quad i \in \mathcal{B} \ , \ \gamma\|d\|^2 \leq 2 \ \right\} \end{aligned} \tag{2.10}$$

which exposes both a proximal term weighted with ρ and a trust region one governed by γ, at the only cost of one more (among the many) quadratic constraint. The corresponding dual

$$\min_{\lambda,\mu} \ \left\{ \ \frac{\left\|\sum_{i\in\mathcal{B}} \lambda_i \hat{g}_i\right\|^2}{2\left(\mu + \rho + \sum_{i\in\mathcal{B}} \lambda_i \epsilon_i\right)} + \sum_{i\in\mathcal{B}} \lambda_i \hat{\alpha}_i + \frac{\mu}{\gamma} \ : \ \lambda \in \Lambda \ , \ \mu \geq 0 \ \right\} \tag{2.11}$$

looks pretty similar to (2.9), has analogous primal-dual relationships centered upon

$$d^* = -\frac{\sum_{i\in\mathcal{B}} \lambda_i^* \hat{g}_i}{\mu^* + \rho + \sum_{i\in\mathcal{B}} \lambda_i^* \epsilon_i} \tag{2.12}$$

and can be reformulated as a SOCP in exactly as (1.5) could, i.e.,

$$\begin{cases} \min_{\lambda,\mu,t,s} \ t + \sum_{i\in\mathcal{B}} \hat{\alpha}_i \lambda_i + \mu/\gamma \\ ts \geq \left\|\sum_{i\in\mathcal{B}} \lambda_i \hat{g}_i\right\|^2 \\ s = 2\left(\mu + \rho + \sum_{i\in\mathcal{B}} \lambda_i \epsilon_i\right) \\ \lambda \in \Lambda \ , \ \mu \geq 0 \end{cases} . \tag{2.13}$$

Thus, one can equivalently solve the Quadratically-Constrained Quadratic Program (2.10), and collect λ_i^* as the dual optimal multiplier of the i-th constraint for each $i \in \mathcal{B}$, or solve its SOCP dual and regain the primal optimal solution via (2.12). Plenty of options are currently available for both approaches. Because (2.10)/(2.11) generalize both the "pure" proximal ($\gamma = 0$) and trust region ($\rho = 0$) approaches, in the following we will provide a unified convergence analysis for a version of the algorithm this flexible master problem. This is not surprising, since [7] (cf. in particular Theorem 3.2) shows that stabilizing terms can be used which "look like a proximal term, a trust region term, or both" with little impact on the overall convergence of the approach.

An interesting feature of the new model $\breve{f}_{\mathcal{B}}$ is that it is somewhat "self-stabilized"; the ϵ_is play a role similar to that of ρ and γ. This is made more evident by rewriting

(2.10) as

$$\min_{v,d} \left\{ v \ : \ v \geq \ \epsilon_i' \|d\|^2/2 + \hat{g}_i d - \hat{\alpha}_i \quad i \in \mathcal{B} \ , \ \gamma \|d\|^2 \leq 2 \right\}$$

where

$$\epsilon_i' = \epsilon_i + \rho \tag{2.14}$$

and noting that the fixed ρ, the variable μ ("controlled" by γ) and the variable

$$\epsilon(\lambda) = \textstyle\sum_{i\in\mathcal{B}} \lambda_i \epsilon_i \qquad \text{and/or} \qquad \epsilon(\lambda)' = \sum_{i\in\mathcal{B}} \lambda_i \epsilon_i' = \epsilon(\lambda) + \rho \tag{2.15}$$

basically play the same role in (2.11). Indeed, provided that at least one of the ϵ_i is strictly positive, one could even take $\rho = \gamma = 0$ (i.e., remove any "external" stabilization) while ensuring that the master problems always have a solution. Indeed, the classical example of instability of the "pure" (non-stabilized) cutting-plane algorithm [16] uses $f(x) = x^2/2$ with initial iterates $x_1 = 1$ and $x_2 = -\varepsilon$; it is immediate to realize that for this example $\breve{f}_{\mathcal{B}}(x) = x^2/2 = f(x)$, and the pure cutting-plane algorithm with the new model instead terminates at the third iteration. A slightly more interesting example is

$$\min \left\{ \ f(y,\eta) \ : \ (y,\eta) \in \mathcal{C} \ \right\}$$

where $f(y,\eta) = \max\{|\eta|, -1 + 2\varepsilon + \|y\|\} \ : \ \mathbb{R}^n \times \mathbb{R} \to \mathbb{R}$ for some $\varepsilon \in (0, 1/2)$, and $\mathcal{C}$ is the unit ball $B^{(n+1)}$. The function f attains its minimum on $\mathcal{C}$ at all points of the set

$$\left\{ \ (y,0) \ : \ y \in B^{(n)}_{1-2\varepsilon} \ \right\}$$

with minimum value $f^* = 0$, and the pure cutting-plane algorithm requires a large number of iteration before finding an optimal solution given $x_1 = (y_1, \eta_1) = (0, 1)$ as starting point [16]. When using the piecewise-quadratic model instead, one has $\breve{f}_{\mathcal{B}}(x) = \eta$ at the first iteration, and consequently the first master problem

$$\min \left\{ \ v \ : \ v \geq \eta \ , \ \|y\|^2 + \eta^2 \leq 1 \ \right\}$$

has minimal value $v^* = -1$ attained at $x_2 = (y^*, \eta^*) = (0, -1)$. This gives

$$\breve{f}_{\mathcal{B}}(x) = (\|y\|^2 + \eta^2 + 1)/2$$

at the second iteration, and consequently the second master problem

$$\min \left\{ \ v \ : \ v \geq (\|y\|^2 + \eta^2 + 1)/2 \ , \ \|y\|^2 + \eta^2 \leq 1 \ \right\}$$

whose minimal value $v^* = 1/2$ is attained at $x_3 = (y^*, \eta^*) = (0, 0)$, which is an optimum of the problem. Thus, for a few selected examples the new model, even without stabilization, does improve on the classical cutting-plane one.

3. The algorithm. We now present the algorithm, which depends on

- the descent parameter $m \in (0,1)$,
- the upper threshold T on the scaling factors ϵ_i,
- the stopping parameters $\eta \geq 0$, $\kappa > 0$, and $\delta \in (0,1)$.

Let us indicate with $v(\epsilon)$ the optimal value of the dual master problem (2.11), which is the opposite of the optimal value of the primal master problem (2.10). Under (2.7) one has that $v(\epsilon) \geq 0$; this is crucial, because that value is used for the "approximate" stopping criterion of the algorithm, i.e.,

$$v(\epsilon) = \frac{\left\|\sum_{i\in\mathcal{B}} \lambda_i^* \hat{g}_i\right\|^2}{2\left(\mu^* + \rho + \sum_{i\in\mathcal{B}} \lambda_i^* \epsilon_i\right)} + \sum_{i\in\mathcal{B}} \lambda_i^* \hat{\alpha}_i + \frac{\mu^*}{\gamma} \leq \eta(1-\delta) \quad . \tag{3.1}$$

Accordingly, the "true" stopping criterion is

$$v(0) = \frac{\left\|\sum_{i\in\mathcal{B}} \lambda_i^* g_i\right\|^2}{2(\mu^* + \rho + \kappa)} + \sum_{i\in\mathcal{B}} \lambda_i^* \alpha_i \leq \eta \tag{3.2}$$

with the obvious property that

$$v(0) \leq \lim_{\|\epsilon\|\to 0} v(\epsilon) \quad ; \tag{3.3}$$

the "$\leq$" is due to the extra term "κ", which is there to avoid any problem with $\mu^* = 0$ (a possible occurrence), and to the missing non-negative term μ^*/γ.

The algorithm is initialized with an arbitrary starting point $x_0 \in \mathbb{R}^n$. The initial stability center y is set equal to x_0, the initial bundle is $\mathcal{B} = \{ (x_0, f(x_0), g_0) \}$, where $g_0 \in \partial f(x_0)$, and $\mathcal{I} = \mathcal{B}$ $(= \{y\})$. The parameters ρ and γ are initialized to any non-negative value, and a parameter t is initialized to any value in $(0, T]$. The algorithm then executes the following steps:

Step 1. Solve (2.10)/(2.11) for the optimal solutions $(d^*, v^*)/(\lambda^*, \mu^*)$.

Step 2. If (3.1) is not satisfied, then go to Step 4.

Step 3. If (3.2) holds then stop, else set $t := t/2$ and $\epsilon_i := \min\{\ \epsilon_i, t\ \}$ for all $i \in \mathcal{B}$. Possibly increase ρ and/or γ. Go to Step 1.

Step 4. Define the tentative point $x_+ = y + d^*$. Evaluate $f(x_+)$ and some $g_+ \in \partial f(x_+)$. Calculate ϵ_+ at x_+ according to (2.2). Set $\epsilon_+ := \min\{\ \epsilon_+, t\ \}$. Add the triple $(x_+, f(x_+), g_+)$ to $\mathcal{B}$, and optionally to $\mathcal{I}$, with the scaling factor ϵ_+. If

$$f(x_+) - f(y) > -mv(\epsilon) \tag{3.4}$$

then possibly increase ρ and/or γ. Go to Step 1.

Step 5. Set $y = x_+$. Possibly update $\mathcal{I}$, ensuring that (2.7) holds. Adjust the ϵ_is according to (2.2) with respect to new stability center and $\mathcal{I}$. Recompute the $\hat{\alpha}_i$s and $\hat{g}_i$s according to (2.5) with the new y and ϵ_is. Possibly reset t to any value in $(0, T]$, and ρ and γ to any non-negative value. Go to Step 1.

The core of the algorithm is the "main iteration", i.e., a sequence of consecutive steps 1. – 4. where the stability center remains unchanged. Within the main iteration one can have several "inner iterations", corresponding to sequences of consecutive steps where step 3. is never executed, i.e., the ϵ_is also are unchanged and only $\mathcal{B}$ (and possibly ρ) varies. The fact that the ϵ_is need not be updated during a main iteration, even if the newly obtained point is inserted in $\mathcal{I}$ (which is possible, although not mandatory) is not entirely obvious, but it can be easily proven since (3.4) gives

$$f(x_+) - f(y) > -mv(\epsilon) \geq \check{f}_{\mathcal{B}}(d^*) = \check{f}_{\mathcal{B}}(x_+) - f(y)$$

(using $m \leq 1$ and $-v(\epsilon) \geq \check{f}_{\mathcal{B}}(d^*)$, which implies $f(x_+) > \check{f}_{\mathcal{B}}(x_+)$. This gives

$$f(x_+) \geq \check{f}_{\mathcal{B}}(x_+) \geq f(x_i) + g_i(x_+ - x_i) + \epsilon_i \|x_+ - x_i\|^2/2$$

for all $i \in \mathcal{B}$, and therefore

$$\epsilon_{i+} = \frac{2\alpha_{i+}}{\|x_+ - x_i\|^2} = \frac{2(f(x_+) - f(x_i) - g_i(x_+ - x_i))}{\|x_+ - x_i\|^2} \geq \epsilon_i \ .$$

The result is easy to explain intuitively: all q_i support $\check{f}_{\mathcal{B}}$ (their pointwise maximum) in x_+, but f is well above $\check{f}_{\mathcal{B}}$ there, for otherwise a descent step would have been obtained. Therefore, within a main iteration the ϵ_is for the items already in $\mathcal{B}$ do not increase. Hence, it is immediate to verify that, within the same main iteration,

$$\epsilon_i \leq \bar{t}/2^{p-1} \tag{3.5}$$

where $\bar{t}$ is the value of t at the beginning of the main iteration (as set in step 5.), and p is the number of inner iterations within the main iteration, i.e., the number of times step 3. has been executed. This means that all the ϵ_is eventually converge to zero if infinitely many inner iterations are performed within the same main iteration.

3.1. Convergence of the main iteration. As customary in bundle-type methods, the convergence analysis uses two different arguments; first one proves that the main iteration eventually terminates, then one moves to examining what happens when the stability center is updated (step 5. of the algorithm, usually referred to as a *Serious Step*). Hence, here we focus on a single main iteration, and we denote by the index "$_k$" all the quantities at the k-th pass through steps 1.–4., removing the superscript "*" for notational simplicity. A first (albeit obvious) assumption is needed to ensure that the master problem is well-defined:

ASSUMPTION 3.1. *At least one among ρ, γ and the ϵ_is is strictly positive.*

Under Assumption 3.1, the objective function of (2.10) is strongly convex, and therefore the problem admits a (unique) optimal solution. Due to accumulation of information in $\mathcal{B}$, within the same *inner* iteration, one would expect the optimal value of the master problem to be monotone, i.e., that $v_+(\epsilon) \leq v_k(\epsilon)$ ("$_+$" again indicating the subsequent pass). However, this standard property, at the cornerstone of classical convergence arguments in bundle methods [7], is no longer true when the ϵ_is are reduced in Step 3., i.e., whenever more than one inner iteration is performed within the same main iteration. In fact, it is easy to verify that $-v_k(\epsilon) \geq -v_k(\epsilon')$ for $\epsilon' \leq \epsilon$, as the value of $\breve{f}_{\mathcal{B}}$ decreases as the ϵ_is do. Therefore, Step 3. may cause an increase in $v_k(\epsilon)$, whose effect is not easy to bound. This is relevant for the boundedness arguments that are technically important in the convergence analysis.

LEMMA 3.2. *If either $\gamma_1 > 0$, or $\rho_1 > 0$ and there exist a linear function $l(d) = gd - \alpha$ such that $gd - \alpha \leq \breve{f}_k(d)$ for all k and $d \in \Re^n$, then the sequence $\{d_k\}$ is bounded.*

Proof. Needless to say, the hypothesis means that Assumption 3.1 is satisfied. If $\gamma_1 > 0$ then $\gamma_k > 0$ for all k (as the sequence γ_k is nondecreasing), and $||d_k|| \leq \sqrt{2/\gamma_k} \leq \sqrt{2/\gamma_1}$. In the other case, it is clear that $d_k \in \{\ d\ :\ v_k + \rho_k \|d_k\|^2/2 \leq 0\ \}$. Since $v_k = \breve{f}_k \geq gd - \alpha$, one has $\rho_k \|d_k\|^2/2 \leq -v_k \leq \alpha - gd_k$. In other words, $d_k \in D = \{\ d\ :\ \rho_1 \|d\|^2/2 \leq \alpha - gd\ \}$ (as the sequence ρ_k is nondecreasing), and D is clearly a compact set. □

Lemma 3.2 shows the advantage of having an explicit trust region term: without it ($\gamma = 0$), boundedness requires an extra assumption. It is worth remarking that without adjustments of the ϵ_is—i.e., within the same inner iteration—the assumption is not necessary, and boundedness of $\{d_k\}$ is a consequence of monotonicity of $v_k(\epsilon)$; one can e.g. easily copy the arguments of [7, Lemma 5.5] (that Lemma may appear to require that $f_{\mathcal{B}} \leq f$, but this is actually not necessary). Yet, that line of proof fails when one cannot bound the optimal value of the master problem, as it may be the

case when the ϵ_is decrease. The assumption itself is not overly strong. For instance, in applications like Lagrangian relaxation one has that f is bounded below and a lower bound $\underline{f} > -\infty$ is *explicitly known*; typically, this is provided by the (best) feasible solution of the original problem found so far [8]. In this case, a (linear) constraint $\breve{f}_{\mathcal{B}}(y+d) \geq \underline{f}$ can be *explicitly added* to (2.8) with no difficulties, as it corresponds to a "flat" subgradient with $g = 0$, coupled with a scaling factor $\epsilon = 0$ which eliminates any need to define a corresponding iterate x. This would have the added bonus to guarantee well-posedness of the master problem even without Assumption 3.1, a-la [7, condition (P3′)], but such information is not available in all applications (and even combinatorial problems can be empty). Alternatively, it is enough to ensure that *one single subgradient always survives in* $\mathcal{B}$ in all iterations. This seems to be a given, since handling of $\mathcal{B}$ has not been discussed thus far, which may have left the reader with the impression that $\mathcal{B}$ is a monotonically increasing set. However, in general removing items from $\mathcal{B}$ is important to keep the cost of the master problem low enough (even more so in this case), as discussed in the following, which makes satisfying the assumption not entirely obvious.

LEMMA 3.3. *Under the hypotheses of Lemma 3.2, if infinitely many inner iterations are performed within a main iteration, then*

$$\lim_{k\to\infty} v_k(0) \leq \lim_{k\to\infty} v_k(\epsilon) \ .$$

Proof. Under the hypotheses, $p \to \infty$ in (3.5), and therefore $\epsilon_i \to 0$. Lemma 3.2 ensures the existence of some $\infty > D \geq \|d_k\|$ such that

$$\|\hat{g}_i\| \leq \|g_i\| + \epsilon_i D \quad , \quad \hat{\alpha}_i \geq \alpha_i - \epsilon_i D^2/2$$

uniformly for all $i \in \mathcal{B}_k$ (that is, D does not depend on k). Thus, as $k \to \infty$ one has $\hat{g}_i \to g_i$ and $\hat{\alpha}_i \to \alpha_i$, hence the result follows as in (3.3). □

Clearly, more convoluted schemes for updating the ϵ_is and t could be devised, provided that they serve the same purpose. The above Lemma is crucial for proving that, eventually, the "true" stopping criterion (3.2) holds, at least if a very conservative strategy is adopted for the handling of $\mathcal{B}$.

LEMMA 3.4. *Assume that no item is ever removed from* $\mathcal{B}$*, and that either* $\gamma_1 > 0$ *or* $\rho_1 > 0$*; if an inner iteration of infinitely many steps is performed, then* $\lim_{k\to\infty} v_k(\epsilon) = 0$.

Proof. As already discussed, keeping everything in $\mathcal{B}$ means that at least the fixed linear function $l_1(d) = g_1 d - \alpha_1$ underestimates $\breve{f}_k$ for all k, thus the hypotheses of Lemma 3.2 hold. During an infinitely long inner iteration, the ϵ_is are never changed. Since the descent criterion at step 4 has not been met by hypothesis, one has for all k (using (2.6) with $d = d_k$)

$$\begin{aligned} \breve{f}_+(d_k) &\geq g_+ d_k - \alpha_+ = f(y+d_k) - f(y) \\ &> -m v_k(\epsilon) \geq -v_k(\epsilon) \geq \breve{f}_k(d_k) = v_k \end{aligned} \tag{3.6}$$

which implies that the new constraint entering the master problem at iteration $k+1$ is not satisfied by the pair (d_k, v_k). Consequently, as we previously remarked, the sequence $\{v_k(\epsilon)\}$ is monotonically nonincreasing; since, due to (2.7), $v_k(\epsilon) \geq 0$ one has that $v_\infty(\epsilon) = \lim_{k\to\infty} v_k(\epsilon) \geq 0$. Furthermore, from Lemma 3.2 $\{d_k\}$ belongs to a compact set and there exists a convergent subsequence, say $\{d_k\}_{k\in K}$. Now, let i and s be two successive indices in K: necause no item is ever removed from $\mathcal{B}$, $i \in \mathcal{B}_s$. Note that s is not in principle $i+1$ but rather the—unknown a priori—following iteration in the convergent subsequence, whence the need for removing nothing from $\mathcal{B}$. Both inequalities

$$\epsilon_i \|d_i\|^2/2 + d_i \hat{g}_i - \hat{\alpha}_i > -m v_i(\epsilon)$$
$$\epsilon_i \|d_s\|^2/2 + d_s \hat{g}_i - \hat{\alpha}_i \leq v_s \leq -v_s(\epsilon)$$

hold, from which we obtain

$$v_s(\epsilon) - m v_s(\epsilon) < \hat{g}_i(d_i - d_s) + \epsilon_i(\|d_i\|^2 - \|d_s\|^2)/2 \ .$$

Taking the limit one obtains $v_\infty(\epsilon) \leq 0$, and therefore $v_\infty(\epsilon) = 0$. □

THEOREM 3.5. *Assume that no item is ever removed from $\mathcal{B}$: under the hypotheses of Lemma 3.3, the main iteration terminates.*

Proof. Assume by contradiction that the main iteration does not terminate. If the main iteration consists of only one inner iteration, we can apply Lemma 3.4 to prove that $v_k(\epsilon) \to 0$; however, this clearly implies that the condition (3.1) at step 2. can not be satisfied for all k, hence step 3. must be occasionally entered. The same line of proof works if the number of inner iterations is finite, by just waiting long enough for the last one to occur. Therefore, we are left with examining the case of infinitely many inner iterations. Then, Lemma 3.3 ensures that, since (3.1) is satisfied infinitely many times, eventually (3.2) must also be satisfied, a contradiction. □

Several modifications to this basic scheme are possible which may be useful in practice without requiring hardly any change in the convergence analysis. Since only infinitely long sequences matter, anything that "does not happen infinitely often" can be tolerated. For instance, *decreasing* ρ within a main iteration is also possible, e.g. to accommodate *curved searches* along y in the style of [32], provided that this is done only finitely many times; ditto for not evaluating f at x_+ (e.g. to perform a line search instead) and/or not inserting the new subgradient in $\mathcal{B}$ [7].

However, Theorem 3.5 is hardly satisfactory, since it implies a monotonically increasing $\mathcal{B}$. This makes the algorithm hardly implementable both in theory and in practice, considering that each new item in $\mathcal{B}$ corresponds to a quadratic constraint in the master problem. Strategies to reduce the size of $\mathcal{B}$ without hindering convergence have indeed been developed for the methods based on the standard cutting-plane model. The first of these is based on the observation that the dual optimal multipliers

λ_i^* provide a useful "measure of importance" of the corresponding points $i \in \mathcal{B}$; in particular, if $\lambda_i^* = 0$ then the corresponding item is useless for (the current) master problem, and can be eliminated without changing its solution. This leads to proving that eliminating all these items does not impair convergence. The same result can be proven, with somewhat more convoluted arguments, for the current setting. To do so, it is convenient to introduce, in analogy with (2.15), the *aggregated data*

$$\hat{g} = \textstyle\sum_{i\in\mathcal{B}} \lambda_i^* \hat{g}_i \quad , \quad \hat{\alpha} = \textstyle\sum_{i\in\mathcal{B}} \lambda_i^* \hat{\alpha}_i \quad , \quad \epsilon = \textstyle\sum_{i\in\mathcal{B}} \lambda_i^* \epsilon_i \tag{3.7}$$

with the property that the *aggregated primal and dual master problems*

$$\min_{v,d} \left\{ v + \rho\|d\|^2/2 \ : \ v \geq \ \epsilon\|d\|^2/2 + \hat{g}d - \hat{\alpha} \ , \ \gamma\|d\|^2 \leq 2 \right\} \tag{3.8}$$

$$\min_{\lambda,\mu} \left\{ \frac{\|\lambda\hat{g}\|^2}{2\,(\mu+\rho+\lambda\epsilon)} + \hat{\alpha}\lambda \ : \ \lambda = 1 \ , \ \mu \geq 0 \right\} \tag{3.9}$$

have the same optimal value (and, in case of (3.8), the same optimal solution) to (2.10)/(2.11). More to the point, they have the same optimal value to the modified (2.10)/(2.11) *in which all items such that* $\lambda_i^* = 0$ *have been removed from* $\mathcal{B}$. This is easily gauged by looking at (3.9), and it is transported to (3.8) by standard duality arguments. Therefore, one can consider the simplified problem

$$-\bar{v}_+(\epsilon) = \begin{cases} \min_{v,d} v + \rho\|d\|^2/2 \\ v \geq \epsilon\|d\|^2/2 + \hat{g}d - \hat{\alpha} \\ v \geq \epsilon_+\|d\|^2/2 + \hat{g}_+ d - \hat{\alpha}_+ \\ \gamma\|d\|^2 \leq 2 \end{cases} \tag{3.10}$$

and be ensured that $0 \leq v_+(\epsilon) \leq \bar{v}_+(\epsilon)$, even if—possibly—all items such that $\lambda_i^* = 0$ have been removed from $\mathcal{B}$. These tools allow us to prove the following weakened form of Lemma 3.4.

LEMMA 3.6. *Assume that the hypotheses of Lemma 3.2 hold and that at all iterations no item with* $\lambda_i^* > 0$ *is ever removed from* $\mathcal{B}$. *If* $\rho_k > 0$ *for at least one iteration* k, *then any inner iteration must finitely terminate.*

Proof. Note that, unlike in Lemma 3.4, for $\gamma_k = 0$ and $\rho_k > 0$ the hypothesis of Lemma 3.2 is no longer automatically guaranteed: without a trust region, compactness has to be ensured by external means. We will show that $\bar{v}_+(\epsilon)$ is "significantly lower" than $v(\epsilon)$—the value before the insertion of the new item in $\mathcal{B}$—in a way that guarantees that the sequence has finitely terminate. We prove this for the case where the stabilization parameters do not change during the iteration (i.e., $\rho_k = \rho_+ = \rho$ and $\gamma_k = \gamma_+ = \gamma$), knowing that the result holds *a fortiori* if ρ and/or γ increase.

The first remark is that the optimal solution (d_k, v_k) of (2.8) is the same as that of (3.8), and therefore cannot be still optimal for (3.10), for otherwise (3.6) would give $v_k = \breve{f}_+(d_k) > v_k$. Therefore

$$\breve{f}_+(d_+) = v_+ = \epsilon_+\|d_+\|^2/2 + \hat{g}_+ d_+ - \hat{\alpha}_+ \ , \tag{3.11}$$

where (v_+, d_+) is the optimal solution to (3.10), i.e., the newly added quadratic constraints is always active in the new optima (note that this argument works without changes for the trust region case). We denote by $s_+ = d_+ - d_k$ the effect of the introduction of the new constraint on the optimal primal solution; for technical reasons, we analyze separately the two mutually exclusive cases

$$(i)\ \ 2(\|\hat{g}_+\| + \epsilon_+\|d_k\|)\|s_+\| < \eta(1-m)(1-\delta) \quad \text{and} \quad (ii)\ \ \|s_+\| \geq \frac{\eta(1-m)(1-\delta)}{2(\|\hat{g}_+\| + \epsilon_+\|d_k\|)}$$

of "small" and "large" s_+, respectively, where the threshold between the two (which uses the tolerances η and δ of the stopping criterion, cf. (3.1)) is only a technicality for the proof.

Assume (i) holds; note that both $\|\hat{g}_+\| + \epsilon_+\|d_k\| = 0$ and $d_+ = d_k \Rightarrow \|s_+\| = 0$ fall into this case. Using (3.11) one has

$$-\bar{v}_+(\epsilon) \geq v_+ = \epsilon_+\|d_+\|^2/2 + \hat{g}_+ d_+ - \hat{\alpha}_+$$

with equality holding if $\rho_k = 0$. We can then continue the inequality chain as

$$\begin{aligned}
-\bar{v}_+(\epsilon) \geq v_+ &= \epsilon_+\|d_k + s_+\|^2/2 + \hat{g}_+(d_k + s_+) - \hat{\alpha}_+ \\
&= \big(\epsilon_+\|d_k\|^2/2 + \hat{g}_+ d_k - \hat{\alpha}_+\big) + \hat{g}_+ s_+ + \epsilon_+ d_k s_+ + \epsilon_+\|s_+\|^2/2 \\
&> -mv(\epsilon) + (\hat{g}_+ + \epsilon_+ d_k)s_+ + \epsilon_+\|s_+\|^2/2 \\
&\geq -mv(\epsilon) + (\hat{g}_+ + \epsilon_+ d_k)s_+ \\
&= -v(\epsilon) + (1-m)v(\epsilon) + (\hat{g}_+ + \epsilon_+ d_k)s_+ \\
&\geq -v(\epsilon) + (1-m)v(\epsilon) - \|s_+\|\big(\|\hat{g}_+\| + \epsilon_+\|d_k\|\big)\ .
\end{aligned}$$

Now, using (i) and the not satisfaction of the stopping rule (3.1) we finish it off as

$$\begin{aligned}
-\bar{v}_+(\epsilon) &> -v(\epsilon) + \eta(1-m)(1-\delta) - \eta(1-m)(1-\delta)/2 \\
&= -v(\epsilon) + \eta(1-m)(1-\delta)/2\ .
\end{aligned}$$

Thus, at each iteration the optimal value increases by at least a fixed amount, which rules out infinitely many steps.

We now move to case (ii). Here we start from the fact that

$$v_+ \geq \epsilon\|d_+\|^2/2 + \hat{g} d_+ - \hat{\alpha}$$

(cf. (3.11), the definition of $\breve{f}_+$ and (2.6)) to write

$$\begin{aligned}
&-\bar{v}_+(\epsilon) = v_+ + \rho\|d_+\|^2/2 \\
&\geq \epsilon\|d_k + s_+\|^2/2 + \hat{g}(d_k + s_+) - \hat{\alpha} + \rho\|d_k + s_+\|^2/2 \\
&= -v(\epsilon) + \big(\hat{g} + (\rho+\epsilon)d_k\big)s_+ + (\rho+\epsilon)\|s_+\|^2/2 \\
&= -v(\epsilon) - \mu_k d_k s_+ + (\rho_+ + \epsilon)\|s_+\|^2/2
\end{aligned} \tag{3.12}$$

where in the last passage we have used $(\mu_k + \rho + \epsilon)d_k = -\hat{g}$ (cf. (2.12)). Now, if $\mu_k = 0$ then $\mu_k d_k s_+ = 0$. Otherwise, the constraint $\gamma\|d\|^2 \leq 2$ is active in d_k, i.e., $\gamma\|d_k\|^2 = 2$. Since one also has $\gamma\|d_+\|^2 \leq 2$, we obtain that

$$\gamma\|d_k\|^2 + \gamma\|s_+\|^2 + 2\gamma d_k s_+ \leq 2 \quad \Rightarrow \quad \gamma\|s_+\|^2 + 2\gamma d_k s_+ \leq 0 \ .$$

Hence, $-\mu_k d_k s_+ \geq \mu_k\|s_+\|^2/2$ is true whatever the value of μ_k is. We can now exploit the hypothesis (ii) in (3.12) to conclude that

$$-\bar{v}_+(\epsilon) \geq -v(\epsilon) + (\mu_k + \rho + \epsilon)\|s_+\|^2/2 \geq -v(\epsilon) + (\mu_k + \rho + \epsilon)\frac{\eta^2(1-m)^2(1-\delta)^2}{8(\|\hat{g}_+\| + \epsilon_+\|d_k\|)^2} \ .$$

Now, the numerator in the fraction of the rightmost term is constant, and the denominator $\|\hat{g}_+\| + \epsilon_+\|d_k\|$ is bounded above. To prove that, consider that from Lemma 3.2 all primal solutions, and hence in particular d_k, belong to a compact set. Hence so do all the g_i (the image of a compact set under the subdifferential mapping is compact for a function that is finite everywhere), and the term $y - x_i$ in (2.5) is likewise bounded. Finally, all ϵ_i (and hence in particular ϵ_+) are bounded above by t, which gives boundedness of all the $\hat{g}_i$s, and hence in particular of $\hat{g}_+$. Thus, the fraction is bounded away from zero. Since $\mu_k + \rho + \epsilon \geq \rho = \rho_k$ (with the other two terms very possibly being zero), if any ρ_h is strictly positive then (since the sequence is nondecreasing) at length $\rho_k \geq \rho_h > 0$: summing over k infinitely many times would contradict $\bar{v}_+(\epsilon) \geq 0$, hence the inner iteration is finite. □

Lemma 3.6 can be used in Theorem 3.4 instead of Lemma 3.4 to prove convergence of the main iteration under the more relaxed handling of $\mathcal{B}$. It may be work remarking that this result sharply distinguish between the two forms of stabilization: while the trust region is a handy mean to ensure compactness, but otherwise inessential, the proximal term is necessary, i.e., setting $\rho_k = 0$ is not an option. This appears to be inherent rather than a flaw in the analysis. Indeed, convergence under aggregation for the standard cutting plane method requires the *dual stabilizing term to be smooth, i.e., the primal stabilizing term to be strictly convex* [7, condition (P3″)]. Needless to say, the trust region function corresponds to a primal stabilizing term with the form of an indicator function, and therefore *not* strictly convex, whose conjugate is in fact not differentiable (in 0).

While this result may be sufficient for practical purposes, the set of items such that $\lambda_i^* > 0$ can still be very large in practice, leading to computationally very expensive master problems. Indeed, while in case of the standard cutting plane model one can prove that $|\mathcal{B}| \leq n + 1$ (still not a "small" number for large-scale optimization) suffice, in the quadratic case even this bound is not given. Fortunately, an even better approach exist.

3.2. Convergence with aggregation. The above analysis suggests an interesting possibility: *if it were possible to replace $\mathcal{B}$ with just the aggregated pair $(\hat{g}, \hat{\alpha})$*, with multiplier ϵ, then the convergence would still be assured. This is in fact possible

when using the cutting plane model, as the *aggregated subgradient and linearization error* (z^*, σ^*) (cf. 1.3) can indeed be legally added to $\mathcal{B}$, possibly removing all the rest of the points in exchange. Doing so at avery iteration yields the so-called "poor-man" version of bundle methods, that are characterized by solving at each step a master problem with only *two* subgradients (for which closed formulae can be devised), and that closely resemble subgradient approaches [1]. Achieving the same feat for the quadratic model, however, is substantially more complex, due to the fact that $(\hat{g}, \hat{\alpha}) \neq (z^*, \sigma^*)$, and in particular that $\hat{g}$ is *not*, in general, a(n approximated) subgradient to f.

The catch, therefore, is the need—peculiar of our new quadratic model—to exhibit a potential new bundle element $(\bar{x}, f(\bar{x}), \bar{g})$ and its multiplier $\bar{\epsilon}$, derived from existing information, which, when plugged into (2.5), *exactly reproduce* $\hat{g}$, $\hat{\alpha}$ and ϵ. This might at first seem easy, because

$$\begin{aligned} \hat{g} = \textstyle\sum_{i\in\mathcal{B}} \lambda_i^* \hat{g}_i = \sum_{i\in\mathcal{B}} \lambda_i^* (\, g_i + \epsilon_i(y - x_i)\,) \\ = z^* + \epsilon \Big(\textstyle\sum_{i\in\mathcal{B}} \frac{\lambda_i^* \epsilon_i}{\epsilon} (y - x_i) \Big) = z^* + \epsilon(y - \tilde{x}) \end{aligned}$$

where $\tilde{x} = \sum_{i\in\mathcal{B}} \eta_i x_i$ and $\eta_i = \lambda_i^* \epsilon_i / \epsilon$, with the obvious property that $\eta \in \Lambda$. Hence, combining the original convex multipliers λ_i^* and the weights ϵ_i provides new convex multipliers η_i which would seem to produce a good candidate for defining the "center" $\bar{x}$ of the usual aggregate subgradient z^*. Note that while the η_i are undefined if $\epsilon = 0$, that case requires $\lambda_i^* \epsilon_i = 0$ for all $i \in \mathcal{B}$, which immediately gives $\hat{g} = z^*$ and $\hat{\alpha} = \sigma^*$; so, that one actually is the "easy" case in which everything falls back to the standard aggregate model (formally, one can then take $\bar{x} = y$ and $\bar{\epsilon} = 0$). Unfortunately, things are not so easy: in fact, while plugging σ^*, ϵ and $\tilde{x}$ in (2.5) gives

$$\alpha^* = \sigma^* - \epsilon \|y - \tilde{x}\|^2/2 = \sigma^* - \epsilon \left\| \textstyle\sum_{i\in\mathcal{B}} \eta_i (y - x_i) \right\|^2 /2$$

one has

$$\hat{\alpha} = \textstyle\sum_{i\in\mathcal{B}} \lambda_i^* \hat{\alpha}_i = \sum_{i\in\mathcal{B}} \lambda_i^* (\alpha_i - \epsilon_i \|y - x_i\|^2/2) = \sigma^* - \epsilon \big(\sum_{i\in\mathcal{B}} \eta_i \|y - x_i\|^2 \big)/2 \ .$$

In plain words, using z^*, σ^* and $\tilde{x}$, while correctly reproducing $\hat{g}$, fails to exactly reproduce $\hat{\alpha}$; in particular, it is easy to verify that the obtained α^* is larger than $\hat{\alpha}$, in that

$$\xi = \frac{\|y - \tilde{x}\|^2}{\sum_{i\in\mathcal{B}} \eta_i \|y - x_i\|^2} = \frac{\left\| \sum_{i\in\mathcal{B}} \eta_i (y - x_i) \right\|^2}{\sum_{i\in\mathcal{B}} \eta_i \|y - x_i\|^2} \leq 1$$

(use e.g. the definition of convexity for $\|y - \cdot\|^2$). Fortunately, there is more than one way to obtain $\hat{g}$, at least if one is willing to play with ϵ. Indeed, take any $\bar{\epsilon} \in (0, \epsilon)$ (reminding that $\epsilon > 0$), and consider

$$\bar{x} = \frac{(\bar{\epsilon} - \epsilon)}{\bar{\epsilon}} y + \frac{\epsilon}{\bar{\epsilon}} \tilde{x} \quad \iff \quad \bar{\epsilon}(y - \bar{x}) = \epsilon(y - \tilde{x}) \ .$$

It is then immediate to verify that

$$z^* + \bar{\epsilon}(y - \bar{x}) = z^* + \epsilon(y - \tilde{x}) = \hat{g} \ ;$$

in plain words, for any chosen $\bar{\epsilon}$ the corresponding $\bar{x}$ allows to reproduce $\hat{g}$. For the specific choice

$$\bar{\epsilon} = \xi\epsilon$$

one has

$$\begin{aligned}\sigma^* - \frac{\bar{\epsilon}}{2}\|y - \bar{x}\|^2 = \sigma^* - \frac{\bar{\epsilon}}{2}\left\|\frac{\epsilon}{\bar{\epsilon}}(y - \tilde{x})\right\|^2 &= \sigma^* - \frac{\epsilon}{2\xi}\|y - \tilde{x}\|^2 \\ &= \sigma^* - \epsilon\big(\ \textstyle\sum_{i\in\mathcal{B}} \eta_i\|y - x_i\|^2\ \big)/2 = \hat{\alpha} \ .\end{aligned}$$

The case $\xi = 0 \Rightarrow \bar{\epsilon} = 0$ is also consistent, since it gives $y = \tilde{x}$ and, again, $\hat{g} = z^*$ and $\hat{\alpha} = \sigma^*$. Thus, in all cases one can pretend that the linear lower approximation to f given by z^* and σ^* has been obtained by the oracle in $\bar{x}$; assigning it weight $\bar{\epsilon} = \xi\epsilon$ reproduces both $\hat{g}$ and $\hat{\alpha}$. Imposing that

$$\sigma^* = f(y) - f(\bar{x}) - z^*(y - \bar{x})$$

is equivalent to assuming that

$$f(\bar{x}) = f(y) + z^*(\bar{x} - y) - \sigma^* \ .$$

Thus, the value to be used as $f(\bar{x})$ for the aggregated element to be inserted in $\mathcal{B}$ is simply that of the aggregated linearization, which is a *lower bound* on the true function value. Clearly, if the corresponding ϵ eventually goes to zero during a main iteration, what remains is a perfectly legal linear function underestimating f, which cannot cause any problem to the convergence of the algorithm. The only issue with using a lower bound instead of the true value of $f(\bar{x})$ is the possibility of *negative* α_{ij}, and therefore negative $\epsilon_{ij} \Rightarrow \epsilon_i$, for subgradients obtained after the aggregation step. There could be ways of dealing with this: for instance, once a negative α_{ij} is detected, then the point that generates it (the one where the linear approximation lies above the alleged function value) can be updated by increasing its function value so as to obtain $\alpha_{ij} = 0$. This is legal, since one has just obtained a better lower bound on the true function value, that can just be used to replace the initial one. Alternatively, one may just update (2.1) to ignore negative elements (i.e., set $\epsilon_{ij} = \max\{\epsilon_{ij}, 0\}$). All this would require some analysis, and it may have a negative impact in practice, since it would tend to decrease the size of the weights ϵ_i, as the quadratic models would be forced to support (possibly crude) lower approximations to true function values. Fortunately, our setting allows for an easier solution: simply *avoid to insert the aggregated point into* $\mathcal{I}$. This is possible, since $\bar{x}$ will never be the current point except by chance (cf. the case $\bar{\epsilon} = 0$ above), and no issues arise. By ensuring that the

aggregated point never belongs to $\mathcal{I}$, none of the corresponding α_{ij} and ϵ_{ij} will ever be computed, and the fact that the estimate of $f(\bar{x})$ used to construct the corresponding quadratic function is a(n even crude) lower bound on the true value is immaterial.

One catch remains in the above approach: to reproduce $\hat{\alpha}$ one has to decrease the weight of the aggregated piece from the expected ϵ. Without any other action, the optimal value of the master problems may increase, and the primal optimal solution would be different, as it is easy to verify from (2.12). However, there is an easy fix for this: *update the stabilization parameters*. This can be done independently for both, considering that the aggregated dual master problem (3.9) only has one variable μ, and that the optimal solution to the aggregated primal master problem (3.8) always has the form $\bar{d} = -\beta\hat{g}$ for $\beta = 1/(\mu^* + \rho + \epsilon) > 0$ (cf. (2.12)).

1. *Changing* ρ is actually very simple, since it is easy to verify that

$$\rho' = \rho + \epsilon - \bar{\epsilon} > \rho \tag{3.13}$$

leads to an aggregated dual master problem (3.9) with exactly the same optimal solution μ^* as the original one, in that $\rho' + \bar{\epsilon} = \rho + \epsilon$. Consequently, the aggregated primal master problem (3.8) has exactly the same optimal solution d^* (and optimal value) of the original one.

2. *Changing* γ is instead rather more complex, as the role of ρ is taken by the extra variable μ, which cannot be directly set and only "indirectly" reacts to changes of γ. The issue is then that of finding a new value for γ so that the optimal value of the aggregated problem reproduces that of the original one, which is

$$\epsilon'\|d^*\|^2/2 + \hat{g}d^* - \hat{\alpha} = \left(\frac{\epsilon'}{2(\mu^* + \epsilon')^2} - \frac{1}{\mu^* + \epsilon'}\right)\|\hat{g}\|^2 - \hat{\alpha}$$

since all constraints corresponding to dual multipliers $\lambda_i^* > 0$ are active; note that we have used the "alternative" form of the problem, cf. (2.14). Imposing that the new optimal solution $\bar{d} = -\beta\hat{g}$ reproduces the same value, i.e.,

$$\bar{\epsilon}'\|\bar{d}\|^2/2 + \hat{g}\bar{d} - \hat{\alpha} = \left(\frac{\bar{\epsilon}\beta^2}{2} - \beta\right)\|\hat{g}\|^2 - \hat{\alpha}$$

(where obviously $\bar{\epsilon}' = \bar{\epsilon}' + \rho$) leads to the equation

$$\frac{\epsilon'}{2(\mu^* + \epsilon')^2} - \frac{1}{\mu^* + \epsilon'} = \frac{\bar{\epsilon}'\beta^2}{2} - \beta \ .$$

This has to be studied separately for $\bar{\epsilon}' = 0$ (which implies $\rho = 0$) and $\bar{\epsilon}' > 0$. The former case gives

$$\bar{\beta} = \frac{2\mu^* + \epsilon'}{2(\mu^* + \epsilon')^2} \ \ (\geq 0)$$

while the latter has two roots

$$\beta_{\pm} = \frac{1 \pm \sqrt{1-\delta}}{\bar{\epsilon}'} \quad \text{where} \quad 0 < \delta = \bar{\epsilon}' \frac{2\mu^* + \epsilon'}{(\mu^* + \epsilon')^2} = 2\bar{\epsilon}'\bar{\beta} < 1$$

as it is easy to verify algebraically (use $\epsilon' = \epsilon + \rho > \bar{\epsilon} + \rho = \epsilon'$ and $\mu^* \geq 0$). One thus wants to select $\gamma' \geq \gamma$ such that

$$\gamma' \|\bar{d}\|^2 = 2 \quad \Rightarrow \quad \gamma' = \frac{2}{\beta^2 \|\hat{g}\|^2}$$

where note that $\gamma \|d^*\|^2 \leq 2$ gives $\gamma \leq 2(\mu^* + \epsilon')^2 / \|\hat{g}\|^2$. For $\bar{\epsilon}' = 0$ this gives

$$\gamma' = \frac{8(\mu^* + \epsilon')^4}{(2\mu^* + \epsilon')^2 \|\hat{g}\|^2} \leq \frac{8(\mu^* + \epsilon')^2}{\|\hat{g}\|^2} \tag{3.14}$$

which if $\mu^* > 0 \Rightarrow \gamma = 2(\mu^* + \epsilon')^2 / \|\hat{g}\|^2$ also gives $\gamma' \leq 4\gamma$. For $\bar{\epsilon}' > 0$ the root β_+ cannot be chosen in general, as if $\mu^* > 0$ one would have

$$\gamma = 2\left(\frac{\mu^* + \epsilon'}{\|\hat{g}\|}\right)^2 > 2\left(\frac{\bar{\epsilon}'}{\|\hat{g}\|}\right)^2 > 2\left(\frac{\bar{\epsilon}'}{(1 + \sqrt{1-\delta})\|\hat{g}\|}\right)^2 = \gamma'$$

since $\bar{\epsilon}' < \mu^* + \epsilon'$ and $1 + \sqrt{1-\delta} > 1$. This finally leads to

$$\gamma' = 2\left(\frac{\bar{\epsilon}'}{(1 - \sqrt{1-\delta})\|\hat{g}\|}\right)^2 \tag{3.15}$$

being the chosen value, and indeed it can be verified algebraically that

$$\gamma' = 2\left(\frac{\bar{\epsilon}'}{(1 - \sqrt{1-\delta})\|\hat{g}\|}\right)^2 \geq 2\left(\frac{\mu^* + \epsilon'}{\|\hat{g}\|}\right)^2 \geq \gamma \ .$$

In fact, the relationship

$$(\mu^* + \epsilon')^2 - \bar{\epsilon}(2\mu^* + \epsilon') > (\mu^* + \epsilon')^2 - \epsilon'(2\mu^* + \epsilon') = (\mu^*)^2$$

gives $\sqrt{(\mu^* + \epsilon)^2 - \bar{\epsilon}(2\mu^* + \epsilon')} > \mu^*$, whence

$$\begin{aligned}
\gamma' = 2\left(\frac{\bar{\epsilon}'}{(1 - \sqrt{1-\delta})\|\hat{g}\|}\right)^2 &= 2\left(\frac{\bar{\epsilon}'}{\|\hat{g}\|} \frac{\mu^* + \epsilon'}{\mu^* + \epsilon' - \sqrt{(\mu^* + \epsilon')^2 - \bar{\epsilon}(2\mu^* + \epsilon')}}\right)^2 \\
&= 2\left(\frac{(\mu^* + \epsilon')\bar{\epsilon}'}{\|\hat{g}\|} \frac{\mu^* + \epsilon' + \sqrt{(\mu^* + \epsilon')^2 - \bar{\epsilon}(2\mu^* + \epsilon')}}{(\mu^* + \epsilon')^2 - (\mu^* + \epsilon')^2 + \bar{\epsilon}(2\mu^* + \epsilon')}\right)^2 \\
&> 2\left(\frac{\mu^* + \epsilon'}{\|\hat{g}\|} \frac{\mu^* + \epsilon' + \mu^*}{2\mu^* + \epsilon'}\right)^2 = 2\left(\frac{\mu^* + \epsilon'}{\|\hat{g}\|}\right)^2 \geq \gamma
\end{aligned}$$

as claimed. This derivation only works for $\bar{\epsilon}' > 0$, which is the most likely case as $\rho > 0$ is needed to ensure convergence with aggregation; nonetheless, it is easy to verify that (3.14) is nothing but the limit for $\bar{\epsilon}' \to 0$ of (3.15), and therefore the property is extended to that case, too.

The above analysis shows that one can aggregate while retaining the convergence of the approach, as increasing ρ and/or γ during Null Steps is allowed. A last issue remains, though: while ρ and/or γ can become arbitrarily large as far as "local" convergence is concerned, some discipline has to be exercised on the stabilizing terms if "global" convergence has to be attained, as it is clear that (say) shrinking the trust region exponentially fast may lead to the algorithm to stall far from the optimum. The simplest form of discipline requires insisting that $\rho_k \leq \rho_{max} < +\infty$ and $\gamma_k \leq \gamma_{max} < +\infty$ (cf. Theorem 3.9); however, one may then find himself between a rock and a hard place when ρ and/or γ have to be increased due to aggregation.

Fortunately, increasing ρ and/or γ is a reaction to the fact that the ϵ obtained by aggregation is "too small"; yet, reducing ϵ is a standard step in our algorithm, and it is actually necessary for convergence. Hence, the only required trick is to properly co-ordinate the increase of the stabilization parameters and the decrease of ϵ. In this respect, (3.13) comes in very handy, because $\epsilon_i \leq t$ for all $i \in \mathcal{B}$ implies that $\epsilon \leq t$, and therefore $\rho' \leq \rho + t$. Thus, one may impose any arbitrary upper bound ρ_{max} on ρ and still be able to perform aggregations with the following simple modifications to the algorithm:

- initialize t such that $t \leq (\rho_{max} - \rho_1)/4$;
- *never* increase ρ by more than t at a time (this is free for aggregation, but not necessarily so for regular ρ-handling heuristics, cf. §4.1);
- each time that $\rho_{max} - \rho_k < 2t$ set $t := t/4$ and $\epsilon_i := \min\{ \epsilon_i, t \}$ for all $i \in \mathcal{B}$ (cf. Step 3. of the algorithm).

This ensures that $\rho_{max} - \rho_k \geq 2t$ at all iterations k, and therefore that "there is always enough room to increase ρ" when an aggregation has to be performed. Of course this also implies that $\epsilon \to 0$ whenever $\rho_k \to \rho_{max}$. Doing a similar trick for γ appears to be more difficult, as bounding the increase of γ in terms of ϵ (hence t) does not seem obvious. Thus, perhaps the most promising setting is *one large and fixed trust region* to guarantee compactness arguments, and then using ρ as the real driver of the stabilization tuning. In so doing, the maximum size of $\mathcal{B}$ can be kept limited to any fixed number ≥ 2 by inserting the aggregated constraints into $\mathcal{B}$, deleting any subset of the current bundle elements (possibly all) and replacing ρ by ρ'. Clearly this does not impact on the proof of Lemma 3.6, and therefore leads to a converging (albeit "poorman") version of the approach.

3.3. Global convergence. We are now in the position to prove finiteness of the algorithm for any $\eta > 0$. Since Theorem 3.5 rules out infinitely long main iterations, we only need to prove that an infinite number of descents (main iterations) cannot occur. For this we can disregard whatever happens during a main iteration and only consider the state of the algorithm at the end of each; therefore, from now on the

index "$_k$" denotes to the iteration where step 5. is executed for the k-th time, and we can assume $k \to \infty$ for otherwise nothing has to be proven. Of course, in this context the stability center also has an index.

THEOREM 3.7. *Either $f_\infty = \lim_{k\to\infty} f(y_k) = -\infty$ (and therefore f is unbounded below and $\{y_k\}$ is a minimizing sequence), or the algorithm stops after a finite number of main iterations.*

Proof. Since (3.4) is not satisfied at iteration k, then

$$f(y_{k+1}) \leq f(y_k) - mv_k(\epsilon_k)$$

where $v_k(\epsilon_k) > \eta(1-\delta)$ since (3.1) is not satisfied; summing that over k gives

$$f(y_k) < f(y_0) - km\eta(1-\delta) \ .$$

For $k \to \infty$ this gives $f_\infty = -\infty$, thus either a minimizing sequence is constructed which proves that f is unbounded below, or the algorithm terminates in a finite number of main iterations. □

Thus, for any fixed $\eta > 0$ the algorithm eventually terminates, and the obtained stability center satisfies the approximate optimality conditions (3.2). However, running the algorithm with $\eta = 0$ is not, in principle, possible. Yet, one can resort to an obvious trick: for a sequence $\{\eta_k\} \to 0$, run the algorithm with $\eta = \eta^k$ and collect y_k, z_k and σ_k as, respectively, the stability center, the aggregated subgradient and the aggregated linearization error when the algorithm terminates. It is easy to show that, provided that the optimal value is not "artificially" reduced by sending ρ_k and/or $\gamma_k \to \infty$, $||z_k||$ and σ_k can be made "as small as desired". Thus $\{y_k\}$ "looks like" a minimizing sequence, and it actually is so under weak assumptions on f, such as:

DEFINITION 3.8. *A function f is *-compact if $\forall\ L \geq l > f^* \geq -\infty$*

$$e(l,L) = \sup_x\{\ \mathrm{dist}(x, S_l(f))\ :\ x \in S_L(f)\ \} < \infty \ .$$

*-compact functions are *asymptotically well-behaved*, which precisely means that any sequence like $\{y_k\}$ is minimizing. Many functions are *-compact, e.g. all the inf-compact ones; see [7] for further discussion.

THEOREM 3.9. *Assume that $\rho_k \leq \rho_{max} < +\infty$, $\gamma_k \leq \gamma_{max} < +\infty$, and f is *-compact; then, $f_\infty = f^*$.*

Proof. It is easy to realize that boundedness of ρ_k and γ_k implies that $\{\|z_k\|\} \to 0$ and $\sigma_k \to 0$; just look to (3.2) and consider that (3.1) implies $\mu_k \leq \gamma_{max}\eta(1-\delta)$ (that is, boundedness of γ implies boundedness of μ). Now, assume by contradiction that $f^* < l = f_\infty - \lambda$ for some $\lambda > 0$, and take $\hat{y}_k$ as the projection of y_k on $S_l(f)$, i.e., $\hat{y}_k = \arg\inf\{\ \|y_k - x\|\ :\ x \in S_l(f)\ \}$. Since $f(y_k)$ is nonincreasing, $f(y_k) \leq f(y_1) = L$ for all k. Hence, since z_k is a σ_k-subgradient of f at y_k

$$f_\infty - \lambda = f(\hat{y}_k) \geq f(y_k) + z_k(\hat{y}_k - y_k) - \sigma_k \geq f_\infty - \|z_k\|\ \|\hat{y}_k - y_k\| - \sigma_k \ .$$

From *-compactness $\|\hat{y}_k - y_k\| \leq e(l, L) < \infty$ and, taking into account that $\{z_k\} \to 0$ and $\{\sigma_k\} \to 0$, we get the desired contradiction. □

4. Implementation and numerical results.

4.1. Implementation issues. The proposed algorithmic scheme has several implementations details which may significantly impact the practical performances of the algorithm. In the following we describe several of them, detailing onto the choices that were used to obtain the results reported in §4.2.

- The state-of-the-art, general-purpose, commercial solver `Cplex` version 12.2 was used to solve the Master Problem. `Cplex` can solve both Quadratically-Constrained Quadratic Programs such as the primal master problem (2.10) and Second-Order Cone Programs such as the formulation (2.13) of the dual master problem. As the computational results will show, choosing the "right" formulation definitely has an impact on the running time of the approach.
- The set of "important" subgradients was chosen *at every SS* as $\mathcal{I} = \{\ i \in \mathcal{B}\ :\ \alpha_i \leq \zeta\sigma^*\ \}$, where $\zeta \geq 0$ is a parameter and σ^* is that of the *previous iteration* (since $\mathcal{I}$ has to be chosen *before* solving the master problem). The set is kept fixed during the following NS, up until the next SS, except possibly for inserting the (subgradient corresponding to the) newly obtained point x_+ if $\alpha_+ \leq \zeta\sigma^*$ (where σ^* is now that of the current iteration, which is known when the condition is checked). On the contrary, aggregated subgradients are *never* inserted in $\mathcal{I}$, as doing so actually results in negative linearization errors in practice. Since $\zeta \geq 0$, all subgradients that are "active" at the stability center y (i.e., have $\alpha_j = 0$) are surely comprised in $\mathcal{I}$, and therefore (2.7) holds. A "small" value of ζ keeps in $\mathcal{I}$ only those subgradients that are "close" to being active in y, while larger and larger values of ζ imply larger sets of "important" subgradients.
- Rather than checking the stopping conditions (3.1) and (3.2) for the current value of ρ, and then ensuring that eventually ρ decreases to a "small enough" value to attain a solution with the required accuracy, as implied by the results in §3.3, one could rather use the tests

$$\frac{1}{2\left(\mu^* + \bar{\rho} + \sum_{i\in\mathcal{B}} \lambda_i^* \epsilon_i\right)} \left\|\sum_{i\in\mathcal{B}} \lambda_i^* \hat{g}_i\right\|^2 + \sum_{i\in\mathcal{B}} \lambda_i^* \hat{\alpha}_i + \frac{\mu^*}{\gamma} \leq \eta(1-\delta) f(y)$$

$$\frac{1}{2(\mu^* + \bar{\rho} + \kappa)} \left\|\sum_{i\in\mathcal{B}} \lambda_i^* g_i\right\|^2 + \sum_{i\in\mathcal{B}} \lambda_i^* \alpha_i \leq \eta f(y) \tag{4.1}$$

for a properly chosen "small" value of $\bar{\rho}$. This does not require any substantial change to the convergence analysis, and choosing an appropriate (not too small) value of $\bar{\rho}$ such that the attained solution is actually η-optimal (in *relative* sense, thanks to the scaling factor $f(y)$) in the end is usually easy enough.

- Decreases of t at Step 3 of the algorithm are actually implemented as $t := \min\{\ t, \sigma^*\ \}/2$ to ensure that the ϵ_i actually change (which may not happen if t is still "large") and therefore that the master problem at the subsequent iteration is actually significantly different.
- Heuristics for increasing/decreasing ρ are of utmost importance for the practical effectiveness of the approach. Following [6] both "short-term" approaches, only based on information gathered in the current iteration (or at most in the few preceding ones), and "long-term" ones which take into account data pertaining to the overall convergence behavior of the algorithm, were implemented. For the former, one can basically copy the approaches in [23, 6] by considering the two-piece quadratic model of (3.10) restricted along the previous direction d^* and computing its minimum ξd^*, with $\xi \in (0, 1]$. This can be done in constant time by evaluating the two-piece quadratic model in the only five value of ξ where the minimum can be (the minimum of each individual quadratic function, if any, the intersection between of the two functions, and $\xi = 1$). Then, one can set ρ to the value that would place the minimum of the aggregated model there *assuming that d^* would remain the same* (which we know it won't), that is, the ρ' which solves

$$\xi d^* = -\frac{\xi \hat{g}}{\mu^* + \rho + \epsilon} = -\frac{\hat{g}}{\mu^* + \rho' + \epsilon}$$

  (again a $O(1)$ computation). Since this value may be either too large, or too small, than the previous one, this approach is typically "damped" by projecting ρ' onto the interval $[\underline{m}\rho, \overline{m}\rho]$ centered on the previous value for some fixed $0 < \underline{m} < 1 < \overline{m}$. As far as "long-term" approaches go, the idea is to monitor that ρ never becomes "too large too rapidly", thereby causing long sequences of very "short" SSs which do not actually improve the objective value much, nor "remains too small too long" thereby causing long sequences of NSs between any two SSs. One way in which this can de done is decreasing ρ, or at least inhibiting further increases, if $\|z^*\|^2$ is already "much smaller" than σ^*, as both terms eventually needs to become "small" for the algorithm to stop (cf. (4.1)). This can be done in different ways; one, for instance, is to try to ensure that the ratio $\sigma^*/\|z^*\|^2$ is comprised into some interval $[\pi, 1/\pi]$ for some fixed $0 < \pi < 1$, and inhibiting decreases/increases of ρ (whichever is appropriate) if the ratio is already outside the interval. This of course can be done on bundle approaches based on the standard cutting-plane model, too, and in fact the heuristics implemented for the newly proposed bundle method closely mirrors these already present in a "standard" code already used with success in several applications ([4, 8, 9, 10] among the others).
- As far as control of the bundle size is concerned, a classical approach (again inspired from the classical cutting-plane version) is to keep track of the number of consecutive iterations that any given subgradient is "useless" (i.e., has

$\lambda_i^* = 0$) and remove all subgradients for which this count is larger than a given threshold. This can already contribute to keep the bundle size controlled by discarding information that seem to have few chances to ever return to be significant again. In order to further decrease the master problem cost one can also impose any given hard limit the maximum bundle size; as soon as the limit is hit, first all subgradients with $\lambda_i^* = 0$ in the current solution are discarded (in order of their count). If this is not enough, aggregation is performed (cf. §3.2) and *two* subgradients are discarded (in reverse order of λ_i^*, i.e., starting from those with smallest multiplier) in order to make space for the aggregated subgradient and the newly added one.

4.2. Numerical results. The proposed algorithm has been coded in `C++` and compared with a `C++` code based on the standard cutting-plane model [4, 8, 9, 10] on a 2.10GHz Intel T8100 CPU with 2Gb of RAM, under a i686 GNU/Linux (Ubuntu 10.04 LTS), compiled with `g++` version 4.4.3. We have fixed $\eta =$ `1e-6`, in (4.1), i.e., required six significant digits of precision in the optimal function value. The (numerous) algorithmic parameters were tuned (simultaneously for all functions, but) individually for each algorithm to find the most performing settings for the given test set. Also, comparison with the variable metric algorithm of [33] has been possible using results reported in the literature.

We have first tested the algorithms on 14 standard convex nondifferentiable functions, described in Table 4.1; for more details (comprised optimal value, optimal solution and starting point) the interested reader can consult [26] for 1–9, [24] for 10–11, and [22] for 12 (the "very easy" functions 13 and 14 need little explanation).

The results are reported in the following Table 4.2. In the Table, columns "CPB" refer to the standard bundle approach using the cutting plane model, columns "VMNC" refer to the variable metric algorithm of [33], and columns "QPB" to the algorithm proposed in this paper. For all algorithms, column "$\#f$" reports the total number of function evaluations and column "gap" reports the final relative gap w.r.t. the "true" optimal value (either known beforehand, or obtained by running CPB with very high required accuracy and unlimited available running time). For CPB and QPB, column "time" reports the total CPU time required. Finally, for QPB the column "MP" reports the total number of master problems solved (which may be larger than the number of function evaluations due to inner iterations), column "SS" reports the total number of Serious Steps, and column "ptime" reports the running time of the algorithm if the primal master problem (2.10) is solved instead of the dual (2.13).

The Table shows that the newly proposed algorithm is not particularly effective. The number of function evaluations is only occasionally better than that of the other contenders, and some times significantly worse. The total number of iterations (solutions of the master problem) is even larger due to the need of computing inner iterations; most often these are few, but on occasion they represent a significant fraction of the iterations. Furthermore, the running time is even worse due to the need

	name	n	function
1	CB2	2	$f(x) = \max\{\, x_1^2 + x_2^4\,,\ (2-x_1)^2 + (2-x_2)^2\,,\ 2e^{-x_1+x_2}\,\}$
2	CB3	2	$f(x) = \max\{\, x_1^4 + x_2^2\,,\ (2-x_1)^2 + (2-x_2)^2\,,\ 2e^{-x_1+x_2}\,\}$
3	DEM	2	$f(x) = \max\{\, 5x_1 + x_2\,,\ -5x_1 + x_2\,,\ x_1^2 + x_2^2 + 4x_2\,\}$
4	QL	2	$f(x) = \max\{\, x_1^2 + x_2^2\,,\ x_1^2 + x_2^2 + 10(-4x_1 - x_2 + 4)\,,\ x_1^2 + x_2^2 + 10(-x_1 - 2x_2 + 6)\,\}$
5	LQ	2	$f(x) = \max\left\{\, -x_1 - x_2\,,\ -x_1 - x_2 + (x_1^2 + x_2^2 - 1)\,\right\}$
6	Mifflin1	2	$f(x) = -x_1 + 20\max\left\{\, x_1^2 + x_2^2 - 1\,,\ 0\,\right\}$
7	Rosen	4	$f(x) = \max\{\, f_1(x)\,,\ f_1(x) + 10f_2(x)\,,\ f_1(x) + 10f_3(x)\,, f_1(x) + 10f_4(x)\,\}$ where $f_1(x) = x_1^2 + x_2^2 + 2x_3^2 + x_4^2 - 5x_1 - 5x_2 - 21x_3 + 7x_4$ $f_2(x) = x_1^2 + x_2^2 + x_3^2 + x_4^2 + x_1 - x_2 + x_3 - x_4 - 8$ $f_3(x) = x_1^2 + 2x_2^2 + x_3^2 + 2x_4^2 - x_1 - x_4 - 10$ $f_4(x) = x_1^2 + x_2^2 + x_3^2 + 2x_1 - x_2 - x_4 - 5$
8	Maxq	20	$f(x) = \max_{1\le i\le 20} x_i^2$
9	Maxl	20	$f(x) = \max_{1\le i\le 20} \mid x_i \mid$
10	Maxquad	10	$f(x) = \max_{1\le k\le 5} (\, xA_kx - b_kx\,)$
11	TR48	48	$f(x) = \sum_{j=1}^{48} d_j \max_{1\le i\le 48}(x_i - a_{ij}) - \sum_{i=1}^{48} s_i x_i$
12	Shor	5	$f(x) = \max_{1\le i\le 10}\left\{b_i \sum_{j=1}^{5}(x_j - a_{ij})^2\right\}$
13	Smooth	n	$f(x) = \sum_{i=1}^{n} x_i^2$
14	AbsVal	n	$f(x) = \sum_{i=1}^{n} \mid x_i \mid$

TABLE 4.1
Standard test functions

to solving a more complex master problem; the slowdown can be relevant if the dual master problem is addressed, but can be downright dramatic if the primal formulation is solved instead.

It therefore appears that insertion of the "artificial" second-order information in the model does little to improve the actual convergence rate, which is not entirely surprising since this information can be thought to have little to do with the actual second-order behavior of the function. Particularly indicative in this respect are the "very easy" functions 13 and 14, that are solved extremely efficiently by the standard bundle approach and much less so by the newly proposed one. For 13, this is likely due to the fact that first-order information "by chance" points directly towards the optimum, and the "noise" provided by the extra quadratic terms in the model deviates the algorithm from the extremely promising direction it'd have when using the standard cutting plane model. Function 14 may be thought to lack any

		CPB			VMNC		QPB					
	n	$\#f$	time	gap	$\#f$	gap	MP	$\#f$	SS	time	ptime	gap
1	2	21	0.01	3e−7	16	3e−7	19	18	12	0.14	0.54	3e−7
2	2	34	0.01	3e−7	17	0e+0	19	17	12	0.08	0.49	6e−7
3	2	12	0.01	0e+0	20	1e−7	66	43	40	0.48	2.74	6e−7
4	2	21	0.01	1e−7	18	3e−7	26	25	14	0.18	1.58	1e−7
5	2	10	0.01	3e−8	10	2e−7	34	19	15	0.19	0.79	4e−7
6	2	30	0.01	2e−7	59	8e−6	36	28	21	0.26	2.89	2e−7
7	4	43	0.01	2e−7	32	6e−7	64	63	15	0.63	5.08	3e−7
8	20	141	0.01	1e−6	111	9e−6	135	134	52	1.44	94.07	2e−7
9	20	31	0.01	0e+0	23	0e+0	91	73	64	1.18	32.40	6e−6
10	10	116	0.02	6e−7	89	3e−6	118	117	33	1.52	24.27	3e−7
11	48	140	0.01	0e+0	295	4e−6	175	166	53	4.98	3016.11	9e−7
12	5	51	0.01	7e−7	30	1e−6	62	61	20	0.63	5.72	5e−7
13	100	2	0.01	0e+0	—	—	21	20	11	0.13	76.20	6e−9
14	100	3	0.01	0e+0	—	—	6	5	3	0.06	0.71	4e−7
13	200	2	0.01	0e+0	—	—	20	19	9	0.17	12.99	1e−7
14	200	3	0.01	0e+0	—	—	6	5	3	0.04	1.93	5e−5

TABLE 4.2
Results for standard test functions

meaningful second-order information everywhere, and the surrogate provided by the quadratic model proves to be worse than just relying on the first-order information alone, which, analogously to the previous case, turns out to be "quite exact" already.

It therefore appears the introduction of second-order information on the model can only be effective if it actually "has something to do" with the actual second-order behavior of the objective function. To test this hypothesis we developed a new class of functions, called "QR(n, m)", with the form

$$f(x) = \max_{j=1,\dots,m} \left\{ b_j \|x - x_j\|^2 + a_j \right\}$$

where each a_j and all the components of each fixed center x_j is a random number uniformly drawn in $[-100, 100]$, while each b_j is a random number uniformly drawn in $[0, 100]$. That is, these functions "have a similar shape" to that of the quadratic model employed in QPB, but of course the actual data characterizing each function is unknown to the algorithm and it is only approximated by using information iteratively extracted from the oracle. We have tested both CPB and QPB on a set of functions constructed as follows: for each $n \in \{10, 100, 200, 1000\}$ we have considered two pairs (n, m) (variables and quadratic components) of the type (n, n) and $(n, 10n)$. For any pair (n, m) we have generated 5 QR(n, m) functions for five different values of the seed to the random number generator. Results of these experiments are reported in

Table 4.3, with each row representing the average of all the 5 functions with the same (n, m).

		CPB			QPB				
n	m	$\#f$	time	gap	MP	$\#f$	SS	time	gap
10	10	33	0.01	3e−7	44	38	23	0.59	4e−7
10	100	52	0.01	2e−7	53	48	27	0.81	3e−7
100	100	90	0.01	3e−7	72	67	35	1.09	4e−7
100	1000	158	0.14	4e−7	100	95	46	2.05	5e−7
200	200	121	0.04	6e−7	97	91	47	1.93	4e−7
200	2000	286	1.18	3e−7	174	168	61	7.04	5e−7
1000	1000	291	3.15	4e−7	178	173	66	9.52	4e−7
1000	10000	541	64.30	5e−7	323	317	87	54.55	5e−7

TABLE 4.3
Results for $QR(n, m)$ test functions

Table 4.3 paints quite a different picture than Table 4.2: QPB converges consistently faster than the CPB for all but the smallest values of n, up to the point that for the largest values of n and m the running time is actually (slightly) smaller, despite the more complex master problem solved at each iteration. While one may argue that the QR(n, m) functions are "too good a fit" for QPB, and that this computational advantage does not seem to be replicated on more varied functions, we believe that these results are an indication that a piecewise-quadratic model containing "appropriate" second-order information can actually result in a more efficient algorithm. Therefore, variants of the proposed algorithm which incorporate less "rigid" forms of second-order information than a scalar multiple of the identity matrix could turn out to be interesting.

5. Concluding remarks. We have developed a new version of bundle method based on a piecewise-quadratic model which does not necessarily support the objective function on below. We have shown that the quadratic terms in the model can be adjusted in such a way that it supports the objective function on a properly chosen set of "important" points, and that this is enough to ensure convergence. A nice feature of the algorithm is that it naturally allows for a hybrid stabilization which uses both a trust-region term (useful for ensuring compactness in spite of variation of the weights of the quadratic terms in the model) and a proximal term (useful for online tuning of the stabilization parameters). The convergence analysis of the approach allows for incorporation of important practical aspects such as heuristics for handling the stabilization parameter(s) and aggregation, that turns out to be surprisingly more complex in this case than when the usual cutting plane model is employed. Numerical results on the newly proposed method show promise only for a special class of functions for which the piecewise-quadratic model is "a natural fit"; while this means that the algorithm, in its current form, does not seem to be particularly useful for solving

many problems (that do not have that form), we believe that the results indicate that versions using richer forms of second-order information could actually prove to be competitive. Of particular interest in this sense is the fact that aggregation allows restricting the number of quadratic pieces to any fixed value (downto two), which may ease concerns about dealing with many dense quadratic constraints in the master problem. We also believe that the use of quadratic models could be usefully extended to bundle methods designed to jointly deal with nonconvexity and nonsmoothness [11, 13, 14].

REFERENCES

[1] L. Bahiense, N. Maculan, and C. Sagastizábal, *The volume algorithm revisited: relation with bundle methods*, Mathematical Programming, 94 (2002), pp. 41–70.

[2] H. Ben Amor, J. Desrosiers, and A. Frangioni, *On the Choice of Explicit Stabilizing Terms in Column Generation*, Discrete Applied Mathematics, 157 (2009), pp. 1167–1184.

[3] R. Cominetti and R. Correa, *A Generalized Second Order Derivative in Nonsmooth Optimization*, SIAM Journal on Control and Optimization, 28 (1990), pp. 789–809.

[4] T. Crainic, A. Frangioni, and B. Gendron, *Bundle-based Relaxation Methods for Multicommodity Capacitated Fixed Charge Network Design Problems*, Discrete Applied Mathematics, 112 (2001), pp. 73–99.

[5] A. Frangioni, *Solving semidefinite quadratic problems within nonsmooth optimization algorithms*, Computers & Operations Research, 21 (1996), pp. 1099–1118.

[6] ———, *Dual-Ascent methods and Multicommodity flow problems*, PhD thesis, TD 5/97, Dipartimento di Informatica, Università di Pisa, Pisa, Italy, 1997.

[7] ———, *Generalized Bundle Methods*, SIAM Journal on Optimization, 13 (2002), pp. 117–156.

[8] ———, *About Lagrangian Methods in Integer Optimization*, Annals of Operations Research, 139 (2005), pp. 163–193.

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

[10] A. Frangioni, A. Lodi, and G. Rinaldi, *New approaches for optimizing over the semimetric polytope*, Mathematical Programming, 104 (2005), pp. 375–388.

[11] A. Fuduli, M. Gaudioso, and G. Giallombardo, *Minimizing nonconvex nonsmooth functions via cutting planes and proximity control*, SIAM Journal on Optimization, 14 (2004), pp. 743–756.

[12] M. Gaudioso, G. Giallombardo, and G. Miglionico, *An incremental method for solving convex finite min-max problems*, Mathematics of Operations Research, 31 (2006), pp. 173–187.

[13] M. Gaudioso and E. Gorgone, *Gradient set splitting in nonconvex nonsmooth numerical optimization*, Optimization Methods and Software, 25 (2010), pp. 59–74.

[14] M. Gaudioso, E. Gorgone, and M. F. Monaco, *Piecewise linear approximations in nonconvex nonsmooth optimization*, Numerische Mathematik, 113 (2009), pp. 73–88.

[15] M. Gaudioso and M. F. Monaco, *Quadratic approximations in convex nondifferentiable optimization*, SIAM Journal Control and Optimization, 29 (1991), pp. 58–70.

[16] J. Hiriart-Urruty and C. Lemaréchal, *Convex analysis and minimization algorithms Vol. I-II*, Springer-Verlag, 1993.

[17] J.-B. Hiriart Urruty, J.-J. Strodiot, and V. Hien Nguyen, *Generalized Hessian Matrix and Second Order Optimality Conditions for Problems with $C^{1,1}$ data*, Applied Mathematics and Optimization, 11 (1984), pp. 43–53.

[18] K. Kiwiel, *Efficiency of Proximal Bundle Methods*, Journal of Optimization Theory and Ap-

plications, 104 (2000), pp. 589–603.
[19] ———, *A proximal bundle method with approximate subgradient linearizations*, SIAM Journal on Optimization, 16 (2006), pp. 1007–1023.
[20] ———, *A proximal-projection bundle method for Lagrangian relaxation, including semidefinite programming*, SIAM Journal on Optimization, 17 (2006), pp. 1015–1034.
[21] K. Kiwiel and C. Lemaréchal, *An inexact bundle variant suited to column generation*, Mathematical Programming, 118 (2009), pp. 177–206.
[22] K. C. Kiwiel, *Methods of descent for nondifferentiable optimization*, vol. 1133 of Lecture notes in mathematics, Springer-Verlag, 1985.
[23] ———, *Proximity control in bundle methods for convex nondifferentiable minimization*, Mathematical Programming, 46 (1990), pp. 105–122.
[24] C. Lemaréchal and R. Mifflin, *Nonsmooth optimization*, in Proceedings of a IIASA Workshop, Oxford, 1977, Pergamon Press.
[25] C. Lemaréchal, A. Nemirovskii, and Y. Nesterov, *New variants of bundle methods*, Mathematical Programming, 69 (1995), pp. 111–147.
[26] M. Mäkelä and P. Neittaanmäki, *Nonsmooth optimization*, World Scientific, 1992.
[27] O. L. Mangasarian, *Nonlinear programming*, McGraw-Hill, New York, 1969.
[28] R. Mifflin and C. Sagastizábal, *On $\mathcal{VU}$-theory for functions with primal-dual gradient structure*, SIAM Journal on Optimization, 11 (2000), pp. 547–571.
[29] ———, *A $\mathcal{VU}$-algorithm for convex minimization*, Mathematical Programming, 104 (2005), pp. 583–608.
[30] R. Mifflin, D. Sun, and L. Qi, *Quasi-Newton bundle-type methods for nondifferentiable convex optimization*, SIAM Journal on Optimization, 8 (1998), pp. 583–603.
[31] A. Ouorou, *A proximal cutting plane method using Chebychev center for nonsmooth convex optimization*, Mathematical Programming, 119 (2009), pp. 239–271.
[32] H. Schramm and J. Zowe, *A version of the bundle idea for minimizing a nonsmooth function: conceptual idea, convergence analysis, numerical results*, SIAM Journal on Optimization, 1 (1992), pp. 121–152.
[33] J. Vlček and Luksăn, *Globally convergent variable metric method for nonconvex nondifferentiable unconstrained minimization*, Journal of Optimization Theory and Applications, 111 (2001), pp. 407–430.