SIAM J. Optim.
Vol. 21, No. 4, pp. 1418–1438

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 the bundle of information. Numerical results are also reported.

Key words. nondifferentiable optimization, bundle methods, quadratic model

AMS subject classifications. 90C26, 65K05

DOI. 10.1137/100817930

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 necessarily 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 when x has to belong to a known and "easy" convex set X or, alternatively, when f is an extended-valued 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., [21, 8, 23, 13]) that will not be discussed here for the sake of notational simplicity. Also, techniques developed to cope with inexact computation of the objective function [22] and/or the constraints [24] 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, with a slight abuse of notation, the set of their indices. In what follows we also denote by $\|\cdot\|$ the Euclidean norm in $\mathbb{R}^n$, and by ab the standard inner product of the vectors a and b. The *bundle* $\mathcal{B}$ is typically used for 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., $\hat{f}_{\mathcal{B}} \le f$). This is used to drive the choice of the next iterate, clearly in the region where $\hat{f}_{\mathcal{B}}$ improves over the

[*] Received by the editors December 13, 2010; accepted for publication (in revised form) August 26, 2011; published electronically December 8, 2011.
http://www.siam.org/journals/siopt/21-4/81793.html

[†] 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 (astorino@icar.cnr.it).

[‡] Dipartimento di Informatica, Università di Pisa, Polo Universitario della Spezia, Via dei Colli 90, 19121 La Spezia, Italy (frangio@di.unipi.it).

[§] Dipartimento di Elettronica Informatica e Sistemistica, Università della Calabria, 87036 Rende (CS), Italia (gaudioso@deis.unical.it, egorgone@deis.unical.it).

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 unbounded below. In *proximal* bundle methods, one selects a *stability center* $y \in \mathbb{R}^n$ (e.g., the best iterate found so far) which leads to the corresponding 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 y. Then, for an appropriately chosen *proximity parameter* $\rho > 0$, one finds the optimal solution d^* of the *master problem*

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

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

$$\text{(1.2)} \qquad \min_{\lambda}\left\{\, \frac{1}{2\rho}\left\|\sum_{i\in\mathcal{B}} \lambda_i g_i\right\|^2 + \sum_{i\in\mathcal{B}} \lambda_i \alpha_i \;:\; \lambda \in \Lambda \,\right\}$$

(where $\Lambda = \{\, \lambda \geq 0 \;:\; \sum_{i\in\mathcal{B}} \lambda_i = 1\,\}$ is the unitary simplex of appropriate dimension) is also relevant. From the algorithmic viewpoint, the optimal solution λ^* of (1.2) reveals the *aggregated subgradient and linearization error*

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

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. $|\mathcal{B}|$ [6]. From the analytic viewpoint it is easy to verify that g_i belongs to the ϵ-subdifferential of f at y for $\epsilon = \alpha_i$ (i.e., $g_i \in \partial_{\alpha_i} f(y)$), and consequently one has that $z^* \in \partial_{\sigma^*} f(y)$; thus, whenever both $\|z^*\|$ and σ^* are "small," an approximate optimality condition is reached. Let us mention here that different forms of stabilization ([26, 8] and many others) can be used with only slight modifications to the master problems and next to none to the convergence theory [8]. In particular, (1.1) with the proximal term $\rho\|d\|^2/2$ in the objective function replaced by a *trust-region* constraint $\gamma\|d\|^2 \leq 2$ would lead to an algorithm with basically the same convergence properties. This has not received much attention in the past, most likely because the master problem then becomes a quadratically constrained problem, hence potentially more difficult to solve in practice than the linearly constrained quadratic problem (1.1); as we will see, trust region constraints are instead basically "free" in our case.

Despite being useful in several applications ([3, 5, 10] among many others), bundle 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 an optimum, than that of the second-order model of Newton-type approaches. Whence the push toward second-order bundle-type algorithms [32, 36] 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 into the subspace where f is essentially "kinky" and therefore accumulation of linear inequalities is efficient. This $\mathcal{VU}$-theory [30] allows us to develop, under appropriate assumptions, second-order-type approaches that are rapidly convergent both in theory and in practice [31]. 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 [4, 18], one may want to develop a piecewise-smooth model of f. The most natural form is that of a piecewise-quadratic (convex) model [16]. This is the kind of model envisioned in [27], where the quadratic components are second-order matrices produced by the oracle (or approximated by finite differences) at the sample points. Yet, the master problem solved in that paper does not use a piecewise-quadratic model, but one similar to [32] with a single quadratic term obtained by averaging the second-order matrices. In contrast, we aim at keeping the structure of the piecewise-quadratic model intact, although we also allow aggregation whenever appropriate; yet, our aggregation generates another quadratic function in the piecewise-quadratic model, rather than being applied only to the quadratic part. An issue with piecewise-quadratic models is that by necessity they lose the property of the cutting plane model, that is, being a lower approximation of f *everywhere*. Yet, the latter property is not strictly necessary: the recent paper [33] 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 $\psi_y(x) > 0$. We will show that one can work with a model that, while actually overestimating f somewhere, never does so *knowingly* at least on a (potentially very small) set of points. We obtain this by working on the scaling parameters of the quadratic part; our approach is therefore significantly different from that in [27] in this respect, as there the result was obtained by modifying the linear components of the model using *proximity measures*. 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, in the manner of [27].

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.

2. The piecewise-quadratic model.

For every (ordered) pair $(i, j) \in \mathcal{B} \times \mathcal{B}$, the *mutual linearization error* computed in x_j for the ith 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 $q_i(x) = f(x_i) + g_i(x - x_i) + \epsilon_i \|x - x_i\|^2/2$, the quadratic expansion of f generated at x_i, one has that

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

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 no q_i *knowingly overestimates* f on the points in $\mathcal{I}$, that is, $f(x_j) \geq q_i(x_j)$ for all $(i, j) \in \mathcal{B} \times \mathcal{I}$. We can take $\epsilon_{ii} = +\infty$ in (2.1), as when $\mathcal{B} = \{i\}$

the property clearly holds for any ϵ_i. Of course, the property is then transmitted from the individual q_i to the natural piecewise-quadratic model of f,

$$\breve{f}_{\mathcal{B}}(x) = \max\{\, q_i(x) \;:\; i \in \mathcal{B} \,\}, \tag{2.3}$$

which, since $q_i(x_i) = f(x_i)$ by definition, therefore (like $\hat{f}_{\mathcal{B}}$) satisfies

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

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

$$\begin{aligned} q_i(d) &= f(y) + \hat{g}_i d - \hat{\alpha}_i + \epsilon_i \|d\|^2/2, \qquad \text{where} \\ \hat{\alpha}_i &= \alpha_i - \epsilon_i \|y - x_i\|^2/2 \qquad \text{and} \qquad \hat{g}_i = g_i + \epsilon_i (y - x_i). \end{aligned} \tag{2.4}$$

Note 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,

$$\epsilon_i \|d\|^2/2 + \hat{g}_i d - \hat{\alpha}_i \geq g_i d - \alpha_i \qquad \forall\, i \in \mathcal{B} \text{ and } \forall\, d \in \mathbb{R}^n. \tag{2.5}$$

In our development we will assume that y *is one of the tentative points* x_i, and we denote by c the index such that $y = x_c$. It is easy to verify (use $\alpha_i = \alpha_{ic}$) that

$$c \in \mathcal{I} \qquad \Rightarrow \qquad \hat{\alpha}_i \geq 0 \qquad \forall\, i \in \mathcal{B}. \tag{2.6}$$

This property is essential, and $\mathcal{I} = \{c\}$ is the minimal possible set of interpolating points for our analysis to work. In fact, with the corresponding translated model $\breve{f}_{\mathcal{B}}(d) = \breve{f}_{\mathcal{B}}(y + d) - f(y)$ we can define a *proximal/trust region* master problem

$$\begin{aligned} &\min\nolimits_d \;\left\{\; \breve{f}_{\mathcal{B}}(d) \;+\; \rho\|d\|^2/2 \;:\; \gamma\|d\|^2 \leq 2 \;\right\} \\ &= \min\nolimits_{v,d} \;\left\{\; v \;+\; \rho\|d\|^2/2 \;:\; v \geq \; \epsilon_i\|d\|^2/2 + \hat{g}_i d - \hat{\alpha}_i, \quad i \in \mathcal{B} \;,\; \gamma\|d\|^2 \leq 2 \;\right\}, \end{aligned} \tag{2.7}$$

which exposes both a proximal term weighted with ρ and a trust region term governed by γ. Its dual (use, e.g., the strict converse duality theorem [29, p. 117])

$$\min_{\lambda,\mu} \left\{ \frac{\left\|\sum_{i\in\mathcal{B}} \lambda_i \hat{g}_i\right\|^2}{2\left(\mu + \rho + \sum_{i\in\mathcal{B}} \lambda_i \epsilon_i\right)} + \sum_{i\in\mathcal{B}} \lambda_i \hat{\alpha}_i + \frac{\mu}{\gamma} \;:\; \lambda \in \Lambda \;,\; \mu \geq 0 \right\} \tag{2.8}$$

has similar primal-dual relationships to (1.3)

$$d^* = -\frac{\sum_{i\in\mathcal{B}} \lambda_i^* \hat{g}_i}{\mu^* + \rho + \sum_{i\in\mathcal{B}} \lambda_i^* \epsilon_i} \;,\; v^* = -\|d^*\|^2 \left(\mu^* + \rho + \frac{\sum_{i\in\mathcal{B}} \lambda_i^* \epsilon_i}{2} \right) - \sum_{i\in\mathcal{B}} \lambda_i^* \hat{\alpha}_i, \tag{2.9}$$

which show that, under (2.6), the optimal value of (2.8) is nonnegative, and therefore the optimal value of (2.7) is nonpositive. It is easy to check that this means that $v^* = \breve{f}_{\mathcal{B}}(d^*) \leq 0$, which implies 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. Note that both the "pure" proximal ($\gamma = 0$) and trust region ($\rho = 0$) versions are unavoidably quadratically constrained problems, so there is no longer any reason to prefer one to the other; furthermore, we will see that having both stabilizing terms not only creates no problems in the convergence analysis (this is not surprising since [8, Theorem 3.2] shows that stabilizing terms can "look like a proximal term, a trust region term, or both"), but it actually helps. The apparently nasty fractional term in the objective function of (2.8) can be dealt with by reformulating it as

$$\begin{aligned} &\min_{\lambda,\mu,t,s} \quad t + \sum_{i\in\mathcal{B}} \lambda_i \hat{\alpha}_i + \mu/\gamma \\ (2.10)\qquad & ts \geq \left\| \sum_{i\in\mathcal{B}} \lambda_i \hat{g}_i \right\|^2 , \quad s = 2\left(\mu + \rho + \sum_{i\in\mathcal{B}} \lambda_i \epsilon_i \right), \ \lambda \in \Lambda, \ \mu \geq 0, \end{aligned}$$

which is a rotated second-order cone program (SOCP) solvable by standard approaches (it can be transformed into a standard SOCP with well-known trick $ts = (t+s)^2/4 - (t-s)^2/4$ if need be).

An interesting feature of the new model $\breve{f}_{\mathcal{B}}$ is that it is somewhat "self-stabilized," with the ϵ_is playing a role similar to that of ρ and γ: just rewrite (2.7) as

$$\begin{aligned} &\min_{v,d} \left\{ v \ : \ v \geq \ \epsilon_i' \|d\|^2/2 + \hat{g}_i d - \hat{\alpha}_i, \quad i \in \mathcal{B}, \ \gamma\|d\|^2 \leq 2 \right\}, \\ (2.11)\qquad & \qquad\qquad \text{where} \qquad \epsilon_i' = \epsilon_i + \rho, \end{aligned}$$

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

$$(2.12)\qquad \epsilon(\lambda) = \sum_{i\in\mathcal{B}} \lambda_i \epsilon_i \qquad \text{and/or} \qquad \epsilon(\lambda)' = \sum_{i\in\mathcal{B}} \lambda_i \epsilon_i' = \epsilon(\lambda) + \rho$$

basically play the same role in (2.8). Indeed, provided that at least one of the ϵ_i is strictly positive, one could even take $\rho = \gamma = 0$ while ensuring that the master problems always have a solution. Interestingly, the classical example of instability of the (nonstabilized) cutting-plane algorithm [17] 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. Similarly, for any fixed $\varepsilon \in (0, 1/2)$ the minimization of $f(y,\eta) = \max\{|\eta|, -1 + 2\varepsilon + \|y\|\}$ on the unit ball with the cutting-plane algorithm given $(y_1, \eta_1) = (0, 1)$ as starting point requires a large number of iterations [17], while when using $\breve{f}_{\mathcal{B}}$ instead, only two iterations are required (cf. [1] for details). Thus, for a few selected examples the new model, even without stabilization, does improve on the classical cutting-plane model.

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 (2.8), which is the opposite of the optimal value of the primal master problem (2.7). Under (2.6) one has that $v(\epsilon) \geq 0$; this is crucial, because that value is used for the "approximate" stopping criterion of the algorithm:

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

Accordingly, the "true" stopping criterion is

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

with the obvious property that

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

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 nonnegative term μ^*/γ.

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

Step 1. Solve (2.7)/(2.8) 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 $i \in \mathcal{B}$. 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 increase ρ and/or γ and go to Step 1.

Step 5. Set $y = x_+$. Update $\mathcal{I}$ ensuring that (2.6) holds. Compute the ϵ_i according to (2.2) with the new y and $\mathcal{I}$. Compute the $\hat{\alpha}_i$ and $\hat{g}_i$ according to (2.4) with the new y and ϵ_i. Reset t to any value in $(0, T]$ and ρ and γ to any nonnegative value. Go to Step 1.

The core of the algorithm is the *main iteration*, consisting of 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; in this case the ϵ_i also are unchanged and only $\mathcal{B}$ (and possibly ρ/γ) varies. The fact that the ϵ_i 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 proved since (3.4) gives

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

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

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

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 $\breve{f}_{\mathcal{B}}$ (their pointwise maximum) in x_+, but f is well above $\breve{f}_{\mathcal{B}}$ there, for otherwise a descent step would have been obtained. Therefore, within a main iteration the ϵ_i 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_i 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 first step is to prove that the main iteration eventually terminates. Hence we focus on a single main iteration, denoting by the index "$_k$" all the quantities at the kth pass through Steps 1–4, removing the superscript "*" for notational simplicity. A first assumption is needed to ensure that the master problem is well defined.

Assumption 3.1. At least one among ρ, γ, and the ϵ_i is strictly positive.

Under Assumption 3.1, the objective function of (2.7) 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 [8], is no longer true when the ϵ_i 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 $\epsilon' \leq \epsilon$: therefore, Step 3 may cause an increase in $v_k(\epsilon)$, whose effect is not easy to bound. We therefore need the following result.

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

Proof. Clearly, Assumption 3.1 is satisfied. In the first case $\gamma_k \geq \gamma_1 > 0$ and therefore $||d_k|| \leq \sqrt{2/\gamma_k} \leq \sqrt{2/\gamma_1}$ for all k. In the second 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 $\rho_k \|d_k\|^2/2 \leq -v_k \leq \alpha - gd_k$. Hence $d_k \in \{\ d\ :\ \rho_1 \|d\|^2/2 \leq \alpha - gd\ \}$ ($\rho_k \geq \rho_1$), 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_i the assumption is not necessary: boundedness of $\{d_k\}$ is a consequence of monotonicity of $v_k(\epsilon)$ (cf. [8, Lemma 5.5], which does not require $f_\mathcal{B} \leq f$ it may appear that [10, Lemma 5.5] requires $f_\mathcal{B} \leq f$, but in fact it does not require it). Yet, that line of proof fails when one cannot bound the optimal value of the master problem, as may be the case when the ϵ_i decrease. The assumption itself is not overly strong. For instance, in Lagrangian relaxation often some $l > -\infty$ (the value of the best feasible solution to the original problem found so far [9]) is known such that $f \geq l$. In this case, a (linear) constraint $\breve{f}_\mathcal{B}(y + d) \geq l$ can be explicitly added to (2.7); 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 also guarantees the well-posedness of (2.7) without Assumption 3.1, in the manner of [8, Condition (P3$'$)]. Alternatively, it is enough to ensure that *one single subgradient always survives in* $\mathcal{B}$ in all iterations. However, 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* $\lim_{k\to\infty} v_k(0) \le \lim_{k\to\infty} v_k(\epsilon)$.

Proof. Under the hypotheses, $p \to \infty$ in (3.5), and therefore $\epsilon_i \to 0$. Lemma 3.2 ensures the existence of some $\infty > D \ge \|d_k\|$ such that $\|\hat{g}_i\| \le \|g_i\| + \epsilon_i D$ and $\hat{\alpha}_i \ge \alpha_i - \epsilon_i D^2/2$ 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, different schemes for updating the ϵ_i and t could be used, provided that $\epsilon_i \to 0$ holds. Lemma 3.3 allows us to prove that, eventually, the stopping criterion (3.2) holds, at least if a very conservative strategy is adopted for handling $\mathcal{B}$.

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

Proof. As already discussed, no removals imply that the 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_i are never changed. Since the descent criterion at Step 4 has not been met by hypothesis, one has for all k (using (2.5) with $d = d_k$)

$$\breve{f}_+(d_k) \ge g_+ d_k - \alpha_+ = f(y + d_k) - f(y) > -m v_k(\epsilon) \ge -v_k(\epsilon) \ge \breve{f}_k(d_k) = v_k, \tag{3.6}$$

which implies that the new constraint entering the master problem at iteration $k+1$ is not satisfied by the pair (d_k, v_k). Hence, the sequence $\{v_k(\epsilon)\}$ is monotonically nonincreasing; since, due to (2.6), $v_k(\epsilon) \ge 0$ one has that $v_\infty(\epsilon) = \lim_{k\to\infty} v_k(\epsilon) \ge 0$. 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: because 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

$$\begin{aligned}
&\epsilon_i \|d_i\|^2/2 + d_i \hat{g}_i - \hat{\alpha}_i > -m v_i(\epsilon), \\
&\epsilon_i \|d_s\|^2/2 + d_s \hat{g}_i - \hat{\alpha}_i \le v_s \le -v_s(\epsilon)
\end{aligned}$$

hold, from which we obtain $v_s(\epsilon) - m v_s(\epsilon) < \hat{g}_i(d_i - d_s) + \epsilon_i(\|d_i\|^2 - \|d_s\|^2)/2$; thus $v_\infty(\epsilon) \le 0$, 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. Either infinitely many inner iterations are performed, or the last inner iteration is infinite. In the former case (3.1) is satisfied infinitely many times, but due to Lemma 3.3 eventually (3.2) must also be satisfied, a contradiction. In the latter case Lemma 3.4 gives that $v_k(\epsilon) \to 0$; however, this clearly implies that condition (3.1) at Step 2 cannot be satisfied for all k, again a contradiction. □

Several modifications to this basic scheme are possible which may be useful in practice without requiring any significant 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 [35]), provided that this is done only finitely many times. Similarly, the convergence analysis allows for not evaluating f at x_+ (e.g., to perform a line search instead) and/or not inserting the new subgradient in $\mathcal{B}$ [8].

However, Theorem 3.5 requires a monotonically increasing $\mathcal{B}$; this makes the algorithm hardly implementable, both in theory and in practice. Yet, the strategies to reduce the size of $\mathcal{B}$ that have been developed for the methods based on the standard cutting-plane model can be adapted to our setting. 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 proved, with somewhat more convoluted arguments, for the current setting. To do so, it is convenient to introduce, in analogy with (2.12), the *aggregated data*

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

and consider the corresponding *aggregated primal master problem*

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

It is immediate to realize that (3.8) has the same optimal solution (and therefore optimal value) as (2.7): just construct its dual, whose single "variable" λ can only achieve value 1, and use (2.9). More to the point, (3.8) has the same optimal value to the modified (2.7) in which all items such that $\lambda_i^* = 0$ have been removed from $\mathcal{B}$. Therefore, let $\bar{v}_+(\epsilon)$ be the (opposite of the) optimal value to the simplified problem

$$\begin{array}{ll} \min_{v,d} & v + \rho \|d\|^2/2 \\ & v \geq \epsilon \|d\|^2/2 + \hat{g}d - \hat{\alpha}, \ v \geq \epsilon_+ \|d\|^2/2 + \hat{g}_+ d - \hat{\alpha}_+, \ \gamma \|d\|^2 \leq 2. \end{array} \tag{3.9}$$

It is easy to check that $0 \leq v_+(\epsilon) \leq \bar{v}_+(\epsilon)$, even if—possibly—all items such that $\lambda_i^* = 0$ have been removed from $\mathcal{B}$. We can then 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 to 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.

Let (d_k, v_k) and (v_+, d_+) be the optimal solution of (3.8) (and (2.7)) and (3.9), respectively: $v_+ \neq v_k$, for otherwise (3.6) would give $v_k = v_+ = \breve{f}_+(d_k) > v_k$. Hence

$$\breve{f}_+(d_+) = v_+ = \epsilon_+ \|d_+\|^2/2 + \hat{g}_+ d_+ - \hat{\alpha}_+, \tag{3.10}$$

which means that the newly added quadratic constraint is always active in the new optima. Let $s_+ = d_+ - d_k$ be the effect of the introduction of the new constraint on the optimal primal solution; we analyze separately the two mutually exclusive cases

$$\text{(i)} \ \ 2(\|\hat{g}_+\| + \epsilon_+ \|d_k\|)\|s_+\| < \tau, \qquad \text{(ii)} \ \ 2(\|\hat{g}_+\| + \epsilon_+ \|d_k\|)\|s_+\| \geq \tau$$

of "small" and "large" s_+, respectively, where the threshold $\tau = \eta(1 - m)(1 - \delta)$ uses the tolerances η and δ of the stopping criterion (3.1) and m from (3.4). In case (i), which holds in particular if either $\|\hat{g}_+\| + \epsilon_+ \|d_k\| = 0$ or $d_+ = d_k$ (i.e., $\|s_+\| = 0$), using (3.10) one has

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

where in the last passage we have used (i) and the fact that the stopping rule (3.1) is not satisfied. This rules out infinitely many steps, since at each iteration the optimal value increases by at least a fixed amount. For case (ii), we start from

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

(cf. (3.10), the definition of $\check{f}_+$, and (2.5)) to write

$$\begin{aligned}
-\bar{v}_+(\epsilon) &= v_+ + \rho\|d_+\|^2/2 \geq \epsilon\|d_k + s_+\|^2/2 + \hat{g}(d_k + s_+) - \hat{\alpha} + \rho\|d_k + s_+\|^2/2\\
&= -v(\epsilon) + \big(\hat{g} + (\rho+\epsilon)d_k\big)s_+ + (\rho+\epsilon)\|s_+\|^2/2\\
&= -v(\epsilon) - \mu_kd_ks_+ + (\rho+\epsilon)\|s_+\|^2/2, \qquad (3.11)
\end{aligned}$$

where in the last passage we have used $(\mu_k+\rho+\epsilon)d_k = -\hat{g}$ (cf. (2.9)). If $\mu_k = 0$, then $\mu_kd_ks_+ = 0$; 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

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

Hence, $-\mu_kd_ks_+ \geq \mu_k\|s_+\|^2/2$ is always true. Using (ii) in (3.11), we conclude that

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

Now, τ is constant, and $\|\hat{g}_+\| + \epsilon_+\|d_k\|$ is bounded above. In fact, 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 (2.4) is likewise bounded. Finally, all ϵ_i (comprising ϵ_+) are bounded above by t, which gives boundedness of all the $\hat{g}_i$ (comprising $\hat{g}_+$). Since $\mu_k + \rho + \epsilon \geq \rho = \rho_k$, if any ρ_h is strictly positive, then 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 worth remarking that this result sharply distinguishes the two forms of stabilization: while the trust region is a handy means of ensuring compactness but is otherwise inessential, the proximal term is necessary (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 [8, condition (P3″)]). The trust region 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).

The number of points such that $\lambda_i^* > 0$ can be very large in practice, leading to computationally expensive master problems. Indeed, while with $\hat{f}_{\mathcal{B}}$ 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, one can do better.

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 every iteration yields the so-called poorman versions of bundle methods, which are characterized by solving at each step a master problem with only *two* subgradients (for which closed formulae can be devised), and which closely resemble subgradient approaches [2].

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 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 (2.4), *exactly reproduce* $\hat{g}$, $\hat{\alpha}$, and ϵ. At first this might seem easy, because

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

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 $\hat{\alpha} = \sigma^*$; thus, that is actually 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 $\tilde{x}$ into (2.4) gives

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

one has

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

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 $\alpha^* \geq \hat{\alpha}$, 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., convexity of $\|y - \cdot\|^2$). Fortunately, there are other ways to obtain $\hat{g}$, at least if one is willing to play with ϵ. Indeed, for any $\bar{\epsilon} \in (0, \epsilon)$ (recall that $\epsilon > 0$),

$$\bar{x} = \frac{(\bar{\epsilon} - \epsilon)}{\bar{\epsilon}} y + \frac{\epsilon}{\bar{\epsilon}} \tilde{x} \qquad \Longleftrightarrow \qquad \bar{\epsilon}(y - \bar{x}) = \epsilon(y - \tilde{x})$$

has the property that

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

For the specific choice $\bar{\epsilon} = \xi\epsilon$ one has

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

The case $\xi = 0$, which implies $\bar{\epsilon} = 0$, 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 $\hat{\alpha}$. Further, 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 into $\mathcal{B}$ is simply that of the aggregated linearization, which is a *lower bound* on the true function value. 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 for 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 ϵ_{ij}/ϵ_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, which 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 inserting 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 is easy to verify from (2.9). However, there is an easy fix for this: *update the stabilization parameters*. This can be done independently for both, considering 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. (2.9)).

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

$$\rho' = \rho + \epsilon - \bar{\epsilon} > \rho \tag{3.12}$$

leads to exactly the same optimal solution μ^* as the original value ρ, 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) as the original value ρ.

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 reacts only "indirectly" to changing 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. (2.11)). Imposing that the new optimal solution $\bar{d} = -\beta\hat{g}$ reproduces the same value, i.e., that

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

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

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

For $\bar{\epsilon}' = 0$ (which implies $\rho = 0$) the only solution is

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

while for $\bar{\epsilon}' > 0$ the solution 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 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

$$\gamma'\|\bar{d}\|^2 = 2 \quad \Rightarrow \quad \gamma' = 2/(\beta^2\|\hat{g}\|^2),$$

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

$$\gamma' = \frac{8(\mu^* + \epsilon')^4}{(2\mu^* + \epsilon')^2\|\hat{g}\|^2} \leq \frac{8(\mu^* + \epsilon')^2}{\|\hat{g}\|^2}, \tag{3.13}$$

which, in case $\mu^* > 0$ and consequently $\gamma = 2(\mu^* + \epsilon')^2/\|\hat{g}\|^2$, 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

$$\gamma' = 2 \left(\frac{\bar{\epsilon}'}{(1 - \sqrt{1-\delta}) \|\hat{g}\|} \right)^2 \tag{3.14}$$

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

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

(the verification is tedious although not difficult; see [1] for details).

The above analysis shows that one can aggregate while retaining the convergence of the approach, as increasing ρ and/or γ during a main iteration 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 stalling 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 oneself between a rock and a hard place when ρ and/or γ must 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 coordinate the increase of the stabilization parameters and the decrease of ϵ. In this respect, (3.12) 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 $\rho_{\max}$ on ρ and still be able to perform aggregations as follows:

- 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. section 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'.

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 need only prove that an infinite number of descents cannot occur. For this we can disregard whatever happens during a main iteration and consider only the state of the algorithm at the end of each; therefore, from now on the index "$_k$" denotes the iteration where Step 5 is executed for the kth time, and $k \to \infty$, for otherwise nothing has to be proved. Of course, here 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 finitely stops.*

Proof. Since (3.4) is not satisfied at iteration k, then $f(y_{k+1}) \leq f(y_k) - mv_k$ where $v_k(\epsilon_k) > \eta(1-\delta)$, since (3.1) is not satisfied. Summing over k gives $f(y_k) < f(y_0) - km\eta(1-\delta)$, which for $k \to \infty$ 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. □

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 in the following definition.

DEFINITION 3.8. *Let $S_\delta(f) = \{\ x\ :\ f(x) \leq \delta\ \}$ be the level set corresponding to the f-value δ: a function f is *-compact if for all $L \geq l > f^* \geq -\infty$*

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

*-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 [8] 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 onto $S_l(f)$ (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 performance of the algorithm. In the following we describe several of them, detailing the choices that were used to obtain the results reported in section 4.2.

- The 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 (2.7) and SOCPs such as (2.10). As the computational results will show, choosing the "right" formulation definitely has an impact on the running time of the approach.
- The set $\mathcal{I}$ was chosen at Step 5 as $\mathcal{I} = \{\ i \in \mathcal{B}\ :\ \|x_i - y\| \leq \zeta\|d^*\|\ \}$, where $\zeta \geq 1$ is a parameter, which clearly guarantees (2.6).
- It appears to be beneficial to set the ϵ_i to a slightly smaller value than that dictated by (2.2), thus having the model strictly minorize f on $\mathcal{I}$.

- Rather than checking (3.1), (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 required by the results in section 3.3, we use the tests

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

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

  for a properly chosen "small" value of $\bar{\rho}$. This does not require any substantial change to the convergence analysis, and choosing a 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.
- Heuristics for increasing/decreasing ρ are of utmost importance for the practical effectiveness of the approach. Following [7] both "short-term" approaches, based only on information gathered in the current iteration (or at most in the few preceding ones) and "long-term" approaches, 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 [20, 7] by considering the two-piece quadratic model of (3.9) restricted along the previous direction d^* and computing its minimum ξd^*, with $\xi \in (0, 1]$. This can be done in constant time, since the minimum can only lie at five different values of ξ (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 would not), 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, compared to 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 neither becomes "too large too rapidly," thereby causing long sequences of very "short" descents which do not actually improve the objective value much, nor "remains too small too long," thereby causing long sequences of nondescent steps. One way in which this can be done is by decreasing ρ, or at least inhibiting further increases, if $\|z^*\|^2$ is already "much smaller" than σ^*, as both terms eventually need 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 $\sigma^*/\|z^*\|^2$ lies into some interval $[\pi, 1/\pi]$ for some fixed $0 < \pi < 1$, and inhibit decreases/increases of ρ (whichever is appropriate) if the ratio is already outside the interval.
- 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 in which 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 keeping the bundle size controlled by discarding information that seems to have few chances of ever returning to be significant again. In order to further decrease the master problem cost one can also impose any given hard limit on 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. section 3.2), and *two* subgradients are discarded (in reverse order of λ_i^*, i.e., starting with those with the 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 [5, 9, 10, 11] on a 2.10GHz Intel T8100 CPU with 2Gb of RAM, under an i686 GNU/Linux (Ubuntu 10.04 LTS) compiled with `g++` version 4.4.3. We have fixed $\eta =$ `1e-6` in (4.1), hence requiring 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 best performing settings for the given test set. Also, comparison with the variable metric algorithm of [36] has been possible using results reported in the literature.

We first tested the algorithms on 14 standard convex nondifferentiable functions, described in Table 4.1; for more details (e.g., optimal value, optimal solution, and

TABLE 4.1
Standard test functions.

	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	$f(x) = \sum_{j=1}^{48} d_j \max_{1\le i\le 48}(x_i - a_{ij}) - \sum_{i=1}^{48} s_ix_i$
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 4.2
Results for standard test functions.

		CPB			VMNC		BNL		QPB					
	n	$\#f$	time	gap	$\#f$	gap	$\#f$	gap	MP	$\#f$	SS	time	ptime	gap
1	2	21	0.01	3e−7	16	3e−7	10	0e+0	16	15	13	0.10	0.37	1e−7
2	2	34	0.01	3e−7	17	0e+0	15	0e+0	18	17	11	0.27	0.94	2e−7
3	2	12	0.01	0e+0	20	1e−7	16	0e+0	21	21	14	0.22	0.68	9e−7
4	2	21	0.01	1e−7	18	3e−7	6	0e+0	15	15	11	0.07	0.17	2e−7
5	2	10	0.01	3e−8	10	2e−7	17	3e−8	23	22	11	0.45	0.58	3e−7
6	2	30	0.01	2e−7	59	8e−6	13	0e+0	30	30	16	0.26	0.75	9e−7
7	4	43	0.01	2e−7	32	6e−7	15	0e+0	25	25	13	0.14	0.56	1e−7
8	20	141	0.01	1e−6	111	9e−6	39	3e−9	141	141	55	1.59	33.88	2e−7
9	20	31	0.01	0e+0	23	0e+0	25	5e−9	52	51	38	1.11	13.98	4e−7
10	10	116	0.02	6e−7	89	3e−6	14	7e−8	47	44	24	0.38	5.08	6e−7
11	48	140	0.01	0e+0	295	4e−6	—	—	184	184	59	9.32	1662.07	6e−7
12	5	51	0.01	7e−7	30	1e−6	8	6e−7	21	21	12	0.36	0.69	2e−7
13	100	2	0.01	0e+0	—	—	—	—	13	13	8	0.08	1.70	5e−9
14	100	3	0.01	0e+0	—	—	—	—	14	14	13	0.06	0.62	8e−7
13	200	2	0.01	0e+0	—	—	—	—	13	13	8	0.05	3.22	1e−8
14	200	3	0.01	0e+0	—	—	—	—	14	14	13	0.07	1.45	5e−8

starting point) the interested reader can consult [28] for functions 1–9, [25] for 10–11, and [19] for 12 (the "very easy" functions 13–14 need little explanation).

The results are reported in Table 4.2. In the table, columns "CPB" refer to the standard bundle approach using the cutting-plane model of [11], columns "VMNC" refer to the variable metric algorithm of [36], columns "BNL" refer to the bundle-Newton method of [27], and columns "QPB" refer 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 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 descent steps (serious steps), and column "ptime" reports the running time of the algorithm if the master problem (2.7) is solved instead of the dual (2.10).

The table shows that BNL performs uniformly better than any other among the tested algorithms. It is worth noting, however, that BNL requires computation of the Hessian matrix whenever a new element is inserted into the bundle, and consequently the number of function evaluations does not completely represent the computational burden of the method. The newly proposed algorithm often requires fewer function evaluations than CPB or VMNC on the first 12 test functions; however, the running time is worse due to the need to solve a more complex master problem. The slowdown can be relevant if the dual master problem is addressed, but can be downright disastrous if the primal formulation is solved instead. The idea of inserting this "simplified" second-order information into the model does not appear to be particularly fruitful for test functions such as TR48 (function 11), which is in fact polyhedral, nor for the "very easy" functions 13 and 14, which 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 toward the optimum, and the "noise" provided by the extra quadratic terms in the model deviates the algorithm from the extremely promising direction it would have when

Table 4.3
Results for QR(n, m) *test functions.*

		CPB			QPB				
n	m	$\#f$	time	gap	MP	$\#f$	SS	time	gap
10	10	33	0.01	3e−7	25	24	13	0.16	5e−7
10	100	52	0.01	2e−7	31	31	13	0.22	4e−7
100	100	90	0.01	3e−7	56	55	21	0.81	6e−7
100	1000	158	0.14	4e−7	88	86	34	2.03	6e−7
200	200	121	0.04	6e−7	84	83	29	1.82	5e−7
200	2000	286	1.18	3e−7	164	162	48	9.68	6e−7
1000	1000	291	3.15	4e−7	173	172	54	14.19	6e−7
1000	10000	541	64.30	5e−7	300	298	66	106.12	7e−7

using the standard cutting-plane model. Function 14 may be thought to lack any 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. By contrast, the algorithm is quite efficient on Maxquad (function 10), which is the pointwise maximum of quadratic functions. It therefore appears that any second-order information introduced in the model should be somewhat related to the actual second-order behavior of f. To further 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 every component 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 shape similar to that of the quadratic model employed in QPB, but of course the actual data characterizing each function is unknown to the algorithm and 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 the two values $m = n$ and $m = 10n$. For each pair (n, m) we have generated five 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 five functions with the same (n, m).

Table 4.3 indicates that QPB requires a consistently smaller number of function evaluations than CPB; although the running time is still considerably larger, it is less so than in the previous cases, especially for large n and m. While one may argue that the QR(n, m) functions are "too good a fit" for QPB, these results seem to be 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 from 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 on-line tuning of the stabilization parameters). The convergence analysis of the approach allows for the incorporation of important practical aspects such as heuristics for handling the stabilization parameter(s) and aggregation, which 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 for only a special class of functions for which the piecewise-quadratic model is "a natural fit". Hence, while the current form of the algorithm does not seem to be particularly useful for general functions, these results seem to 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 (as low as 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 [34, 12, 14, 15].

REFERENCES

[1] A. Astorino, A. Frangioni, M. Gaudioso, and E. Gorgone, *Piecewise Quadratic Approximations in Convex Numerical Optimization*, Technical report 2/10, DEIS, Università della Calabria, Rende, Italy, 2010.

[2] L. Bahiense, N. Maculan, and C. Sagastizábal, *The volume algorithm revisited: Relation with bundle methods*, Math. Program., 94 (2002), pp. 41–70.

[3] H. Ben Amor, J. Desrosiers, and A. Frangioni, *On the choice of explicit stabilizing terms in column generation*, Discrete Appl. Math., 157 (2009), pp. 1167–1184.

[4] R. Cominetti and R. Correa, *A generalized second-order derivative in nonsmooth optimization*, SIAM J. Control Optim., 28 (1990), pp. 789–809.

[5] T. Crainic, A. Frangioni, and B. Gendron, *Bundle-based relaxation methods for multicommodity capacitated fixed charge network design problems*, Discrete Appl. Math., 112 (2001), pp. 73–99.

[6] A. Frangioni, *Solving semidefinite quadratic problems within nonsmooth optimization algorithms*, Comput. Oper. Res., 21 (1996), pp. 1099–1118.

[7] A. Frangioni, *Dual-Ascent Methods and Multicommodity Flow Problems*, Ph.D. thesis, TD 5/97, Dipartimento di Informatica, Università di Pisa, Pisa, Italy, 1997.

[8] A. Frangioni, *Generalized bundle methods*, SIAM J. Optim., 13 (2002), pp. 117–156.

[9] A. Frangioni, *About Lagrangian methods in integer optimization*, Ann. Oper. Res., 139 (2005), pp. 163–193.

[10] A. Frangioni, C. Gentile, and F. Lacalandra, *Solving unit commitment problems with general ramp contraints*, Int. J. Elect. Power Energy Syst., 30 (2008), pp. 316–326.

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

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

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

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

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

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

[17] J.-B. Hiriart-Urruty and C. Lemaréchal, *Convex Analysis and Minimization Algorithms* Vols. I–II, Springer-Verlag, Berlin, 1993.

[18] 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*, Appl. Math. Optim., 11 (1984), pp. 43–53.

[19] K. C. Kiwiel, *Methods of Descent for Nondifferentiable Optimization*, Lecture Notes in Math. 1133, Springer-Verlag, Berlin, 1985.

[20] K. C. Kiwiel, *Proximity control in bundle methods for convex nondifferentiable minimization*, Math. Programming, 46 (1990), pp. 105–122.

[21] K. C. Kiwiel, *Efficiency of proximal bundle methods*, J. Optim. Theory Appl., 104 (2000), pp. 589–603.

[22] K. C. Kiwiel, *A proximal bundle method with approximate subgradient linearizations*, SIAM J. Optim., 16 (2006), pp. 1007–1023.

[23] K. C. Kiwiel, *A proximal-projection bundle method for Lagrangian relaxation, including semidefinite programming*, SIAM J. Optim., 17 (2006), pp. 1015–1034.

[24] K. C. Kiwiel and C. Lemaréchal, *An inexact bundle variant suited to column generation*, Math. Program., 118 (2009), pp. 177–206.

[25] C. Lemaréchal and R. Mifflin, eds., *Nonsmooth Optimization*, Proc. Ser. 3, Oxford, New York, 1978.

[26] C. Lemaréchal, A. Nemirovskii, and Y. Nesterov, *New variants of bundle methods*, Math. Programming, 69 (1995), pp. 111–147.

[27] L. Lukšăn and J. Vlček, *A bundle–Newton method for nonsmooth unconstrained minimization*, Math. Programming, 83 (1998), pp. 373–391.

[28] M. Mäkelä and P. Neittaanmäki, *Nonsmooth Optimization*, World Scientific, River Edge, NJ, 1992.

[29] O. L. Mangasarian, *Nonlinear Programming*, McGraw-Hill, New York, 1969.

[30] R. Mifflin and C. Sagastizábal, *On $\mathcal{VU}$-theory for functions with primal-dual gradient structure*, SIAM J. Optim., 11 (2000), pp. 547–571.

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

[32] R. Mifflin, D. Sun, and L. Qi, *Quasi-Newton bundle-type methods for nondifferentiable convex optimization*, SIAM J. Optim., 8 (1998), pp. 583–603.

[33] A. Ouorou, *A proximal cutting plane method using Chebychev center for nonsmooth convex optimization*, Math. Program., 119 (2009), pp. 239–271.

[34] D. Pallaschke and S. Rolewicz, *Foundations of Mathematical Optimization. Convex Analysis without Linearity*, Kluwer Academic Publishers, Dordrecht, The Netherlands, 1997.

[35] H. Schramm and J. Zowe, *A version of the bundle idea for minimizing a nonsmooth function: Conceptual idea, convergence analysis, numerical results*, SIAM J. Optim., 2 (1992), pp. 121–152.

[36] J. Vlček and Lukšăn, *Globally convergent variable metric method for nonconvex nondifferentiable unconstrained minimization*, J. Optim. Theory Appl., 111 (2001), pp. 407–430.