HAL open science

An alternating linearization bundle method for convex optimization and nonlinear multicommodity flow problems

Krzysztof C. Kiwiel

To cite this version:

Krzysztof C. Kiwiel. An alternating linearization bundle method for convex optimization and nonlinear multicommodity flow problems. [Research Report] RR-6420, INRIA. 2008. inria-00208074v2

HAL Id: inria-00208074

https://inria.hal.science/inria-00208074v2

Submitted on 21 Jan 2008

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

An alternating linearization bundle method for convex optimization and nonlinear multicommodity flow problems

Krzysztof C. Kiwiel

N° 6420

Janvier 2008

Thème NUM

Rapport de recherche

ISSN 0249-6399 ISRN INRIA/RR--6420--FR+ENG

An alternating linearization bundle method for convex optimization and nonlinear multicommodity flow problems

Krzysztof C. Kiwiel *

Thème NUM — Systèmes numériques
Projet Bipop

Rapport de recherche n° 6420 — Janvier 2008 —21 pages

Abstract: We give a bundle method for minimizing the sum of two convex functions, one of them being known only via an oracle of arbitrary accuracy. Each iteration involves solving two subproblems in which the functions are alternately represented by their linearizations. Our approach is motivated by applications to nonlinear multicommodity flow problems. Encouraging numerical experience on large scale problems is reported.

Key-words: Nondifferentiable optimization, convex programming, proximal bundle methods, approximate subgradients, network flow problem

* Systems Research Institute, Polish Academy of Sciences, Newelska 6, 01–447 Warsaw, Poland, kiwiel@ibspan.waw.pl

Unité de recherche INRIA Rhône-Alpes
655, avenue de l'Europe, 38334 Montbonnot Saint Ismier (France)
Téléphone : +33 4 76 61 52 00 — Télécopie +33 4 76 61 52 52

Un algorithm de faisceaux avec linéarisations alternées pour l'optimisation convexe et les multiflots non linéaires

Résumé : Nous donnons une méthode de faisceaux pour minimiser la somme de deux fonctions convexes, dont l'une n'est connue que par un oracle arbitrairement imprécis. Chaque itération considère deux sous-problèmes, dans lesquels les fonctions sont alternativement représentées par leur linéarisation. Notre approche est motivée par l'application au problème du multiflot non linéaire. Des expériences numériques sur des problèmes de grande taille se révèlent encourageantes.

Mots-clés : Optimisation non différentiable, optimisation convexe, méthode de faisceaux proximale, sous-gradients approchés, problèmes de flot dans un réseau

1 Introduction

We give a bundle method for the structured convex minimization problem

$$\theta_* := \inf\{\, \theta(\cdot) := \sigma(\cdot) + \pi(\cdot) \,\}, \tag{1.1}$$

where $\sigma : \mathbb{R}^m \to (-\infty, \infty]$ and $\pi : C \to \mathbb{R}$ are closed proper convex functions, and $C := \operatorname{dom}\sigma := \{u : \sigma(u) < \infty\}$ is the effective domain of σ. Such problems may appear via duality when the primal has a certain structure. For instance, consider the minimization problems

$$f_* := \inf\{\, f(Ax) : x \in X \,\} = \inf\{\, f(y) : y = Ax, x \in X \,\}, \tag{1.2}$$

where $X \subset \mathbb{R}^n$ and A is an $m \times n$ matrix. For the Lagrangian $L(x,y;u) := f(y) + \langle u, Ax - y\rangle$, minimization over $(x,y) \in X \times \mathbb{R}^m$ yields a dual problem of the form (1.1) with

$$\sigma(u) := f^*(u) := \sup_y\{\, \langle u,y\rangle - f(y) \,\} \quad \text{and} \quad \pi(u) := \sup\{\, \langle -A^T u, x\rangle : x \in X \,\}. \tag{1.3}$$

We assume that σ is "simple", i.e., minimizing σ plus a separable convex quadratic function is "easy". On the other hand, π is only known via an oracle, which at any $u \in C$ delivers an affine minorant of π (e.g., $\langle -Ax, \cdot\rangle$ for a possibly inexact maximizer x in (1.3)).

Our method is an approximate version of the proximal point algorithm [17, 20] which generates a sequence

$$\hat{u}^{k+1} = \arg\min \sigma(\cdot) + \pi(\cdot) + \tfrac{1}{2t_k}|\cdot - \hat{u}^k|^2 \quad \text{for } k = 1, 2, \ldots, \tag{1.4}$$

starting from a point $\hat{u}^1 \in C$, where $|\cdot|$ is the Euclidean norm and $t_k > 0$ are stepsizes. It combines two basic ideas: *bundling* from the proximal bundle methods [8], [6, Sect. XV.3] and their extensions [11, 12] to inexact oracles, and *alternating linearization* (AL for short) from [10, 12, 15]. Here bundling means replacing π in (1.4) by its polyhedral model $\check{\pi}_k \le \pi$ derived from the past oracle answers. Since the resulting subproblem may still be too difficult, we follow the AL approach in which a subproblem involving the sum of two functions (here σ and $\check{\pi}_k$) is replaced by two subproblems in which the functions are alternately represented by linear models. Thus, (1.4) is replaced by the two easier subproblems

$$\check{u}^{k+1} := \arg\min \bar{\sigma}_{k-1}(\cdot) + \check{\pi}_k(\cdot) + \tfrac{1}{2t_k}|\cdot - \hat{u}^k|^2, \tag{1.5}$$

$$u^{k+1} := \arg\min \sigma(\cdot) + \bar{\pi}_k(\cdot) + \tfrac{1}{2t_k}|\cdot - \hat{u}^k|^2. \tag{1.6}$$

The first subproblem (1.5) employs a linearization $\bar{\sigma}_{k-1} \le \sigma$ obtained at the previous iteration. Its solution yields by the usual optimality condition a linearization $\bar{\pi}_k \le \check{\pi}_k$ which may a posteriori replace $\check{\pi}_k$ in (1.5) without changing its optimal value and solution. Similarly, the solution of (1.6) provides a linearization $\bar{\sigma}_k \le \sigma$ which may replace σ in (1.6).

Our method coincides with that of [12] in the special case of σ being the indicator function i_C of C ($i_C(u) = 0$ if $u \in C$, ∞ otherwise). Then u^{k+1} in (1.6) is the projection onto C of $\hat{u}^k - t_k \nabla \bar{\pi}_k$; this projection is straightforward if the set C is "simple". For more difficult cases, it is crucial to allow for approximate solutions in (1.6). We show (cf. Sect. 4.2) that such solutions can be obtained by solving the Fenchel dual of (1.6) approximately; this is conceptually related to the use of Fenchel's duality in [6, Prop. XV.2.4.3 and p. 306].

For dual applications, we restrict our attention to the setup of (1.2)–(1.3) with f closed proper convex and X compact and convex (since other examples of [15] could be treated in similar ways). As in [12], even when the dual has no solutions, our method can still asymptotically find ε_π-optimal

primal solutions, where ε_π is an upper bound on the oracle's errors; in fact only the asymptotic oracle errors matter, as discussed in [12, Sect. 4.2].

Actually, our theoretical contributions outlined above were motivated by applications to nonlinear multicommodity flow problems (NMFP for short); more concretely, by the good experimental results of [1], where the analytic center cutting plane method (ACCPM for short) exploited "nice" second-order properties of σ in (1.1). We show that our method can exploit such properties as well, obtaining significant speedups on most instances used in [1].

As for the state-of-the-art in NMFP, we refer the reader to [1] for the developments subsequent to the review of [18], adding the more recent references of [13, 16].

The paper is organized as follows. In Sect. 2 we present our method for general models of π. Its convergence is analyzed in Sect. 3. Useful modifications, including approximate solutions of (1.6), are given in Sect. 4. Application to the Lagrangian relaxation of (1.2) is studied in Sect. 5. Specializations to NMFP are given in Sect. 6. Implementation issues are discussed in Sect. 7. Finally, numerical comparisons with ACCPM are given in Sect. 8.

2 The alternating linearization bundle method

We first explain our use of approximate objective values in (1.5), (1.6). Our method generates a sequence of *trial points* $\{u^k\}_{k=1}^\infty \subset C$ at which the oracle is called. We assume that for a fixed *accuracy tolerance* $\varepsilon_\pi \geq 0$, at each $u^k \in C$ the oracle delivers an *approximate value* π_u^k and an *approximate subgradient* g_π^k of π that produce the *approximate linearization* of π:

$$\pi_k(\cdot) := \pi_u^k + \langle g_\pi^k, \cdot - u^k \rangle \leq \pi(\cdot) \quad \text{with} \quad \pi_k(u^k) = \pi_u^k \geq \pi(u^k) - \varepsilon_\pi. \tag{2.1}$$

Thus $\pi_u^k \in [\pi(u^k) - \varepsilon_\pi, \pi(u^k)]$, whereas g_π^k lies in the ε_π-subdifferential of π at u^k

$$\partial_{\varepsilon_\pi} \pi(u^k) := \left\{ g_\pi : \pi(\cdot) \geq \pi(u^k) - \varepsilon_\pi + \langle g_\pi, \cdot - u^k \rangle \right\}.$$

Then $\theta_u^k := \sigma_u^k + \pi_u^k$ is the approximate value of θ at u^k, where $\sigma_u^k := \sigma(u^k)$.

At iteration $k \geq 1$, the current *prox* (or *stability*) *center* $\hat{u}^k := u^{k(l)} \in C$ for some $k(l) \leq k$ has the value $\theta_{\hat{u}}^k := \theta_u^{k(l)}$ (usually $\theta_{\hat{u}}^k = \min_{j=1}^k \theta_u^j$); note that, by (2.1),

$$\theta_{\hat{u}}^k \in [\theta(\hat{u}^k) - \varepsilon_\pi, \theta(\hat{u}^k)]. \tag{2.2}$$

If $\pi_{\hat{u}}^k < \bar{\pi}_k(\hat{u}^k)$ in (1.6) due to evaluation errors, the *predicted descent*

$$v_k := \theta_{\hat{u}}^k - \left[\sigma(u^{k+1}) + \bar{\pi}_k(u^{k+1})\right] \tag{2.3}$$

may be nonpositive; hence, if necessary, t_k is increased and (1.5)–(1.6) are solved again until $v_k \geq |u^{k+1} - \hat{u}^k|^2/2t_k$ as in [11, 12, 14]. A *descent* step to $\hat{u}^{k+1} := u^{k+1}$ is taken if

$$\theta_u^{k+1} \leq \theta_{\hat{u}}^k - \kappa v_k \tag{2.4}$$

for a fixed $\kappa \in (0,1)$. Otherwise, a *null* step $\hat{u}^{k+1} := \hat{u}^k$ occurs; then $\bar{\pi}_k$ and the new linearization π_{k+1} are used to produce a better model $\check{\pi}_{k+1} \geq \max\{\bar{\pi}_k, \pi_{k+1}\}$.

Specific rules of our method will be discussed after its formal statement below.

Algorithm 2.1

Step 0 (*Initiation*). Select $u^1 \in C$, a *descent parameter* $\kappa \in (0,1)$, a *stepsize bound* $t_{\min} > 0$ and a *stepsize* $t_1 \geq t_{\min}$. Call the oracle at u^1 to obtain π_u^1 and g_π^1 of (2.1). Set $\bar{\pi}_0 := \pi_1$ by (2.1), and $\bar{\sigma}_0(\cdot) := \sigma(u^1) + \langle p_\sigma^0, \cdot - u^1\rangle$ with $p_\sigma^0 \in \partial\sigma(u^1)$. Set $\hat{u}^1 := u^1$, $\theta_{\hat{u}}^1 := \theta_u^1 := \sigma_u^1 + \pi_u^1$ with $\sigma_u^1 := \sigma(u^1)$, $i_t^1 := 0$, $k := k(0) := 1$, $l := 0$ ($k(l)-1$ will denote the iteration of the lth descent step).

Step 1 (*Model selection*). Choose $\check{\pi}_k : \mathbb{R}^m \to \mathbb{R}$ convex and such that

$$\max\{\bar{\pi}_{k-1}, \pi_k\} \leq \check{\pi}_k \leq \pi. \tag{2.5}$$

Step 2 (*Solving the π–subproblem*). Find $\check{u}^{k+1}$ of (1.5) and the *aggregate linearization* of $\check{\pi}_k$

$$\bar{\pi}_k(\cdot) := \check{\pi}_k(\check{u}^{k+1}) + \langle p_\pi^k, \cdot - \check{u}^{k+1}\rangle \quad \text{with} \quad p_\pi^k := (\hat{u}^k - \check{u}^{k+1})/t_k - p_\sigma^{k-1}. \tag{2.6}$$

Step 3 (*Solving the σ–subproblem*). Find u^{k+1} of (1.6) and the *aggregate linearization* of σ

$$\bar{\sigma}_k(\cdot) := \sigma(u^{k+1}) + \langle p_\sigma^k, \cdot - u^{k+1}\rangle \quad \text{with} \quad p_\sigma^k := (\hat{u}^k - u^{k+1})/t_k - p_\pi^k. \tag{2.7}$$

Compute v_k of (2.3), and the *aggregate subgradient* and *linearization error* of θ

$$p^k := (\hat{u}^k - u^{k+1})/t_k \quad \text{and} \quad \varepsilon_k := v_k - t_k|p^k|^2. \tag{2.8}$$

Step 4 (*Stopping criterion*). If $\max\{|p^k|, \varepsilon_k\} = 0$, stop ($\theta_{\hat{u}}^k \leq \theta_*$).

Step 5 (*Noise attenuation*). If $v_k < -\varepsilon_k$, set $t_k := 10t_k$, $i_t^k := k$ and go back to Step 2.

Step 6 (*Oracle call*). Call the oracle at u^{k+1} to obtain π_u^{k+1} and g_π^{k+1} of (2.1).

Step 7 (*Descent test*). If the descent test (2.4) holds with $\theta_u^{k+1} := \sigma(u^{k+1}) + \pi_u^{k+1}$, set $\hat{u}^{k+1} := u^{k+1}$, $\theta_{\hat{u}}^{k+1} := \theta_u^{k+1}$, $i_t^{k+1} := 0$, $k(l+1) := k+1$ and increase l by 1 (*descent step*); otherwise, set $\hat{u}^{k+1} := \hat{u}^k$, $\theta_{\hat{u}}^{k+1} := \theta_{\hat{u}}^k$, and $i_t^{k+1} := i_t^k$ (*null step*).

Step 8 (*Stepsize updating*). If $k(l) = k+1$ (i.e., after a descent step), select $t_{k+1} \geq t_{\min}$; otherwise, either set $t_{k+1} := t_k$, or choose $t_{k+1} \in [t_{\min}, t_k]$ if $i_t^{k+1} = 0$.

Step 9 (*Loop*). Increase k by 1 and go to Step 1.

Several comments on the method are in order. Step 1 may choose the simplest model $\check{\pi}_k = \max\{\bar{\pi}_{k-1}, \pi_k\}$. More efficient choices are discussed in [12, Sect. 4.4] and [14, Sect. 2.3]. For polyhedral models, Step 2 may use the QP methods of [3, 7, 9], which can handle efficiently sequences of subproblems (1.5).

We now use the relations of Steps 2 and 3 to derive an optimality estimate, which involves the *aggregate linearization* $\bar{\theta}_k := \bar{\sigma}_k + \bar{\pi}_k$ and the *optimality measure*

$$V_k := \max\{|p^k|, \varepsilon_k + \langle p^k, \hat{u}^k\rangle\}. \tag{2.9}$$

Lemma 2.2 (1) *The vectors p_π^k and p_σ^k defined in* (2.6) *and* (2.7) *are in fact subgradients*:

$$p_\pi^k \in \partial\check{\pi}_k(\check{u}^{k+1}) \quad \textit{and} \quad p_\sigma^k \in \partial\sigma(u^{k+1}), \tag{2.10}$$

and the linearizations $\bar{\pi}_k$ and $\bar{\sigma}_k$ defined in (2.6) *and* (2.7) *provide the minorizations*

$$\bar{\pi}_k \leq \check{\pi}_k, \quad \bar{\sigma}_k \leq \sigma \quad \textit{and} \quad \bar{\theta}_k := \bar{\pi}_k + \bar{\sigma}_k \leq \theta. \tag{2.11}$$

(2) *The aggregate subgradient* p^k *defined in* (2.8) *and the linearization* $\bar{\theta}_k$ *above satisfy*

$$p^k = p^k_\pi + p^k_\sigma = (\hat{u}^k - u^{k+1})/t_k, \tag{2.12}$$

$$\bar{\theta}_k(\cdot) = \bar{\theta}_k(u^{k+1}) + \langle p^k, \cdot - u^{k+1}\rangle. \tag{2.13}$$

(3) *The predicted descent* v_k *of* (2.3) *and the aggregate linearization error* ε_k *of* (2.8) *satisfy*

$$v_k = \theta^k_{\hat{u}} - \bar{\theta}_k(u^{k+1}) = t_k|p^k|^2 + \varepsilon_k \quad \textit{and} \quad \varepsilon_k = \theta^k_{\hat{u}} - \bar{\theta}_k(\hat{u}^k). \tag{2.14}$$

(4) *The aggregate linearization* $\bar{\theta}_k$ *is expressed in terms of* p^k *and* ε_k *as follows*:

$$\theta^k_{\hat{u}} - \varepsilon_k + \langle p^k, \cdot - \hat{u}^k\rangle = \bar{\theta}_k(\cdot) \le \theta(\cdot). \tag{2.15}$$

(5) *The optimality measure* V_k *of* (2.9) *satisfies* $V_k \le \max\{|p^k|, \varepsilon_k\}(1+|\hat{u}^k|)$ *and*

$$\theta^k_{\hat{u}} \le \theta(u) + V_k\big(1+|u|\big) \quad \textit{for all } u. \tag{2.16}$$

(6) *We have* $v_k \ge -\varepsilon_k \Leftrightarrow t_k|p^k|^2/2 \ge -\varepsilon_k \Leftrightarrow v_k \ge t_k|p^k|^2/2$. *Moreover,* $v_k \ge \varepsilon_k$, $-\varepsilon_k \le \varepsilon_\pi$ *and*

$$v_k \ge \max\big\{t_k|p^k|^2/2, |\varepsilon_k|\big\} \qquad \textit{if} \quad v_k \ge -\varepsilon_k, \tag{2.17}$$

$$V_k \le \max\big\{(2v_k/t_k)^{1/2}, v_k\big\}\big(1+|\hat{u}^k|\big) \quad \textit{if} \quad v_k \ge -\varepsilon_k, \tag{2.18}$$

$$V_k < (2\varepsilon_\pi/t_k)^{1/2}\big(1+|\hat{u}^k|\big) \qquad \textit{if} \quad v_k < -\varepsilon_k. \tag{2.19}$$

Proof. (1) Let ϕ^k_π and ϕ^k_σ denote the objectives of (1.5) and (1.6). By (2.6), the optimality condition $0 \in \partial\phi^k_\pi(\check{u}^{k+1})$ for (1.5) with $\nabla\bar{\sigma}_{k-1} = p^{k-1}_\sigma$ by Step 0 and (2.7), i.e.,

$$0 \in \partial\phi^k_\pi(\check{u}^{k+1}) = \partial\check{\pi}_k(\check{u}^{k+1}) + p^{k-1}_\sigma + (\check{u}^{k+1} - \hat{u}^k)/t_k = \partial\check{\pi}_k(\check{u}^{k+1}) - p^k_\pi,$$

and the equality $\bar{\pi}_k(\check{u}^{k+1}) = \check{\pi}_k(\check{u}^{k+1})$ yield $p^k_\pi \in \partial\check{\pi}_k(\check{u}^{k+1})$ and $\bar{\pi}_k \le \check{\pi}_k$. Similarly, by (2.7),

$$0 \in \partial\phi^k_\sigma(u^{k+1}) = p^k_\pi + \partial\sigma(u^{k+1}) + (u^{k+1} - \hat{u}^k)/t_k = \partial\sigma(u^{k+1}) - p^k_\sigma$$

(using $\nabla\bar{\pi}_k = p^k_\pi$) and $\bar{\sigma}_k(u^{k+1}) = \sigma(u^{k+1})$ give $p^k_\sigma \in \partial\sigma(u^{k+1})$ and $\bar{\sigma}_k \le \sigma$. Combining both minorizations, we obtain that $\bar{\pi}_k + \bar{\sigma}_k \le \check{\pi}_k + \sigma \le \theta$ by (2.5) and (1.1).

(2) Use the linearity of $\bar{\theta}_k := \bar{\pi}_k + \bar{\sigma}_k$, (2.6), (2.7) and (2.8).

(3) Rewrite (2.3), using the fact that $\bar{\theta}_k(\hat{u}^k) = \bar{\theta}_k(u^{k+1}) + t_k|p^k|^2$ by (2).

(4) We have $\theta^k_{\hat{u}} - \varepsilon_k = \bar{\theta}_k(\hat{u}^k)$ by (3), $\bar{\theta}_k$ is affine by (2) and minorizes θ by (1).

(5) Use the Cauchy–Schwarz inequality in the definition (2.9) and in (4).

(6) The equivalences follow from the expression of $v_k = t_k|p^k|^2 + \varepsilon_k$ in (3); in particular, $v_k \ge \varepsilon_k$. Next, by (2.14), (2.11) and (2.2), we have

$$-\varepsilon_k = \bar{\theta}_k(\hat{u}^k) - \theta^k_{\hat{u}} \le \theta(\hat{u}^k) - \theta^k_{\hat{u}} \le \varepsilon_\pi.$$

Finally, to obtain the bounds (2.17)–(2.19), use the equivalences together with the facts that $v_k \ge \varepsilon_k$, $-\varepsilon_k \le \varepsilon_\pi$ and the bound on V_k from assertion (5). □

The optimality estimate (2.16) justifies the stopping criterion of Step 4: $V_k = 0$ yields $\theta^k_{\hat{u}} \le \inf\theta = \theta_*$; thus, the point $\hat{u}^k$ is ε_π-*optimal*, i.e., $\theta(\hat{u}^k) \le \theta_* + \varepsilon_\pi$ by (2.2). If the oracle is exact ($\varepsilon_\pi = 0$), we have $v_k \ge \varepsilon_k \ge 0$ by Lemma 2.2(6), and Step 5 is redundant. When inexactness is discovered at Step 5 via $v_k < -\varepsilon_k$ and the stepsize t_k is increased, the *stepsize indicator* $i^k_t \ne 0$ prevents Step 7 from decreasing t_k after null steps until the next descent step occurs (cf. Step 6). At Step 6, we have $u^{k+1} \in C$ and $v_k > 0$ (by (2.17), since $\max\{|p^k|, \varepsilon_k\} > 0$ at Step 4), so that $\hat{u}^{k+1} \in C$ and $\theta^{k+1}_{\hat{u}} \le \theta^k_{\hat{u}}$ for all k.

3 Convergence

With Lemma 2.2 replacing [12, Lem. 2.2], it is easy to check that the convergence results of [12, Sect. 3] will hold once we prove [12, Lem. 3.2] for our method. To this end, as usual in bundle methods, we assume that the oracle's subgradients are *locally bounded*:

$$\{g_\pi^k\} \text{ is bounded if } \{u^k\} \text{ is bounded.} \tag{3.1}$$

Further, as in [12], we assume that the model subgradients $p_\pi^k \in \partial\check{\pi}_k(\check{u}^{k+1})$ in (2.10) satisfy

$$\{p_\pi^k\} \text{ is bounded if } \{u^k\} \text{ is bounded.} \tag{3.2}$$

Remark 3.1 Note that (3.1) holds if $C = \mathbb{R}^m$ or if π can be extended to become finite-valued on a neighborhood of C, since $g_\pi^k \in \partial_{\varepsilon_\pi}\pi(u^k)$ by (2.1), whereas the mapping $\partial_{\varepsilon_\pi}\pi$ is locally bounded on C in both cases [6, Sect. XI.4.1]. As discussed in [12, Rem. 4.4], typical models $\check{\pi}_k$ satisfy condition (3.2) automatically when (3.1) holds.

A suitable modification of the proof of [12, Lem. 3.2] follows.

Lemma 3.2 *Suppose there exists $\bar{k}$ such that for all $k \geq \bar{k}$, only null steps occur and Step* 5 *doesn't increase t_k. Then $V_k \to 0$.*

Proof. Let ϕ_π^k and ϕ_σ^k denote the objectives of subproblems (1.5) and (1.6). First, using partial linearizations of these subproblems, we show that their optimal values $\phi_\pi^k(\check{u}^{k+1}) \leq \phi_\sigma^k(u^{k+1})$ are nondecreasing and bounded above for $k \geq \bar{k}$.

Fix $k \geq \bar{k}$. By the definitions in (1.5) and (2.6), we have $\bar{\pi}_k(\check{u}^{k+1}) = \check{\pi}_k(\check{u}^{k+1})$ and

$$\check{u}^{k+1} = \arg\min\{\, \bar{\phi}_\pi^k(\cdot) := \bar{\pi}_k(\cdot) + \bar{\sigma}_{k-1}(\cdot) + \tfrac{1}{2t_k}|\cdot - \hat{u}^k|^2 \,\} \tag{3.3}$$

from $\nabla\bar{\phi}_\pi^k(\check{u}^{k+1}) = 0$. Since $\bar{\phi}_\pi^k$ is quadratic and $\bar{\phi}_\pi^k(\check{u}^{k+1}) = \phi_\pi^k(\check{u}^{k+1})$, by Taylor's expansion

$$\bar{\phi}_\pi^k(\cdot) = \phi_\pi^k(\check{u}^{k+1}) + \tfrac{1}{2t_k}|\cdot - \check{u}^{k+1}|^2. \tag{3.4}$$

Similarly, by the definitions in (1.6) and (2.7), we have $\bar{\sigma}_k(u^{k+1}) = \sigma(u^{k+1})$,

$$u^{k+1} = \arg\min\{\, \bar{\phi}_\sigma^k(\cdot) := \bar{\pi}_k(\cdot) + \bar{\sigma}_k(\cdot) + \tfrac{1}{2t_k}|\cdot - \hat{u}^k|^2 \,\}, \tag{3.5}$$

$$\bar{\phi}_\sigma^k(\cdot) = \phi_\sigma^k(u^{k+1}) + \tfrac{1}{2t_k}|\cdot - u^{k+1}|^2. \tag{3.6}$$

Next, to bound the objective values of the linearized subproblems (3.3) and (3.5) from above, we use the minorizations $\bar{\pi}_k \leq \pi$ and $\bar{\sigma}_{k-1}, \bar{\sigma}_k \leq \sigma$ of (2.11) for $\theta := \pi + \sigma$:

$$\phi_\pi^k(\check{u}^{k+1}) + \tfrac{1}{2t_k}|\check{u}^{k+1} - \hat{u}^k|^2 = \bar{\phi}_\pi^k(\hat{u}^k) \leq \theta(\hat{u}^k), \tag{3.7a}$$

$$\phi_\sigma^k(u^{k+1}) + \tfrac{1}{2t_k}|u^{k+1} - \hat{u}^k|^2 = \bar{\phi}_\sigma^k(\hat{u}^k) \leq \theta(\hat{u}^k), \tag{3.7b}$$

where the equalities stem from (3.4) and (3.6). Due to the minorization $\bar{\sigma}_{k-1} \leq \sigma$, the objectives of subproblems (3.3) and (1.6) satisfy $\bar{\phi}_\pi^k \leq \phi_\sigma^k$. On the other hand, since $\hat{u}^{k+1} = \hat{u}^k$, $t_{k+1} \leq t_k$ (cf. Step

7), and $\bar{\pi}_k \le \check{\pi}_{k+1}$ by (2.5), the objectives of (3.5) and the next subproblem (1.5) satisfy $\bar{\phi}_\sigma^k \le \phi_\pi^{k+1}$. Altogether, by (3.4) and (3.6), we see that

$$\phi_\pi^k(\check{u}^{k+1}) + \tfrac{1}{2t_k}|u^{k+1} - \check{u}^{k+1}|^2 = \bar{\phi}_\pi^k(u^{k+1}) \le \phi_\sigma^k(u^{k+1}), \tag{3.8a}$$

$$\phi_\sigma^k(u^{k+1}) + \tfrac{1}{2t_k}|\check{u}^{k+2} - u^{k+1}|^2 = \bar{\phi}_\sigma^k(\check{u}^{k+2}) \le \phi_\pi^{k+1}(\check{u}^{k+2}). \tag{3.8b}$$

In particular, the inequalities $\phi_\pi^k(\check{u}^{k+1}) \le \phi_\sigma^k(u^{k+1}) \le \phi_\pi^{k+1}(\check{u}^{k+2})$ imply that the nondecreasing sequences $\{\phi_\pi^k(\check{u}^{k+1})\}_{k\ge\bar{k}}$ and $\{\phi_\sigma^k(u^{k+1})\}_{k\ge\bar{k}}$, which are bounded above by (3.7) with $\hat{u}^k = \hat{u}^{\bar{k}}$ for all $k \ge \bar{k}$, must have a common limit, say $\phi_\infty \le \theta(\hat{u}^{\bar{k}})$. Moreover, since the stepsizes satisfy $t_k \le t_{\bar{k}}$ for all $k \ge \bar{k}$, we deduce from the bounds (3.7)–(3.8) that

$$\phi_\pi^k(\check{u}^{k+1}), \phi_\sigma^k(u^{k+1}) \uparrow \phi_\infty, \quad \check{u}^{k+2} - u^{k+1} \to 0, \tag{3.9}$$

and the sequences $\{\check{u}^{k+1}\}$ and $\{u^{k+1}\}$ are bounded. Then the sequences $\{g_\pi^k\}$ and $\{p_\pi^k\}$ are bounded by (3.1) and (3.2).

We now show that the *approximation error* $\bar{\varepsilon}_k := \pi_u^{k+1} - \bar{\pi}_k(u^{k+1})$ vanishes. Using the form (2.1) of π_{k+1}, the minorization $\pi_{k+1} \le \check{\pi}_{k+1}$ of (2.5), the Cauchy–Schwarz inequality, and the optimal values of subproblems (1.5) and (1.6) with $\hat{u}^k = \hat{u}^{\bar{k}}$ for $k \ge \bar{k}$, we estimate

$$\begin{aligned}
\bar{\varepsilon}_k &:= \pi_u^{k+1} - \bar{\pi}_k(u^{k+1}) = \pi_{k+1}(\check{u}^{k+2}) - \bar{\pi}_k(u^{k+1}) + \langle g_\pi^{k+1}, u^{k+1} - \check{u}^{k+2}\rangle \\
&\le \check{\pi}_{k+1}(\check{u}^{k+2}) - \bar{\pi}_k(u^{k+1}) + |g_\pi^{k+1}||u^{k+1} - \check{u}^{k+2}| \\
&= \phi_\pi^{k+1}(\check{u}^{k+2}) - \phi_\sigma^k(u^{k+1}) + \Delta_u^k + \Delta_\sigma^k + |g_\pi^{k+1}||u^{k+1} - \check{u}^{k+2}|,
\end{aligned} \tag{3.10}$$

where $\Delta_u^k := |u^{k+1} - \hat{u}^{\bar{k}}|^2/2t_k - |\check{u}^{k+2} - \hat{u}^{\bar{k}}|^2/2t_{k+1}$ and $\Delta_\sigma^k := \sigma_u^{k+1} - \bar{\sigma}_k(\check{u}^{k+2})$; in fact, we have $\Delta_\sigma^k = -\langle p_\sigma^k, \check{u}^{k+2} - u^{k+1}\rangle$ by (2.7). To see that $\Delta_u^k \to 0$, note that

$$|\check{u}^{k+2} - \hat{u}^{\bar{k}}|^2 = |u^{k+1} - \hat{u}^{\bar{k}}|^2 + 2\langle \check{u}^{k+2} - u^{k+1}, u^{k+1} - \hat{u}^{\bar{k}}\rangle + |\check{u}^{k+2} - u^{k+1}|^2,$$

$|u^{k+1} - \hat{u}^{\bar{k}}|^2$ is bounded, $\check{u}^{k+2} - u^{k+1} \to 0$ by (3.9), and $t_{\min} \le t_{k+1} \le t_k$ for $k \ge \bar{k}$ by Step 7. These properties also give $\Delta_\sigma^k \to 0$, since by (2.7) and the Cauchy–Schwarz inequality,

$$|\Delta_\sigma^k| \le |p_\sigma^k||\check{u}^{k+2} - u^{k+1}| \quad \text{with} \quad |p_\sigma^k| \le |u^{k+1} - \hat{u}^{\bar{k}}|/t_k + |p_\pi^k|,$$

where $\{p_\pi^k\}$ is bounded. Hence, using (3.9) and the boundedness of $\{g_\pi^{k+1}\}$ in (3.10) yields $\overline{\lim}_k\, \bar{\varepsilon}_k \le 0$. On the other hand, $\bar{\varepsilon}_k = \theta_u^{k+1} - \bar{\theta}_k(u^{k+1})$ from $\bar{\sigma}_k(u^{k+1}) = \sigma_u^{k+1}$ in (2.7), while for $k \ge \bar{k}$ the null step condition $\theta_u^{k+1} > \theta_{\hat{u}}^k - \kappa v_k$ gives

$$\bar{\varepsilon}_k = \left[\theta_u^{k+1} - \theta_{\hat{u}}^k\right] + \left[\theta_{\hat{u}}^k - \bar{\theta}_k(u^{k+1})\right] > -\kappa v_k + v_k = (1-\kappa)v_k \ge 0$$

by (2.14), where $\kappa < 1$ by Step 0; we conclude that $\bar{\varepsilon}_k \to 0$ and $v_k \to 0$. Finally, since $v_k \to 0$, $t_k \ge t_{\min}$ (cf. Step 7) and $\hat{u}^k = \hat{u}^{\bar{k}}$ for $k \ge \bar{k}$, we have $V_k \to 0$ by (2.18). □

We may now state our principal result on the *asymptotic objective value* $\theta_{\hat{u}}^\infty := \lim_k \theta_{\hat{u}}^k$

Theorem 3.3 (1) *We have* $\theta_{\hat{u}}^k \downarrow \theta_{\hat{u}}^\infty \le \theta_*$, *and additionally* $\underline{\lim}_k V_k = 0$ *if* $\theta_* > -\infty$.
(2) $\theta_* \le \underline{\lim}_k\, \theta(\hat{u}^k) \le \overline{\lim}_k\, \theta(\hat{u}^k) \le \theta_{\hat{u}}^\infty + \varepsilon_\pi$.

Proof. Use the proof of [12, Thm. 3.5], with obvious modifications. □

4 Modifications

4.1 Looping between subproblems

To obtain a more accurate solution to the prox subproblem (1.4) with π replaced by $\check{\pi}_k$, we may cycle between subproblems (1.5) and (1.6), updating their data as if null steps occured without changing the model $\check{\pi}_k$. Specifically, for a given *subproblem accuracy threshold* $\check{\kappa} \in (0,1)$, suppose that the following step is inserted after Step 5.

Step 5′ (*Subproblem accuracy test*). If

$$\sigma(u^{k+1}) + \check{\pi}_k(u^{k+1}) > \theta^k_{\hat{u}} - \check{\kappa} v_k, \tag{4.1}$$

set $\bar{\sigma}_{k-1}(\cdot) := \bar{\sigma}_k(\cdot)$, $p^{k-1}_\sigma := p^k_\sigma$ and go back to Step 2.

The main aim of this modification is to avoid "unnecessary" null steps. Namely, if the test (4.1) holds with $\check{\kappa} \le \kappa$ and the oracle is exact enough to deliver $\pi^{k+1}_u \ge \check{\pi}_k(u^{k+1})$, then the descent test (2.4) can't hold and a null step must occur, which is bypassed by Step 5′.

When the oracle is expensive, the optional use of Step 5′ with $\check{\kappa} \in [\kappa, 1)$ gives room for deciding whether to continue working with the current model $\check{\pi}_k$ before calling the oracle.

Convergence for this modification can be analyzed as in [12, Rem. 4.1]. Omitting details for brevity, here we just observe that for the test (4.1) written as (cf. (2.14))

$$\check{\varepsilon}_k := \check{\pi}_k(u^{k+1}) - \bar{\pi}_k(u^{k+1}) > (1-\check{\kappa}) v_k,$$

the $\check{\varepsilon}_k$ above may play the role of $\bar{\varepsilon}_k$ in (3.10).

4.2 Solving the σ-subproblem approximately

For a given tolerance $\kappa_N \in (0, 1-\kappa)$, suppose Step 3 is replaced by the following.

Step 3′ (*Solving the σ–subproblem approximately*). Find a linearization $\bar{\sigma}_k \le \sigma$ such that

$$\phi^k_\pi(\check{u}^{k+1}) \le \bar{\phi}^k_\sigma(u^{k+1}), \tag{4.2}$$

$$\sigma(u^{k+1}) - \bar{\sigma}_k(u^{k+1}) \le \kappa_N v_k, \tag{4.3}$$

for u^{k+1} given by (3.5) and v_k by (2.14). Set p^k and ε_k by (2.8), and $p^k_\sigma := \nabla \bar{\sigma}_k$.

Before discussing implementations, we show that Step 3′ does not spoil convergence. In Sect. 2, $\bar{\sigma}_k(u^{k+1})$ replaces $\sigma(u^{k+1})$ in (2.3), (2.7) and (2.10). In Sect. 3, it suffices to validate Lemma 3.2.

Lemma 4.1 *Lemma 3.2 still holds for Step 3 replaced by Step 3′ above.*

Proof. We only sketch how to modify the proof of Lemma 3.2. First, referring to (3.5) instead of (1.6), replace ϕ^k_σ by $\bar{\phi}^k_\sigma$ throughout, and (3.8a) by (4.2). Second, let $\Delta^k_\sigma := \bar{\sigma}_k(u^{k+1}) - \bar{\sigma}_k(\check{u}^{k+2})$ in (3.10). Third, by (4.3), the null step condition yields

$$\bar{\sigma}_k(u^{k+1}) + \pi^{k+1}_u > \theta^k_{\hat{u}} - \kappa v_k + \bar{\sigma}_k(u^{k+1}) - \sigma(u^{k+1}) \ge \theta^k_{\hat{u}} - \tilde{\kappa} v_k$$

for $\tilde{\kappa} := \kappa + \kappa_N < 1$, and hence

$$\bar{\varepsilon}_k = \bar{\sigma}_k(u^{k+1}) + \pi^{k+1}_u - \bar{\theta}_k(u^{k+1}) > (1-\tilde{\kappa}) v_k \ge 0,$$

so that the proof may finish as before. □

Step 3′ can be implemented by solving the Fenchel dual of (1.6) approximately. Indeed, using the representation $\sigma(\cdot) = \sup_z\{\langle z,\cdot\rangle - \sigma^*(z)\}$ in (1.6), consider the Lagrangian

$$L(u,z) := \langle z,u\rangle - \sigma^*(z) + \bar{\pi}_k(u) + \tfrac{1}{2t_k}|u-\hat{u}^k|^2, \tag{4.4}$$

and associate with each dual point $z \in \operatorname{dom}\sigma^*$ the following quantities:

$$\bar{u}(z) := \arg\min_u L(u,z) = \hat{u}^k - t_k\big(p^k_\pi + z\big), \tag{4.5}$$

$$\bar{\sigma}(\cdot;z) := \langle z,\cdot\rangle - \sigma^*(z), \tag{4.6}$$

$$\varepsilon(z) := \sigma(\bar{u}(z)) - \bar{\sigma}(\bar{u}(z);z) = \sigma(\bar{u}(z)) + \sigma^*(z) - \langle z,\bar{u}(z)\rangle, \tag{4.7}$$

$$v(z) := \theta^k_u - \big[\bar{\pi}_k(\bar{u}(z)) + \bar{\sigma}(\bar{u}(z);z)\big], \tag{4.8}$$

where $\bar{u}(z)$ is the Lagrangian solution (with $p^k_\pi = \nabla\bar{\pi}_k$), $\bar{\sigma}(\cdot;z)$ is the linearization of σ, $\varepsilon(z)$ is its linearization error at $\bar{u}(z)$, and $v(z)$ is the predicted descent. Maximizing $L(\bar{u}(z),z)$ or equivalently minimizing $w(z) = -L(\bar{u}(z),z)$ leads to the following dual problem:

$$w_* := \min_z\big\{\, w(z) := \sigma^*(z) + \tfrac{t_k}{2}|p^k_\pi + z|^2 - \langle z,\hat{u}^k\rangle - \bar{\pi}_k(\hat{u}^k)\,\big\}, \tag{4.9}$$

with a unique solution z^* giving $u^* := \bar{u}(z^*)$ such that $u^* \in \partial\sigma^*(z^*)$, $z^* \in \partial\sigma(u^*)$ and

$$\sigma(u^*) + \sigma^*(z^*) - \langle z^*,u^*\rangle = 0; \tag{4.10}$$

not suprisingly, u^* is the exact solution of (1.6) and z^* is the corresponding p^k_σ in (2.7). Note that (4.9) can be restricted to the set $D := \operatorname{dom}\partial\sigma^* := \{z : \partial\sigma^*(z) \neq \emptyset\}$, which contains z^*.

Now, suppose that we have a method for solving (4.9) with the following properties:

(1) It starts from the point $z^1 := p^{k-1}_\sigma \in D$ such that $\sigma_{k-1}(\cdot) = \langle z^1,\cdot\rangle - \sigma^*(z^1)$; thus, by (3.3), (3.4) and (4.4)–(4.6), the initial $w(z^1) = -\phi^k_\pi(\check{u}^{k+1})$ from $w(z^1) = -L(\bar{u}(z^1),z^1)$.

(2) It generates points $z^i \in D$ with $w(z^i) \le w(z^1)$ such that $z^i \to z^*$, $\sigma^*(z^i) \to \sigma^*(z^*)$ and $\sigma(\bar{u}(z^i)) \to \sigma(u^*)$, where $\bar{u}(z^i) \to u^*$ by (4.5).

Then $\varepsilon(z^i) \to 0$ by (4.7) and (4.10), whereas $v(z^i) \to v(z^*)$ by (4.8). Thus, if $v(z^*) > 0$, we will eventually have $\varepsilon(z^i) \le \kappa_N v(z^i)$. Then the method may stop with $u^{k+1} := \bar{u}(z^i)$, $v_k := v(z^i)$, $\bar{\sigma}_k(\cdot) := \bar{\sigma}(\cdot;z^i)$ and $p^k_\sigma := z^i$ to meet the requirements of Step 3′, with (4.2) following from $-\phi^k_\sigma(u^{k+1}) = w(z^i) \le w(z^1) = -\phi^k_\pi(\check{u}^{k+1})$; see (1) above and (3.5).

As for the assumptions in (2) above, note that $\sigma^*(z^i) \to \sigma^*(z^*)$ if σ^* is continuous on $D := \operatorname{dom}\partial\sigma^*$ (e.g., in Sect. 6.3). Similarly, $\sigma(\bar{u}(z^i)) \to \sigma(u^*)$ holds if σ is continuous on $\operatorname{dom}\partial\sigma$ and $\bar{u}(z^i) \in \operatorname{dom}\partial\sigma$ for large i.

5 Lagrangian relaxation

We now consider the application of our method to (1.2) treated as the *primal problem*

$$\varphi_* := \sup\,\{\varphi(y) := -f(y)\} \quad \text{s.t.} \quad \psi(x,y) := y - Ax = 0,\ x \in X, \tag{5.1}$$

assuming that f is closed proper convex and the set $X \neq \emptyset$ is compact and convex. In view of (1.3) and (2.1), suppose that, at each $u^k \in C$, the oracle delivers

$$g^k_\pi := -Ax^k \quad \text{and} \quad \pi_k(\cdot) := \langle -Ax^k,\cdot\rangle \quad \text{for some } x^k \in X. \tag{5.2}$$

For simplicity, let Step 1 retain only selected past linearizations for its kth model

$$\check{\pi}_k(\cdot) := \max_{j\in J_k} \pi_j(\cdot) \quad \text{with} \quad k \in J_k \subset \{1,\dots,k\}. \tag{5.3}$$

Then (see (2.10) and [12, Sect. 4.4]) there are convex weights $\nu_j^k \geq 0$ such that

$$(\bar{\pi}_k, p_\pi^k, 1) = \sum_{j\in\hat{J}_k} \nu_j^k (\pi_j, g_\pi^j, 1) \quad \text{with} \quad \hat{J}_k := \{j \in J_k : \nu_j^k > 0\}, \tag{5.4}$$

and for convergence it suffices to choose $J_{k+1} \supset \hat{J}_k \cup \{k+1\}$. Using these weights and (2.7), we may estimate a solution to (5.1) via the *aggregate primal solution* $(\hat{x}^k, \hat{y}^k)$ with

$$\hat{x}^k := \sum_{j\in J_k} \nu_j^k x^j \quad \text{and} \quad \hat{y}^k := p_\sigma^k. \tag{5.5}$$

We first derive useful expressions of $\varphi(\hat{y}^k)$ and $\psi(\hat{x}^k, \hat{y}^k)$.

Lemma 5.1 *We have $\hat{x}^k \in X$, $\varphi(\hat{y}^k) = \theta_{\hat{u}}^k - \varepsilon_k - \langle p^k, \hat{u}^k\rangle$ and $\psi(\hat{x}^k, \hat{y}^k) = p^k$.*

Proof. First, $\hat{x}^k \in \mathrm{co}\{x^j\}_{j\in\hat{J}_k} \subset X$, $\bar{\pi}_k(\cdot) = \langle -A\hat{x}^k, \cdot\rangle$ and $p_\pi^k = -A\hat{x}^k$ by convexity of X, (5.2), (5.4) and (5.5). Then $p^k = \hat{y}^k - A\hat{x}^k = \psi(\hat{x}^k, \hat{y}^k)$ by (2.12), (5.1) and (5.5). Next, by [19, Thm. 23.5], the inclusion $\hat{y}^k := p_\sigma^k \in \partial\sigma(u^{k+1})$ of (2.10) with $\sigma := f^*$ in (1.3) yields $\sigma(u^{k+1}) = \langle u^{k+1}, \hat{y}^k\rangle - f(\hat{y}^k)$; thus $\varphi(\hat{y}^k) := -f(\hat{y}^k) = \bar{\sigma}_k(0)$ by (5.1) and (2.7). Since $\bar{\pi}_k(0) = 0$ in (2.11), (2.15) gives $\bar{\sigma}_k(0) = \bar{\theta}_k(0) = \theta_{\hat{u}}^k - \varepsilon_k + \langle p^k, \hat{u}^k\rangle$, as required. □

In terms of the optimality measure V_k of (2.9), the expressions of Lemma 5.1 imply

$$\hat{x}^k \in X \quad \text{with} \quad \varphi(\hat{y}^k) \geq \theta_{\hat{u}}^k - V_k, \quad |\psi(\hat{x}^k, \hat{y}^k)| \leq V_k. \tag{5.6}$$

We now show that $\{(\hat{x}^k, \hat{y}^k)\}$ has cluster points in the set of ε_π-*optimal solutions* of (5.1)

$$Z_{\varepsilon_\pi} := \{\, (x,y) \in X \times \mathbb{R}^m : \varphi(y) \geq \varphi_* - \varepsilon_\pi, \psi(x,y) = 0 \,\}, \tag{5.7}$$

unless $\varphi_* = -\infty$, i.e., the primal problem is infeasible. Note that (5.2) with X compact and (5.4) yield (3.1)–(3.2), as required for Theorem 3.3.

Theorem 5.2 *Either $\theta_* = -\infty$ and $\theta_{\hat{u}}^k \downarrow -\infty$, in which case the primal problem (5.1) is infeasible, or $\theta_* > -\infty$, $\theta_{\hat{u}}^k \downarrow \theta_{\hat{u}}^\infty \in [\theta_* - \varepsilon_\pi, \theta_*]$, $\overline{\lim}_k\, \theta(\hat{u}^k) \leq \theta_{\hat{u}}^\infty + \varepsilon_\pi$ and $\underline{\lim}_k V_k = 0$. In the latter case, let $K \subset \mathbb{N}$ be a subsequence such that $V_k \xrightarrow{K} 0$. Then:*

(1) *The sequence $\{(\hat{x}^k, \hat{y}^k)\}_{k\in K}$ is bounded and all its cluster points lie in the set $X \times \mathbb{R}^m$.*

(2) *Let $(\hat{x}^\infty, \hat{y}^\infty)$ be a cluster point of the sequence $\{(\hat{x}^k, \hat{y}^k)\}_{k\in K}$. Then $(\hat{x}^\infty, \hat{y}^\infty) \in Z_{\varepsilon_\pi}$.*

(3) $d_{Z_{\varepsilon_\pi}}((\hat{x}^k, \hat{y}^k)) := \inf_{(x,y)\in Z_{\varepsilon_\pi}} |(\hat{x}^k, \hat{y}^k) - (x,y)| \xrightarrow{K} 0$.

(4) *If $\varepsilon_\pi = 0$, then $\theta_{\hat{u}}^k \downarrow \theta_*$, $\varphi(\hat{y}^k) \xrightarrow{K} \varphi_* = \theta_*$, and $\psi(\hat{x}^k, \hat{y}^k) \xrightarrow{K} 0$.*

Proof. The first assertion follows from Theorem 3.3 (since $\theta_* = -\infty$ implies primal infeasibility by weak duality). In the second case, using $\theta_{\hat{u}}^k \downarrow \theta_{\hat{u}}^\infty \geq \theta_* - \varepsilon_\pi$ and $V_k \xrightarrow{K} 0$ in the bounds of (5.6) yields $\underline{\lim}_{k\in K}\, \varphi(\hat{y}^k) \geq \theta_* - \varepsilon_\pi$ and $\lim_{k\in K} \psi(\hat{x}^k, \hat{y}^k) = 0$.

(1) By (5.6), $\{\hat{x}^k\}$ lies in the compact X; then $\{\hat{y}^k\}_{k\in K}$ is bounded by (5.1) and (5.6).

(2) We have $\hat{x}^\infty \in X$, $\varphi(\hat{y}^\infty) \geq \theta_* - \varepsilon_\pi$ and $\psi(\hat{x}^\infty, \hat{y}^\infty) = 0$ by closedness of φ and continuity of ψ. Since $\theta_* \geq \varphi_*$ by weak duality (cf. (1.1), (1.3), (5.1)), we get $\varphi(\hat{y}^\infty) \geq \varphi_* - \varepsilon_\pi$. Thus $(\hat{x}^\infty, \hat{y}^\infty) \in Z_{\varepsilon_\pi}$ by the definition (5.7).

(3) This follows from (1), (2) and the continuity of the distance function $d_{Z_{\varepsilon_\pi}}$.

(4) In the proof of (2), $\theta_* \geq \varphi_* \geq \varphi(\hat{y}^\infty) \geq \theta_*$ yields $\varphi_* = \varphi(\hat{y}^\infty) = \theta_*$, and for $K' \subset K$ such that $\hat{y}^k \xrightarrow{K'} \hat{y}^\infty$ we have $\varphi(\hat{y}^\infty) \geq \overline{\lim}_{k\in K'} \varphi(\hat{y}^k) \geq \underline{\lim}_{k\in K'} \varphi(\hat{y}^k) \geq \theta_*$, i.e., $\varphi(\hat{y}^k) \xrightarrow{K'} \varphi_*$. So considering convergent subsequences in (1) gives $\varphi(\hat{y}^k) \xrightarrow{K} \varphi_*$. □

6 Application to multicommodity network flows

6.1 The nonlinear multicommodity fbw problem

Let $(\mathscr{N}, \mathscr{A})$ be a directed graph with $N := |\mathscr{N}|$ nodes and $m := |\mathscr{A}|$ arcs. Let $E \in \mathbb{R}^{N\times m}$ be its node-arc incidence matrix. There are n commodities to be routed through the network. For each commodity i there is a *required flow* $r_i > 0$ from its *source* node o_i to its *sink* node d_i. Let s_i be the *supply* N-vector of commodity i, having components $s_{io_i} = r_i$, $s_{id_i} = -r_i$, $s_{il} = 0$ if $l \neq o_i, d_i$. Our *nonlinear multicommodity flow problem* (NMFP for short) is:

$$\min \quad f(y) := \sum_{j=1}^{m} f_j(y_j) \tag{6.1a}$$

$$\text{s.t.} \quad y = \sum_{i=1}^{n} x_i, \tag{6.1b}$$

$$x_i \in X_i := \{x_i : Ex_i = s_i, 0 \leq x_i \leq \bar{x}_i\}, \quad i = 1\colon n, \tag{6.1c}$$

where each *arc cost* function f_j is closed proper convex, y is the *total flow* vector, x_i is the *flow vector* of commodity i, and $\bar{x}_i$ is a fixed positive vector of *flow bounds* for each i.

Our assumptions seem to be weaker than those employed in the literature. We add that if $\operatorname{dom} f^* \subset \mathbb{R}^m_+$, then the flow bounds $\bar{x}_i$ are not needed in (6.1c): Even if they are absent, our algorithm will proceed as if we had $\bar{x}_{ij} = r_i$ for all i and j; cf. [13, Sect. 7.2].

6.2 Primal recovery

We may treat problem (6.1) as (5.1) with $Ax = \sum_{i=1}^n x_i$, $X = \prod_{i=1}^n X_i$, and the oracle solving shortest path problems to evaluate $\pi(u^k) = -\sum_{i=1}^n \min\{\langle u^k, x_i\rangle : x_i \in X_i\}$ at each u^k. Thus the results of Sect. 5 hold. Yet, as in [13, Sect. 7.3], for stopping criteria it is useful to employ another aggregate solution $(\hat{x}^k, \check{y}^k)$ with $\hat{x}^k$ given by (5.5) and

$$\check{y}^k := A\hat{x}^k = \sum_{i=1}^{n} \hat{x}_i^k, \tag{6.2}$$

which satisfies the constraints of (6.1). Thus $f(\check{y}^k) \geq f_*$, where the optimal value f_* of (6.1) satisfies $-f_* = \varphi_* \leq \theta_*$ by weak duality. Hence, if the oracle is exact, $\theta^k_{\hat{u}} \geq \theta_*$ implies that the method may stop when $f(\check{y}^k) + \theta^k_{\hat{u}} \leq \varepsilon$ for a given tolerance $\varepsilon > 0$, in which case $(\hat{x}^k, \check{y}^k)$ is an *ε-solution* of (6.1). This stopping criterion will be met for some k under conditions similar to those in [13, Prop. 7.1].

Proposition 6.1 *Suppose problem* (6.1) *is feasible and has a unique optimal total flow y^* (e.g., f is strictly convex on $\mathbb{R}^m_+ \cap \operatorname{dom} f$) that satisfies $y^* \in [0, c) \subset \operatorname{dom} f$ for some $c \in \mathbb{R}^m_+$. Further, let*

$\varepsilon_\pi = 0$ (*i.e., the oracle is exact*), *and let* $K \subset \mathbb{N}$ *be a subsequence such that* $V_k \xrightarrow{K} 0$. *Then* $\check{y}^k \xrightarrow{K} y^*$, $f(\check{y}^k) \xrightarrow{K} f_* = -\theta_*$ *and* $f(\check{y}^k) + \theta^k_{\hat{u}} \xrightarrow{K} 0$.

Proof. By Theorem 5.2(3) and the uniqueness of y^*, $\hat{y}^k \xrightarrow{K} y^*$. Hence $\check{y}^k \xrightarrow{K} y^*$ from $\hat{y}^k - \check{y}^k = \psi(\hat{x}^k, \hat{y}^k) \xrightarrow{K} 0$ (cf. Theorem 5.2(4)), where $\check{y}^k \geq 0$ by (6.2) with $\hat{x}^k \in X$ (Lem. 5.1). Consequently, $y^* \in [0, c)$ gives $\check{y}^k \in [0, c)$ for all large $k \in K$. Since each function f_j in (6.1a) is continuous on $\operatorname{dom} f_j \supset [0, c_j)$, we have $f(\check{y}^k) \xrightarrow{K} f(y^*) = f_*$. The conclusion follows from Theorem 5.2(4) with $\theta_* = \varphi_* = -f_*$. □

An extension to the case where some arc costs are linear follows.

Proposition 6.2 *Let problem* (6.1) *be feasible. Suppose that the first* $\check{m}$ *components of any optimal total flow* y^* *are unique* (*e.g.,* f_j *are strictly convex on* $\mathbb{R}^m_+ \cap \operatorname{dom} f_j$ *for* $j \leq \check{m}$) *and satisfy* $y^*_j \in [0, c_j) \subset \operatorname{dom} f_j$ *for some* $c_j > 0$, *whereas the costs* f_j *are linear for* $j > \check{m}$. *Further, let* $\varepsilon_\pi = 0$ (*i.e., the oracle is exact*), *and let* $K \subset \mathbb{N}$ *be a subsequence such that* $V_k \xrightarrow{K} 0$. *Then* $\check{y}^k_j \xrightarrow{K} y^*_j$ *for* $j \leq \check{m}$, $f(\check{y}^k) \xrightarrow{K} f_* = -\theta_*$ *and* $f(\check{y}^k) + \theta^k_{\hat{u}} \xrightarrow{K} 0$.

Proof. The proof of Proposition 6.1 gives $\hat{y}^k_j, \check{y}^k_j \xrightarrow{K} y^*_j$ and $f_j(\hat{y}^k_j), f_j(\check{y}^k_j) \xrightarrow{K} f_j(y^*_j)$ for $j \leq \check{m}$, since $\hat{y}^k \in \operatorname{dom} f$ by (5.6). For $j > \check{m}$, $f_j(y_j) = \alpha_j y_j$ for some $\alpha_j \in \mathbb{R}$; thus $\sigma_j(u_j) := f^*_j(u_j) = i_{\{\alpha_j\}}(u_j)$. Then $u^{k+1}_j = \hat{u}^k_j = \alpha_j$ in (1.6) yields $p^k_j = 0$ in (2.8), so $\psi_j(\hat{x}^k, \hat{y}^k) = 0$ by Lemma 5.1; since $\hat{y}^k - \check{y}^k = \psi(\hat{x}^k, \hat{y}^k)$, we have $\hat{y}^k_j = \check{y}^k_j$ for $j > \check{m}$. Therefore, by (6.1a),

$$f(\check{y}^k) = f(\hat{y}^k) + \sum_{j \leq \check{m}} [f_j(\check{y}^k_j) - f_j(\hat{y}^k_j)],$$

where the sum vanishes as $k \xrightarrow{K} \infty$; Theorem 5.2(4) with $\varphi := -f$ gives the conclusion. □

6.3 Specific arc costs

For specific arc costs, as in [1, 13], we shall consider *Kleinrock*'s average delays

$$f_j(y_j) := \begin{cases} \infty & \text{if } y_j \geq c_j, \\ y_j/(c_j - y_j) & \text{if } y_j \in [0, c_j), \\ y_j/c_j & \text{if } y_j < 0, \end{cases} \tag{6.3a}$$

$$f^*_j(u_j) := \begin{cases} \left(\sqrt{c_j u_j} - 1\right)^2 & \text{if } u_j \geq 1/c_j, \\ \infty & \text{if } u_j < 1/c_j, \end{cases} \tag{6.3b}$$

with arc capacities $c_j > 0$, the *BPR* (Bureau of Public Roads) nonlinear delays

$$f_j(y_j) := \begin{cases} \alpha_j y_j + \beta_j y_j^{\gamma_j} & \text{if } y_j \geq 0, \\ \alpha_j y_j & \text{if } y_j < 0, \end{cases} \tag{6.4a}$$

$$f^*_j(u_j) := \begin{cases} \frac{\gamma_j - 1}{\gamma_j} (u_j - \alpha_j)^{\gamma_j/(\gamma_j - 1)} / (\beta_j \gamma_j)^{1/(\gamma_j - 1)} & \text{if } u_j \geq \alpha_j, \\ \infty & \text{if } u_j < \alpha_j, \end{cases} \tag{6.4b}$$

with parameters $\alpha_j \geq 0$, $\beta_j > 0$, $\gamma_j \geq 2$, as well as BPR *linear* delays with $\alpha_j \geq 0$:

$$f_j(y_j) := \alpha_j y_j \quad \text{for all } y_j, \tag{6.5a}$$

$$f_j^*(u_j) := \begin{cases} 0 & \text{if } u_j = \alpha_j, \\ \infty & \text{if } u_j \neq \alpha_j. \end{cases} \tag{6.5b}$$

Our costs are *linearly extrapolated* versions of the "standard" costs used in [13], where $f_j(y_j)$ is set to ∞ for $y_j < 0$, so that $f_j^*(u_j)$ becomes 0 instead of ∞ for $u_j < f_j'(0)$. Note that the value of f_j at $y_j < 0$ does not matter for (6.1), where the constraints yield $y_j \geq 0$. Further, if (6.1) is feasible, the assumptions of Propositions 6.1 and 6.2 hold for our Kleinrock and nonlinear BPR costs, and for a mixture of our nonlinear and linear BPR costs, respectively. Finally, since $\operatorname{dom}\sigma = \operatorname{dom} f^* \subset \mathbb{R}^m_+$ for our costs, the oracle has to solve shortest path problems with nonnegative arc lengths u^k only; hence, we may assume that $\varepsilon_\pi = 0$.

6.4 Solving the σ-subproblem for specific arc costs

We now specialize the results of Sect. 4.2 with $\sigma^* := f$ for the costs of Sect. 6.3. Since σ^* is separable, we may handle (4.9) by solving m one-dimensional subproblems to determine components of an approximate solution, say $\tilde{z}$. Thus we need a stopping criterion for each subproblem. To this end, we replace the criterion $\varepsilon(z^i) \leq \kappa_N v(z^i)$ by $\varepsilon(\tilde{z}) \leq \kappa_N \bar{v}(\tilde{z})$ for

$$\bar{v}(z) := \sigma_u^k - \bar{\sigma}(\hat{u}^k; z) + t_k |p_\pi^k + z|^2 = v(z) - \left[\pi_u^k - \bar{\pi}_k(\hat{u}^k)\right], \tag{6.6}$$

where the second equality follows from (4.5), (4.6) and (4.8) with $\theta_u^k = \sigma_u^k + \pi_u^k$. Moreover, $\sigma_u^k - \bar{\sigma}(\hat{u}^k; z) \geq 0$ yields $\bar{v}(z) \geq 0$, whereas by the results of Sect.4.2, $\bar{v}(z) = 0$ only if $z = z^* = -p_\pi^k$; since checking if $\bar{v}(-p_\pi^k) = 0$ is easy, we may assume that $\bar{v}(z^*) > 0$. Finally, $\bar{v}(z) \leq v(z)$ from $\varepsilon_\pi = 0$. The resulting "natural" subproblem criteria are discussed below.

To simplify notation, we assume temporarily that $m = 1$, drop the subscript j in (6.3)–(6.5) and let $t := t_k$ in (4.5). We first consider the Kleinrock and nonlinear BPR costs in (6.3)–(6.4). For finding an approximate solution $\tilde{z}$, we exploit the following properties:

- $f(z) = f'(0)z$ for $z \leq 0$ with $f'(0) \geq 0$;
- $f''(z) > 0$ for $z > 0$ in $F := \operatorname{dom} f = (-\infty, c)$, with $c := \infty$ in the BPR case;
- $\sigma^* = f$ is continuous on F with $\operatorname{dom}\partial\sigma^* = F$;
- $\sigma := f^*$ is continuous on $\operatorname{dom}\sigma = [f'(0), \infty)$ with $\operatorname{dom}\partial\sigma = \operatorname{dom}\sigma$;
- $w'(z) = f'(z) - \bar{u}(z)$ and $w''(z) = f''(z) + t$ for $z \in F$ in (4.9) by (4.5).

If $w'(0) \geq 0$, then $\tilde{z} := -w'(0)/t$ is optimal ($w'(\tilde{z}) = 0$), $\varepsilon(\tilde{z}) = 0$ and $\bar{u}(\tilde{z}) = f'(0)$.
If $w'(0) < 0$, then $z^* \in (0, -w'(0)/t)$, since for $z \geq -w'(0)/t$, $f'(z) > f'(0)$ yields

$$w'(z) = f'(z) - \bar{u}(z) > f'(0) - \bar{u}(z) = w'(0) + tz \geq 0.$$

Further, $z^* \in (0, z^{\text{up}})$ for $z^{\text{up}} := \min\{-w'(0)/t, c\}$ from $z^* \in F$, and $\bar{u}(z) \in \operatorname{dom}\sigma$ for $z \in (0, z^{\text{up}})$, since $\bar{u}(z) > f'(0)$ iff $z < -w'(0)/t$. These properties and the results of Sect. 4.2 yield the following. Suppose we minimize w over $(0, z^{\text{up}})$ via a descent method, starting from $z^1 := p_\sigma^{k-1}$ if $p_\sigma^{k-1} \in (0, z^{\text{up}})$ or any $z^1 \in (0, z^{\text{up}})$ otherwise, which generates points $z^i \in (0, z^{\text{up}})$ such that $z^i \to z^*$. Then $\varepsilon(z^i) \to 0$ and $\bar{v}(z^i) \to \bar{v}(z^*) > 0$ in (6.6) imply that we will eventually have $\varepsilon(z^i) \leq \kappa_N \bar{v}(z^i)$, in which case the method may stop with $\tilde{z} := z^i$.

Next, for the linear BPR costs in (6.5) with $w'(z) = f'(0) - \bar{u}(z)$, $\tilde{z} := -w'(0)/t$ is optimal ($w'(\tilde{z}) = 0$), $\varepsilon(\tilde{z}) = 0$ and $\bar{u}(\tilde{z}) = f'(0)$ (as in the case of $w'(0) \geq 0$ above).

For $m > 1$, expressing $\varepsilon(z)$ in (4.7), $w(z)$ in (4.9) and $\bar{v}(z)$ in (6.6) as sums of $\varepsilon_j(z_j)$, $w_j(z_j)$ and $\bar{v}_j(z_j)$ respectively over $j = 1, \dots, m$, for each j we may find $\tilde{z}_j$ as above so that $\varepsilon_j(\tilde{z}_j) \le \kappa_N \bar{v}_j(\tilde{z}_j)$, and $w(\tilde{z}) \le w(p_\sigma^{k-1})$; since $\bar{v}(z) \le v(z)$ in (6.6), we also have $\varepsilon(\tilde{z}) \le \kappa_N v(\tilde{z})$. Thus, as in Sect. 4.2, we may set $u^{k+1} := \bar{u}(\tilde{z})$, $\chi := v(\tilde{z})$, $\bar{\sigma}_k(\cdot) := \bar{\sigma}(\cdot; \tilde{z})$ and $p_\sigma^k := \tilde{z}$.

7 Implementation issues

We now describe the main issues in our implementation of each step of Algorithm 2.1 for the network applications of Sect. 6. We also highlight aspects where our implementation could be less efficient than that of [1]; improving these aspects is left for future work.

7.1 Initial settings

In the Kleinrock case of (6.3), the initial $u_j^1 := (1-\rho_*)^{-2}/c_j$ for all j, with $\rho_* := \frac{1}{4}$ estimating the maximum traffic intensity $\max_j y_j^*/c_j$ as in [5, 13]; then $p_\sigma^0 := \nabla\sigma(u^1)$. In the BPR case of (6.4)–(6.5), $u_j^1 := \alpha_j$ for all j, and we let $p_\sigma^0 := 0$.

As usual in bundle methods, we use the descent parameter $\kappa = 0.1$ in (2.4). We set the initial stepsize to $t_1 := 1$, corresponding to the inverse of the initial proximal coefficient of [1], and let $t_{\min} := 10^{-20} t_1$.

7.2 Subproblem solution

For the polyhedral models $\check{\pi}_k$ of (5.3), subproblem (1.5) is solved by the QP routine of [9]. This routine has at least two drawbacks. First, being designed for bound-constrained problems, it employs data structures that are not efficient in the unconstrained case. Second, its linear algebra is behind the current state of the art (in contrast with the MATLAB implementation of [1], where linear equations are solved more efficiently).

The one-dimensional subproblems of Sect. 6.4 are solved for the tolerance $\kappa_N = 10^{-3}$ by Newton's method with Armijo's backtracks for a descent tolerance of 10^{-6}, where at each iteration the initial unit stepsize is reduced if necessary to 0.9 times the maximum feasible stepsize, and the stepsize is divided by 2 for each Armijo's failure. This works quite well, but implementations based on self-concordant ideas (as in [1]) could be more efficient.

The looping Step 5′ of Sect. 4.1 employs the tolerance $\check{\kappa} = 0.2$, but the number of loops at any iteration is limited to 30.

7.3 Shortest-path oracle

Let $S \le n$ be the number of common sources (different source nodes) in (6.1). To evaluate $\pi(u^{k+1})$, we call S times subroutine L2QUE of [4], which finds shortest paths from a given source to all other nodes. Being quite old, L2QUE is unlikely to be competitive with Dijkstra's algorithm with binary heap structures employed in [1].

7.4 Termination criterion

In view of Sect. 6.2, we stop after Step 6 when the relative optimality gap is small enough:

$$\gamma_{\text{rel}}^k := \big(f_{\text{up}}^k - f_{\text{low}}^k \big) / \max\{ f_{\text{low}}^k, 1\} \le \varepsilon_{\text{opt}}, \tag{7.1}$$

where $\varepsilon_{\text{opt}} = 10^{-5}$ as in [1], whereas f^k_{up} and f^k_{low} are the best upper and lower bounds on f_* obtained so far. Specifically, $f^k_{\text{low}} := -\min_{j\le k+1} \theta^j_u$, whereas f^k_{up} is the minimum of $f(\check{y}^j)$ over iterations $j \le k$, $j = 10, 20, \ldots$, at which $f(\check{y}^j)$ is computed. A more frequent computation of $f(\check{y}^j)$ could save work on small instances.

7.5 Stepsize updating

Our implementation of Step 8 uses the following procedure, in which γ^k_{rel} is the relative gap of (7.1), $\gamma_k := f^k_{\text{up}} - f^k_{\text{low}}$ is the absolute gap, l_k is the number of loops made on iteration k, and n_k counts descent or null steps since the latest change of t_k, with $n_1 := 1$.

Procedure 7.1 (Stepsize updating)

(1) Set $t_{k+1} := t_k$.
(2) If $\hat{u}^{k+1} = \hat{u}^k$ or $l_k > 0$ go to (5).
(3) If $n_k \ge 10$, or $v_k < \gamma_k/2$ and $\gamma^k_{\text{rel}} \le 0.01$, set $t_{k+1} := 2t_k$.
(4) Set $n_{k+1} := \max\{n_k + 1, 1\}$. If $t_{k+1} \ne t_k$, set $n_{k+1} := 1$. Exit.
(5) If $i^{k+1}_t = 0$, $n_k \le -10$, and either $v_k > \gamma_k/2$ or $\gamma^k_{\text{rel}} > 0.01$, set $t_{k+1} := \max\{t_k/5, t_{\min}\}$.
(6) Set $n_{k+1} := \min\{n_k - 1, -1\}$. If $t_{k+1} \ne t_k$, set $n_{k+1} := -1$. Exit.

The counter n_k introduces some inertia, which smooths out the stepsize updating. In general, t_k should be increased (respectively decreased) if "too many" descent (respectively null) steps are occuring, but v_k should be of order γ_k, since descent steps with $v_k \ll \gamma_k$ bring little. Of course, our procedure is just an example and there is still room for improvement.

8 Numerical illustrations

To get a feeling for the practical merits and drawbacks of our approach, we benchmark our AL implementation against the ACCPM results of [1].

8.1 Test problems

We used the same four sets of test problems as in [1]. Their features are given in Table 8.1, where N is the number of nodes, m is the number of arcs, n is the number of commodities, S is the number of common sources, and $f^{\text{Kleinrock}}_*$ and f^{BPR}_* are the optimal values of (6.1) for the Kleinrock and BPR costs respectively, with relative optimality gaps of at most 10^{-5}. Table 8.1 corrects some values of [1, Tab. 2]; see [2] and below.

For the first two sets of planar and grid problems[1], the cost functions are generated as in [1, Sect. 8.1]; we add that problem planar150 is missing in [1].

The third set of telecommunication problems includes a corrected version of problem ndo22 [2]; the BPR costs are generated as in [1].

The fourth set of transportation problems[2] uses original BPR costs, and Kleinrock costs generated as in [1]. To clarify the description of [1], we add that in the Kleinrock case the demands are divided by 2 for Sioux-Falls, 2000 for Winnipeg, 5100 for Barcelona, 2.5 for Chicago-sketch, 6 for Chicago-region, and 7 for Philadelphia. We also observe that although [1, Tab. 2] gives wrong Kleinrock

[1] Available at http://www.di.unipi.it/di/groups/optimize/Data/MMCF.html.
[2] Available at http://www.bgu.ac.il/bargera/tntp/.

Table 8.1: Test problems

Problem	N	m	n	S	$f_*^{\text{Kleinrock}}$	f_*^{BPR}
Planar problems						
planar30	30	150	92	29	40.5668	4.44549×10^7
planar50	50	250	267	50	109.478	1.21236×10^8
planar80	80	440	543	80	232.321	1.81906×10^8
planar100	100	532	1085	100	226.299	2.29114×10^8
planar150	150	850	2239	150	715.309	5.27985×10^8
planar300	300	1680	3584	300	329.120	6.90748×10^8
planar500	500	2842	3525	500	196.394	4.83309×10^9
planar800	800	4388	12756	800	354.008	1.16952×10^9
planar1000	1000	5200	20026	1000	1250.92	3.41859×10^9
planar2500	2500	12990	81430	2500	3289.05	1.23827×10^{10}
Grid problems						
grid1	25	80	50	23	66.4002	8.33599×10^5
grid2	25	80	100	25	194.512	1.72689×10^6
grid3	100	360	50	40	84.5618	1.53241×10^6
grid4	100	360	100	63	171.331	3.05543×10^6
grid5	225	840	100	83	236.699	5.07921×10^6
grid6	225	840	200	135	652.877	1.05075×10^7
grid7	400	1520	400	247	776.566	2.60669×10^7
grid8	625	2400	500	343	1542.15	4.21240×10^7
grid9	625	2400	1000	495	2199.83	8.36394×10^7
grid10	625	2400	2000	593	2212.89	1.66084×10^8
grid11	625	2400	4000	625	1502.75	3.32475×10^8
grid12	900	3480	6000	899	1478.93	5.81488×10^8
grid13	900	3480	12000	900	1760.53	1.16933×10^9
grid14	1225	4760	16000	1225	1414.39	1.81297×10^9
grid15	1225	4760	32000	1225	1544.15	3.61568×10^9
Telecommunication problems						
ndo22	14	22	23	5	103.412	1.86767×10^3
ndo148	61	148	122	61	151.926	1.40233×10^5
904	106	904	11130	106	33.4931	1.29197×10^7
Transportation problems						
Sioux-Falls	24	76	528	24	600.679	4.23133×10^6
Winnipeg	1067	2836	4344	135	1527.41	8.25673×10^5
Barcelona	1020	2522	7922	97	845.872	1.22856×10^6
Chicago-sketch	933	2950	93135	386	614.726	1.67484×10^7
Chicago-region	12982	39018	2296227	1771	3290.49	2.58457×10^7
Philadelphia	13389	40003	1149795	1489	2557.42	2.24926×10^8

Table 8.2: Peformance of AL for Kleinrock costs

Problem	k	l	Sigma	Newton	CPU	%Si	%Or	AC/AL
planar30	125	62	4.7	1.9	0.1	60	0	11.0
planar50	214	73	3.2	2.2	0.2	31	10	11.0
planar80	308	80	3.0	2.2	0.6	28	28	10.8
planar100	312	75	3.9	2.4	0.8	24	28	7.5
planar150	979	95	1.7	2.1	12.2	3	17	10.8
planar300	303	84	6.4	2.7	4.7	27	46	4.7
planar500	253	77	8.3	2.6	9.7	23	55	2.5
planar800	341	82	7.7	2.7	28.1	16	69	2.7
planar1000	648	104	4.1	3.0	74.8	8	73	4.1
planar2500	1530	103	2.5	2.6	1092.1	2	86	2.2
grid1	92	65	8.2	2.3	0.1	20	20	5.0
grid2	185	62	2.9	2.4	0.0	0	0	8.0
grid3	222	74	6.7	2.2	0.4	43	13	5.7
grid4	247	79	5.3	2.7	0.4	43	9	7.7
grid5	290	82	5.5	2.3	1.2	40	19	10.0
grid6	453	89	2.9	2.5	2.3	17	26	10.6
grid7	646	98	3.0	2.4	8.3	12	32	11.0
grid8	940	102	2.1	2.3	21.0	8	42	18.3
grid9	900	99	2.2	2.4	24.3	7	49	12.6
grid10	730	100	2.8	2.7	22.0	9	54	9.1
grid11	424	85	5.6	3.3	14.0	19	51	6.9
grid12	458	96	5.8	3.4	26.9	16	59	4.0
grid13	423	94	6.4	3.7	26.0	20	58	4.8
grid14	470	106	7.1	3.9	49.2	18	62	3.4
grid15	451	102	7.7	4.1	49.4	19	62	3.3
ndo22	361	187	17.9	2.0	0.1	30	0	2.0
ndo148	94	53	2.3	2.0	0.0	0	0	8.0
904	240	58	7.5	3.1	1.5	53	22	5.1
Sioux-Falls	497	252	2.4	2.1	0.1	8	0	16.0
Winnipeg	1298	482	4.6	1.8	123.7	4	10	1.1
Barcelona	2611	434	1.7	1.6	127.6	2	17	0.6
Chicago-sketch	375	92	8.1	2.5	18.3	18	60	1.6
Chicago-region	303	73	7.7	2.1	901.0	4	88	9.6
Philadelphia	433	89	8.4	3.2	1431.3	5	85	9.1

values for Chicago-sketch, Chicago-region and Philadelphia, their entries in [1, Tab. 5] are apparently correct. In contrast, for the BPR versions of Barcelona and Philadelphia, [1, Tab. 6] must be corrected as in [2].

8.2 Numerical results

Tables 8.2 and 8.3 give our results for the problems of Sect. 8.1. In these tables,

- k and l are the numbers of iterations and descent steps respectively;
- *Sigma* is the average number of subproblems solved at Step 3 per iteration;
- *Newton* is the average number of Newton's iterations for the one-dimensional subproblems solved approximately at Step 3 (cf. Sect. 7.2);
- *CPU* is the total CPU time in seconds;
- *%Si* is the percentage of CPU time spent on the subproblems of Step 3;

Table 8.3: Peformance of AL for BPR costs

Problem	k	l	Sigma	Newton	CPU	%Si	%Or	AC/AL
planar30	75	69	1.3	1.1	0.0	66	33	12.0
planar50	105	64	1.4	1.3	0.0	66	33	29.0
planar80	150	59	1.1	1.3	0.2	8	73	33.5
planar100	108	44	1.4	1.3	0.2	20	54	21.5
planar150	194	52	1.1	1.5	0.9	12	67	24.7
planar300	97	31	1.3	1.2	1.4	8	86	9.0
planar500	50	23	1.7	1.0	3.3	4	92	2.6
planar800	108	33	1.9	1.2	25.4	2	94	1.3
planar1000	209	41	1.4	1.3	32.6	2	88	4.1
planar2500	264	52	1.3	1.6	411.8	0	97	4.0
grid1	48	29	3.6	2.2	0.0	25	0	4.0
grid2	61	27	1.7	2.2	0.0	0	0	8.0
grid3	43	23	2.5	1.3	0.0	100	0	7.0
grid4	59	26	1.8	2.2	0.1	50	50	15.0
grid5	86	28	2.1	1.7	0.3	44	35	8.0
grid6	150	33	2.0	2.0	0.6	47	35	11.3
grid7	108	31	2.1	2.3	0.9	35	56	10.2
grid8	143	36	1.6	2.3	2.4	22	58	12.8
grid9	183	37	1.7	2.4	4.0	21	60	11.6
grid10	200	34	2.3	2.5	5.4	22	57	8.4
grid11	120	32	4.2	3.2	4.1	40	48	7.3
grid12	122	31	5.8	3.4	8.8	40	47	3.9
grid13	140	30	5.5	3.6	10.1	37	48	4.4
grid14	111	28	8.0	4.0	16.1	42	45	3.3
grid15	115	26	8.0	4.3	17.1	44	45	3.5
ndo22	11	8	2.2	2.2	0.0	0	0	1.0
ndo148	14	11	2.4	2.1	0.0	0	100	2.0
904	116	32	1.2	2.8	0.5	27	62	12.4
Sioux-Falls	105	37	6.3	2.6	0.1	83	0	14.0
Winnipeg	127	31	8.4	1.8	4.5	53	37	2.4
Barcelona	92	24	14.3	3.0	5.5	72	18	1.4
Chicago-sketch	129	32	7.0	2.2	7.1	32	55	2.6
Chicago-region	300	51	3.6	2.6	891.0	5	89	9.2
Philadelphia	671	62	2.7	1.9	3239.7	2	94	2.6

- *%Or* is the percentage of CPU time spent on the oracle's shortest path subproblems;
- *AC/AL* is the ratio of the CPU times of ACCPM from [1, Tabs. 5 and 6][3], [2, Tab. 1] and our AL, with our times increased to 0.1 if necessary.

As for CPU comparisons, we used a Dell M60 notebook (Pentium M 755 2 GHz, 1.5 GB RAM) under MS Windows XP and Fortran 77, with SPECint2000 of 1541 and SPECfp2000 of 1088. On the other hand, [1] used a desktop PC (P4 2.8 GHz, 2 GB RAM) under Linux, Matlab for linear algebra and C for the shortest path computation, with SPECint2000 of 1254 and SPECfp2000 of 1327. Hence our CPU times are comparable with those of [1].

Thus it is interesting to compare the CPU performance of ACCPM and AL. Here we ignore the smallest problem ndo22. In the Kleinrock case (Tab. 8.2), AL is substantially faster than ACCPM on most instances, and slower than ACCPM on a single instance of Barcelona. In the BPR case (Tab. 8.3), AL is substantially faster than ACCPM on all instances except for planar800 and Barcelona, where its speedups over ACCPM are quite modest.

In conclusion, AL is competitive with ACCPM.

Acknowledgements. I would like to thank F. Babonneau, H. Bar-Gera and J.-P. Vial for numerous discussions and help with the test problems.

References

[1] Babonneau, F., Vial, J.Ph.: ACCPM with a nonlinear constraint and an active set strategy to solve nonlinear multicommodity flow problems. Math. Programming **?** (2007). To appear

[2] Babonneau, F., Vial, J.Ph.: ACCPM with a nonlinear constraint and an active set strategy to solve nonlinear multicommodity flow problems: A corrigendum. Math. Programming **?** (2007). To appear

[3] Frangioni, A.: Solving semidefinite quadratic problems within nonsmooth optimization algorithms. Comput. Oper. Res. **23**, 1099–1118 (1996)

[4] Gallo, G., Pallotino, S.: Shortest path algorithms. Ann. Oper. Res. **13**, 3–79 (1988)

[5] Goffin, J.L.: The ellipsoid method and its predecessors. In: L. Contesse, R. Correa, A. Weintraub (eds.) Recent Advances in System Modelling and Optimization, Lecture Notes in Control and Information Sciences 87, pp. 127–141. Springer-Verlag, Berlin (1987)

[6] Hiriart-Urruty, J.B., Lemaréchal, C.: Convex Analysis and Minimization Algorithms. Springer, Berlin (1993)

[7] Kiwiel, K.C.: A method for solving certain quadratic programming problems arising in nonsmooth optimization. IMA J. Numer. Anal. **6**, 137–152 (1986)

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

[9] Kiwiel, K.C.: A Cholesky dual method for proximal piecewise linear programming. Numer. Math. **68**, 325–340 (1994)

[3] The CPU times for problem planar150 were provided by F. Babonneau.

[10] Kiwiel, K.C.: A projection-proximal bundle method for convex nondifferentiable minimization. In: M. Théra, R. Tichatschke (eds.) Ill-posed Variational Problems and Regularization Techniques, Lecture Notes in Economics and Mathematical Systems 477, pp. 137–150. Springer-Verlag, Berlin (1999)

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

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

[13] Kiwiel, K.C., Larsson, T., Lindberg, P.O.: Lagrangian relaxation via ballstep subgradient methods. Math. Oper. Res. **32**, 669–686 (2007)

[14] Kiwiel, K.C., Lemaréchal, C.: An inexact bundle variant suited to column generation. Math. Programming **?** (2007). DOI 10.1007/s10107-007-0187-4. To appear

[15] Kiwiel, K.C., Rosa, C.H., Ruszczyński, A.: Proximal decomposition via alternating linearization. SIAM J. Optim. **9**, 668–689 (1999)

[16] Lemaréchal, C., Ouorou, A., Petrou, G.: A bundle-type algorithm for routing in telecommunication data networks. Comput. Optim. Appl. **?** (2007). To appear

[17] Martinet, B.: Régularisation d'inéquations variationelles par approximations successives. RAIRO Rech. Opér. **4(R3)**, 154–158 (1970)

[18] Ouorou, A., Mahey, P., Vial, J.Ph.: A survey of algorithms for convex multicommodity flow problems. Management Sci. **46**, 126–147 (2000)

[19] Rockafellar, R.T.: Convex Analysis. Princeton University Press, Princeton, NJ (1970)

[20] Rockafellar, R.T.: Monotone operators and the proximal point algorithm. SIAM J. Control Optim. **14**, 877–898 (1976)

Unité de recherche INRIA Rhône-Alpes
655, avenue de l'Europe - 38334 Montbonnot Saint-Ismier (France)

Unité de recherche INRIA Futurs : Parc Club Orsay Université - ZAC des Vignes
4, rue Jacques Monod - 91893 ORSAY Cedex (France)
Unité de recherche INRIA Lorraine : LORIA, Technopôle de Nancy-Brabois - Campus scientifi que
615, rue du Jardin Botanique - BP 101 - 54602 Villers-lès-Nancy Cedex (France)
Unité de recherche INRIA Rennes : IRISA, Campus universitaire de Beaulieu - 35042 Rennes Cedex (France)
Unité de recherche INRIA Rocquencourt : Domaine de Voluceau - Rocquencourt - BP 105 - 78153 Le Chesnay Cedex (France)
Unité de recherche INRIA Sophia Antipolis : 2004, route des Lucioles - BP 93 - 06902 Sophia Antipolis Cedex (France)

Éditeur
INRIA - Domaine de Voluceau - Rocquencourt, BP 105 - 78153 Le Chesnay Cedex (France)
http://www.inria.fr
ISSN 0249-6399