Math. Program., Ser. A 102: 407–456 (2005)

Digital Object Identifier (DOI) 10.1007/s10107-004-0553-4

Aharon Ben-Tal · Arkadi Nemirovski

Non-euclidean restricted memory level method for large-scale convex optimization*

Received: January 27, 2003 / Accepted: July 26, 2004
Published online: December 29, 2004 –

Abstract. We propose a new subgradient-type method for minimizing extremely large-scale nonsmooth convex functions over "simple" domains. The characteristic features of the method are (a) the possibility to adjust the scheme to the geometry of the feasible set, thus allowing to get (nearly) dimension-independent (and nearly optimal in the large-scale case) rate-of-convergence results for minimization of a convex Lipschitz continuous function over a Euclidean ball, a standard simplex, and a spectahedron (the set of positive semidefinite symmetric matrices, of given size, with unit trace); (b) flexible handling of accumulated information, allowing for tradeoff between the level of utilizing this information and iteration's complexity. We present extensions of the scheme for the cases of minimizing non-Lipschitzian convex objectives, finding saddle points of convex-concave functions and solving variational inequalities with monotone operators. Finally, we report on encouraging numerical results of experiments with test problems of dimensions up to 66,000.

1 Introduction

With the success of Interior Point Methods (IPMs) to solve nonlinear convex optimization problems came also the realization that these methods have their limitations when encountering problems with design dimension n of order 10^4 – 10^5 or more. Indeed, when n is of this size, the arithmetic cost of an iteration of an IPM, being at least quadratic in n, becomes prohibitively large. The unavoidable conclusion is that for very large-scale problems, we can only use *simple* methods with linear in n arithmetic cost of an iteration. It follows also that we cannot utilize anymore our a priori knowledge of the analytical structure of the problem since, for the time being, all known ways to utilize this knowledge result in at least quadratic in n arithmetic cost of an iteration. This observation implies the second unavoidable conclusion: we are enforced to restrict ourselves to "black-box-oriented" methods – those using at each iteration function values and (sub)gradients only. In Convex Optimization, just two types of "cheap" black-box-oriented optimization techniques are known:

- techniques for *unconstrained* minimization of *smooth* convex functions (Gradient Descent, Conjugate Gradients, quasi-Newton methods with restricted memory, etc.);
- various subgradient-type techniques for constrained and/or *nonsmooth* convex programs.

A. Ben-Tal, A. Nemirovski: MINERVA Optimization Center, Faculty of IE&M, Technion – Israel Institute of Technology, Israel. e-mail: abental@ie.technion.ac.il; nemirovs@ie.technion.ac.il

* This research was supported by the Technion Fund for Promotion of Research

In this paper we propose a new subgradient-type method – *Non-Euclidean restricted Memory Level* (NERML) – adhering to the above restrictions, and aimed at solving very large-scale convex nonsmooth optimization problems in the form

$$\min_x \{f(x) : x \in X\}, \tag{1}$$

where X is a convex compact set in $\mathbf{R}^n$ and f is a Lipschitz continuous convex function with $X \subseteq \mathrm{Dom}\, f$. To get an impression of the performance of NERML, we list some of the results obtained later in this paper:

A. For $X = B_n$ (the unit Euclidean ball in $\mathbf{R}^n$) and when f is Lipschitz continuous, with constant L w.r.t. the Euclidean norm $\|\cdot\|_2$, for every $\epsilon > 0$ NERML finds an ϵ-solution of (1) (i.e., a point $x_\epsilon \in X$ such that $f(x_\epsilon) - \min_X f \le \epsilon$) in no more than $O(1)\frac{L^2}{\epsilon^2}$ iterations[1)], with a single computation of the value and a subgradient of f and $O(1)n$ additional arithmetic operations per iteration.

B. For $X = \Delta_n \equiv \{x \in \mathbf{R}^n_+ : \sum_i x_i = 1\}$ and when f is Lipschitz continuous, with constant L w.r.t. the norm $\|x\|_1 = \sum_i |x_i|$, for every $\epsilon > 0$ NERML finds an ϵ-solution to (1) in no more than $O(1)\frac{L^2 \ln(n)}{\epsilon^2}$ iterations of the same complexity as in A.

In the context of extremely large-scale optimization, the good news reported by these results is that the NERML algorithm is simple and its rate of convergence is (nearly) independent of the dimension n of the problem. A bad news is that the rate of convergence is rather slow – sublinear. The latter fact, however, is a "law of nature" rather than a shortcoming of the algorithm. Indeed, it is known [10, 1] that in the "large-scale case", specifically, $n \ge \frac{L^2}{\epsilon^2}$, in every one of the situations A, B, *no* "black-box-oriented" optimization method[2)] is capable to minimize within accuracy ϵ *all* convex objectives from the corresponding family in less than $O(1)\frac{L^2}{\epsilon^2}$ steps. It follows that *in large-scale cases of A and B, the NERML algorithm possesses the best possible (or nearly so) rate of convergence.*

By itself, the outlined optimality of the NERML algorithm, being an attractive theoretical property, should not be overestimated. First, this property is shared by many well-known simple optimization techniques. For example, in the case of A it is possessed by the simple Subgradient Descent method ([13, 12]; for a comprehensive overview, see [5]), by bundle-Level and many other bundle algorithms (see [6, 9, 3, 7, 14, 8, 4] and references therein), and by several analytic center cutting plane methods [11]. In the case of B, the corresponding nearly dimension-independent rate of convergence is shared by the $\|\cdot\|_1$-Mirror Descent ([10]; for a more comprehensive presentation, see [1])[3)].

[1)] From now on, all $O(1)$'s are appropriate positive absolute constants.

[2)] A method which collects information on a particular objective f by computing values and subgradients of f at subsequent search points, with the next search point being built solely on the basis of information collected at the previous points.

[3)] In fact, there exists a natural spectrum of cases "linking" A and B and sharing the properties of the "endpoints", specifically the case when $X = \{x \in \mathbf{R}^n : \|x\|_p \le 1\}$ for certain $p \in [1, 2]$, and f is convex and Lipschitz continuous with constant L w.r.t. $\|\cdot\|_p$, where $\|x\|_p = (\sum_i |x_i|^p)^{1/p}$. In these cases, appropriate versions of Mirror Descent [10] find ϵ-solution in no more than $O(1)\frac{L^2 \min[\frac{p}{p-1}, \ln n]}{\epsilon^2}$ steps, and no black-box-oriented method can solve all problems under consideration in less than $O(1)\frac{L^2}{\epsilon^2}$ steps, provided that $n > \frac{L^2}{\epsilon^2}$.

Second, the above "optimal" rate of convergence is very poor, so that *by itself* it promises nearly nothing good. What we would like to have, is *a method with rate of convergence which is guaranteed never to be worse, and "typically" is much better than the aforementioned "optimal" rate.* How much can be achieved in this direction, this is clearly demonstrated by comparing bundle methods to the Subgradient Descent. In the case of A, all these methods share the same theoretical rate of convergence – inaccuracy after t steps is at most $O(1)L/\sqrt{t}$. However, in practice the bundle methods outperform the Subgradient Descent by far. The reason is that the Subgradient Descent is "memoryless" – at every step t, the method operates with a linear "model" $f(x_t) + (x - x_t)^T f'(x_t)$ of the objective, with all previous information "compressed" in the current search point x_t. In contrast to this, at a step t in a bundle method one operates with a richer model $f^t(x) = \max\limits_{\tau \in I(t)} [f(x_\tau) + (x - x_\tau)^T f'(x_\tau)]$ of the objective, where $I(t)$ is a certain (perhaps, "large") set of indices $\tau \leq t$; better utilization of accumulated information usually results in better convergence. Apart from the difference in the use of "memory", bundle methods and Subgradient Descent are of the same kind, in the sense that they are intrinsically "linked" to the specific "Euclidean ball" geometry of case A. For problems with essentially different geometry (e.g., in the case of B), no dimension-independent rate-of-convergence results for these methods are known, and in fact the practical behaviour of bundle methods in the large-scale case may become pretty poor.

The NERML algorithms we are about to develop are in the same relation to the Mirror Descent as the bundle methods are to Subgradient Descent. Same as the Mirror Descent algorithms, the NERML scheme can be adjusted, to some extent, to the geometry of the domain X of a convex minimization problem (1); same as a bundle method, the NERML is a method "with memory". The essence of the difference between the usual bundle method and NERML can be seen from the following rough description of a step:

- In a bundle method, the next search point x_{t+1} is given by

$$x_{t+1} = \operatorname*{argmin}_x \left\{ \frac{1}{2}\|x - p_t\|_2^2 : x \in X,\, A_t x \leq b_t \right\} \tag{2}$$

  where p_t is the current *prox-center*, and the linear inequalities $A_t x \leq b_t$ ($[A_t, b_t]$ is the current *bundle*) are such that outside of the set $X_t = \{x \in X,\, A_t x \leq b_t\}$ the objective f is $\geq \ell_t$, where ℓ_t is the current *level*. Various versions of bundle methods differ from each other by rules, explicit or implicit, for updating the prox-center, the bundle and the level.
- In a NERML method, x_{t+1} is given by

$$x_{t+1} = \operatorname*{argmin}_x \left\{ \omega(x) - x^T \nabla\omega(p_t) : x \in X,\, A_t x \leq b_t \right\} \tag{3}$$

  where $\omega(x)$ is a continuously differentiable strongly convex function on X, and p_t, $[A_t, b_t]$ (and the "implicitly present" level ℓ_t) are similar to those in (2). Various versions of the NERML methods differ from each other mainly by the choice of $\omega(\cdot)$, as well as by rules for updating the prox-center, the bundle and the level.

It is immediately seen that (2) is a particular case of (3) corresponding to $\omega(x) = \frac{1}{2}x^T x$. What allows to adjust NERML to the geometry of X, is the freedom in the choice of ω. For example, it turns out that in the case of A a good choice of this function is $\omega(x) = \frac{1}{2}x^T x$, so that in this case NERML becomes a usual bundle method. In contrast to this, in the case of B the latter "Euclidean" choice of ω does not result in a nearly dimension-independent rate of convergence and does not exhibit good practical performance in the large-scale case. A good choice of ω in the case of B here is, e.g., the regularized entropy $\omega(x) = \sum_{i=1}^{n}(x_i + \delta n^{-1})\ln(x_i + \delta n^{-1})$ with, say, δ =1.e-16. Other elements of the NERML, that is, the rules for updating the prox-center, the bundle and the level, are, essentially, the same as in the Restricted Memory Prox-Level method of Kiwiel [4] (which, in turn, is a significant improvement of the Prox-Level method proposed in [8]). In particular, the NERML scheme allows for full control of the cardinality of the bundle (the column size of the matrix $[A_t, b_t]$); this control (essentially the same as the one in Kiwiel's method) allows for tradeoff between the complexity of solving the auxiliary problems (3) and the utilization of the information accumulated so far.

The rest of the paper is organized as follows. In Section 2, we present the generic NERML algorithm for solving problems (1) with compact convex domain X and Lipschitz continuous convex objective f and carry out the complexity analysis of the algorithm. Further, we explain how to adjust the algorithm to the aforementioned cases A, B and the "semidefinite analogy" of case B – the case C where the domain X of (1) is a *spectahedron* (the part of the positive semidefinite cone in the space of symmetric matrices of a given dimension cut off the cone by the constraint $\mathrm{Tr}(x) = 1$), and the convex objective f is Lipschitz continuous with constant L w.r.t. the norm $|x|_1 = \|\lambda(x)\|_1$ (x is symmetric matrix, $\lambda(x)$ is the vector of eigenvalues of x). It turns out that as far as NERML scheme is concerned, the geometry of the spectahedron is completely similar to the one of the simplex, so that the complexity results in the case C are exactly the same as in the case of B. In Section 3, we explain how to solve the auxiliary problems (3) and address several other implementation issues. Section 4 is devoted to several extensions of the NERML scheme, specifically, to problems (1) with non-Lipschitzian convex objectives, to finding saddle points of convex-concave functions and to solving variational inequalities with monotone operators. In the concluding Section 5, we report on a number of preliminary numerical experiments with the NERML algorithm as applied to large-scale problems (1) of various dimensions reaching up to 66,000. The applications we are considering are the relaxations of Uncapacitated Facility Location problems (3,000 and 6,000 variables), and 2D Tomography Image Reconstruction problems (16,641 and 66,049 variables). To the best of our judgement, the results we have obtained are quite encouraging and, in particular, demonstrate the importance of adjusting the method to problem's geometry.

Notation. • $B_n = \{x \in \mathbf{R}^n : \|x\|_2 \le 1\}$ is the unit Euclidean ball, $\Delta_n = \{x \in \mathbf{R}^n : x \ge 0, \sum_i x_i = 1\}$ is the standard "flat" simplex in $\mathbf{R}^n$, $\Delta_n^+ = \{x \in \mathbf{R}^n : x \ge 0, \sum_i x_i \le 1\}$, is the standard full-dimensional simplex.

• $\mathbf{S}^n$ is the space of $n \times n$ symmetric matrices equipped with the Frobenius inner product $\langle A, B\rangle = \mathrm{Tr}(AB)$. For $A \in \mathbf{S}^n$, $\lambda(A)$ is the vector of eigenvalues of A (taken with their multiplicities and arranged in the non-ascending order), and the relation $A \succ 0$

($A \succeq 0$) means that A is positive (semi)definite. $\Sigma_n = \{x \in \mathbf{S}^n : x \succeq 0, \mathrm{Tr}(x) = 1\}$ and $\Sigma_n^+ = \{x \in \mathbf{S}^n : x \succeq 0, \mathrm{Tr}(x) \leq 1\}$ are the "flat" and the full-dimensional spectahedrons, respectively.

• For a convex lower semicontinuous function f, its *subgradient mapping* $x \mapsto \partial f(x)$ is defined as follows: at a point x from the relative interior of the domain X of f, $\partial f(x)$ is comprised of all subgradients g of f at x which are in the linear span of $X - X$. For a point $x \in X \backslash \mathrm{rint}\, X$, the set $\partial f(x)$ is comprised of all vectors g, if any, such that there exist $x_i \in \mathrm{rint}\, X$ and $g_i \in \partial f(x_i)$, $i = 1, 2, ...$ with $x = \lim\limits_{i\to\infty} x_i$, $g = \lim\limits_{i\to\infty} g_i$. Finally, $\partial f(x) = \emptyset$ for $x \notin X$. Note that with this definition for a convex function f which is Lipschitz continuous, with constant L w.r.t. a norm $\|\cdot\|$, on $X = \mathrm{Dom}\, f$, for every $x \in X$ the set $\partial f(x)$ is nonempty, and

$$\xi \in \partial f(x) \Rightarrow |\xi^T h| \leq L\|h\| \ \forall h \in \mathrm{Lin}(X - X). \tag{4}$$

In other words, if $\mathrm{int}\, X \neq \emptyset$ and $\xi \in \partial f(x)$, then $\|\xi\|_* \leq L$, where

$$\|\xi\|_* = \max_x \left\{\xi^T x : \|x\| \leq 1\right\} \tag{5}$$

is the norm conjugate to $\|\cdot\|$. If X is "flat" ($\mathrm{Lin}(X - X) \neq \mathbf{R}^n$) and $\xi \in \partial f(x)$, then $\|\xi + \delta\|_* \leq L$ for a proper "correction" $\delta \in [\mathrm{Lin}(X - X)]^\perp$.

To streamline the exposition, all proofs are moved to Appendix.

2 The basic NERML algorithm

2.1 The algorithm

The purpose. The basic NERML method is aimed at solving optimization problem (1) which is assumed to possess the following properties:

(P.1): X is a nonempty convex compact subset of $\mathbf{R}^n$;

(P.2): f is convex and Lipschitz continuous on X.

To quantify assumption (P.2), we fix a norm $\|\cdot\|$ on $\mathbf{R}^n$ and associate with f the Lipschitz constant of $f\big|_X$ w.r.t. the norm $\|\cdot\|$:

$$L_{\|\cdot\|}(f) = \min\{L : |f(x) - f(y)| \leq L\|x - y\| \ \forall x, y \in X\}.$$

Finally, we assume that

(P.3) We have access to a First Order oracle which, given as input a point $x \in X$, returns the value $f(x)$ and a subgradient $f'(x) \in \partial f(x)$ of f at x.

Note that

$$|h^T f'(x)| \leq L_{\|\cdot\|}(f)\|h\| \quad \forall h \in \mathrm{Lin}(X - X), \tag{6}$$

see (4).

The setup for the generic NERML method is given by the following triplet: the set X, a norm $\|\cdot\|$ *and* a continuously differentiable function $\omega(x) : X \to \mathbf{R}$ which is *strongly convex* on X, with parameter $\kappa > 0$, w.r.t. the norm $\|\cdot\|$:

$$\omega(y) \geq \omega(x) + (y-x)^T \nabla\omega(x) + \frac{\kappa}{2}\|y-x\|^2 \quad \forall x, y \in X. \tag{7}$$

To make the NERML algorithm implementable, the pair $(X, \omega(\cdot))$ should be simple enough to allow for rapid solving of auxiliary problems of the form

$$x[p] = \operatorname*{argmin}_{x \in X}[\omega(x) + p^T x] \tag{8}$$

We will be especially interested in the following *standard setups*:

1. "Ball setup": X is a convex compact subset of the unit Euclidean ball B_n, $\|\cdot\| = \|\cdot\|_2$, $\omega(x) = \frac{1}{2}x^T x$;
2. "Simplex setup": X is a convex compact subset of the standard "full-dimensional" simplex

   $$\Delta_n^+ = \{x \in \mathbf{R}^n : x \geq 0, \sum_i x_i \leq 1\},$$

   $\|\cdot\| = \|\cdot\|_1$, and $\omega(x)$ is the "regularized entropy"

   $$\omega(x) = \sum_{i=1}^n (x_i + \delta n^{-1}) \ln(x_i + \delta n^{-1}) : \Delta_n^+ \to \mathbf{R}, \tag{9}$$

   where $\delta \in (0, 1)$ is a fixed "regularization parameter";
3. "Spectahedron setup": this setup deals with the special case when the underlying "universe" is the space $\mathbf{S}^n$ of $n \times n$ symmetric matrices rather than $\mathbf{R}^n$; $\mathbf{S}^n$ is equipped with the Frobenius inner product $\langle A, B\rangle = \mathrm{Tr}(AB)$. The *spectahedron* is the set in $\mathbf{S}^n$ defined as

   $$\Sigma_n^+ = \{x \in \mathbf{S}^n : x \succeq 0, \mathrm{Tr}(x) \leq 1\}$$

   (we are using lowercase notation for the elements of $\mathbf{S}^n$ in order to be consistent with the rest of the text). In the spectahedron setup, X is a convex compact subset of Σ_n^+, $\|\cdot\|$ is the norm

   $$|x|_1 \equiv \|\lambda(x)\|_1$$

   on $\mathbf{S}^n$, where $\lambda(x)$ stands for the vector of eigenvalues of a symmetric matrix x, and the function $\omega(x)$ is the "regularized matrix entropy"

   $$\omega(x) = \mathrm{Tr}((x + \delta n^{-1} I_n) \ln(x + \delta n^{-1} I_n)) : \Sigma_n^+ \to \mathbf{R}, \tag{10}$$

   where $\delta \in (0, 1)$ is a fixed regularization parameter.

   Note that the simplex setup is, in fact, a particular case of the Spectahedron one corresponding to the case when X is comprised of *diagonal* positive semidefinite matrices.

One can verify that for these setups, $\omega(\cdot)$ is indeed continuously differentiable on X and satisfies (7) with $\kappa = O(1)$. More specifically, one has

$$\kappa = \begin{cases} 1, & \text{ball setup} \\ (1+\delta)^{-1}, & \text{simplex setup} \\ 0.5(1+\delta)^{-1}, & \text{spectahedron setup,} \end{cases} \tag{11}$$

see Appendix.

The generic algorithm NERML works as follows.

A. The algorithm generates a sequence of search points, all belonging to X, where the First Order oracle is called, and at every step builds the following entities:

1. the best value of f found so far, along with the corresponding search point; the latter is treated as the current approximate solution built by the method;
2. a (valid) lower bound on the optimal value of the problem.

B. The execution is split in subsequent *phases*. Phase s, $s = 1, 2, \ldots$, is associated with a *prox-center* $c_s \in X$ and a *level* $\ell_s \in \mathbf{R}$ such that

- when starting the phase, we already know $f(c_s)$, $f'(c_s)$;
- $\ell_s = f_s + \lambda(f^s - f_s)$, where
  - f^s is the best value of the objective known at the time when the phase starts;
  - f_s is the lower bound on f_* we have at our disposal when the phase starts;
  - $\lambda \in (0, 1)$ is a parameter of the method.

The prox-center c_1 corresponding to the very first phase can be chosen in X in an arbitrary fashion. We start the entire process with computing f, f' at this prox-center, which results in

$$f^1 = f(c_1)$$

and set

$$f_1 = \min_{x \in X}[f(c_1) + (x - c_1)^T f'(c_1)],$$

thus getting the initial lower bound on f_*.

C. The description of a particular phase s is as follows. Let

$$\omega_s(x) = \omega(x) - (x - c_s)^T \nabla\omega(c_s);$$

note that (7) implies that

$$\omega_s(y) \geq \omega_s(x) + (y - x)^T \nabla\omega_s(x) + \frac{\kappa}{2}\|y - x\|^2 \quad \forall x, y \in X. \tag{12}$$

Note also that $c_s = \operatorname{argmin}_{x \in X} \omega_s(\cdot)$.

At phase s, the search points $x_t = x_{t,s}$, $t = 1, 2, \ldots$ are generated according to the following rules:

1. When generating x_t, we already have in our disposal x_{t-1}, a valid lower bound $\tilde{f}_t = \tilde{f}_{s,t}$ on f_* and a *localizer* X_{t-1} – a convex compact set $X_{t-1} \subseteq X$ such that

$$\begin{aligned} &(a_{t-1}) \quad x \in X \backslash X_{t-1} \Rightarrow f(x) > \ell_s; \\ &(b_{t-1}) \quad x_{t-1} \in \operatorname*{argmin}_{X_{t-1}} \omega_s. \end{aligned} \tag{13}$$

   Here $x_0 = c_s$, $\tilde{f}_0 = f_s$ and, say, $X_0 = X$, which ensures (13.a_0–b_0).
2. To update (x_{t-1}, X_{t-1}) into (x_t, X_t), we solve the auxiliary problem

$$\tilde{f} = \min_x \left\{ g_{t-1}(x) \equiv f(x_{t-1}) + (x - x_{t-1})^T f'(x_{t-1}) : x \in X_{t-1} \right\}. \tag{L_{t-1}}$$

   Observe that the quantity

$$\hat{f} = \min[\tilde{f}, \ell_s]$$

   is a lower bound on f_*. Indeed, in $X \backslash X_{t-1}$ we have $f(x) > \ell_s$ by (13.a_{t-1}), while on X_{t-1} we have $f(x) \geq \tilde{f}$ due to the inequality $f(x) \geq g_{t-1}(x)$ given by the convexity of f. Thus, $f(x) \geq \min[\ell_s, \tilde{f}]$ everywhere on X, so that the quantity

$$\tilde{f}_t = \max[\tilde{f}_{t-1}, \min[\ell_s, \tilde{f}]]$$

   is a lower bound on f_*.

Our subsequent actions depend on the results obtained when solving (L_{t-1}), specifically:

(a) In the case of "significant progress in the lower bound", specifically,

$$\tilde{f}_t \geq \ell_s - \theta(\ell_s - f_s), \tag{14}$$

where $\theta \in (0, 1)$ is a parameter of the method, we terminate phase s, set

$$f^{s+1} = \min[f^s, \min_{0 \leq \tau \leq t-1} f(x_\tau)], \quad f_{s+1} = \tilde{f}_t$$

and pass to phase $s + 1$. The prox-center c_{s+1} for the new phase can be chosen in X in an arbitrary fashion.

(b) In the case of no significant progress in the lower bound, we solve the optimization problem

$$\min_x \{\omega_s(x) : x \in X_{t-1},\ g_{t-1}(x) \leq \ell_s\}. \tag{P_{t-1}}$$

This problem is feasible, since otherwise $\tilde{f} = \infty$, whence $\tilde{f}_t = \ell_s$, and therefore (14) would take place, which in the case of (b) is impossible. When solving (P_{t-1}), we get the optimal solution x_t of this problem and compute $f(x_t)$, $f'(x_t)$. It is possible that

- (b.1) We get a "significant" progress in the objective, specifically,

$$f(x_t) - \ell_s \leq \theta(f^s - \ell_s). \tag{15}$$

  In this case, we again terminate the phase, set

$$f^{s+1} = \min[f^s, \min_{0 \leq \tau \leq t} f(x_t)], \quad f_{s+1} = \tilde{f}_t$$

  and pass to phase $s + 1$. The prox-center c_{s+1} for the new phase, same as above, can be chosen in X in an arbitrary fashion.

- (b.2) When (P_{t-1}) is feasible and (15) is *not* valid, we continue the phase s, choosing as X_t an arbitrary convex compact set such that

$$\begin{aligned} \underline{X}_t &\equiv \{x \in X_{t-1} : g_{t-1}(x) \le \ell_s\} \subseteq X_t \subseteq \overline{X}_t \\ &\equiv \{x \in X : (x - x_t)^T \nabla \omega_s(x_t) \ge 0\}, \end{aligned} \tag{16}$$

see Fig. 2.1.

Note that in the case of (b.2) problem (P_{t-1}) is feasible and x_t is its optimal solution; it follows that

$$\emptyset \neq \underline{X}_t \subseteq \overline{X}_t,$$

so that (16) indeed allows to choose X_t. Moreover, every choice of X_t compatible with (16) ensures $(13.a_t)$ and $(13.b_t)$; the first relation is clearly ensured by the left inclusion in (16) combined with $(13.a_{t-1})$ and the fact that $f(x) \ge g_{t-1}(x)$, while the second relation $(13.b_t)$ follows from the right inclusion in (16) due to the convexity of $\omega_s(\cdot)$.

The summary of the NERML algorithm is as follows:

NERML algorithm

Parameters: $\lambda \in (0, 1), \theta \in (0, 1), \epsilon > 0$.

Initialization: Choose $c_1 \in X$, compute $f(c_1)$ and $f'(c_1) \in \partial f(c_1)$ and set

$$f^1 = f(c_1), \quad f_1 = \min_{x \in X}[f(c_1) + (x - c_1)^T f'(c_1).$$

Phase s ($s = 1, 2, ...$): If $f^s - f_s \le \epsilon$ (required tolerance), terminate. Otherwise set $\ell_s = f_s + \lambda(f^s - f_s)$, start inner iterations:

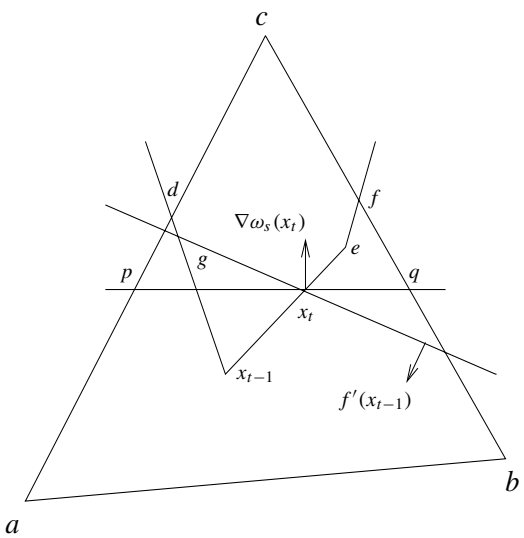

Fig. 1. Geometry of a step. X:abc; X_{t-1}: $dx_{t-1}efc$; $\{x : g_t(x) = \ell_s\}$: gx_t; $\{x : (x - x_t)^T \nabla w_s(x_t) = 0\}$: pq; $\underline{X}_t$: dgx_tefc; $\overline{X}_t$: pqc. X_t should be in-between $\underline{X}_t$ and $\overline{X}_t$, e.g., dgx_tqc.

Initialization: $x_0 = c_s$, $\tilde{f}_0 = f_s$, $X_0 = X$;
Inner iteration t (t = 1, 2, ...):
Compute

$$\tilde{f} = \min_{x \in X_{t-1}} \left[f(x_{t-1}) + (x - x_{t-1})^T f'(x_{t-1}) \right],$$
$$\tilde{f}_t = \max[\tilde{f}_{t-1}, \min[\ell_s, \tilde{f}]].$$

If $\tilde{f}_t \geq \ell_s - \theta(\ell_s - f_s)$,
set

$$f^{s+1} = \min\{f^s, \min_{0 \leq \tau \leq t-} f(x_\tau)\}, \ f_{s+1} = \tilde{f}_t,$$

choose $c_{s+1} \in X$ and pass to phase $s + 1$,
else
compute

$$x_t = \operatorname*{argmin}_{x \in X_{t-1}} \left\{ \omega(x) - (x - c_s)^T \nabla\omega(c_s) : f(x_{t-1} + (x - x_{t-1})^T f'(x_{t-1}) \leq \ell_s \right\}$$

and $f(x_t)$, $f'(x_t) \in \partial f(x_t)$.
If $f(x_t) - \ell_s \leq \theta(f^s - \ell_s)$,
choose $c_{s+1} \in X$ and pass to phase $s + 1$,
else
choose X_t as any convex compact set satisfying (16)
and pass to step $t + 1$ of phase s.

2.2 Convergence Analysis

Let us define *s-th gap* as the quantity

$$\epsilon_s = f^s - f_s$$

By its origin, the gap is nonnegative, nonincreasing in s, and is a valid upper bound on the inaccuracy, in terms of the objective, of the approximate solution z^s we have at the beginning of phase s (i.e., $f(z^s)$ is the smallest value of the objective found so far).

The convergence and the complexity properties of the NERML algorithm are given by the following result. (The proof is given in the Appendix.)

Theorem 2.1 (i) *The number N_s of oracle calls at a phase s is bounded from above as follows:*

$$N_s \leq \frac{4\Omega L^2_{\|\cdot\|}(f)}{\theta^2(1-\lambda)^2\kappa\epsilon_s^2}, \tag{17}$$

where

$$\Omega = \max_{x,y \in X} [\omega(y) - \omega(x) - (y - x)^T \nabla\omega(x)]. \tag{18}$$

(ii) *Consequently, for every $\epsilon > 0$, the total number of oracle calls, before the first phase s for which $\epsilon_s \leq \epsilon$ is started (i.e., before an ϵ-solution to the problem is built) does not exceed*

$$N(\epsilon) = c(\theta, \lambda)\frac{\Omega L^2_{\|\cdot\|}(f)}{\kappa\epsilon^2} \tag{19}$$

with an appropriate $c(\theta, \lambda)$ depending solely and continuously on $\theta, \lambda \in (0, 1)$[4)].

2.3 Optimality of NERML in the case of standard setups

Let us look what the complexity analysis says in the case of the standard setups.

Ball setup and optimization over the ball. We recall that for the case of the ball setup the parameter of strong convexity of $\omega(\cdot)$ is $\kappa = 1$. Also, it is immediately seen that here

$$\Omega = \frac{1}{2}D^2_{\|\cdot\|_2}(X),$$

where $D_{\|\cdot\|_2}(X) = \max\limits_{x,y\in X} \|x - y\|_2$ if the $\|\cdot\|_2$-diameter of X. Since with the ball setup X is a subset of the unit Euclidean ball, we conclude that $\Omega \leq 2$. Thus, (19) becomes

$$N(\epsilon) \leq 2c(\theta, \lambda)\frac{L^2_{\|\cdot\|_2}(f)}{\epsilon^2}. \tag{20}$$

Now let $L > 0$, and let $\mathcal{P}_{\|\cdot\|_2,L}(X)$ be the family of all convex problems (CP) with objective functions which are Lipschitz continuous on X with constant L w.r.t. $\|\cdot\|_2$. It is known [10] that *if X is the unit n-dimensional Euclidean ball and $n \geq \frac{L^2}{\epsilon^2}$, then the information-based complexity of the family $\mathcal{P}_{\|\cdot\|_2,L}(X)$ (the minimal number of calls to the First Order oracle in which a black-box-oriented method can solve every problem from the family within accuracy ϵ) is at least $O(1)\frac{L^2}{\epsilon^2}$*. Comparing this result with (20), we arrive at the following conclusion on the optimality of NERML with the ball setup:

> *If X is the unit n-dimensional Euclidean ball, then the complexity of the family $\mathcal{P}_{\|\cdot\|_2,L}(X)$ w.r.t. the NERML algorithm with the ball setup in the "large-scale case" (the one of $n \geq \frac{L^2}{\epsilon^2}$) coincides (within a factor depending solely on θ, λ) with the information-based complexity of the family.*

Simplex setup and minimization over the simplex. Here one has $\kappa = (1 + \delta)^{-1}$, where $\delta \in (0, 1)$ is the regularization parameter for the entropy, and

$$\Omega \leq (1 + \delta)\left[1 + \ln\left(\frac{n(1 + \delta)}{\delta}\right)\right]. \tag{21}$$

(see Appendix).

[4)] The specific formula of $c(\theta, \lambda)$ is given at the end of the proof of Theorem 2.1 in the Appendix.

We see that *for the simplex setup, Ω is of order of* $\ln n$*, provided that δ is not extremely small.* E.g., when δ =1.e-16 is the "machine zero" (so that for all computational purposes, our regularized entropy is, essentially, the same as the usual entropy), we have $\Omega \le 37 + \ln n$, whence $\Omega \le 6 \ln n$, provided that $n \ge 1000$.

With the above bounds for κ and Ω, the complexity bound (19) becomes

$$N(\epsilon) \le \widehat{c}(\theta, \lambda) \frac{L^2_{\|\cdot\|_1}(f) \ln n}{\epsilon^2} \tag{22}$$

(provided that $\delta \ge$ 1.e-16). On the other hand, for the family $\mathcal{P}_{\|\cdot\|_1, L}(X)$ of all convex problems (CP) with objective functions which are Lipschitz continuous, with constant L w.r.t. $\|\cdot\|_1$. It is known that *if X is the n-dimensional simplex Δ_n (or the full-dimensional simplex Δ_n^+) and $n \ge \frac{L^2}{\epsilon^2}$, then the information-based complexity of the family $\mathcal{P}_{\|\cdot\|_1, L}(X)$ is at least $O(1)\frac{L^2}{\epsilon^2}$* (see [1]). Comparing this result with (20), we conclude that

> *If X is the n-dimensional simplex Δ_n (or the full-dimensional simplex Δ_n^+), then the complexity of the family $\mathcal{P}_{\|\cdot\|_1, L}(X)$ w.r.t. the NERML algorithm with the simplex setup, in the "large-scale case" $n \ge \frac{L^2}{\epsilon^2}$, coincides within a factor of order of* $\ln n$ *with the information-based complexity of the family.*

Spectahedron setup and large-scale semidefinite optimization. All the conclusions we have made for the case of the simplex setup and $X = \Delta_n$ (or $X = \Delta_n^+$) remain valid in the case of the spectahedron setup and X defined as the set of all *block-diagonal* matrices of a given block-diagonal structure contained in $\Sigma_n^+ = \{x \in \mathbf{S}^n : x \succeq 0, \mathrm{Tr}(x) \le 1\}$ (or contained in Σ_n).

We see that *with every one of our standard setups, the* NERML *algorithm under appropriate conditions possesses dimension independent (or nearly dimension independent) complexity bound and, moreover, is nearly optimal in the sense of Information-based complexity theory, provided that the dimension is large.*

Why the standard setups? "The contribution" of $\omega(\cdot)$ to the performance estimate (19) is in the factor $\Theta = \frac{\Omega}{\kappa}$; the smaller it is, the better. In principle, given X and $\|\cdot\|$, we could adjust $\omega(\cdot)$ so as to minimize Θ. The standard setups are given by a kind of such optimization for the cases when X is the ball and $\|\cdot\| = \|\cdot\|_2$ ("the ball case"), when X is the simplex and $\|\cdot\| = \|\cdot\|_1$ ("the simplex case"), and when X is the spectahedron and $\|\cdot\| = |\cdot|_1$ ("the spectahedron case"), respectively. We did not try to solve the corresponding variational problems exactly; however, it can be proved in all three cases that the value of Θ we have reached (i.e., $O(1)$ in the ball case and $O(\ln n)$ in the simplex and the spectahedron cases) cannot be reduced by more than an absolute constant factor. Note that in the simplex case the (regularized) entropy is not the only reasonable choice; similar complexity results can be obtained for, say, $\omega(x) = \sum_i x_i^{p(n)}$ or $\omega(x) = \|x\|^2_{p(n)}$ with $p(n) = 1 + O\left(\frac{1}{\ln n}\right)$.

3 Implementation issues

3.1 Solving auxiliary problems (L_t), (P_t).

The major issue in the implementation of the NERML algorithm is *how to solve efficiently the auxiliary problems* (L_t), (P_t). Formally, these problems are of the same design dimension as the problem of interest; what then is gained by reducing the solution of a *single* large-scale problem (CP) to a long series of auxiliary problems of the same dimension? To answer this crucial question, observe first that *we have control on the complexity of the domain X_t which,* up to a single linear constraint, *is the feasible domain of* (L_t), (P_t). Indeed, assume that X_{t-1} is a part of X given by a finite list of linear inequalities. Then the sets $\underline{X}_t$ and $\overline{X}_t$ in (16) are also cut off X by finitely many linear inequalities, so that we may enforce X_t to be cut off X by finitely many linear inequalities as well. Moreover, we have full control of the number of inequalities in the list. Indeed,

A. Setting all the time $X_t = \overline{X}_t$, we ensure that X_t is cut off X by a *single* linear inequality;
B. Setting all the time $X_t = \underline{X}_t$, we ensure that X_t is cut off X by t linear inequalities (so that the larger is t, the "more complicated" is the description of X_t);
C. We can choose something in-between the above extremes. Assume that we have chosen a positive integer m and we want to work with X_t's cut off X by at most m linear inequalities. In this case, we could use the policy B at the initial steps of a phase, until the number of linear inequalities in the description of X_{t-1} reaches the maximum allowed value m, so that

$$X_{t-1} = \{x \in X : h_j^{t-1}(x) \le 0,\ j = 1, \ldots, m\}.$$

At step t, we should choose X_t in-between the two sets $\underline{X}_t$, $\overline{X}_t$, where

$$\begin{aligned} \underline{X}_t &= \{x \in X : h_1^{t-1}(x) \le 0, \ldots, h_m^{t-1}(x) \le 0, h_{m+1}^{t-1}(x) \le 0\}, \\ \overline{X}_t &= \{x \in X : h_m^t(x) \le 0\}, \\ h_{m+1}^{t-1}(x) &\equiv g_{t-1}(x) - \ell_s, \\ (*) \quad h_m^t(x) &\equiv (x_t - x)^T \nabla \omega_s(x_t). \end{aligned} \tag{23}$$

To this end, we can set

$$X_t = \{x \in X : h_j^t(x) \le 0,\ j = 1, \ldots, m\},$$

where $h_m^t(x)$ is given by (23.(∗)), and every one of the inequalities $h_j^t(x) \le 0$, $j = 1, \ldots, m-1$, is a convex combination of the inequalities $h_1^{t-1}(x) \le 0, \ldots, .h_{m+1}^{t-1}(x) \le 0, h_m^t(x) \le 0$.

The bottom line is: *we can always ensure that X_{t-1} is cut off X by at most m linear inequalities $h_j^{t-1}(x) \le 0$, $j = 1, \ldots, m$, where $m \ge 1$ is a desirable bound.* Consequently, we may assume that *the feasible set of* (P_{t-1}) *is cut off X by $m+1$ linear inequalities $h_j(x) \le 0$, $j = 1, \ldots, m+1$.* The crucial point is that *with this approach,*

we can reduce (L_{t-1}), (P_{t-1}) *to convex programs with at most* $m+1$ *decision variables.* Indeed, let us start with problem (P_{t-1}) and assume that it is strictly feasible:

$$\exists(\bar{x} \in \operatorname{rint} X) : h_j(\bar{x}) \le 0,\ j = 1, \ldots, m+1 \quad (\Leftrightarrow \underline{X}_t \cap \operatorname{rint} X \neq \emptyset).$$

By standard Lagrange Duality, the optimal value in (P_{t-1}) is equal to the one in its dual problem

$$\max_{\lambda \ge 0} L(\lambda), \quad L(\lambda) \equiv \min_{x \in X}[\omega_s(x) + \sum_{j=1}^{m+1} \lambda_j h_j(x)]. \tag{D_{t-1}}$$

Note that the objective in (D_{t-1}) is concave and "computable" at every given λ. Indeed, to compute the value $L(\lambda)$ and a supergradient $L'(\lambda)$ of L at a given λ is the same as to find the optimal solution x_λ to the optimization program

$$\min_{x \in X}[\omega_s(x) + \sum_{j=1}^{m+1} \lambda_j h_j(x)]; \tag{D[λ]}$$

after x_λ is found, we set

$$L(\lambda) = \omega_s(x_\lambda) + \sum_{j=1}^{m+1} \lambda_j h_j(x_\lambda), \quad L'(\lambda) = (h_1(x_\lambda), \ldots, h_{m+1}(x_\lambda))^T.$$

It remains to note that to solve (D[λ]) means to minimize over X a sum of $\omega(\cdot)$ and a linear function, and we have assumed that $(X, \omega(\cdot))$ is simple enough for problems of this type to be rapidly solved.

The summary of our observations is that (D[λ]) is a convex optimization program with $m+1$ decision variables, and we have in our disposal a First Order oracle for this problem, so that we can solve it efficiently, provided that m is not too large, e.g., by the Ellipsoid method. We can indeed enforce the latter requirement – m is in our full control!

After (D_{t-1}) is solved to high accuracy and we have in our disposal the corresponding maximizer λ_*, we can choose, as x_t, the point x_{λ_*}, since by the Lagrange Duality theorem the optimal solution x_t of (P_{t-1}) is among the optimal solutions to (D[λ_*]), and the set of the optimal solutions to the latter problem is a singleton, since ω_s is strongly convex.

It remains to understand how to solve (L_{t-1}) and how to ensure the strict feasibility of (P_{t-1}) (the latter is a sufficient condition for the above construction to work). Here again we can apply the Lagrange Duality. Indeed, assuming that (L_{t-1}) is strictly feasible (e.g., $X_{t-1} \cap \operatorname{rint} X \neq \emptyset$), we have

$$\begin{aligned} &\min_x \left\{ g_{t-1}(x) : x \in X_{t-1} = \{x \in X : h_j(x) \le 0,\ j = 1, \ldots, m\} \right\} \\ &\quad = \max_\lambda \left\{ L(\lambda) \equiv \min_{x \in X} \left[g_{t-1}(x) + \sum_{j=1}^{m} \lambda_j h_j(x) \right] : \lambda \ge 0 \right\}. \end{aligned}$$

Same as above, we have in our disposal a First Order oracle for $L(\cdot)$ and can therefore minimize the (low-dimensional) function L by the Ellipsoid or the bundle methods.

Now, what we want is just the optimal value, not an optimal solution, hence the fact that the objective in (L_{t-1}) is not strongly convex does not cause any difficulties. If the optimal value in (L_{t-1}) is $\geq \ell_s$, we must terminate the phase and hence do not need to solve (P_{t-1}) at all, otherwise the set $\underline{X}_t = \{x \in X_{t-1} : g_{t-1}(x) \leq \ell_s\}$ clearly intersects rint X (since X_{t-1} is assumed to possess this property) and we are in a good position to solve (P_{t-1}) via duality. Note also that in the latter case the set $X_t \supset \underline{X}_t$ intersects rint X (since $\underline{X}_t$ does so). Assuming that X_0 intersects rint X (which is for sure so when $X_0 = X$), we conclude that all the auxiliary problems to be solved are strictly feasible, and thus can be processed via duality.

When are the standard setups implementable? As we have seen, the possibility to implement the NERML algorithm depends on the ability to solve rapidly optimization problems of the form (8). Let us look at several important cases when this indeed is possible.

Ball setup. Here problem (8) becomes $\min\limits_{x\in X}\left[\frac{1}{2}x^Tx - p^Tx\right]$, or, equivalently, $\min\limits_{s\in X}\left[\frac{1}{2}\|x-p\|_2^2\right]$. We see that to solve (8) is the same as to *project* on X – to find the point in X which is as close as possible, in the usual $\|\cdot\|_2$-norm, to a given point p. This problem is easy to solve for several simple solids X, e.g.,

- a ball $\{x : \|x - a\|_2 \leq r\}$,
- a box $\{x : a \leq x \leq b\}$,
- the simplex $\Delta_n = \{x : x \geq 0, \sum\limits_i x_i = 1\}$.

In the first two cases, it takes $O(n)$ operations to compute the solution which is given by explicit formulas. The third case is a bit more involved: the projection is given by the relations $x_i = x_i(\lambda_*)$, where $x_i(\lambda) = \max[0, p_i - \lambda]$ and λ_* is the unique root of the equation

$$\sum_i x_i(\lambda) = 1.$$

The left hand side of this equation is nonincreasing and continuous in λ and, as it is immediately seen, its value varies from something ≥ 1 when $\lambda = \min\limits_i p_i - 1$ to 0 when $\lambda = \max\limits_i p_i$. It follows that one can easily approximate λ_* by bisection, and that it takes a moderate absolute constant number of bisection steps to compute λ_* (and thus – the projection) within the machine precision. The arithmetic cost of a bisection step is $O(n)$, and so the overall arithmetic complexity of finding the projection is also $O(n)$.

Simplex setup. Consider the two simplest cases:

S.A: X is the standard simplex Δ_n;
S.B: X is the standard full-dimensional simplex Δ_n^+.

Case S.A. When $X = \Delta_n$, problem (8) becomes

$$\min\left\{\sum_i (x_i+\sigma)\ln(x_i+\sigma) - p^Tx : x \geq 0, \sum_i x_i = 1\right\} \qquad [\sigma = \delta n^{-1}] \qquad (24)$$

It can be worked out that the solution to (24) is $x_i = x_i(\lambda_*)$, where

$$x_i(\lambda) = \max[\exp\{\widehat{p}_i - \lambda\} - \sigma, 0] \qquad [\widehat{p}_i = p_i - \min_j p_j] \tag{25}$$

and λ_* is the solution to the equation

$$\sum_i x_i(\lambda) = 1.$$

Here again the left hand side of the equation is nonincreasing and continuous in λ and its value varies from something which is ≥ 1 when $\lambda = -\sigma$ to something which is < 1 when $\lambda = \ln n$, hence we again can compute λ_* (and thus $x(\lambda_*)$) within machine precision in a moderate absolute constant number of bisection steps. As a result, the arithmetic cost of solving (24) is again $O(n)$.

"Numerically speaking", we should not be concerned about bisection at all. Indeed, let us set δ to something really small, say, $\delta = 1.e\text{-}16$. Then $\sigma = \delta n^{-1} << 1.e\text{-}16$, while (at least some of) $x_i(\lambda_*)$ should be of order of $1/n$ (since their sum should be 1). It follows that with actual (i.e., finite precision) computations, the quantity σ in (25) is negligible. Omitting σ in (24) (i.e., replacing in (8) the regularized entropy by the usual one), we can explicitly write down the solution x_* to (24):

$$x_i = \frac{\exp\{-\widehat{p}_i\}}{\sum_j \exp\{-\widehat{p}_j\}},\ i = 1, \ldots, n.$$

Case S.B. The case of $X = \Delta_n^+$ is very close to the one of $X = \Delta_n$. The only difference is that now we first should check whether

$$\sum_i \max\left[\exp\{-1 - p_i\} - \delta n^{-1}, 0\right] \leq 1;$$

if it is the case, then the optimal solution to (8) is given by

$$x_i = \max\left[\exp\{-1 - p_i\} - \delta n^{-1}, 0\right],\ i = 1, \ldots, n,$$

otherwise the optimal solution to (8) is exactly the optimal solution to (24).

Spectahedron setup. Consider two simple cases of the spectahedron setup:

Sp.A: X is comprised of all block-diagonal matrices of a given block-diagonal structure belonging to Σ_n,
or

Sp.B: X is comprised of all block-diagonal matrices of a given block-diagonal structure belonging to Σ_n^+.

Case Sp.A. Here the problem (8) becomes

$$\min_{x \in X} \{\mathrm{Tr}((x + \sigma I_n)\ln(x + \sigma I_n)) + \mathrm{Tr}(px)\} \qquad [\sigma = \delta n^{-1}]$$

We lose nothing by assuming that p is a symmetric block-diagonal matrix of the same block-diagonal structure as the one of matrices from X. Let $p = U\pi U^T$ be the eigenvalue decomposition of p with orthogonal U and diagonal π of the same block-diagonal structure as that of p. Passing from x to the new matrix variable ξ according to $x = U\xi U^T$, we convert our problem to the problem

$$\min_{\xi\in X}\{\mathrm{Tr}((\xi+\sigma I_n)\ln(\xi+\sigma I_n)) + \mathrm{Tr}(\pi\xi)\} \tag{26}$$

We claim that the unique (due to strong convexity of the function ω) optimal solution ξ^* to the latter problem is a diagonal matrix. Indeed, for every diagonal matrix D with diagonal entries ± 1 and for every feasible solution ξ to our problem, the matrix $D\xi D$ is again a feasible solution with the same value of the objective (recall that π is diagonal). It follows that the optimal set $\{\xi^*\}$ of our problem is invariant w.r.t. the aforementioned transformations $\xi \mapsto D\xi D$, which is possible if and only if ξ^* is a diagonal matrix. Thus, when solving (26), we may from the very beginning restrict ourselves with diagonal ξ, and with this restriction the problem becomes

$$\min_{\xi\in\mathbf{R}^n}\left\{\sum_i(\xi_i+\sigma)\ln(\xi_i+\sigma) + \pi^T\xi : \xi\geq 0, \sum_i\xi_i = 1\right\}, \tag{27}$$

which is exactly the problem we have considered in the case of the simplex setup with $X = \Delta_n$. We see that the only extra work needed in the case of the spectahedron setup, as compared to the simplex one, is in the necessity to find the eigenvalue decomposition of p. The latter task is easy, provided that the diagonal blocks in the matrices in question are of small sizes. Note that this favourable situation does occur in several important applications, e.g., in Structural Design.

Case Sp.B. This case is completely similar to the previous one; the only difference is that the role of (27) is now played by the problem

$$\min_{\xi\in\mathbf{R}^n}\left\{\sum_i(\xi_i+\sigma)\ln(\xi_i+\sigma) + \pi^T\xi : \xi\geq 0, \sum_i\xi_i \leq 1\right\},$$

which we have already considered discussing the simplex setup.

Updating prox-centers. The complexity results stated in Theorem 2.1 are independent of how the prox-centers are updated, so that in this respect one, *in principle*, is completely free. It is reasonable, however, to choose as the prox-center at every stage the best (with the smallest value of f) solution obtained up to the current stage.

Accumulating information. The set X_t summarizes, in a sense, all the information on f accumulated so far and to be used in the sequel. Relation (16) allows for a tradeoff between the quality (and the volume) of this information and the computational effort required to solve the auxiliary problems (P_{t-1}). With no restrictions on this effort, the most promising policy for updating X_t's would be to set $X_t = \underline{X}_t$ ("collecting information without compressing it"). With this policy the NERML algorithm *with the ball setup* is basically identical to the *Prox-Level Algorithm* of Lemarechal, Nemirovski and Nesterov [8]; the "restricted memory" version of the latter method (that is, the generic NERML algorithm with ball setup) was proposed by Kiwiel [4].

4 Extensions

The NERML algorithm and the results presented so far are limited to the case of Lipschitz continuous objective, and what is worse, the complexity bound (19) is proportional to the squared Lipschitz constant of the objective, and thus becomes very bad for "rapidly varying" objectives. To some extent, this is "a law of nature", as it follows from the optimality results mentioned in Section 2.3. However, *if X possesses symmetry, the complexity bounds can be improved*. For example, assume that X is the unit Euclidean ball. It is known [10] that in this case an appropriate version of the usual Subgradient Descent method guarantees that the inaccuracy, in terms of the objective, after N steps does not exceed $O(1)\frac{V[f]}{\sqrt{N}}$, where $V[f] = \max\limits_X f - \min\limits_X f$ is the variation of the objective (assumed to be convex continuous, but not necessarily Lipschitz continuous) on the feasible domain. In other words, the number of steps sufficient to minimize f within accuracy ϵ does not exceed $M(\epsilon) = O(1)\frac{V^2[f]}{\epsilon^2}$. In the case of Lipschitz continuous f we have $V[f] \le 2L_{\|\cdot\|_2}(f)$ (since $X = B_n$), so that $M(\epsilon)$ is, up to an absolute constant factor, less than the complexity bound $N(\epsilon)$ given by (19), and the ratio $N(\epsilon)/M(\epsilon)$ can be arbitrarily large (look at the case of $f(x) = -\sqrt{1+\delta - x^Tx}$).

We are about to demonstrate that the NERML method can be modified to ensure improved, as compared to (19), complexity bounds. Furthermore, we extend the algorithm from convex optimization problems in the form of (1) to other problems "with convex structure", including finding saddle points of convex-concave functions and solving variational inequalities with monotone operators.

4.1 Semi-bounded monotone mappings

We start with developing an appropriate general framework. Let $\mathcal{X}$ be a Euclidean space with inner product $\langle\cdot,\cdot\rangle$ and associated norm $\|\cdot\|_{\mathcal{X}}$.

Definition 4.1 *Let $X \subseteq \mathcal{X}$ be a nonempty convex set, let $c \in X$, and let $\Theta \in (0,1]$. Let F be a multi-valued monotone mapping on $\mathcal{X}$ (i.e., $F(x)$ is a subset in $\mathcal{X}$, the set $\mathrm{Dom}\, F = \{x : F(x) \neq \emptyset\}$ is nonempty and convex, and $\langle \xi - \eta, x - y\rangle \ge 0$ for all $x, y \in \mathrm{Dom}\, F$ and all $\xi \in F(x)$, $\eta \in F(y)$). We say that F is semi-bounded w.r.t. (X, c, Θ) (notation: $\mathbf{V}_{X,c,\Theta}[F] < \infty$), if $X \subseteq \mathrm{Dom}\, F$ and the quantity*

$$\mathbf{V}_{X,c,\Theta}[F] = \sup_{x,y\in X,\zeta\in F(x),\kappa=\pm 1} \langle \zeta, c + \kappa\Theta(c-y) - x\rangle \tag{28}$$

is finite.

Note that the functional $\mathbf{V}_{X,c,\Theta}[\cdot]$ clearly is nonnegative on its domain (set $x = y = c$ in the right hand side of (28)).

Example 4.1 [Bounded monotone mapping/Subgradient of Lipschitz continuous function] Let F be a bounded monotone mapping (i.e., $\sup\limits_{\substack{x\in \mathrm{Dom}\, F\\ \zeta\in F(x)}} \|\zeta\|_{\mathcal{X}} < \infty$) and X be a convex set such that $X \subseteq \mathrm{Dom}\, F$. Then for every norm $\|\cdot\|$ on $\mathcal{X}$ and every $c \in X$, $\Theta \in (0,1]$ one has

$$\mathbf{V}_{X,c,\Theta}[F] \le 2D_{\|\cdot\|}(X)V_{\|\cdot\|,X}[F] \tag{29}$$

where

- $D_{\|\cdot\|}(X) = \max\limits_{x,y\in X} \|x-y\|$ is the diameter of X w.r.t. $\|\cdot\|$;
- $V_{\|\cdot\|,X}[F] = \sup\limits_{x\in X,\zeta\in F(x)} \|\zeta\|_*$, (as always, $\|\cdot\|_*$ is the norm conjugate to $\|\cdot\|$).

In particular, if function f is convex Lipschitz continuous, with constant $L_{\|\cdot\|}(f)$ w.r.t. a norm $\|\cdot\|$, $F(x) = \partial f(x)$ is the corresponding subdifferential mapping, and $X \subseteq \mathrm{Dom}\, f$ is a convex set, then for all $c \in X$ and all $\Theta \in (0, 1]$ one has

$$\mathbf{V}_{X,c,\Theta}[F] \le 2D_{\|\cdot\|}(X)L_{\|\cdot\|}(f). \tag{30}$$

Example 4.2 [Semi-bounded monotone mapping/Subgradient of a bounded function in a "nearly symmetric" domain] Let F be a monotone mapping with convex domain X. Assume that F can be extended from X to a monotone mapping Φ with convex domain $Y = \mathrm{Dom}\,\Phi$ (so that $F(x) = \Phi(x)$ for $x \in X$) in such a way that

1. The quantity
$$V_Y[\Phi] = \sup_{\substack{x,y\in Y\\ z\in\partial\Phi(x)}} \langle y-x, z\rangle$$
is finite;
2. For certain $\Theta \in (0, 1]$, the set Y is Θ-symmetric w.r.t. a certain point $c \in X$, i.e.,
$$c + \Theta(c - Y) \subseteq Y$$
(geometrically: the point c splits every segment passing through c and with endpoints on the boundary of Y in the ratio not exceeding $1 : \Theta$).

Then

$$\mathbf{V}_{X,c,\Theta}[F] \le V_Y[\Phi]. \tag{31}$$

In particular, if a convex function $f : X \to \mathbf{R}$ can be extended to a lower semicontinuous convex function ϕ with $X \subseteq Y \subseteq \mathrm{rint}\,\mathrm{Dom}\,\phi$ and the convex set Y is Θ-symmetric w.r.t. certain point $c \in X$, then

$$\mathbf{V}_{X,c,\Theta}[\partial f] \le V_Y[\partial\phi] \equiv \sup_{x,y\in Y, z\in\partial\phi(x)} \langle y-x, z\rangle \le \sup_Y \phi - \inf_Y \phi. \tag{32}$$

Indeed, given $x, y \in X$, we have $z_\pm \equiv c \pm \Theta(y-c) \in Y$ due to $X \subseteq Y$ and to Θ-symmetry of Y w.r.t. c. By the definition of $V_Y[\Phi]$ we have

$$\sup_{\zeta\in\partial F(x)} \langle \zeta, z_\pm - x\rangle \le V_Y[\Phi],$$

i.e., for all $x, y \in X$ and all $\zeta \in F(x)$ one has

$$\langle \zeta, c \pm \Theta(c-y) - x\rangle \le V_Y[\Phi],$$

as claimed in (31). To get (32), if suffices to apply (31) to the monotone mapping $\Phi(x) = \partial\phi(x)$ restricted to the domain Y and to note that since ϕ is a convex function on Y, we have

$$\sup_{\substack{x,y\in Y\\ \zeta\in\partial\phi(x)}} \langle y-x, \zeta\rangle \le \sup_{x,y\in Y} [\phi(y) - \phi(x)] = \sup_Y \phi - \inf_Y \phi.$$

Many additional examples of semi-bounded monotone mappings can be generated due to the fact that the functional $\mathbf{V}_{X,c,\Theta}[\cdot]$ is "well-behaved" under operations preserving monotonicity:

Proposition 4.1 *Let $X \subseteq \mathcal{X}$ be a nonempty convex set, let $c \in X$, and let $\Theta \in (0, 1]$. Then*

1. [Homogeneity and sublinearity] *If F_i are semi-bounded w.r.t. (X, c, Θ) and $\lambda_i \geq 0$, $i = 1, \ldots, m$, then*
$$\mathbf{V}_{X,c,\Theta}\left[\sum_i \lambda_i F_i\right] \leq \sum_i \lambda_i \mathbf{V}_{X,c,\Theta}[F_i];$$
2. [Stability w.r.t. affine substitutions of argument] *Let $y \mapsto Ay + b : \mathcal{F} \to \mathcal{X}$ be an affine mapping such that $c = Ad + b$ for certain d, let F be semi-bounded w.r.t. (X, c, Θ), and let*
$$Y = \{y : Ay + b \in X\}, \quad G(y) = A^* F(Ay + b).$$
Then
$$\mathbf{V}_{Y,d,\Theta}[G] \leq \mathbf{V}_{X,c,\Theta}[F].$$
3. [Monotonicity w.r.t. Θ] *Whenever $\Theta' \in (0, \Theta]$ and F is semi-bounded w.r.t. X, c, Θ, one has*
$$\mathbf{V}_{X,c,\Theta'}[F] \leq \mathbf{V}_{X,c,\Theta}[F].$$
4. [Monotonicity w.r.t. X] *Whenever $c \in X' \subseteq X$ with convex X', one has*
$$\mathbf{V}_{X',c,\Theta}[F] \leq \mathbf{V}_{X,c,\Theta}[F].$$
5. [Stability w.r.t. c] *Let $c' \in X$ be such that*
$$\pi_c(c' - c) \equiv \inf\{t > 0 : c \pm t^{-1}(c' - c) \in X\} < \Theta,$$
and let
$$\Theta' = \frac{\Theta - \pi_c(c' - c)}{1 - \pi_c(c' - c)}.$$
Then
$$\mathbf{V}_{X,c',\Theta'}[F] \leq \mathbf{V}_{X,c,\Theta}[F].$$

We are about to demonstrate that the NERML scheme can be extended from the case when the objective in (1) has a bounded subgradient mapping to the case of a *semi*bounded gradient mapping.

4.2 The general NERML scheme

Setup for the general NERML scheme (GNERML) is given by:

1. A solid (convex compact set with a nonempty interior) $X \subseteq \mathcal{X}$,
2. A strongly convex and continuously differentiable function $\omega(\cdot)$ on X.

We set
$$\Omega[\omega(\cdot)] = \max_{x,y \in X} \left[\omega(y) - \omega(x) - \langle \omega'(x), y - x\rangle\right].$$

The data for the GNERML scheme are given by a *bounded* vector field

$$g(x) : X \mapsto \mathcal{X}.$$

Given a nonempty finite subset S of X, we set

$$F_S(y) = \max_{x \in S} \langle g(x), y - x \rangle, \quad g_*[S] = \min_{y \in X} F_S(y).$$

The goal is, given $\delta > 0$, to build a set S such that

$$g_*[S] > -\delta.$$

To give a (preliminary) motivation to our goal, consider the case when we are interested to minimize over X a Lipschitz continuous convex function $f(x)$. Setting $g(x) = f'(x)$, observe that the relation $g_*[S] \geq -\epsilon$ means that for every $y \in X$ there exists $x \in S$ such that $\langle f'(x), y - x \rangle \geq -\epsilon$, whence $f(x) - f(y) \leq \epsilon$ by convexity of f. Thus, $\min_{x \in S} f(x) - f(y) \leq \epsilon$ for all $y \in X$, so that the best (with the smallest value of f) of the points from S is an ϵ-minimizer of f on X.

The GNERML scheme is as follows. We build a *search sequence* $x_0, x_1, ...$, thus defining finite sets $S^t = \{x_0, x_1, \ldots, x_t\} \subseteq X$. The search sequence is built according to the following rules.

A. We choose an arbitrary $x_0 \in X$ and set $f_1 = \min_{x \in X} \langle g(x_0), x - x_0 \rangle$. We clearly have $f_1 \leq 0$; the case when $f_1 = 0$ is trivial, since here $g_*[\{x_0\}] = 0$, which is even more than we need.

Our subsequent actions are split into *phases* enumerated 1, 2,... Let us describe a particular phase s.

B. Phase s starts at a certain moment t_s ($t_1 = 1$). Let us set $S_s = \{x_0, \ldots, x_{t_s - 1}\}$, and let $f_s < 0$ be a valid lower bound, available at the beginning of phase s, on the quantity $g_*[S_s]$. We set

$$\ell_s = (1 - \lambda) f_s$$

($\lambda \in (0, 1)$ is a once forever fixed parameter); note that $\ell_s < 0$ along with f_s.

To save notation, we denote the subsequent search points generated at phase s as $u_1, u_2, ...$, so that $x_{t_s + \tau} = u_{\tau + 1}$, $\tau = 0, 1, ...$ We choose $u_1 \in X$ in an arbitrary fashion, set

$$\omega_s(x) = \omega(x) - \langle \omega'(u_1), x \rangle$$

and use finitely many linear inequalities to cut off X a *localizer* X_1 such that $u_1 \in X_1$ and

$$x \in X \backslash X_1 \Rightarrow F_{S_s}(x) > \ell_s.$$

C. The situation at the beginning of step τ of phase s is as follows: we have $u_\tau \in X$ and $X_\tau \subseteq X$ (X_τ is cut off X by finitely many linear inequalities) such that

$$\begin{aligned} x \in X \backslash X_\tau &\Rightarrow F_{S_s^{\tau - 1}}(x) > \ell_s \quad (a_\tau) \\ u_\tau &= \operatorname*{argmin}_{x \in X_\tau} \omega_s(x) \quad (b_\tau) \end{aligned}$$

where $S_s^{\tau - 1} = S_s \cup \{u_1, \ldots, u_{\tau - 1}\}$, $\tau > 1$, and $S_s^0 = S_s$.

D. At step τ, we act as follows:

1. compute $g(u_\tau)$ and set $h_\tau(x) = \langle g(u_\tau), x - u_\tau\rangle$,
2. solve the auxiliary problem

$$\tilde{f} = \min_{x\in X_\tau} h_\tau(x), \tag{33}$$

and compute the quantity

$$f_{s,\tau} = \max\left[f_{s,\tau-1}, \min\left[\tilde{f}, \ell_s\right]\right]$$

where $f_{s,0} = f_s$.
Note: $f_{s,\tau}$ is a valid lower bound on $g_*[S_s^\tau]$ along with $f_{s,\tau-1}$, since in $X\backslash X_\tau$ one has

$$F_{S_s^\tau}(x) \geq F_{S_s^{\tau-1}}(x) > \ell_s,$$

while on X_τ one has

$$F_{S_s^\tau}(x) \geq h_\tau(x) \geq \tilde{f} = \min_{x\in X_\tau} h_\tau(x),$$

so that everywhere on X one has

$$F_{S_s^\tau}(x) \geq \min\left[\tilde{f}, \ell_s\right]$$

and, besides this, everywhere on X one has

$$F_{S_s^\tau}(x) \geq F_{S_s^{\tau-1}}(x) \geq f_{s,\tau-1}.$$

3. We check whether

$$\ell_s - f_{s,\tau} \leq \theta(\ell_s - f_s)$$

where $\theta \in (0, 1)$. If this is the case, phase s is terminated, and we set

$$f_{s+1} = f_{s,\tau}.$$

4. If phase s is not terminated yet, the set $\{x \in X_\tau : h_\tau(x) \leq \ell_s\}$ is nonempty (since otherwise we would have $\min_{X_\tau} h_\tau(\cdot) \geq \ell_s \geq (1+\theta)\ell_s$ and therefore $f_{s,\tau} \geq \ell_s$, which is impossible, since phase s was not terminated yet). We set

$$u_{\tau+1} = \operatorname{argmin}\{\omega_s(x) : x \in X_\tau, h_\tau(x) \leq \ell_s\}$$

and choose as $X_{\tau+1}$ an arbitrary set (cut off X by finitely many linear inequalities) such that

$$\{x \in X : \langle \omega_s'(u_{\tau+1}), x - u_{\tau+1}\rangle \geq 0\} \subseteq X_{\tau+1} \subseteq \{x \in X_\tau : h_\tau(x) \leq \ell_s\}. \tag{34}$$

With this approach, we clearly satisfy the requirements $(a_{\tau+1})$, $(b_{\tau+1})$.

The summary of the General NERML scheme is as follows:

General NERML scheme

Parameters: $\lambda \in (0, 1), \theta \in (0, 1), \epsilon > 0$.

Initialization: Choose $x_0 \in X$, compute $g(x_0)$ and set $f_1 = \min\limits_{x \in X} \langle g(x_0), x - x_0 \rangle$, $t_1 = 1$.

Phase s ($s = 1, 2, ...$): If $f_s \geq -\epsilon$ (required tolerance) terminate; otherwise set $S_s = \{x_0, \ldots, x_{t_s - 1}\}$, $\ell_s = (1 - \lambda)(f_s)$, choose an arbitrary $u_1 \equiv x_{t_s} \in X$, and set

$$\omega_s(x) = \omega(x) - \langle \omega'(u_1), x \rangle,$$
$$f_{s,0} = f_s.$$

Choose a subset $X_1 \subseteq X$ cut off X by finitely many linear inequalities such that $u_1 \in X_1$ and

$$x \in X \backslash X_s \Rightarrow F_{S_s}(x) > \ell_s.$$

Start inner iterations:

Inner iteration τ ($\tau = 1, 2, ...$):
Compute $g(u_\tau)$ and set

$$h_\tau(x) = \langle g(u_\tau), x - u_\tau \rangle,$$
$$\tilde{F} = \min_{x \in X_\tau} h_\tau(x),$$
$$f_{s,\tau} = \max \left[f_{s,\tau - 1}, \min[\tilde{F}, \ell_s[\right].$$

If $\ell_s - f_{s,\tau} \leq \theta(\ell_s - f_s)$,
set $f_{s+1} = f_{s,\tau}$ and pass to phase $s + 1$,
else
set

$$x_{t_s + \tau} \equiv u_{\tau + 1} = \operatorname*{argmin}_{x \in X_\tau} \{\omega_s(x) : h_\tau(x) \leq \ell_s\},$$

choose $X_{\tau+1}$ cut off X by finitely many linear inequalities and satisfying (34), and pass to step $\tau + 1$ of phase s.

Implementation issues for the GNERML scheme can be resolved in the same way as for the basic NERML algorithm, see Section 3.

Convergence properties of the outlined scheme are described in the following statement:

Theorem 4.1 *Let a field $g(\cdot)$ be processed by a GNERML scheme associated with $(X, \omega(\cdot))$, let $\|\cdot\|$ be a norm on $\mathcal{X}$, and let $\kappa > 0$ be such that $\omega(\cdot)$ is κ-strongly convex w.r.t. $\|\cdot\|$:*

$$\forall (x, y \in X): \quad \omega(y) \geq \omega(x) + \langle \omega'(x), y - x \rangle + \frac{\kappa}{2} \|y - x\|^2. \tag{35}$$

Finally, let $M < \infty$ be such that

$$\|g(x_t)\|^* \le M \quad \forall t, \tag{36}$$

where $\|\cdot\|^*$ is the norm conjugate to $\|\cdot\|$. Then

(i) *The number N_s of steps at a phase s is bounded above as follows:*

$$N_s \le \frac{4\Omega[\omega(\cdot)]M^2}{\theta^2(1-\lambda)^2\kappa\delta_s^2},$$
$$\delta_s \equiv -f_s. \tag{37}$$

(ii) *Consequently, for every $\epsilon > 0$, the total number of steps, before the phase s for which $\delta_s \le \epsilon$ is started does not exceed*

$$N(\epsilon) = c(\theta,\lambda)\frac{\Omega[\omega(\cdot)]M^2}{\kappa\epsilon^2} \tag{38}$$

with an appropriate $c(\theta,\lambda)$ depending solely and continuously on $\theta, \lambda \in (0,1)$.

We conclude this section with the following result which is important for the sequel:

Proposition 4.2 *Let $X \subseteq \mathcal{X}$ be a solid, let $F(\cdot)$ be semi-bounded on X:*

$$\mathbf{V}_{X,c,\Theta}[F] \le L < \infty, \tag{39}$$

and let a field $g(\cdot)$ be given by

$$g(x) = [\|h(x)\|_X^*]^{-1}h(x), \quad h(x) \in F(x), \tag{40}$$

where $\|\cdot\|_X$ is the norm on $\mathcal{X}$ with the unit ball $\frac{1}{2}[X-X]$:

$$\|x\|_X = \inf\left\{t > 0 : t^{-1}x \in \frac{1}{2}[X-X]\right\}$$

and $\|\cdot\|_X^*$ is the norm conjugate to $\|\cdot\|_X$:

$$\|\xi\|_X^* = \max\{\langle\xi, x\rangle : \|x\|_X \le 1\} = \frac{1}{2}\left[\max_{x\in X}\langle\xi,x\rangle - \min_{x\in X}\langle\xi,x\rangle\right].$$

Whenever points $x_0, \ldots, x_T \in X$ are such that

$$g_*[\{x_0,\ldots,x_T\}] \ge -\delta, \tag{41}$$

where

$$\epsilon \equiv \delta/\Theta < 1,$$

one has

$$\min_{x\in X}\max_{t\le T}\langle h(x_t), x - x_t\rangle \ge -\frac{\epsilon}{1-\epsilon}L. \tag{42}$$

In particular, there exist (and can be efficiently found) weights $\lambda_t \geq 0$, $\sum_{t=0}^{T} \lambda_t = 1$ *such that*

$$\min_{x \in X} \sum_{t=0}^{T} \lambda_t \langle h(x_t), x - x_t \rangle \geq -\frac{\epsilon}{1-\epsilon} L; \tag{43}$$

moreover, defining

$$\widetilde{x}_T = \sum_{t=0}^{T} \lambda_t x_t \tag{44}$$

one has

$$\forall (x \in X, \zeta \in F(x)): \quad \langle \zeta, x - \widetilde{x}_T \rangle \geq -\frac{\epsilon}{1-\epsilon} L. \tag{45}$$

4.3 Applications in nonsmooth convex minimization

We show, first, that the GNERML scheme, as applied to the problem of minimizing a Lipschitz continuous convex function, yields the same efficiency guarantees as the basic NERML method:

Proposition 4.3 *Let* f *be a convex function,* $\operatorname{int} \operatorname{Dom} f \supset X$*, which is Lipschitz continuous with constant* $L_{\|\cdot\|}(f)$ *w.r.t. a norm* $\|\cdot\|$ *on* $\operatorname{Dom} f$*, and let the field* $g(x) = f'(x)$ *be processed by the GNERML scheme associated with* $(X, \omega(\cdot))$*. Then*

(i) *For every* T *and* $\epsilon > 0$*, the relation*

$$g_*[\{x_0, \ldots, x_T\}] \geq -\epsilon \tag{46}$$

implies that

$$f(x^T) \leq \min_{x \in X} f(x) + \epsilon, \quad x^T = \operatorname*{argmin}_{x = x_0, \ldots, x_T} f(x). \tag{47}$$

(ii) *For every* $\epsilon > 0$ *the number of steps until* (46) *is satisfied does not exceed the quantity*

$$N(\epsilon) = c(\theta, \lambda) \frac{\Omega[\omega(\cdot)] L^2_{\|\cdot\|}(f)}{\kappa \epsilon^2},$$

where κ *is the constant of strong convexity of* $\omega(\cdot)$ *w.r.t.* $\|\cdot\|$*.*

We are about to demonstrate that in fact the GNERML scheme yields stronger efficiency guarantees, those which were mentioned in the beginning of Section 4.

Proposition 4.4 *Let* $X \subseteq \mathcal{X}$ *be a solid, let* f *be a convex function,* $X \subseteq \operatorname{int} \operatorname{Dom} f$*, and let* $F(x) = \partial f(x)$*. Let, finally,*

$$g(x) = [\|f'(x)\|_X^*]^{-1} f'(x), \quad f'(x) \in F(x) \text{ for } x \in X. \tag{48}$$

(i) *Let* $\epsilon \in (0, 1)$, $\Theta \in (0, 1)$ *and* $c \in X$ *be given, and let the points* $x_0, \ldots, x_T \in X$ *be such that*

$$g_*[x_0, \ldots, x_T] \geq -\epsilon\Theta, \tag{49}$$

one has

$$f(x^T) \leq \min_X f + \frac{\epsilon}{1-\epsilon}\mathbf{V}_{X,c,\Theta}[F],$$

where x^T is the best (with the smallest value of f) of the points $x_0, \ldots, x_T$.

(ii) *When applying to $g(\cdot)$ the GNERML scheme associated with $(X, \omega(\cdot))$, we get a sequence $x_0, x_1, \ldots$ such that*

$$\forall(\epsilon \in (0,1), \Theta > 0):$$
$$T \geq b(\lambda,\theta)\frac{\Omega[\omega(\cdot)]}{\kappa[X,\omega(\cdot)]}\left(\frac{1}{\Theta\epsilon}\right)^2 \Rightarrow g_*[x_0, \ldots, x_T] \geq -\epsilon\Theta, \tag{50}$$

where

- *$\kappa[X, \omega(\cdot)]$ is the constant of strong convexity of $\omega(\cdot)$ w.r.t. the norm $\|\cdot\|_X$, and*
- *b depends solely on the parameters λ and θ of the GNERML scheme.*

4.3.1 Discussion To get an impression of the power of Proposition 4.4, consider several implications of this statement. In the discussion to follow, the parameters $\lambda, \theta \in (0, 1)$ of the GNERML scheme are treated as once for ever fixed absolute constants.

4.3.2 Optimization over "nearly $\|\cdot\|_2$-balls" Let a solid $X \subseteq \mathcal{X} = \mathbf{R}^n$ be such that

- X is contained in the unit Euclidean ball B_n and is Θ_1-symmetric for certain $\Theta_1 \in (0, 1]$;
- the set $\frac{1}{2}[X - X]$ (which clearly is contained in B_n) contains, for certain $\Theta_2 \in (0, 1]$, the ball $\Theta_2 B_n$.
  E.g., when $X \subseteq B_n$ contains ρB_n with certain $\rho > 0$, one can take $\Theta_1 = \Theta_2 = \rho$. Also, when X contains the nonnegative part $\{x \geq 0, \|x\|_2 \leq 1\}$ of the unit ball, one can take $\Theta_2 = \frac{1}{2}$.

Let, further, f be a convex function, $X \subseteq \operatorname{int}\operatorname{Dom} f$, and let

$$V_X[f] \equiv \max_{x,y\in X,\, f'(x)\in\partial f(x)} \langle f'(x), y - x\rangle.$$

Note that the quantity $V_X[F]$ is seemingly the smallest measure compared to some other choices measuring the "magnitude" of $f\big|_X$. For example,

$$V_X[f] \leq \max_X f - \min_X f \leq 2L_{\|\cdot\|_2}(f\big|_X)$$

(the first inequality is valid for all solids X, the second follows from $X \subseteq B_n$). In fact $V_X[f]$ can be much less than the variation $\max_X f - \min_X f$. E.g., for $f(x) = -\ln(1 + \delta - x^T x)$, $\delta > 0$, one has $V_{B_n}[f] \leq 1$, while the variation of f on B_n tends to ∞ as $\delta \to +0$.

Now assume that f is minimized over X by the GNERML scheme associated with $(X, \omega(x) = \frac{1}{2}x^T x)$ and applied to the vector field

$$g(x) = [\|f'(x)\|_X^*]^{-1} f'(x) : X \to \mathbf{R}^n.$$

By (32) (where one should take $Y = X$), one has

$$\mathbf{V}_{X,c,\Theta_1}[\partial f] \le V_X[f], \tag{51}$$

while by Proposition 4.4 one has

$$\forall \epsilon \in (0,1): \quad T \ge b(\theta,\lambda)\frac{\Omega[\omega(\cdot)]}{\kappa[X,\omega(\cdot)]}\frac{1}{\Theta_1^2\epsilon^2} \Rightarrow f(x^T) \le \min_X f + \frac{\epsilon}{1-\epsilon}\mathbf{V}_{X,c,\Theta}[\partial f]. \tag{52}$$

Now, since $X \subseteq B_n$, one clearly has $\Omega[\omega(\cdot)] \le O(1)$, while from the fact that $\frac{1}{2}[X - X]$ is contained in B_n and contains $\Theta_2 B_n$ it follows that $\|x\|_X \le \Theta_2^{-1}\|x\|_2$ for all x, whence $\kappa[X, \omega(\cdot)] \ge \Theta_2^2$. Combining these observations with (51), (52), we arrive at

$$\forall \epsilon \in (0,1): \quad T \ge O(1)\frac{1}{(\Theta_1\Theta_2)^2\epsilon^2} \Rightarrow f(x^T) \le \min_X f + \frac{\epsilon}{1-\epsilon}V_X[f].$$

When Θ_1 and Θ_2 are of order of 1 (as in the case of $X = B_n$), the resulting efficiency estimate

$$\epsilon < 1,\ T \ge O(1)\epsilon^{-2} \Rightarrow f(x^T) \le \min_X f + \frac{\epsilon}{1-\epsilon}V_X[f]$$

is even better than the one mentioned in the beginning of Section 4.

4.3.3 Optimization over "nearly $\|\cdot\|_1$-balls" Let U be a solid in $\mathbf{R}^n$ such that

- U contains the origin, is contained in the set $D_n = \{u \in \mathbf{R}^n : \|u\|_1 \le 1\}$ and is Θ_1-symmetric for certain $\Theta_1 \in (0, 1]$;
- the set $\frac{1}{2}[U - U]$ (which clearly is contained in D_n) contains, for certain $\Theta_2 \in (0, 1]$, the set $\Theta_2 D_n$.
  E.g., when $U \subseteq D_n$ contains ρD_n for certain $\rho > 0$, one can take $\Theta_1 = \Theta_2 = \rho$. Also, when U contains the simplex Δ_n (or the simplex Δ_n^+), one can take $\rho = \frac{1}{2}$.

Assume that we are interested to minimize over U a convex function $h(\cdot)$, $U \subseteq \operatorname{int}\operatorname{Dom} h$. To this end, let us set

$$X = \{x \in \mathbf{R}^{2n} = \mathbf{R}_u^n \times \mathbf{R}_v^n : \|x\|_1 \le 1, x \ge 0, Px \in U\}, \quad P\begin{bmatrix} u \\ v \end{bmatrix} = u - v$$

$$f(u, v) = h(u - v),$$

thus arriving at a convex solid $X \subseteq \mathbf{R}^{2n}$ and a convex function f, $X \subseteq \operatorname{int}\operatorname{Dom} f$, such that the problem of minimizing f over X is equivalent to the problem of minimizing h over U. In order to minimize f over X, let us use the GNERML scheme with the simplex setup as applied to the vector field

$$g(x) = [\|f'(x)\|_X^*]^{-1} f'(x) : X \to \mathbf{R}^{2n}.$$

Note that by items 2 and 4 of Proposition 4.1, one has

$$\mathbf{V}_{X,0,\Theta_1}[\partial f] \le \mathbf{V}_{U,0,\Theta_1}[\partial h] \le V_U[h], \tag{53}$$

the concluding inequality being given by (32) (where one should set $Y = X = U$).

By Proposition 4.4 combined with (53) one has

$$\forall \epsilon \in (0,1): \quad T \ge b(\theta,\lambda)\frac{\Omega[\omega(\cdot)]}{\kappa[X,\omega(\cdot)]}\frac{1}{\Theta_1^2\epsilon^2} \Rightarrow f(x^T) \le \min_X f + \frac{\epsilon}{1-\epsilon}V_U[h]. \tag{54}$$

Now, the set $D = \frac{1}{2}[X - X]$ is contained in D_{2n} (since $X \subseteq D_{2n}$). We claim that D contains $(\Theta_2/6)D_{2n}$.

Indeed, since $0 \in U$, we have $0 \in X$. Assume that D does not contain $(\Theta_2/6)D_{2n}$. Then there exists $\phi = (\phi_u, \phi_v) \in \mathbf{R}^{2n}$ such that $\|(\phi_u, \phi_v)\|_\infty = 1$ and

$$|\phi_u^T u + \phi_v^T v| < \Theta_2/3 \quad \forall (u,v) \in X. \tag{55}$$

Since all vectors of the form (u,u) with $u \ge 0$, $\|u\|_1 \le 1/2$, belong to X, it follows from (55) that $|(\phi_u + \phi_v)u| < \Theta_2/3$ for all $u \ge 0$, $\|u\|_1 \le \frac{1}{2}$, whence

$$\|\phi_u + \phi_v\|_\infty \le 2\Theta_2/3. \tag{56}$$

Since $\|(\phi_u, \phi_v)\|_\infty = 1$, we have $\|\phi_u\|_\infty = 1$, or $\|\phi_v\|_\infty = 1$. Assume that $\|\phi_u\|_\infty = 1$, and let $(u,v) \in X$. We have

$$\begin{aligned} |\phi_u^T(u-v)| &\le |\phi_u^T u + \phi_v^T v| + \|\phi_u + \phi_v\|_\infty \|v\|_1 \\ &< \Theta_2/3 + 2\Theta_2/3 \qquad \text{[by (55), (56)]} \\ &= \Theta_2 \\ \Rightarrow |\phi_u^T u| &< \Theta_2 \quad \forall u \in U. \end{aligned}$$

Since $\|\phi_u\|_\infty = 1$, the concluding relation contradicts the assumption that $\frac{1}{2}[U - U]$ contains $\Theta_2 D_n$. The case of $\|\phi_v\|_\infty = 1$ is completely similar. □

Since X is contained in Δ_{2n}^+, we have $\Omega[\omega(\cdot)] \le O(1)\ln n$, while from $\frac{\Theta_2}{6}D_{2n} \subseteq \frac{1}{2}[X - X] \subseteq D_{2n}$ it follows that $\|x\|_X^* \le O(1)\Theta_2^{-1}\|x\|_1$ for all x. Since the constant of strong convexity of $\omega(\cdot)$ w.r.t. $\|\cdot\|_1$ is $O(1)$, we arrive at $\kappa[X,\omega(\cdot)] \ge O(1)\Theta_2^{-2}$; consequently, (54) implies that

$$\forall \epsilon \in (0,1): T \ge O(1)\frac{\ln n}{(\Theta_1\Theta_2)^2\epsilon^2} \Rightarrow f(x^T) \le \min_X f + \frac{\epsilon}{1-\epsilon}V_U[h]. \tag{57}$$

When Θ_1 and Θ_2 are of order of 1 (as in the case of $U = D_n$), the resulting efficiency estimate is better than the one yielded by Theorem 2.1 for the case of simplex setup (recall that $V_u[h] \le 2L_{\|\cdot\|_1}(h)$ due to $U \subseteq D_n$).

3. Optimization over "matrix balls". Finally, consider the case when U is a solid in the space $\mathbf{M}^n$ of $n \times n$ symmetric matrices of a given block-diagonal structure such that

- U contains the origin, is contained in the set $\mathcal{D}_n = \{x \in \mathbf{M}^n : |x|_1 \leq 1\}$, where $|x|_1 = \|\lambda(x)\|_1$, and is Θ_1-symmetric for certain $\Theta_1 \in (0, 1]$;
- the set $\frac{1}{2}[U - U]$ (which clearly is contained in $\mathcal{D}_n$) contains, for certain $\Theta_2 \in (0, 1]$, the set $\Theta_2 \mathcal{D}_n$.
  E.g., when $X \subseteq B_n$ contains ρB_n with certain $\rho > 0$, one can take $\Theta_1 = \Theta_2 = \rho$. Also, when X contains the nonnegative part $\{x \geq 0, \|x\|_2 \leq 1\}$ of the unit ball, one can take $\Theta_2 = \frac{1}{2}$.

Assume that we are interested to minimize over U a convex function $h(\cdot)$, $U \subseteq \operatorname{int} \operatorname{Dom} h$. To this end, let us set

$$X = \{x \in \mathbf{M}^{2n} = \mathbf{M}^n_u \times \mathbf{M}^n_v : |x|_1 \leq 1, x \succeq 0, Px \in U\}, \quad P \begin{bmatrix} cu \\ v \end{bmatrix} = u - v$$

$$f(u, v) = h(u - v),$$

thus arriving at a convex solid $X \subseteq \mathbf{M}^{2n}$ and a convex function f, $X \subseteq \operatorname{int} \operatorname{Dom} f$, such that the problem of minimizing f over X is equivalent to the problem of minimizing h over U. In order to minimize f over X, we can use the GNERML scheme with the spectahedron setup as applied to the vector field

$$g(x) = [\|f'(x)\|_X^*]^{-1} f'(x) : X \to \mathbf{M}^{2n}.$$

The same reasoning as in the previous case results in the efficiency results as follows:

$$\forall \epsilon \in (0, 1) : T \geq O(1) \frac{\ln n}{(\Theta_1 \Theta_2)^2 \epsilon^2} \Rightarrow f(x^T) \leq \min_X f + \frac{\epsilon}{1 - \epsilon} V_U[h]. \tag{58}$$

4.4 Approximating saddle points

The GNERML scheme can be applied to approximating saddle points of convex-concave functions. The basic result here is as follows:

Proposition 4.5 *Let $X = U \times V \subseteq \mathcal{X} \equiv \mathcal{U} \times \mathcal{V}$, where U and V are solids in the Euclidean spaces $\mathcal{U}$ and $\mathcal{V}$. Let the function $f(u, v)$ be convex in u and concave in v, $X \subseteq \operatorname{int} \operatorname{Dom} f$, and let $F(u, v) = (\partial_u f(u, v)) \times (-\partial_v f(u, v))$. Let, finally,*

$$g(u, v) = [\|(f'_u(u, v), -f'_v(u, v))\|_X^*]^{-1} (f'_u(u, v), -f'_v(u, v)),$$
$$(f'_u(u, v), -f'_v(u, v)) \in F(u, v) \text{ for } (u, v) \in X.$$

(i) *Let $\epsilon \in (0, 1)$, $\Theta \in (0, 1)$ and $c \in X$ be given, and let the points $\{x_t = (u_t, v_t) \in X\}_{t=0}^T$ be such that* (49) *is valid. Then*

$$\epsilon(\widetilde{u}_T, \widetilde{v}_T) \leq \frac{\epsilon}{1 - \epsilon} \mathbf{V}_{X,c,\Theta}[F],$$

where

- $\widetilde{x}_T = (\widetilde{u}_T, \widetilde{v}_T)$ *is the point defined in Proposition 4.2,*
- $\epsilon(u, v) = \left[\overline{f}(u) - \min\limits_U \overline{f}\right] + \left[\max\limits_V \underline{f} - \underline{f}\right]$, *with*

$$\overline{f}(u) = \max_{v \in V} f(u, v), \quad \underline{f}(v) = \min_{u \in U} f(u, v).$$

(ii) *When applying to $g(\cdot)$ the GNERML scheme associated with $(X, \omega(\cdot))$, we get a sequence $x_0 = (u_0, v_0), x_1 = (u_1, v_1), \ldots$ such that*

$$\begin{aligned} &\forall(\epsilon \in (0, 1), \Theta > 0) : \\ &T \geq b(\lambda, \theta) \frac{\Omega[\omega(\cdot)]}{\kappa[X, \omega(\cdot)]} \left(\frac{1}{\Theta\epsilon}\right)^2 \Rightarrow g_*[x_0, \ldots, x_T] \geq -\epsilon\Theta, \end{aligned} \tag{59}$$

where

- $\kappa[X, \omega(\cdot)]$ *is the constant of strong convexity of $\omega(\cdot)$ w.r.t. the norm $\|\cdot\|_X$, and*
- b *depends solely on the parameters λ and θ of the GNERML scheme.*

Assuming that the convex-concave function in question is Lipschitz continuous on its domain, Proposition 4.4 can be modified as follows (cf. Proposition 4.3):

Proposition 4.6 *Let $X = U \times V \subseteq \mathcal{X} \equiv \mathcal{U} \times \mathcal{V}$, where U and V are solids in the Euclidean spaces $\mathcal{U}$ and $\mathcal{V}$, let the function $f(u, v)$ be convex in u and concave in v, $X \subset \operatorname{int}\operatorname{Dom} f$, and let $F(u, v) = (\partial_u f(u, v)) \times (-\partial_v f(u, v))$. Let*

$$\begin{aligned} g(u, v) &= (f'_u(u, v), -f'_v(u, v)), \\ (f'_u(u, v), -f'_v(u, v)) &\in F(u, v) \text{ for } (u, v) \in X, \end{aligned}$$

Further, let $\|\cdot\|$ be a norm on $\mathcal{X}$ and

$$L_{\|\cdot\|}(f) = \max_{(u,x) \in X} \max_{\xi \in F(u,v)} \|\xi\|_*,$$

where $\|\cdot\|_*$ is the norm conjugate to $\|\cdot\|$.

(i) *For all T and $\epsilon > 0$ the relation*

$$g_*[\{x_0, \ldots, x_T\}] \geq -\epsilon \tag{60}$$

implies that

$$\epsilon(\widetilde{u}_T, \widetilde{v}_T) \leq \epsilon, \tag{61}$$

where $\epsilon(u, v)$ and $\widetilde{u}_T, \widetilde{v}_T$ are defined as in Proposition 4.5.

(ii) *For every $\epsilon > 0$ the number of steps until* (60) *is satisfied does not exceed the quantity*

$$N(\epsilon) = c(\theta, \lambda) \frac{\Omega[\omega(\cdot)] L^2_{\|\cdot\|}(f)}{\kappa\epsilon^2},$$

where κ is the constant of strong convexity of $\omega(\cdot)$ w.r.t. $\|\cdot\|$.

5 Numerical results

Test problems. To test the performance of the NERML and the GNERML algorithms as applied to problems (1), we carried out 2 groups of numerical experiments:

- UFL problems (relaxations of the Uncapacitated Facility Location problems),
- TOMO problems (2D Image Reconstruction problems arising in Positron Emission Tomography).

  More details on the test problems are given in the relevant sections below.

The algorithms. The domains X we dealt with were either standard simplexes Δ_n, Δ_n^+, or the boxes $\{x \in \mathbf{R}^n : a \leq x \leq b\}$. For simplex-type domains, we used both the ball and the simplex setups, while for boxes only the ball setups were used.

The "degrees of freedom" in the NERML algorithms, specifically, the policies for updating the prox-centers and the localizers, were resolved as follows:

1. The prox-center c_s for phase s was the best (in terms of the objective) solution we have at our disposal at the beginning of the phase.
2. The localizers X_t were cut off X by at most a given number m of linear inequalities. The policy for handling these inequalities was as follows. Let $u_1, u_2, ...$ be the subsequent search points generated by the algorithm, and let $q_i(x) = f(u_i) + (x - u_i)^T f'(u_i)$.
   (a) *Before $q_m(\cdot)$ is built*, localizer $X_{t-1} = X_{t-1}^s$ (where s is the phase #, t is the # of a step within phase s) is $\{x \in X : h_i(x) \equiv q_i(x) - \ell_s \leq 0,\ i \in I_{t-1}^s\}$, where I_{t-1}^s is the set of indices of $q_i(\cdot)$'s we have built till the beginning of the step t of phase s.
   (b) *After $q_m(\cdot)$ is built*, $X_{t-1}^s = \{x \in X : h_i(x) \equiv g_i(x) - \ell_s \leq 0,\ i \in I_{t-1}^s, h(x) \equiv (x_{t-1} - x)^T \omega_s'(x_{t-1}) \leq 0\}$, where I_{t-1}^s is the set of indices of the $m-1$ *latest* $q_i(\cdot)$'s built till the beginning of the step t of phase s.

   Note that this policy clearly satisfies (16). Moreover, it is immediately seen that with this policy one can replace the auxiliary problem

$$\tilde{f} = \min_x \left\{ g_{t-1}(x) \equiv f(x_{t-1}) + (x - x_{t-1})^T f'(x_{t-1}) : x \in X_{t-1} \right\}, \qquad (\mathrm{L}_{t-1})$$

   responsible for updating the lower bounds on the optimal value in (1), with the problem

$$\tilde{f} = \min_x \left\{ \max[\max_{i \in I_{t-1}^s} q_i(x), g_{t-1}(x)] : x \in X_{t-1} \right\},$$

   thus improving the bounding of f_* from below. We have used this option on our experiments.

Two "polar" policies were tested – the "memoryless" one ($m = 1$), and the "long memory" policy $m = 30$. With the former policy, the algorithms were allowed to run 100 iterations (i.e., to compute f and f' at 100 points), with the latter one – only 40 iterations.

The auxiliary problems (L_{t-1}), (P_{t-1}) were solved by the Level method [8] [5)].

Control parameters. In all our experiments, the parameter θ was set to 0.5. The value of the remaining control parameter λ was somehow adjusted to the type of test problems and never changed in the sequel. Specifically, we used $\lambda = 0.9$ for the UFL test problems, and $\lambda = 0.95$ for the Tomography ones.

Notation in the tables. In the tables, the versions of the algorithms are encoded as `XYYZ`, where

- `X` is either `B` (for ball setup), or `S` (for simplex setup),
- `YY` is the "memory depth" m (either 30, or 01),
- `Z` is either `b` (for NERML), or `g` (for GNERML).

For example, `B01g` denotes the GNERML scheme with the ball setup and no memory.

For every experiment, we display the best values of the objective found at the first iteration (i.e., the value at the starting point), and iterations ## 10, 20, 30, 40 (for the versions with memory depth 30, where 40 iterations were run), or ## 10, 20, 30, 40, 100 (for the versions with no memory, where 100 iterations were run). These values are displayed in the row where the name of the method stands; the values in the subsequent row are the gaps (i.e., the differences between the best value of the objective found so far and the current lower bound on the optimal value, for the basic version, and minus the lower bounds for the quantities $\epsilon_T = g_*[\{x_0, \ldots, x_T\}]$ for the GNERML scheme). Besides these data, we present

- The *progress in the gap* – the ratio `PrgG`$= \dfrac{\mathrm{Gap}_{\mathrm{ini}}}{\mathrm{Gap}_{\mathrm{fin}}}$ of the initial and the final gaps;
- The *progress in the accuracy* `PrgA`$= \dfrac{f^{\mathrm{ini}} - f_{\mathrm{fin}}}{f^{\mathrm{fin}} - f_{\mathrm{fin}}}$, where f^{ini} is the value of the objective at the starting point, f^{fin} is the best value of the objective found in course the run, and f_{fin} is the largest – the last – lower bound on the optimal value found in course of the run. Progress in accuracy is reported for the basic version of the NERML algorithm only (since in the GNERML scheme, no explicit lower bounds on the optimal value are built).
- The CPU time.

All experiments were carried out on Pentium IV 1.3 GHz PC with 256 Mb RAM.

5.1 UFL problems

An Uncapacitated Facility Location problem is the Boolean Programming program as follows:

[5)] Although theoretically slow (with the complexity bound $O(\epsilon^{-2})$), the Level method, as many other bundle algorithms with "full memory", in practice exhibits nice polynomial time convergence: empirically, inaccuracy in terms of the objective goes to 0 at least as fast as $\exp\{-k/m\}$, where m is the design dimension of the problem and k is the number of steps. As a result, in practice the Level method significantly outperforms its "theoretically superior" alternatives, like the Ellipsoid algorithm, provided that the design dimension of the problem is about 5 or more.

$$\min_{x,y}\left\{\sum_{i,j\le n} d(i,j)x_{ij}+\sum_{j\le n} c_j y_j : x_{ij}, y_j \in \{0;1\}, \sum_j x_{ij}=1,\ x_{ij}\le y_{ij}\right\}, \quad (62)$$

where $c_j > 0$ and $d(i, j)$ is a metric on the n-point set $\{1, \ldots, n\}$. Informally, there are n locations of clients to be served. At a location j, a service facility can be installed at the cost c_j. Given the locations of the installed facilities, the clients assign themselves to exactly one facility each. The goal is to decide where to install facilities ($y_j = 1$ iff at the location j a facility is installed) and how to assign the clients to the facilities ($x_{ij} = 1$ iff client i is served by the facility at the location j) in order to minimize the sum of the installation cost $\sum_j c_j y_j$ plus the total service cost $\sum_{i,j} d(i, j)x_{ij}$. UFL is an NP-hard problem. The UFL test problems we dealt with are LP relaxations of (62), specifically, the problems

$$\min_{x,y}\left\{\sum_{i,j\le n} d(i,j)x_{ij}+\sum_{j\le n} c_j y_j : 0\le x_{ij}\le y_j\le 1, \sum_j x_{ij}=1,\ i=1,\ldots,n\right\}. \quad (63)$$

It is shown in [2] that the optimal value of the relaxation (63) coincides, within the factor $1 + 2/3$, with the optimal value of the combinatorial problem.

(63) is just a Linear Programming program with $n^2 + n$ variables; however, when n is few thousands, this program becomes too large (tens of millions of variables) to be solved straightforwardly by the usual LP solvers. Fortunately, the design dimension of (63) can be reduced dramatically by eliminating the x_{ij}-variables: for fixed $y_j \ge 0$ with $\sum_j y_j \ge 1$ one has

$$\min_{x_{ij}}\left\{\sum_{i,j} d(i,j)x_{ij} : 0\le x_{ij}\le y_j, \sum_j x_{ij}=1\right\} = \sum_i \phi_i(y),$$

where $\phi_i(y)$ is the easily computable optimal value in the continuous knapsack problem:

$$\phi_i(y) = \min_u\left\{\sum_j d(i,j)u_j : 0\le u_j\le y_j, \sum_j u_j = 1\right\}.$$

Eliminating x_{ij}, we convert (63) into a nonsmooth convex program

$$\min_y\left\{\sum_i \phi_i(y)+\sum_j c_j y_j : 0\le y_j\le 1, \sum_j y_j\ge 1\right\}. \quad (64)$$

with "only" n variables. We may further extend the feasible domain of (65) to the entire box $\{y : 0 \le y_j \le 1\}$ via "penalizing" the constraint $\sum_j y_j \ge 1$, thus arriving at the problem

$$\min_y \left\{ \sum_i \widehat{\phi}_i(y) + \sum_j c_j y_j : 0 \le y_j \le 1 \right\},$$

$$\widehat{\phi}_i(y) = \min_{u,v} \left\{ \sum_{j=1}^n d(i,j)u_j + D_i v : 0 \le u_j \le y_j,\ j = 1, \ldots, n, 0 \le v,\ \sum_j u_j + v = 1 \right\} \tag{65}$$

where D_i are "big" penalties (it suffices to take $D_i = \max_j (d_{ij} + c_j)$).

Problems (65) were exactly the ULF test problems we used. Following the experiments reported in [2], the data for these problems were generated as follows:

- the n locations $1, \ldots, n$ were chosen at random according to the uniform distribution in the unit square, with the usual Euclidean metric in the role of $d(i, j)$;
- all installation costs c_j were set to $0.1\sqrt{n}$.

Note that with this setup, it is easy to get a "nontrivial" a priori upper bound on the quantity $\sum_j y_j^*$ at an optimal solution y^* to (65). For example, the objective at the feasible solution $y_1 = 1, y_2 = \ldots = y_n = 0$ is at most $\sqrt{2}n$, therefore we should have $0.1\sqrt{n}\sum_j y_j^* = \sum_j c_j y_j^* \le n\sqrt{2}$, whence $\sum_j y_j^* \le 10\sqrt{2n}$, and the resulting upper bound on $\sum_j y_j^*$, for large n, is much less than the trivial bound $\sum_j y_j^* \le n$. Now, if ℓ is a valid upper bound on $\sum_j y_j^*$, then the problem

$$\min_y \left\{ \sum_i \widehat{\phi}_i(y) + \sum_j c_j y_j : 0 \le y_j, \sum_j y_j \le \ell \right\} \tag{66}$$

is equivalent to (65). Thus, the UFL problems can be treated as problems of minimizing both over the box $\{0 \le y_i \le 1\}$ and over the simplex $\{y \ge 0 : \sum_j y_j \le \ell\}$. We used this possibility (with a bit more sophisticated policy for bounding $\sum_j y_j^*$ than the one we have outlined) to test the NERML methods with both ball setup (for problems (65)) and simplex setup (for problems (66)). The starting point for the methods with ball setup was the vector of ones, and for the methods with simplex setup – the vector with the coordinates ℓ/n.

The results of our experiments with the UFL problems at two randomly generated problems, with $n = 3000$ and $n = 6000$ design variables, respectively, are presented in Table 1. The conclusions from the UFL experiments are as follows:

- The basic NERML method *with ball setup* does not work at all even at the smaller problem (see the "`B30b`" row in Table 1). All remaining versions (i.e., those implementing the GNERML scheme with ball setup and all versions with the simplex setup) produce approximate solutions of nearly the same quality. Bearing in mind

Table 1. UFL problems.

Size	Method	Itr#1	Itr#10	Itr#20	Itr#30	Itr#40	Itr#100	PrgG	PrgA	CPU
	B30b	16431.68	16431.68	16431.68	16431.68	16431.68		1.0	1.0	4′57″
		1.6e4	1.6e4	1.6e4	1.6e4	1.6e4				
	B30g	16431.68	632.65	420.75	367.17	363.90		180.1		1′54″
		1.0e0	1.6e-2	8.7e-3	7.1e-3	5.6e-3				
	B01g	16431.68	758.29	377.81	364.98	364.27	361.52	171.1		0′25″
		1.0e0	4.0e-2	9.9e-3	9.9e-3	9.9e-3	5.8e-3			
3000	**S30b**	525.88	374.19	363.30	361.66	361.04		124.5	65.1	2′35″
		3.2e2	3.5e1	7.3e0	3.9e0	2.6e0				
	S30g	525.88	369.59	364.86	363.76	362.55		43.7		2′47″
		9.1e-1	1.5e-1	3.5e-2	2.9e-2	2.1e-2				
	S01b	525.88	374.08	363.54	362.26	361.30	361.05	15.3	8.9	0′27″
		3.2e2	3.4e1	2.3e1	2.2e1	2.1e1	2.1e1			
	S01g	525.88	410.91	366.11	364.53	364.53	362.16	10.9		0′26″
		9.1e-1	4.0e-1	1.7e-1	1.7e-1	1.7e-1	8.3e-2			
	B30g	46475.80	1626.12	971.38	695.81	652.22		295.1		3′3″
		1.0e0	1.6e-2	8.3e-3	4.7e-3	3.4e-3				
	B01g	46475.80	2000.16	763.89	653.13	650.47	646.29	382.0		1′4″
		1.0e0	4.0e-2	1.0e-2	6.0e-3	6.0e-3	2.6e-3			
6000	**S30b**	950.94	667.65	648.57	646.52	645.94		179.1	94.3	4′51″
		5.9e2	6.8e1	1.1e1	4.7e0	3.3e0				
	S30g	950.94	659.37	650.73	647.90	646.96		72.5		4′37″
		9.2e-1	1.5e-1	3.1e-2	1.7e-2	1.3e-2				
	S01b	950.94	668.26	651.03	647.82	647.55	646.02	15.0	8.8	1′16″
		5.9e2	7.2e1	5.4e1	5.1e1	5.1e1	3.9e1			
	S01g	950.94	725.50	653.12	649.20	649.20	646.29	16.7		1′14″
		9.2e-1	3.8e-1	1.2e-1	1.2e-1	1.2e-1	5.5e-2			

that we are executing at most 100 iterations of a first-order method to solve non-smooth convex programs with 3,000 – 6,000 variables coming from LPs with 9,000,000 – 36,000,000 variables, this quality should be qualified as quite satisfactory.

- Among the methods which did work, the clear winner was the NERML with memory depth 30 and simplex setup: it produced the best approximations to the optimal values and the best accuracy guarantees (the smallest gaps in terms of the objective[6)]).
- As far as the proximity to the optimal value is concerned, the 100-iteration basic memoryless methods with simplex setup were nearly as good as their 40-iteration counterparts with memory depth 30, while being approximately 4 times faster in terms of CPU time. As a compensation, the methods with memory were capable "to realize" that they are close to the optimal value. For example, S30b, as applied to the UFL instance with 6,000 locations, reaches in 40 iterations objective's value 645.94 and "knows" that it is within 0.5% of the true optimal value (the final optimality gap reported by the method is 3.3). In contrast to this, S01b on the same instance ends up with nearly the same value 646.02 of the objective, but reports an optimality gap as large as 39.0 (6% of the optimal value).

[6)] To avoid misunderstandings, recall that the gaps reported for the GNERML scheme are minus lower bounds on the "artificial" quantities $g_*[\{x_0, \ldots, x_T\}]$, and not the actual upper bounds on the difference between the best found so far value of the objective and a lower bound on its optimal value. When converted to gaps in terms of the objective according to the recipe from Proposition 4.4.(i), the "g–gaps" become pretty large, like 15-30% of the optimal value

5.2 TOMO problems

The 2D PET (Positron Emission Tomography) imaging problems are as follows. Consider a square plate on the 2D plane partitioned into $n = k \times k$ small squares – *pixels*, filled with a radio-active tracer; let λ_j be the density of the tracer in pixel j. When disintegrating, the tracer emits positrons; every positron annihilates a nearby electron to produce a pair of photons flying at the speed of light in opposite directions along a line ("line of response") with completely random orientation passing through the disintegration point. The plate is encircled by a ring of detectors; when two detectors are (nearly) simultaneously hit by photons, this event is registered, meaning that along certain line crossing both the detectors a disintegration event occurred. The data collected in a tomography study is the collection of the events registered by each *bin* (a pair of detectors), and the problem is to recover from this data the density λ of the tracer at each pixel.

Mathematically speaking, the number y_i of events registered during time t in a bin $\#i$ is a realization of the Poisson random variable with the expectation $t(P\lambda)_i$, where P is a known matrix with nonnegative entries p_{ij} (the probability for a line of response originating in pixel j to be registered in bin i); the random variables y_i with different i's are independent of each other. Estimating λ by the Maximum Likelihood estimator, one ends up with the optimization problem

$$\min_{\lambda \geq 0} \left\{ \sum_i p_j \lambda_j - \sum_i y_i \ln \left(\sum_j p_{ij} \lambda_j \right) \right\}, \quad p_j = \sum_i p_i.$$

From the KKT optimality conditions it follows that every optimal solution λ must satisfy the relation $\sum_j p_j \lambda_j = B \equiv \sum_i y_i$; we lose nothing by adding this constraint, thus arriving at a problem

$$\min_{\lambda \geq 0, \sum_j p_j \lambda_j = B} \left\{ \sum_i p_j \lambda_j - \sum_i y_i \ln \left(\sum_j p_{ij} \lambda_j \right) \right\};$$

passing to the scaled variables $x_j = p_j \lambda_j / B$ and new parameters $q_{ij} = p_{ij} B p_j^{-1}$, we end up with the problem

$$f_* = \min_{x \in \Delta_n} \left\{ f(x) \equiv - \sum_{i=1}^m y_i \ln \left(\sum_j q_{ij} x_j \right) \right\}. \tag{67}$$

In our experiments, we have simulated the tomography data according to the outlined model of the tomography device and then solved the associated problems (67). Below we present the results of two experiments of this type:

- "129 × 360" – 129×129 pixel grid and 360 detectors, which corresponds to $n = 16,641$ design variables and $m = \frac{360 \cdot 359}{2} = 64,630$ log-terms in the objective;
- "257 × 360" – 257×257 pixel grid and 360 detectors (design dimension $n = 66,049$, with $m = 64,630$ log-terms in the objective).

Table 2. TOMO problems.

Size	Method	Itr#1	Itr#10	Itr#20	Itr#30	Itr#40	Itr#100	PrgG	PrgA	CPU
	S30b	10.2478	10.0025	9.8920	9.8429	9.8267		99.7	45.0	6′57″
		9.6e-1	2.9e-1	1.5e-1	5.4e-2	9.6e-3				
	S30g	10.247763	9.9663	9.8613	9.8269	9.8258		134.8		7′21″
		4.9e-1	1.9e-1	7.9e-2	1.5e-2	3.6e-3				
	S01b	10.2478	10.0026	9.8915	9.8368	9.8262	9.8257	170.2	76.2	3′23″
		9.6e-1	2.9e-1	1.5e-1	8.0e-2	2.6e-2	5.6e-3			
129x360	S01g	10.2478	9.9664	9.8614	9.8265	9.8258	9.8258	85.0		3′24″
		4.9e-1	1.9e-1	7.9e-2	3.7e-2	1.9e-2	5.7e-3			
$f_* = 9.8256$	B30b	10.2478	9.9256	9.8811	9.8811	9.8811		6.7	3.6	11′11″
$n = 16,641$		9.6e-1	2.0e-1	1.4e-1	1.4e-1	1.4e-1				
	B30g	10.2478	9.9229	9.8322	9.8257	9.8257		313.8		8′41″
		4.9e-1	1.7e-1	4.8e-2	3.2e-3	1.6e-3				
	B01b	10.2478	9.9256	9.8937	9.8937	9.8937	9.8937	6.2	3.3	3′38″
		9.6e-1	2.0e-1	1.6e-1	1.6e-1	1.6e-1	1.6e-1			
	B01g	10.2478	9.9314	9.8355	9.8259	9.8259	9.8259	26.0		4′59″
		4.9e-1	1.8e-1	5.5e-2	2.7e-2	2.1e-2	1.9e-2			
	S30b	10.2442	10.0001	9.8899	9.8406	9.8256		128.7	57.5	26′39″
		9.5e-1	2.9e-1	1.5e-1	5.4e-2	7.4e-3				
	S30g	10.2442	9.9638	9.8593	9.8258	9.8254		149.8		27′12″
		4.9e-1	1.9e-1	8.0e-2	1.3e-2	3.3e-3				
	S01b	10.2442	10.0000	9.8893	9.8346	9.8256	9.8254	72.4	32.8	11′53″
		9.5e-1	2.9e-1	1.5e-1	8.1e-2	2.3e-2	1.3e-2			
257x360	S01g	10.2442	9.9638	9.8593	9.8238	9.8255	9.8253	42.0		11′28″
		4.9e-1	1.9e-1	8.0e-2	2.3e-2	2.3e-2	1.2e-2			
$f_* < 9.8254$	B30b	10.2442	10.0385	10.0385	10.0385	10.0385		2.3	1.5	15′26″
$n = 66,049$		9.5e-1	4.1e-1	4.1e-1	4.1e-1	4.1e-1				
	B30g	10.2442	9.9899	9.9899	9.9899	9.9899		2.0		7′9″
		4.9e-1	2.5e-1	2.5e-1	2.5e-1	2.5e-1				
	B01b	10.2442	10.0385	10.0385	10.0385	10.0385	10.0385	2.3	1.5	9′51″
		9.5e-1	4.1e-1	4.1e-1	4.1e-1	4.1e-1	4.1e-1			
	B01g	10.2442	9.9899	9.9899	9.9899	9.9899	9.9899	2.0		9′42″
		4.9e-1	2.5e-1	2.5e-1	2.5e-1	2.5e-1	2.5e-1			

The first of the above experiments corresponds to "infinite" observation time – to the noiseless case when $y_i = (P\lambda)_i$. In this case, the optimal value of the objective is known in advance, provided that we know the true image λ^* (which we do know in our simulated experiments); indeed, it is immediately seen that with noiseless observations, the true image is an optimal solution to (67). In the second experiment, the observation time was such that every pixel with unit density of the tracer emitted, in course of the measurements, 40 positrons on the average. For this experiment, the true image λ^* provides us with no more than an upper bound on the optimal value.

Note that the objective in (67) is undefined at a part of the relative boundary of Δ_n and is not Lipschitz continuous on Δ_n [7]. In order to avoid potential numerical difficulties, we have replaced the terms $\ln(\sum_j q_{ij}x_j)$ in the objective with their "regularizations" $\ln(10^{-16}+\sum_j q_{ij}x_j)$. The starting point for all methods was the barycenter of the simplex Δ_n.

The results of our experiments are reported in Table 2; Fig. 2, 3 display the true image and its reconstructions as produced by the methods (the higher is the density, the brighter is the image). The conclusions from the TOMO experiments are as follows:

[7] It should be mentioned that f is semi-bounded on rint Δ_n: $\mathbf{V}_{\text{rint }\Delta_n, c, \frac{1}{n}}[\partial f] \le mn$; this fact, however, is of no practical importance, since the right-hand side in this inequality, although finite, is really huge.

- The versions with simplex setup clearly outperform those with ball setup. Among the methods of the latter group, only `g`-ones (i.e., the GNERML algorithms) were working properly on the smaller problem, and none was working well on the larger problem. This is clearly seen from the data in Table 2 and especially from the pictures on Fig. 2, 3. In contrast to this, all versions with the simplex setup performed pretty well on both problems, reaching inaccuracy in terms of the objective varying from 1.0e-4 (`S01b` as applied to the smaller problem) to 7.3e-3. The advantages of the methods with simplex setup are in full accordance with our theoretical complexity analysis and demonstrate the importance of "adjusting" subgradient-type methods to the geometry of problems to be solved.
- Among the methods with simplex setup, the clear winner on the smaller problem was the basic memoryless method `S01b` – it arrives at the best value of the objective, reports the smallest optimality gap and is the fastest in terms of the CPU time. On a larger problem, this method essentially keeps its superiority in terms of the value of the objective and the CPU time, but reports a relatively big optimality gap, namely, 1.3e-2 vs. the gap 7.4e-3 for `S30b`, thus sharing the "top quality place" with the basic NERML method with memory.

We believe that the numerical results we have presented demonstrate the significant potential of properly adjusted "simple" optimization techniques, such as NERML and GNERML, to solve very large-scale convex programs.

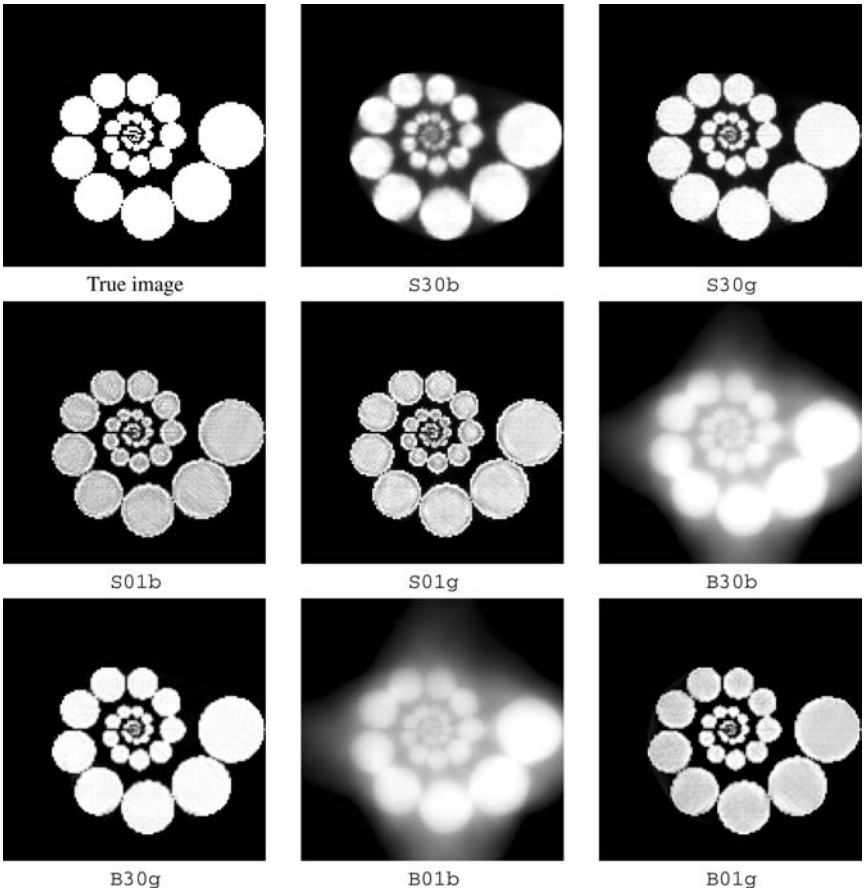

Fig. 2. TOMO 129×360.

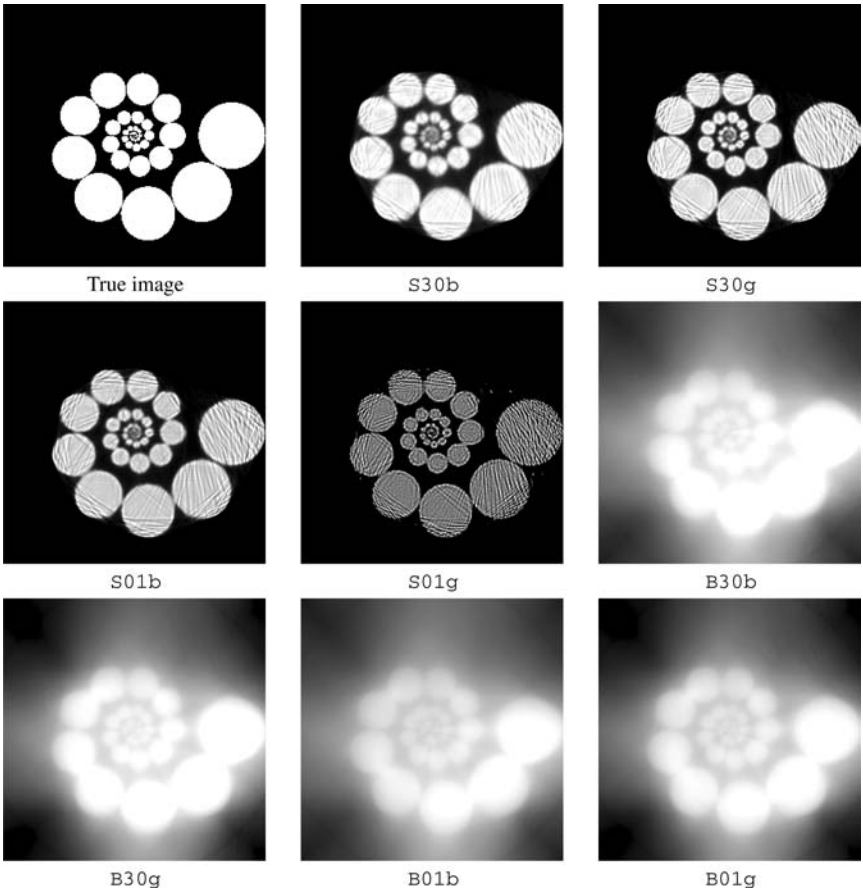

Fig. 3. TOMO 257×360.

References

1. Ben-Tal, A., Margalit, T., Nemirovski, A.: The Ordered Subsets Mirror Descent Optimization Method with Applications to Tomography. SIAM Journal on Optimization **12**, 79–108 (2001)
2. Chudak, F.A.: Improved approximation algorithms for uncapacitated facility location. Lecture Notes on Computer Science **1412**, 180–192 (1998)
3. Kiwiel, K.: An aggregate subgradient method for nonsmooth convex minimization. Mathematical Programming **27**, 320–341 (1983)
4. Kiwiel, K.: Proximal level bundle method for convex nondifferentable optimization, saddle point problems and variational inequalities. Mathematical Programming Series B **69**, 89–109 (1995)
5. Kiwiel K.C., Larson, T., Lindberg, P.O.: The efficiency of ballstep subgradient level methods for convex optimization. Mathematics of Operations Research **24**, 237–254 (1999).
6. Lemaréchal, C.: Nonsmooth optimization and descent methods. Research Report 78–4, IIASA, Laxenburg, Austria, 1978
7. Lemaréchal, C., Strodiot, J.J., Bihain, A.: On a bundle algorithm for nonsmooth optimization. In: O.L. Mangasarian, R.R. Meyer, S.M. Robinson, (eds.), Nonlinear Programming 4 (Academic Press, NY, 1981), pp. 245–282
8. Lemaréchal, C., Nemirovski, A., Nesterov, Yu.: New variants of bundle methods. Mathematical Programming Series B **69**, 111–148 (1995)

9. Mifflin, R.: A modification and an extension of Lemaréchal's algorithm for nonsmooth minimization. Mathematical Programming Study **17**, 77–90 (1982)
10. Nemirovski, A., Yudin, D.: Problem complexity and method efficiency in optimization, J. Wiley & Sons, 1983
11. Nesterov, Yu.: Cutting plane algorithms from analytic centers: complexity estimate. Mathematical Programming **65**, 149–176 (1995)
12. Polyak, B.T.: A general method for solving extremal problems. Soviet Math. Doklady **174**, 33–36 (1967)
13. Shor, N.Z.: Generalized gradient descent with application to block programming. Kibernetika No. 3 (1967) (in Russian)
14. Schramm, H., Zowe, J.: A version of bundle idea for minimizing a non-smooth function: conceptual idea, convergence analysis, numerical results. SIAM Journal on Optimization **2**, 121–152 (1992)

6 Appendix: Proofs

6.1 Proof of Theorem 2.1

(i): Assume that phase s did not terminate in course of N steps. Observe that then

$$\|x_t - x_{t-1}\| \geq \frac{\theta(1-\lambda)\epsilon_s}{L_{\|\cdot\|}(f)}, \quad 1 \leq t \leq N. \tag{68}$$

Indeed, we have $g_{t-1}(x_t) \leq \ell_s$ by construction of x_t and $g_{t-1}(x_{t-1}) = f(x_{t-1}) > \ell_s + \theta(f^s - \ell_s)$, since otherwise the phase would be terminated at the step $t-1$. It follows that $g_{t-1}(x_{t-1}) - g_{t-1}(x_t) > \theta(f^s - \ell_s) = \theta(1-\lambda)\epsilon_s$. Taking into account that $g_{t-1}(\cdot)$, due to (6), is Lipschitz continuous on X w.r.t. $\|\cdot\|$ with constant $L_{\|\cdot\|}(f)$, (68) follows.

Now observe that x_{t-1} is the minimizer of ω_s on X_{t-1} by (13.a_{t-1}), and the latter set, by construction, contains x_t, whence $(x_t - x_{t-1})^T \nabla\omega_s(x_{t-1}) \geq 0$. Applying (12), we get

$$\omega_s(x_t) \geq \omega_s(x_{t-1}) + \frac{\kappa}{2}\left(\frac{\theta(1-\lambda)\epsilon_s}{L_{\|\cdot\|}(f)}\right)^2, \ t = 1, \ldots, N,$$

whence

$$\omega_s(x_N) - \omega_s(x_0) \geq \frac{N\kappa}{2}\left(\frac{\theta(1-\lambda)\epsilon_s}{L_{\|\cdot\|}(f)}\right)^2.$$

The latter relation, due to the evident inequality $\max\limits_X \omega_s(x) - \min\limits_X \omega_s \leq \Omega$ (readily given by the definition of Ω and ω_s) implies that

$$N \leq \frac{2\Omega L^2_{\|\cdot\|}(f)}{\theta^2(1-\lambda)^2\kappa\epsilon_s^2}.$$

Recalling the origin of N, we conclude that

$$N_s \leq \frac{2\Omega L^2_{\|\cdot\|}(f)}{\theta^2(1-\lambda)^2\kappa\epsilon_s^2} + 1.$$

In order to get from this inequality the required relation (17), all we need is to demonstrate that

$$1 \leq \frac{2\Omega L^2_{\|\cdot\|}(f)}{\theta^2(1-\lambda)^2\kappa\epsilon_s^2}. \tag{69}$$

To this end, let $R = \max_{x \in X} \|x - c_1\| = \|\bar{x} - c_1\|$, where $\bar{x} \in X$. We have $\epsilon_s \le \epsilon_1 = f(c_1) - \min_{x \in X}[f(c_1) + (x - c_1)^T f'(c_1)] \le RL_{\|\cdot\|}(f)$, where the last inequality is due to (6). On the other hand, by the definition of Ω and the strong convexity of ω we have

$$\Omega \ge \omega(\bar{x}) - [\omega(c_1) + (\bar{x} - c_1)^T \nabla\omega(c_1)] \ge \frac{\kappa}{2}\|\bar{x} - c_1\|^2 = \frac{\kappa R^2}{2}.$$

Thus, $\Omega \ge \frac{\kappa R^2}{2}$ and $\epsilon_s \le RL_{\|\cdot\|}(f)$, and (69) follows. (i) is proved.

(ii): Assume that $\epsilon_s > \epsilon$ at phases $s = 1, 2, \ldots, S$, and let us bound from above the total number of oracle calls at these S phases. Observe, first, that two subsequent gaps $\epsilon_s, \epsilon_{s+1}$ are linked by the relation

$$\epsilon_{s+1} \le \gamma\epsilon_s, \ \gamma = \gamma(\theta, \lambda) \equiv \max[1 - \theta\lambda, 1 - (1 - \theta)(1 - \lambda)] < 1. \tag{70}$$

Indeed, if phase s was terminated according to the rule 2, then

$$\epsilon_{s+1} = f^{s+1} - f_{s+1} \le f^s - [\ell_s - \theta(\ell_s - f_s)] = (1 - \theta\lambda)\epsilon_s,$$

as required in (70). Otherwise phase s was terminated when relation (15) took place. In this case,

$$\begin{aligned}\epsilon_{s+1} &= f^{s+1} - f_{s+1} \le f^{s+1} - f_s \le \ell_s + \theta(f^s - \ell_s) - f_s = \lambda\epsilon_s + \theta(1 - \lambda)\epsilon_s \\ &= (1 - (1 - \theta)(1 - \lambda))\epsilon_s,\end{aligned}$$

and we again arrive at (70).

>From (70) it follows that $\epsilon_s \ge \epsilon\gamma^{s-S}$, $s = 1, \ldots, S$, since $\epsilon_S > \epsilon$ by the origin of S. We now have

$$\begin{aligned}\sum_{s=1}^{S} N_s &\le \sum_{s=1}^{S} \frac{4\Omega L^2_{\|\cdot\|}(f)}{\theta^2(1 - \lambda)^2\kappa\epsilon_s^2} \le \sum_{s=1}^{S} \frac{4\Omega L^2_{\|\cdot\|}(f)\gamma^{2(S-s)}}{\theta^2(1 - \lambda)^2\kappa\epsilon^2} \le \frac{4\Omega L^2_{\|\cdot\|}(f)}{\theta^2(1 - \lambda)^2\kappa\epsilon^2} \sum_{t=0}^{\infty} \gamma^{2t} \\ &\equiv \underbrace{\left[\frac{4}{\theta^2(1 - \lambda)^2(1 - \gamma^2)}\right]}_{c(\theta,\lambda)} \frac{\Omega L^2_{\|\cdot\|}(f)}{\kappa\epsilon^2}\end{aligned}$$

and (19) follows. □

6.2 Strong convexity of $\omega(\cdot)$ for standard setups

The case of the ball setup is trivial.

The case of the simplex setup: For a C^2 function $\omega(\cdot)$, a sufficient condition for (7) is the relation

$$h^T\omega''(x)h \geq \kappa\|h\|^2 \quad \forall(x,h: x, x+h \in X). \tag{71}$$

For the simplex setup, we have

$$\begin{aligned}\|h\|_1^2 &= \left[\sum_i |h_i|\right]^2 = \left[\sum_i \frac{|h_i|}{\sqrt{x_i+\delta n^{-1}}}\sqrt{x_i+\delta n^{-1}}\right]^2 \\ &\leq \left[\sum_i (x_i+\delta n^{-1})\right]\left[\sum_i \frac{h_i^2}{x_i+\delta n^{-1}}\right] \leq (1+\delta)h^T\omega''(x)h,\end{aligned}$$

and (71) indeed is satisfied with $\kappa = (1+\delta)^{-1}$.

To prove (21), note that for all $x, y \in \Delta_n^+ \supset X$, setting $\bar{x} = x + \delta n^{-1}(1,\ldots,1)^T$, $\bar{y} = y + \delta n^{-1}(1,\ldots,1)^T$, one has

$$\begin{aligned}&\omega(y) - \omega(x) - (y-x)^T\nabla\omega(x) \\ &= \sum_i [\bar{y}_i \ln(\bar{y}_i) - \bar{x}_i\ln(\bar{x}_i) - (\bar{y}_i - \bar{x}_i)(1+\ln(\bar{x}_i))] \\ &= -\sum_i(\bar{y}_i - \bar{x}_i) + \sum_i \bar{y}_i \ln(\frac{\bar{y}_i}{\bar{x}_i}) \\ &\leq 1+\delta + \sum_i \bar{y}_i \ln(\frac{\bar{y}_i}{\bar{x}_i}) \qquad [\text{since } \bar{y}_i \geq 0 \text{ and } \sum_i \bar{x}_i \leq 1+\delta] \\ &\leq 1+\delta + \sum_i \bar{y}_i \ln(\frac{\bar{y}_i}{\delta n^{-1}}) \qquad [\text{since } \bar{x}_i \geq \delta n^{-1}] \\ &\leq 1+\delta c + \max_z \left\{\sum_i z_i \ln(nz_i/\delta) : z \geq 0, \sum_i z_i \leq 1+\delta\right\} \quad [\text{since } \sum_i \bar{y}_i \leq 1+\delta] \\ &= (1+\delta)\left[1 + \ln\left(\frac{n(1+\delta)}{\delta}\right)\right],\end{aligned}$$

and (21) follows.

The case of the spectahedron setup: We again intend to use the sufficient condition (71) for strong convexity, but now it is a bit more involved. First of all, let us compute the second derivative of the regularized matrix entropy

$$\omega(x) = \mathrm{Tr}((x+\sigma I_n)\ln(x+\sigma I_n)) : \Sigma_n^+ \to \mathbf{R} \qquad [\sigma = \delta n^{-1}]$$

Setting $y[x] = x + \sigma I_n$,

$$f(z) = z\ln z$$

(z is a complex variable restricted to belong to the open right half-plane, and $\ln z$ is the principal branch of the logarithm in this half-plane), in a neighbourhood of a given point $\bar{x} \in \Sigma_n^+$ we have, by Cauchy's integral formula,

$$Y(x) \equiv y[x]\ln(y[x]) = \frac{1}{2\pi i}\oint_\gamma f(z)(zI_n - y[x])^{-1}dz, \tag{72}$$

where γ is a closed contour in the right half-plane with all the eigenvalues of $y[\bar{x}]$ inside the contour. Consequently,

$$\begin{aligned} DY(x)[h] &= \frac{1}{2\pi i}\oint_\gamma f(z)(zI_n - y[x])^{-1}h(zI_n - y[x])^{-1}dz, \\ D^2Y(x)[h,h] &= \frac{1}{\pi i}\oint_\gamma f(z)(zI_n - y[x])^{-1}h(zI_n - y[x])^{-1}h(zI_n - y[x])^{-1}dz, \end{aligned} \tag{73}$$

whence

$$D^2\omega(\bar{x})[h,h] = \mathrm{Tr}\left(\frac{1}{\pi i}\oint_\gamma f(z)(zI_n - y[x])^{-1}h(zI_n - y[x])^{-1}h(zI_n - y[x])^{-1}dz\right).$$

Passing to the eigenbasis of $y[\bar{x}]$, we may assume that $y[\bar{x}]$ is diagonal with positive diagonal entries $\mu_1 \le \mu_2 \le ... \le \mu_n$. In this case the formula above reads

$$D^2\omega(\bar{x})[h,h] = \frac{1}{\pi i}\sum_{p,q=1}^n \oint_\gamma h_{pq}^2 \frac{f(z)}{(z-\mu_p)^2(z-\mu_q)}dz. \tag{74}$$

Computing the residuals of the integrands at their poles, we get

$$D^2\omega(\bar{x})[h,h] = \sum_{p,q=1}^n \frac{\ln(\mu_p) - \ln(\mu_q)}{\mu_p - \mu_q}h_{pq}^2, \tag{75}$$

where, by convention, the expression $\frac{\ln(\mu_p)-\ln(\mu_q)}{\mu_p-\mu_q}$ with $\mu_p = \mu_q$ is assigned the value $\frac{1}{\mu_p}$. Since $\ln(\cdot)$ is concave, we have $\frac{\ln(\mu_p)-\ln(\mu_q)}{\mu_p-\mu_q} \ge \frac{1}{\max[\mu_p,\mu_q]}$, so that

$$D^2\omega(\bar{x})[h,h] \ge \sum_{p,q=1}^n \frac{1}{\max[\mu_p,\mu_q]}h_{pq}^2 = \sum_{p=1}^n \frac{1}{\mu_p}\left[h_{pp}^2 + 2\sum_{q=1}^{p-1}h_{pq}^2\right]. \tag{76}$$

It follows that

$$\left(\sum_{p=1}^{n}\sqrt{h_{pp}^2+2\sum_{q=1}^{p-1}h_{pq}^2}\right)^2 = \left(\sum_{p=1}^{n}\frac{\sqrt{h_{pp}^2+2\sum\limits_{q=1}^{p-1}h_{pq}^2}}{\sqrt{\mu_p}}\sqrt{\mu_p}\right)^2$$
$$\leq \left(\sum_{p=1}^{n}\frac{h_{pp}^2+2\sum\limits_{q=1}^{p-1}h_{pq}^2}{\mu_p}\right)\left(\sum_{p=1}^{n}\mu_p\right)$$
$$\leq D^2\omega(\bar{x})[h,h]\mathrm{Tr}(y[\bar{x}]) \quad \text{[see (76)]}$$
$$\leq (1+\delta)D^2\omega(\bar{x})[h,h]. \tag{77}$$

Note that we have

$$h=\sum_{p=1}^{n}h^p, \quad (h^p)_{rs}=\begin{cases}h_{rs}, & (r\leq p\,\&\,s=p) \text{ or } (r=p\,\&s\,\leq p)\\ 0, & \text{otherwise.}\end{cases}$$

Every matrix h^p is of the form $h_{pp}e_pe_p^T+r_pe_p^T+e_pr_p^T$, where $r_p=(h_{1p},\ldots,h_{p-1,p},0,\ldots,0)^T$ and e_p are the standard basic orths. From this representation it is immediately seen that

$$|h^p|_1=\sqrt{h_{pp}^2+4\|r_p\|_2^2}\leq\sqrt{2}\sqrt{h_{pp}^2+2\sum_{q=1}^{p-1}h_{pq}^2},$$

whence

$$|h|_1\leq\sum_{p=1}^{n}|h^p|_1\leq\sqrt{2}\sum_{p=1}^{n}\sqrt{h_{pp}^2+2\sum_{q=1}^{p-1}h_{pq}^2}.$$

Combining this relation with (77), we get

$$D^2\omega(\bar{x})[h,h]\geq\frac{1}{2(1+\delta)}|h|_1^2,$$

so that (71) is satisfied with $\kappa=0.5(1+\delta)^{-1}$.

Now let us bound Ω. Let $x,y\in\Sigma_n^+$, let $\bar{x}=x+\sigma I_n$, $\bar{y}=y+\sigma I_n$, $\sigma=\delta n^{-1}$, and let ξ_p, η_p be the eigenvalues of $\bar{x}$ and $\bar{y}$, respectively. For a contour γ in the open right half-plane such that all ξ_p are inside γ we have (cf. (72) – (73)):

$$D\omega(x)[h]=\mathrm{Tr}\left(\frac{1}{2\pi i}\oint\limits_{\gamma}f(z)(zI_n-\bar{x})^{-1}h(zI_n-\bar{x})^{-1}dz\right).$$

We lose nothing by assuming that $\bar{x}$ is a diagonal matrix; in this case, the latter equality implies that

$$D\omega(x)[h] = \sum_{p=1}^{n} \frac{1}{2\pi i} \oint_{\gamma} f(z)(z-\xi_p)^{-2} h_{pp} dz,$$

whence, computing the residuals of the integrands,

$$D\omega(x)[h] = \sum_{p} (1 + \ln(\xi_p)) h_{pp}.$$

It follows that

$$\begin{aligned}
&\omega(y) - \omega(x) - D\omega(x)[y-x] \\
&\quad = \omega(y) - \sum_p \xi_p \ln \xi_p - \sum_p (1 + \ln(\xi_p))(\bar{y}_{pp} - \xi_p) \\
&\quad = \omega(y) - \sum_p \bar{y}_{pp} \ln(\xi_p) + \sum_p (\xi_p - \bar{y}_{pp}) \\
&\quad \le 1 + \delta + \omega(y) - \sum_p \bar{y}_{pp} \ln(\xi_p) \quad [\text{since } \mathrm{Tr}(\bar{x}) \le 1+\delta,\ \mathrm{Tr}(\bar{y}) \ge 0] \\
&\quad = 1 + \delta + \sum_p \eta_p \ln(\eta_p) + \sum_p \bar{y}_{pp} \ln(1/\xi_p) \\
&\quad \le 1 + \delta + (1+\delta)\ln(1+\delta) + \ln(1/\sigma) \sum_p \bar{y}_{pp} \\
&\qquad [\text{since } \eta \ge 0, \textstyle\sum_p \eta_p = \mathrm{Tr}(\bar{y}) \le 1+\delta \text{ and } 1/\xi_p \le 1/\sigma] \\
&\quad = 1 + \delta + (1+\delta)\ln(1+\delta) + (1+\delta)\ln(n/\delta) = (1+\delta)\left[1 + \ln\left(\frac{n(1+\delta)}{\delta}\right)\right].
\end{aligned}$$

The resulting inequality implies (21).

6.3 Proof of Proposition 4.1

All statements, except for stability w.r.t. c, are straightforward consequences of definitions. Here is the verification of 4.1. Let $L = \mathbf{V}_{X,c,\Theta}[F]$. Clearly, it suffices to prove that if $c' - c = \gamma(z_+ - c)$, $c - c' = \gamma(z_- - c)$ with $z_+, z_- \in X$, and $\Theta'' = \frac{\Theta-\gamma}{1-\gamma}$, then

$$\mathbf{V}_{X,c',\Theta''}[F] \le L. \tag{78}$$

Indeed, let $x, y \in X$. We have

$$\begin{aligned}
c' - \Theta''(c' - y) &= c + \gamma(z_+ - c) - \Theta''(c - y) - \Theta''\gamma(z_+ - c) \\
&= c - \Theta(c - w_+), \\
w_+ &= \left(1 - \frac{\gamma(1-\Theta'')}{\Theta} + \frac{\Theta''}{\Theta}\right) c + \frac{\gamma(1-\Theta'')}{\Theta} z_+ + \frac{\Theta''}{\Theta} y.
\end{aligned}$$

With our Θ'', w_+ is a convex combination of the points c, z_+, y from X and is therefore a point of X, so that

$$\forall(\zeta \in F(x)) : \langle \zeta, c' - \Theta''(c' - y) - x\rangle = \langle \zeta, c - \Theta(c - w_+) - x\rangle \le L.$$

We also have

$$\begin{aligned} c' + \Theta''(c' - y) &= c - \gamma(z_- - c) + \Theta''(c - y) - \Theta''\gamma(z_- - c) \\ &= c + \Theta(c - w_-), \\ w_- &= \left(1 - \frac{\gamma(1 - \Theta'')}{\Theta} - \frac{\Theta''}{\Theta}\right) c + \frac{\gamma(1 - \Theta'')}{\Theta} z_- + \frac{\Theta''}{\Theta} y, \end{aligned}$$

whence, same as above, $w_- \in X$ and therefore

$$\forall(\zeta \in F(x)) : \langle \zeta, c' + \Theta''(c' - y) - x\rangle = \langle \zeta, c + \Theta(c - w_-) - x\rangle \le L.$$

Thus, (78) is true. □

6.4 Proof of Theorem 4.1

Assume that phase s did not terminate in course of N steps. Observe that then

$$\|u_{\tau+1} - u_\tau\| \ge \frac{(1 - \lambda)\delta_s}{M}, \quad 1 \le \tau \le N. \tag{79}$$

Indeed, by construction we have $h_\tau(u_\tau) = 0$ and $h_\tau(u_{\tau+1}) \le \ell_s \equiv -(1 - \lambda)\delta_s$. Besides this, in view of (36) $h_\tau(\cdot)$ is Lipschitz continuous, with constant M, w.r.t. $\|\cdot\|$, and (79) follows. Besides this, we have

$$\delta_{s+1} \le (1 - \lambda(1 - \theta))\delta_s. \tag{80}$$

Indeed, from the rule for terminating a phase it follows that

$$\ell_s + \delta_{s+1} \equiv \ell_s - f_{s+1} \le \theta(\ell_s - f_s) = \theta(\ell_s + \delta_s);$$

since $\ell_s = (1 - \lambda) f_s = -(1 - \lambda)\delta_s$, (80) follows. The rest of the proof repeats word by word the proof of Theorem 2.1, with (79) and (80) in the roles of (68), (70), respectively. □

6.5 Proof of Proposition 4.2

Assume that (41) holds true. Let $\hat{x} \in X$, and let

$$\bar{x} = (1 - \epsilon)\hat{x} + \epsilon c.$$

Since $g_*[\{x_0, \ldots, x_T\}] \ge -\delta$, there exists $t \le T$ such that

$$\langle g(x_t), \bar{x} - x_t\rangle \ge -\delta,$$

or, which is the same,

$$\langle h(x_t), \bar{x} - x_t \rangle \geq -\delta \|h(x_t)\|_X^*.$$

At the same time,

$$\begin{array}{rl} \langle h(x_t), c \pm \Theta(y-c) & -x_t \rangle \leq L \ \forall y \in X \\ & \Downarrow \\ \langle h(x_t), \pm(y-c) \rangle \leq \Theta^{-1} & [L + \langle h(x_t), x_t - c \rangle] \ \forall y \in X \\ & \Downarrow \\ \|h(x_t)\|_X^* \leq \Theta^{-1} & [L + \langle h(x_t), x_t - c \rangle] \end{array}$$

whence

$$\langle h(x_t), \bar{x} - x_t \rangle \geq -\underbrace{\delta\Theta^{-1}}_{\epsilon<1} [L + \langle h(x_t), x_t - c \rangle],$$

or

$$\langle h(x_t), [\bar{x} - \epsilon c] - (1-\epsilon)x_t \rangle \geq -\epsilon L,$$

or, recalling that $\bar{x} = (1-\epsilon)\hat{x} + \epsilon c$,

$$\langle h(x_t), \hat{x} - x_t \rangle \geq -\frac{\epsilon}{1-\epsilon} L.$$

Since $\hat{x} \in X$ is arbitrary, (42) follows.

Now, from (42) it follows that certain convex combination

$$\sum_{t=0}^{T} \lambda_t \langle h(x_t), x - x_t \rangle$$

of the functions $\langle h(x_t), x - x_t \rangle$ is $\geq -\frac{\epsilon}{1-\epsilon} L$ everywhere on X. Defining $\widetilde{x}_T$ according to (44) and taking into account that F is monotone, we have

$$\begin{array}{l} (x \in X, \zeta \in F(x)) \Rightarrow \\ \qquad -\dfrac{\epsilon}{1-\epsilon} L \ \leq \ \displaystyle\sum_{t=0}^{T} \lambda_t \langle h(x_t), x - x_t \rangle \leq \sum_{t=0}^{T} \lambda_t \langle \zeta, x - x_t \rangle \\ \qquad\qquad\qquad = \ \langle \zeta, x - \widetilde{x}_T \rangle, \end{array}$$

as required in (45). □

6.6 Proof of Proposition 4.3

(i): Let x_* be a minimizer of f on X. From (46) it follows that there exists $t \leq T$ such that $\langle f'(x_t), x^* - x_t \rangle = \langle g(x_t), x^* - x_t \rangle \geq -\epsilon$, whence $f(x^*) \geq f(x_t) - \epsilon$, and (47) follows. (ii) is readily given by (38) (note that we are in the situation of $M = L_{\|\cdot\|}(f)$). □

6.7 Proof of Proposition 4.4

(i): Taking into account (42), we get

$$\forall \epsilon \in (0,1): \quad \max_{0\le t\le T} \langle f'(x_t), x - x_t\rangle \ge -\frac{\epsilon}{1-\epsilon}\mathbf{V}_{X,c,\Theta}[F]$$

for all $x \in X$. It remains to note that

$$\begin{aligned}\min_{x\in X} f(x) &\ge \min_{x\in X}\max_{0\le t\le T}\left[f(x_t) + \langle f'(x_t), x - x_t\rangle\right]\\ &\ge f(x^T) + \min_{x\in X}\max_{0\le t\le T}\langle f'(x_t), x - x_t\rangle.\end{aligned}$$

(ii): Specifying the norm $\|\cdot\|$ as $\|\cdot\|_X$, we can apply Theorem 4.1 with $M = 1$ and $\kappa = \kappa[X, \omega(\cdot)]$ to get the implication

$$\forall\left(\epsilon \in (0,1), \Theta > 0, T \ge b(\lambda,\theta)\frac{\Omega[\omega(\cdot)]}{\kappa[X,\omega(\cdot)]}\frac{1}{\epsilon^2\Theta^2}\right): \quad g_*[x_0,\dots,x_T] \ge -\epsilon\Theta.$$

□

6.8 Proof of Proposition 4.5

(i): Under the premise of (i), taking into account Proposition 4.2, we get

$$\inf_{(u,v)\in X}\sum_{t=0}^{T}\lambda_t\left[\langle\xi_t, u - u_t\rangle - \langle\eta_t, v - v_t\rangle\right] \ge -\frac{\epsilon}{1-\epsilon}\mathbf{V}_{X,c,\Theta}[F], \tag{81}$$

where $\lambda_t = \lambda_t(T)$ are as in Proposition 4.2, and $\xi_t = f'_u(u_t, v_t)$, $\eta_t = f'_v(u_t, v_t)$. We have

$$\begin{aligned}\forall(u,v)\in X: \ &\sum_t \lambda_t\Big[\underbrace{\langle\xi_t, u - u_t\rangle}_{\le f(u,v_t)-f(u_t,v_t)} - \underbrace{\langle\eta_t, v - v_t\rangle}_{\ge f(u_t,v)-f(u_t,v_t)}\Big] \ge -\frac{\epsilon}{1-\epsilon}\mathbf{V}_{X,c,\Theta}[F]\\ &\Downarrow\\ \forall(u,v)\in X: \ &\underbrace{\sum_t \lambda_t f(u,v_t)}_{\le f\left(u,\sum_t\lambda_t v_t\right)} - \underbrace{\sum_t \lambda_t f(u_t,v)}_{\ge f\left(\sum_t\lambda_t u_t, v\right)} \ge -\frac{\epsilon}{1-\epsilon}\mathbf{V}_{X,c,\Theta}[F]\\ &\Downarrow\\ \forall(u,v)\in X: \ &f(u,\widetilde{v}_T) - f(\widetilde{u}_T, v) \ge -\frac{\epsilon}{1-\epsilon}\mathbf{V}_{X,c,\Theta}[F]\\ &\Updownarrow\\ &\underline{f}(\widetilde{v}_T) - \overline{f}(\widetilde{u}_T) \ge -\frac{\epsilon}{1-\epsilon}\mathbf{V}_{X,c,\Theta}[F]\\ &\Updownarrow\\ &\overline{f}(\widetilde{u}_T) - \underline{f}(\widetilde{v}_T) \le \frac{\epsilon}{1-\epsilon}\mathbf{V}_{X,c,\Theta}[F]\\ &\Updownarrow\\ &\epsilon(\widetilde{u}_t,\widetilde{v}_T) \le \frac{\epsilon}{1-\epsilon}\mathbf{V}_{X,c,\Theta}[F],\end{aligned}$$

where the concluding $\Updownarrow$ is given by the relation $\min\limits_{U} \overline{f} = \max\limits_{V} \underline{f}$.

(ii): Specifying the norm $\|\cdot\|$ as $\|\cdot\|_X$, we can apply Theorem 4.1 with $M = 1$ and $\kappa = \kappa[X, \omega(\cdot)]$ to get the implication

$$\forall \left(\epsilon \in (0,1), \Theta > 0, T \geq b(\lambda, \theta) \frac{\Omega[\omega(\cdot)]}{\kappa[X, \omega(\cdot)]} \frac{1}{\epsilon^2 \Theta^2} \right) : \quad g_*[x_0, \ldots, x_T] \geq -\epsilon\Theta.$$

□

6.9 Proof of Proposition 4.6

The proof is similar to the one of Proposition 4.4.

(i): Under the premise of (i), we have

$$\inf_{(u,v)\in X} \sum_{t=0}^{T} \lambda_t \left[\langle \xi_t, u - u_t \rangle - \langle \eta_t, v - v_t \rangle \right] \geq -\epsilon, \tag{82}$$

where $\lambda_t = \lambda_t(T)$ are as in Proposition 4.2, and $\xi_t = f'_u(u_t, v_t)$, $\eta_t = f'_v(u_t, v_t)$. Exactly the same computations as in the proof of Proposition 4.5, with (82) in the role of (81), result in the (61).

(ii): This is an immediate consequence of Theorem 4.1, where one should set $M = L_{\|\cdot\|}(f)$. □