ISSN 2281-4299

DIPARTIMENTO DI INGEGNERIA INFORMATICA
AUTOMATICA E GESTIONALE ANTONIO RUBERTI

SAPIENZA
UNIVERSITÀ DI ROMA

New Active-Set Frank-Wolfe Variants for Minimization over the Simplex and the L1-Ball

Andrea Cristofari
Marianna De Santis
Stefano Lucidi
Francesco Rinaldi

Technical Report n. 6, 2016

New Active-Set Frank-Wolfe Variants for Minimization over the Simplex and the ℓ_1-Ball

A. Cristofari*, M. De Santis†, S. Lucidi*, F. Rinaldi†

* Department of Computer, Control and Management Engineering
Sapienza University of Rome
Via Ariosto, 25, 00185 Roma, Italy

† Department of Mathematics
University of Padova
Via Trieste, 63, 35121 Padua, Italy

e-mail (Cristofari): cristofari@dis.uniroma1.it
e-mail (De Santis): desantis@math.unipd.it
e-mail (Lucidi): lucidi@dis.uniroma1.it
e-mail (Rinaldi): rinaldi@math.unipd.it

Abstract

In this paper, we describe a new active-set algorithmic framework for minimizing a function over the simplex. The method is quite general and encompasses different active-set Frank-Wolfe variants. In particular, we analyze convergence (when using Armijo line search in the calculation of the stepsize) for the active-set versions of standard Frank-Wolfe, away-step Frank-Wolfe and pairwise Frank-Wolfe. Then, we focus on convex optimization problems, and prove that all active-set variants converge at a linear rate under weaker assumptions than the classical counterparts. We further explain how to adapt our framework in order to handle the problem of minimizing a function over the ℓ_1-ball. Finally, we report numerical experiments showing the efficiency of the various active-set Frank-Wolfe variants.

Keywords. Active-set methods. Frank-Wolfe algorithm. Unit simplex. ℓ_1-ball.

AMS subject classifications. 65K05. 90C06. 90C30.

1 Introduction

Many real-world applications can be modeled as optimization problems over structured feasible sets. In particular, the problem of minimizing a function over a simple polytope (such as the unit simplex or the ℓ_1-ball) arises in different fields like, e.g., machine learning, statistics and economics. Examples of relevant applications include training of support vector machines, boosting (Adaboost), convex approximation in ℓ_p, mixture density estimation, lasso regression, finding maximum stable sets (maximum cliques) in graphs, portfolio optimization and population dynamics problems (see, e.g., [5, 7, 16] and references therein).

Denoting by $e = (1, \ldots, 1)^T$, the problem we address can be stated as follows:

$$\begin{array}{ll} \min\limits_{x \in \mathbb{R}^n} & f(x) \\ \text{s.t.} & e^T x = 1 \\ & x \geq 0, \end{array} \tag{1}$$

where $f \colon \mathbb{R}^n \to \mathbb{R}$ is continuously differentiable and its gradient $\nabla f(x)$ is Lipschitz continuous over the feasible set.

Note that optimizing an objective function $h(x)$ over a polytope P can be seen as problem (1). Indeed, since any point $x \in P$ can be expressed as a convex combination of the columns of $V = \begin{bmatrix} v_1 & \ldots & v_m \end{bmatrix} \in \mathbb{R}^{n \times m}$, with $v_1, \ldots, v_m$ vertices of P, problem $\min\{h(x) : x \in P\}$ can be rewritten as $\min\{h(Vy) : e^T y = 1, y \geq 0\}$. Thus, each variable y_i represents the *weight* of the i-th vertex in the convex combination.

When dealing with optimization problems over simple feasible sets, Frank-Wolfe type algorithms (see, e.g., [18] for a complete overview) guarantee good scalability, thanks to their ability to nicely handle the constraints, and also give a sparse representation of the iterates in terms of the vertices describing the feasible set. These are the reasons why, in the last few years, those methods have re-gained popularity and now represent an interesting alternative to projected and proximal gradient algorithms.

The original algorithm, described by M. Frank and P. Wolfe [13] in 1956, at each iteration minimizes a linear approximation of the objective function over the given feasible set in order to get a (feasible and descent) search direction, and then minimizes the original function along that direction, thus getting a new iterate. The main drawback of the method is that the convergence rate gets slow (i.e., sublinear) when the solution lies on the boundary of the feasible set. We get this rate mainly because the search directions tend to become orthogonal to the gradient very quickly, thus deteriorating their descent property and getting smaller and smaller stepsizes (this is the so-called "zig-zagging" phenomenon).

In order to reduce the "zig-zagging" effect, Wolfe [28] proposed to use a further search direction that allows to move away from a suitably chosen vertex. Guélat and Marcotte proved in [15] that a linear rate can be established under the assumption that the function is strongly convex, the feasible set is a polytope and the solution satisfies strict complementarity. Jaggi and Lacoste-Julien [18] describe a modified version of the away-step Frank-Wolfe algorithm, which calculates the away vertex by means of a simplex representation of the problem, and prove linear convergence without making assumptions on the solution. They also prove that the method guarantees linear rate for a specific class of non-strongly convex functions. In [1], Beck and Shtern propose a further modification for the away-step variant

that guarantees linear rate for the same class of non-strongly convex functions. Another interesting modification is described in [14], where "in-face" directions are used to boost the algorithm.

In [25], the authors show that the Frank-Wolfe algorithm with away steps is somehow related to the von Neumann algorithm. In particular, they first show that a variant of the von Neumann algorithm converges linearly, then prove that convergence rate and geometric insights also extend to a variant of the Frank–Wolfe algorithm with away steps for minimizing a convex quadratic function over a polytope.

Another variant of the Frank-Wolfe algorithm, the so-called pairwise Frank-Wolfe, was first described by Mitchel et al. in [20] for the polytope distance problem. Here the authors define, at each iteration, a direction that moves the weight from one vertex to another. More specifically, it moves the weight from the away vertex to the Frank-Wolfe vertex and keeps all others weights unchanged. This method is further analyzed in [18] and linear convergence is proved under the same assumptions seen before for the away-step variant. This method is strictly related to classical working set algorithms [22], like SMO algorithms for SVM training (see, e.g., [19, 26]).

In the context of big data, problems usually have very sparse solutions (i.e., solutions with many zero components). Hence, developing methods that allow to quickly build and/or identify the active set (i.e., the subset of zero components in a solution) is getting crucial to guarantee relevant savings in terms of CPU time.

Plenty of active-set methods have been proposed for solving nonlinear optimization problems (see [23] and references therein for further details). In active-set methods, at each iteration, a working set that estimates the set of active constraints at the solution is iteratively updated. Usually, only a single active constraint is added to or deleted from the active set at each iteration. However, when dealing with simple constraints, one can use more sophisticated active-set methods, which can add to or delete from the current estimated active set more than one constraint at each iteration, and eventually find the active set in a finite number of steps if certain conditions hold.

In machine learning, heuristic strategies that try to fix to zero a subset of variables (at each iteration of a given algorithm according to a certain rule), the so-called shrinking techniques, are widely used (see, e.g. [4]). Screening rules, i.e. rules to eliminate optimization variables that do not contribute to any final solution, have also been proposed recently. Those rules can be used either before passing the problem to the optimizer as a preprocessing phase, or in a dynamical way to gradually reduce the problem during the optimization. A nice overview of those methods is given in [27], where some new dynamic rules are also proposed for different classes of (strongly) convex optimization algorithms.

In this paper, we propose an active-set estimate to identify the set of variables that are zero at a stationary point of problem (1). We adapt some specific strategies proposed in the contexts of box-constrained problems (see [3, 6, 8, 9]) to the case of unit simplex. The main features of the active-set strategy developed in this paper are essentially two:

- it does not only focus on the zero variables and keep them fixed, but rather tries to quickly identify as many active variables as possible (including nonzero variables) at a given point;

- it gives a significant reduction in the objective function (when setting to zero those

variables estimated active), while guaranteeing feasibility (i.e. nonzero weights need to be suitably moved from active variables to some other variables in such a way that objective function reduces).

Last property, which is somehow related to the fact that the estimated active variables satisfy an approximate optimality condition, enables us to easily use this strategy into any globally convergent algorithm.

It is easy to see that the proposed strategy is quite different from shrinking techniques and static screening rules. It further differs from dynamic screening rules (like the ones proposed in [27]). Indeed, as we already said, the proposed strategy sets variables to zero in such a way that a sufficient reduction in the objective function is guaranteed. We might say that, in our case, screening and descent are strongly related. Furthermore, all those properties of the active-set estimate are guaranteed without making any strong assumption on the objective function. We can actually use our strategy also when dealing with non-convex objective functions.

In the second part of the paper, we then describe an active-set algorithmic framework that encompasses the classical Frank-Wolfe method and the two variants described so far, that is away-step Frank-Wolfe and pairwise Frank-Wolfe. More specifically, we get a two-step algorithmic framework that combines the active-set strategy with a Frank-Wolfe like procedure. In the first step, the algorithm moves weights from the estimated active variables to a suitably chosen vertex (in order to both keep feasibility and reduce the objective function). Then, in the second step, it defines a search direction in the subspace of the estimated nonactive variables (using one of the Frank-Wolfe variants listed before) and generates a new iterate. We prove convergence of our framework to stationary points of problem (1) using Armijo line searches and Frank-Wolfe like directions (since no convexity assumption is made on the objective function to prove convergence, the algorithm can also be used as a local solver for non-convex optimization problems). Analysis of the convergence rate is carried out under convexity assumptions. Linear convergence is proved for the active-set versions of the Frank-Wolfe variants, when using exact line searches. The main results are listed below:

- thanks to the use of our estimate, we are able to relax the classical assumptions on the objective function needed to prove the linear rate (see, e.g.,[18]). Indeed, we only require that the objective function is strongly convex into a suitably chosen restricted space related to the optimal solution;

- we can also prove that the active-set version of the Frank Wolfe algorithm converges at a linear rate under the additional assumption that strict complementarity holds at the optimal solution (thus relaxing the assumption needed for the original Frank-Wolfe, i.e. optimal solution in the relative interior of the feasible set);

- the rates we obtain depend on the sparsity of the final solution (so, in some way, the sparser the solution is, the better the rate).

We further focus on optimization problems over the ℓ_1-ball. We adapt both the active-set strategy and the algorithmic framework to this case, and analyze their theoretical properties.

The paper is organized as follows. In Section 3, we describe in depth our active-set estimate. In Section 4, we present our algorithmic framework and carry out the convergence analysis for different choices of the search direction. In Section 5, we also analyze the convergence rate of the active-set Frank-Wolfe variants proposed. In Section 6, we show how our algorithm can be easily extended to optimization problems over the ℓ_1-ball. In Section 7, we report our numerical experience. Finally, in Section 8, we draw some conclusions.

2 Notation and Preliminary Results

Throughout the paper, we indicate with $\|\cdot\|$ the Euclidean norm. Given a vector $v \in \mathbb{R}^n$ and an index set $I \subseteq \{1, \ldots, n\}$, we denote with v_I the subvector with components v_i, $i \in I$. We indicate with e_i the i-th unit vector. Given a set of vectors $D = \{v_1, \ldots, v_m\} \subseteq \mathbb{R}^n$, we indicate with $\mathrm{conv}(D)$ the convex hull of D. Finally, the open ball with center x and radius $\rho > 0$ is denoted by $\mathcal{B}(x, \rho)$.

Definition 1. *A feasible point x^* of problem (1) is a stationary point if and only if it satisfies the following first order necessary optimality conditions:*

$$\nabla f(x^*) - \lambda^* e - \mu^* = 0, \tag{2}$$

$$(\mu^*)^T x^* = 0, \tag{3}$$

$$\mu^* \geq 0. \tag{4}$$

where $\lambda^* \in \mathbb{R}$ and $\mu^* \in \mathbb{R}^n$ are the KKT multipliers.

3 Active-Set Estimate

We consider as active set the subset of zero components of the optimal solution.

Definition 2. *Let $x^* \in \mathbb{R}^n$ be a stationary point of problem (1). We define as active set the following set:*

$$\bar{A}(x^*) = \big\{i \in \{1, \ldots, n\} \colon x_i^* = 0\big\}. \tag{5}$$

We further define the nonactive set $\bar{N}(x^*)$ as the complementary set of $\bar{A}(x^*)$:

$$\bar{N}(x^*) = \{1, \ldots, n\} \setminus \bar{A}(x^*) = \big\{i \in \{1, \ldots, n\} \colon x_i^* > 0\big\}. \tag{6}$$

The active-set estimate is computed by following the approach proposed in [10, 11], which requires proper approximations of the KKT multipliers (the so called multiplier functions). Given a stationary point x^* of (1), let (λ^*, μ^*) be the KKT multipliers associated to x^*. By (2), we have

$$\mu^* = \nabla f(x^*) - \lambda^* e,$$

then, multiplying by x^* and taking into account complementarity condition (3), we get

$$0 = (\mu^*)^T x^* = (\nabla f(x^*) - \lambda^* e)^T x^*.$$

From the feasibility of x^*, we obtain the following expressions for the multipliers:

$$\lambda^* = \nabla f(x^*)^T x^*,$$
$$\mu^* = \nabla f(x^*) - \lambda^* e,$$

so that we can introduce the following continuous functions as multiplier functions:

$$\lambda(x) = \nabla f(x)^T x, \tag{7}$$
$$\mu_i(x) = \nabla_i f(x) - \lambda(x), \quad i = 1, \ldots, n. \tag{8}$$

Definition 3. *Let $x \in \mathbb{R}^n$ be a feasible point of problem (1). We define the active-set estimate $A(x)$ and the nonactive-set estimate $N(x)$ as*

$$A(x) = \{i \colon x_i \leq \epsilon\mu_i(x)\} = \{i \colon x_i \leq \epsilon\nabla f(x)^T(e_i - x)\}, \tag{9}$$
$$N(x) = \{i \colon x_i > \epsilon\mu_i(x)\} = \{i \colon x_i > \epsilon\nabla f(x)^T(e_i - x)\}, \tag{10}$$

where ϵ is a positive scalar.

By adapting the results shown in [11], we can state the following result.

Theorem 1. *If (x^*, λ^*, μ^*) satisfies KKT conditions for problem (1), then there exists a neighborhood $\mathcal{B}(x^*, \rho)$ such that, for each x in this neighborhood, we have*

$$\{i \colon x_i^* = 0,\, \mu_i(x^*) > 0\} \subseteq A(x) \subseteq \bar{A}(x^*).$$

Furthermore, if strict complementarity holds, then

$$\{i \colon x_i^* = 0,\, \mu_i(x^*) > 0\} = A(x) = \bar{A}(x^*),$$

for each $x \in \mathcal{B}(x^*, \rho)$.

3.1 A global property of the active-set estimate

Here, we analyze a global property of our active-set estimate. In particular, we show how, given a point $x \in \mathbb{R}^n$ feasible for problem (1), we can obtain a sufficient decrease in the objective function by setting the estimated active variables to zero. In order to keep feasibility, we need to update at least one nonactive variable, so that all variables sum up to 1. Next proposition gives us a hint on how to choose the nonactive variable that will be updated when setting to zero the active variables.

Proposition 1. *Let $J(x)$ be the set:*

$$J(x) = \Big\{ j \colon \; j \in \operatorname*{Argmin}_{i=1,\ldots,n} \big\{ \nabla_i f(x) \big\} \Big\}. \tag{11}$$

Let $x \in \mathbb{R}^n$ be a feasible point of problem (1). Then,

$$N(x) \cap J(x) \neq \emptyset.$$

Proof. see Appendix A. □

In order to state the main result of this section, we need an assumption on the parameter ϵ appearing in Definition 3.

Assumption 1. *Assume that the parameter ϵ appearing in the estimates (9)– (10) satisfies the following conditions:*

$$0 < \epsilon \leq \frac{1}{2Ln}, \tag{12}$$

where L is the Lipschitz constant of $\nabla f(x)$ over the unit simplex.

The main result, reported below, shows that it is possible to get a significant decrease in the objective function when moving weights from the active-set variables to the nonactive variable chosen in the set defined in Proposition 1.

Proposition 2. *Let Assumption 1 hold. Given a feasible point x of problem (1), let $j \in N(x) \cap J(x)$ and $I = \{1, \ldots, n\} \setminus \{j\}$. Let $\hat{A}(x)$ be a set of indices such that*

$$\hat{A}(x) \subseteq A(x).$$

Let $\tilde{x}$ be the feasible point defined as follows:

$$\tilde{x}_{\hat{A}(x)} = 0; \quad \tilde{x}_{I \setminus \hat{A}(x)} = x_{I \setminus \hat{A}(x)}; \quad \tilde{x}_j = x_j + \sum_{h \in \hat{A}(x)} x_h.$$

Then,

$$f(\tilde{x}) - f(x) \leq -L\|\tilde{x} - x\|^2.$$

Proof. See Appendix A. □

4 An Active-Set Algorithmic Framework for Minimization over the Simplex

In this section, we explain how to embed our active-set estimate into an algorithmic framework to minimize a function over the unit simplex. The framework executes two different steps at each iteration: the first one for updating the estimated active variables, and the second one for updating the estimated nonactive variables. The aim is to exploit as much as possible the properties of our estimate. First, the ability to identify those active variables satisfying the strict complementarity after a sufficiently large number of iterations (according to the result in Theorem 1). Second, the ability to get a decrease of the objective function, when moving the weights from the active set to a suitably chosen variable (according to the result in Proposition 2).

In particular, let x^k be the point given at the beginning of a generic iteration k. In the first step, we compute the active and nonactive-set estimates $A(x^k)$, $N(x^k)$, and we generate the new feasible point $\tilde{x}^k$, by setting $\tilde{x}_{A(x^k)}$ to zero and by updating a suitably chosen variable $\tilde{x}^k_j$, $j \in J(x^k)$ (all the other nonactive variables stay the same). Then, in the second step, we compute a search direction in the subspace of the nonactive variables, and, eventually, we execute a line search to get a new iterate x^{k+1}. The detailed scheme of our algorithmic framework is reported in Algorithm 1.

Algorithm 1 `Active-Set algorithmic framework for minimization over the simplex (AS-SIMPLEX)`

1 Choose a feasible point x^0
2 For $k = 0, 1, \ldots$
3 If x^k is a stationary point, then STOP
4 Compute $A^k := A(x^k)$ and $N^k := N(x^k)$
5 Compute $J^k := J(x^k)$, choose $j \in N^k \cap J^k$ and define $\tilde{N}^k = N^k \setminus \{j\}$
6 Set $\tilde{x}^k_{A^k} = 0\,,\ \tilde{x}^k_{\tilde{N}^k} = x^k_{\tilde{N}^k}$ and $\tilde{x}^k_j = x^k_j + \sum_{h \in A^k} x^k_h$
7 Set $d^k_{A^k} = 0$
8 Compute a feasible direction $d^k_{N^k}$ in $\tilde{x}^k$ and a maximum stepsize α^k_{max}
9 If $\nabla f(\tilde{x}^k)^T d^k < 0$ then
10 Compute a stepsize $\alpha^k \in (0, \alpha^k_{max}]$ by means of a line search
11 Else
12 Set $\alpha^k = 0$
13 End if
14 Set $x^{k+1} = \tilde{x}^k + \alpha^k d^k$
15 End for

4.1 Use of Frank-Wolfe type directions in `AS-SIMPLEX`

At every iteration k of Algorithm 1, we need to compute a feasible direction with respect to the nonactive subspace (Step 8), in order to move from $\tilde{x}^k$ and produce the new iterate x^{k+1}. A possibility is that of considering the Frank-Wolfe direction or one of its variants.

The Frank-Wolfe and the away-step directions, computed at $\tilde{x}^k$, in the subspace N^k, are respectively:

$$d^{\mathrm{FW}}_{N^k} = e_{\hat{\imath}} - \tilde{x}^k_{N^k}, \quad \hat{\imath} \in \operatorname*{Argmin}_{i \in N^k} \{\nabla_i f(\tilde{x}^k)\}; \tag{13}$$

$$d^{\mathrm{A}}_{N^k} = \tilde{x}^k_{N^k} - e_{\hat{\jmath}}, \quad \hat{\jmath} \in \operatorname*{Argmax}_{j \in N^k_0} \{\nabla_j f(\tilde{x}^k)\}, \tag{14}$$

where $N^k_0 = \{j \in N^k \colon \tilde{x}^k_j > 0\}$.

According to (13) and (14), we consider the following three search directions d^k:

(FW) $d^k_{N^k}$ is chosen as the Frank-Wolfe direction:

$$\begin{aligned} d^k_{A^k} &= 0, \\ d^k_{N^k} &= d^{\mathrm{FW}}_{N^k}. \end{aligned}$$

(AFW) $d^k_{N^k}$ is chosen as the away-step Frank-Wolfe direction:

$$\begin{aligned} d^k_{A^k} &= 0, \\ d^k_{N^k} = d^{\mathrm{AFW}}_{N^k} &= \begin{cases} d^{\mathrm{FW}}_{N^k}, & \text{if } \nabla_{N^k} f(\tilde{x}^k)^T d^{\mathrm{FW}}_{N^k} \le \nabla_{N^k} f(\tilde{x}^k)^T d^{\mathrm{A}}_{N^k}, \\ d^{\mathrm{A}}_{N^k}, & \text{otherwise.} \end{cases} \end{aligned}$$

(PFW) $d^k_{N^k}$ is chosen as the pairwise Frank-Wolfe direction:

$$d^k_{A^k} = 0,$$
$$d^k_{N^k} = d^{\mathrm{PFW}}_{N^k} = d^{\mathrm{FW}}_{N^k} + d^{\mathrm{A}}_{N^k} = e_{\hat{\imath}} - e_{\hat{\jmath}},$$

where $\hat{\imath}$ and $\hat{\jmath}$ are defined as in (13) and (14), respectively.

In the following, we will refer to d^{FW}, d^{AFW} and d^{PFW} if the direction d^k is chosen according to the Frank-Wolfe (FW), the away-step Frank-Wolfe (AFW) or the pairwise Frank-Wolfe (PFW) rule, respectively.

As stated in the following lemma, all the search directions defined above are non-ascent directions.

Lemma 1. *Let $\tilde{x}^k$ be a feasible point generated by* `AS-SIMPLEX` *(Step 6) at iteration k. Let d^k be a search direction computed according to one among (FW), (AFW) and (PFW) rule. Then,*

$$\nabla f(\tilde{x}^k)^T d^k \leq 0.$$

Proof. See Appendix B. □

In the next lemma, we show that at every point $\tilde{x}^k$ produced by `AS-SIMPLEX`, the directional derivative along d^{PFW} is not larger than the directional derivative along d^{AFW}. This fact will play a crucial role in proving the convergence of the algorithm for all the considered variants of the Frank-Wolfe direction.

Lemma 2. *Let $\tilde{x}^k$ be a feasible point generated by* `AS-SIMPLEX` *at iteration k. Then,*

$$\nabla f(\tilde{x}^k)^T d^{PFW} \leq \nabla f(\tilde{x}^k)^T d^{AFW}.$$

Proof. See Appendix B. □

4.2 Computation of the stepsize

A possibility for the computation of the stepsize, at Step 10 of Algorithm 1, is that of considering the classical Armijo line search (see, e.g., [2] and references therein). This method, which basically performs a successive stepsize reduction, allows to avoid the often considerable computation associated with an exact line search. Indeed, when dealing with some non-convex problems, even finding an approximate local minimizer along the search direction generally requires too many evaluations of the objective function and possibly the gradient.

The detailed scheme of the Armijo line search is reported in Algorithm 2.

Depending on the direction d^k used in the line search procedure, the maximum stepsize $\alpha^k_{\max}$ is set as follows:

(FW) Frank-Wolfe direction: $\alpha^k_{\max} = 1$;

(AFW) away-step Frank-Wolfe direction:

Algorithm 2 `Armijo line search`

0 Choose $\delta \in (0,1)$, $\gamma \in \left(0, \frac{1}{2}\right)$
1 Set initial stepsize $\alpha = \alpha^k_{\max}$
2 While $f(\tilde{x}^k + \alpha d^k) > f(\tilde{x}^k) + \gamma\, \alpha\, \nabla f(\tilde{x}^k)^T d^k$
3 Set $\alpha = \delta\alpha$
4 End while

if $d^k_{N^k} = d^{\mathrm{FW}}_{N^k}$, then $\alpha^k_{\max} = 1$;
if $d^k_{N^k} = d^{\mathrm{A}}_{N^k}$, then $\alpha^k_{\max} = \tilde{x}^k_{\hat{\jmath}}/(1 - \tilde{x}^k_{\hat{\jmath}})$ where $\hat{\jmath}$ is defined as in (14);

(PFW) pairwise Frank-Wolfe direction: $\alpha^k_{\max} = \tilde{x}^k_{\hat{\jmath}}$, where $\hat{\jmath}$ is defined as in (14).

For every considered search direction d^k, this choice guarantees that $\tilde{x}^k + \alpha d^k$ is feasible for all $\alpha \in (0, \alpha^k_{\max}]$. Moreover, it is easy to verify that $\alpha^k_{\max} \leq 1$ for every kind of search direction.

The following proposition follows from classical results on the Armijo line search. It guarantees that $\|\tilde{x}^k - x^k\|$ converges to zero and that the sequence of the directional derivatives along the search direction converges to zero as well, for all the considered search directions. Before stating the proposition, we observe that, from standard results on the Armijo line search, Algorithm 2 computes α^k in a finite number of steps at every iteration k for which $\nabla f(\tilde{x}^k)^T d^k < 0$.

Proposition 3. *Let Assumption 1 hold. Let $\{x^k\}$, $\{\tilde{x}^k\}$ and $\{d^k\}$ be the sequences produced by* `AS-SIMPLEX`*, where d^k is computed at Step 7–8 according to one among (FW), (AFW) and (PFW) rule. If* `AS-SIMPLEX` *does not terminate in a finite number of iterations, then*

$$\lim_{k\to\infty} \|\tilde{x}^k - x^k\| = 0, \tag{15}$$

$$\lim_{k\to\infty} \nabla f(\tilde{x}^k)^T d^k = 0. \tag{16}$$

Proof. See Appendix B. □

Remark 1. *In the proof of Proposition 3 (see Appendix B for further details), computing α^k by the Armijo line search is not essential. Namely, Proposition 3 holds when considering any value $\alpha^k \in (0, \alpha^k_{max}]$ such that $f(\tilde{x}^k + \alpha^k d^k) \leq f(\tilde{x}^k + \alpha^k_A d^k)$, where α^k_A is the value computed by the Armijo line search. In particular, this implies that Proposition 3 holds under the assumption that the stepsize is computed in* `AS-SIMPLEX` *by means of an exact line search, that is, α^k is computed as*

$$\alpha^k \in \operatorname*{Argmin}_{\alpha\in(0,\alpha^k_{max}]} f(\tilde{x}^k + \alpha d^k).$$

4.3 Global convergence analysis

In this subsection, for every considered choice of the direction d^k, we show the global convergence of `AS-SIMPLEX` to stationary points.

Theorem 2. *Let Assumption 1 hold. Let $\{x^k\}$ be the sequence of points produced by* `AS-SIMPLEX`*, where*

- *the search direction d^k is computed according to one among (FW), (AFW) and (PFW) rule;*
- *the stepsize α^k is computed using the Armijo line search.*

Then, either an integer $\bar{k} \geq 0$ exists such that $x^{\bar{k}}$ is a stationary point for problem (1)*, or the sequence $\{x^k\}$ is infinite and every limit point x^* of the sequence is a stationary point for problem* (1)*.*

Proof. See Appendix B. □

5 Convergence Rate Analysis

In this section, following the ideas used in [18], we analyze the convergence rate of the three active-set Frank-Wolfe variants we described in the previous sections. In particular, we first show that the `AS-SIMPLEX` with Frank-Wolfe direction and exact line search converges at a linear rate under the assumptions that the objective function is strongly convex into a suitably chosen restricted space related to the optimal solution and strict complementarity holds at the optimal solution. Then, we get linear convergence for the other active-set variants. In this case, we only assume that the objective function is strongly convex into a suitably chosen restricted space related to the optimal solution.

In order to prove the results, we make an assumption that is pretty common when analyzing the convergence rate of algorithms (see, e.g., [24]).

Assumption 2. *Let $\{x^k\}$ be the infinite sequence generated by* `AS-SIMPLEX`*. We have that*

$$\lim_{k\to\infty} x^k = x^*,$$

where x^* is an optimal point of problem (1)*.*

From now on, we denote with $\bar{A}$ and $\bar{N}$ the index sets defined in (5) and (6), respectively. We denote with $\bar{I}$ the set $\{1, \ldots, n\}$. Finally, we denote with Δ the unit simplex, and with $\Delta_I := \{x \in \Delta \colon x_i = 0, \forall i \notin I\}$, where $I \subseteq \bar{I}$.

5.1 Linear convergence of active-set Frank-Wolfe

Here we show that, when embedding an active-set strategy in the Frank-Wolfe algorithm, one can get linear convergence without assuming that the optimal solution is in the interior of the feasible set. As we will see, this assumption is replaced by strict complementarity in the optimal solution.

Before reporting the theoretical results related to the active-set Frank-Wolfe (i.e., `AS-SIMPLEX` with Frank-Wolfe direction), we need to introduce some constants. Given a minimum

point x^* and an index subset $I \subseteq \bar{I}$, we define:

$$C_f(I) := \sup_{\substack{x,s\in\Delta_I,\\ \alpha\in(0,1],\\ y=x+\alpha(s-x)}} \frac{2}{\alpha^2}\Big[f(y) - f(x) - \nabla f(x)^T(y-x)\Big],$$

$$\mu_f(I) := \inf_{\substack{x\in\Delta_I\setminus\{x^*\},\\ \alpha\in(0,1],\\ \bar{s}=\bar{s}(x,x^*,\Delta),\\ y=x+\alpha(\bar{s}-x)}} \frac{2}{\alpha^2}\Big[f(y) - f(x) - \nabla f(x)^T(y-x)\Big],$$

where $\bar{s}(x, x^*, \Delta) := \text{ray}(x, x^*) \cap \partial(\Delta)$. The curvature constant $C_f(I)$, which measures the non-linearity of the objective function in the subspace Δ_I, is needed to give a quadratic upper bound on the objective function. The strong convexity constant $\mu_f(I)$, which measures the strong convexity of the objective function in Δ_I (and can be interpreted as the lower curvature of the function), is used to give a quadratic lower bound instead. Both bounds are needed for proving the main result reported in this subsection (see [17] for further details).

Remark 2. *The constants given above are similar to the ones introduced in [17]. The main difference is that ours are restricted to a particular subspace. Moreover, for any index subset $I \subseteq \bar{I}$, it is easy to see that*

$$\mu_f(\bar{I}) \leq \mu_f(I) \leq C_f(I) \leq C_f(\bar{I}). \tag{17}$$

In the next theorem, we state the linear convergence rate of `AS-SIMPLEX`, when the search direction d^k is computed according to (FW) rule.

Theorem 3. *Let Assumption 1 and 2 hold, let $f(x)$ be strongly convex on $\Delta_{\bar{N}}$, and let us assume that strict complementarity holds at x^*. Let us further assume that the exact line search is used.*

Then, there exists $\bar{k}$ such that, if d^k is computed according to (FW) rule, we have

$$f(x^{k+1}) - f(x^*) \leq \big(1 - \rho^{AS-FW}\big)\big[f(x^k) - f(x^*)\big], \quad \forall k \geq \bar{k},$$

where

$$\rho^{AS\text{-}FW} = \min\left\{\frac{1}{2}, \frac{\mu_f(\bar{N})}{C_f(\bar{N})}\right\}.$$

Proof. See Appendix C. □

Remark 3. *From* (17)*, it follows that the smaller $\bar{N}$ (i.e., the sparser x^*), the better the convergence rate of `AS-SIMPLEX`. Moreover,*

$$\rho^{AS\text{-}FW} \geq \min\left\{\frac{1}{2}, \frac{\mu_f(\bar{I})}{C_f(\bar{I})}\right\} = \rho^{FW},$$

where ρ^{FW} is the constant given in [17] for the convergence rate of the standard Frank-Wolfe method.

5.2 Linear convergence of active-set Frank-Wolfe variants

In this subsection, we prove that both active-set away-step Frank-Wolfe (i.e., `AS-SIMPLEX` with away-step Frank-Wolfe direction) and active-set pairwise Frank-Wolfe (i.e., `AS-SIMPLEX` with pairwise Frank-Wolfe direction) converge at linear rate. From now on, we denote with

$$N^+ := \bar{N} \cup \{i \in \bar{I} \colon x_i^* = 0,\, \mu_i^* = 0\} \quad \text{and} \quad A^+ := \bar{I} \setminus N^+ = \{i \in \bar{I} \colon x_i^* = 0,\, \mu_i^* > 0\}.$$

Given an index subset $I \subseteq \bar{I}$, we define the following two constants:

$$C_f^{\Delta}(I) := \sup_{\substack{x,s,v \in \Delta_I \\ \alpha \in (0,1], \\ y = x + \alpha(s-v)}} \frac{2}{\alpha^2}\Big[f(y) - f(x) - \alpha \nabla f(x)^T (s-v)\Big],$$

$$\mu_f^{\Delta}(I) := \inf_{x \in \Delta_I} \inf_{\substack{\hat{x} \in \Delta_I \\ \nabla f(x)^T(\hat{x}-x) < 0}} \frac{2}{\alpha_I^{\Delta}(x,\hat{x})^2}\Big[f(\hat{x}) - f(x) - \nabla f(x)^T (\hat{x}-x)\Big],$$

where

$$\alpha_I^{\Delta}(x,\hat{x}) := \frac{\nabla f(x)^T(\hat{x}-x)}{\nabla f(x)^T(s_I(x) - v_I(x))},$$
$$s_I(x) := e_{\hat{\imath}}, \quad \hat{\imath} \in \operatorname*{Argmin}_{i \in I}\{\nabla_i f(x)\},$$
$$v_I(x) := e_{\hat{\jmath}}, \quad \hat{\jmath} \in \operatorname*{Argmax}_{j \in I \colon x_j > 0}\{\nabla_j f(x)\}.$$

These two new constants are motivated in the analysis by the fact that both Frank-Wolfe and away-step directions are used in the variants (see [18] for further details).

Remark 4. *Also in this case, the constants given above are similar to the ones introduced in [18]. Again, the difference is that ours are restricted to a particular subspace. Moreover, for any index subset $I \subseteq \bar{I}$, it is easy to see that the following inequalities hold:*

$$\mu_f^{\Delta}(\bar{I}) \leq \mu_f^{\Delta}(I) \leq C_f^{\Delta}(I) \leq C_f^{\Delta}(\bar{I}). \tag{18}$$

Theorem 8 in [18] shows, for the standard away-step Frank-Wolfe and the standard pairwise Frank-Wolfe methods, that the quantity $f(x^k) - f(x^*)$ decreases linearly at each iteration k that is neither a so-called *drop step* nor a so-called *swap step*.

Iteration k is a drop step when the stepsize $\alpha^k = \alpha_{\max}^k < 1$ and the number of zero components in x^{k+1} increases by one. Iteration k is a swap step when the stepsize $\alpha^k = \alpha_{\max}^k$, but the number of zero components in x^{k+1} does not change. Note that a swap step can occur only in the pairwise Frank-Wolfe method. In the convergence rate analysis, these iterations are troublesome since a geometric decrease cannot be guaranteed.

In our context, these definitions apply when considering the computation of x^{k+1} from $\tilde{x}^k$ and, as to be shown in the next theorem, we can still guarantee that the quantity $f(x^k) - f(x^*)$ decreases linearly at each iteration k that is a *good step* (i.e., not a drop step nor a swap step) with tighter constants (that depend on the sparsity of the optimal solution).

Theorem 4. *Let Assumption 1 and 2 hold, let $f(x)$ be strongly convex on Δ_{N^+}, with $\nabla f(x)$ Lipschitz continuous on $\Delta_{N^+} + (\Delta_{N^+} - \Delta_{N^+})$ (in the Minkowski sense). Let us further assume that the exact line search is used.*

Then, there exists $\bar{k}$ such that, for every iteration $k \geq \bar{k}$ that is a good step (i.e., it is not a drop step nor a swap step), we have

$$f(x^{k+1}) - f(x^*) \leq (1-\rho)\big[f(x^k) - f(x^*)\big], \quad \forall k \geq \bar{k},$$

where

$$\rho = \begin{cases} \rho^{\text{AS-AFW}} = \dfrac{\mu_f^{\Delta}(N^+)}{4C_f^{\Delta}(N^+)}, & \text{if } d^k \text{ is computed by (AFW) rule,} \\ \rho^{\text{AS-PFW}} = \min\left\{\dfrac{1}{2}, \dfrac{\mu_f^{\Delta}(N^+)}{C_f^{\Delta}(N^+)}\right\}, & \text{if } d^k \text{ is computed by (PFW) rule.} \end{cases} \tag{19}$$

Moreover, for $k \geq \bar{k}$, we have that

- *at most $|N^+| - 1$ drop steps can be performed in between two good steps,*
- *at most $3|N^+|!$ swap steps can be performed in between two good steps (this can only happen when (PFW) rule is used).*

Proof. See Appendix C. □

Remark 5. *From* (18)*, it follows that the smaller N^+, the better the convergence rate of* `AS-SIMPLEX`*. Moreover,*

$$\rho^{\text{AS-AFW}} \geq \frac{\mu_f^{\Delta}(\bar{I})}{4C_f^{\Delta}(\bar{I})} = \rho^{\text{AFW}} \quad \text{and} \quad \rho^{\text{AS-PFW}} \geq \min\left\{\frac{1}{2}, \frac{\mu_f^{\Delta}(\bar{I})}{C_f^{\Delta}(\bar{I})}\right\} = \rho^{\text{PFW}},$$

where ρ^{AFW} and ρ^{PFW} are the constants given in [18] for the convergence rate of the standard away-step Frank-Wolfe and the standard pairwise Frank-Wolfe method, respectively. Furthermore, also the upper bound in the number of bad steps between two good steps depends on the cardinality of N^+ (for sufficiently large k). We would like to recall that, in the standard Frank-Wolfe variants, this value is equal to $n-1$ and $3n!$ for drop and swap steps, respectively.

6 Extension to Minimization Problems over the ℓ_1-ball

In this section, we describe how we can adapt our algorithmic framework to solve optimization problems over the ℓ_1-ball. We focus on problems of the following form:

$$\begin{array}{ll} \min\limits_{x \in \mathbb{R}^n} & h(x) \\ \text{s.t.} & \|x\|_1 \leq \tau, \end{array} \tag{20}$$

with $h \in C^1(\mathbb{R}^n)$, $\nabla h(x)$ Lipschitz continuous on the feasible set and $\tau > 0$.

First, we see how any point in the feasible set of problem (20) can be rewritten as a convex combination of $\{\pm\tau e_1, \ldots, \pm\tau e_n\}$. Let $M \in \mathbb{R}^{n\times 2n}$ be the matrix whose columns are the vertices of the feasible region of (20), namely $M = \tau \left[\begin{array}{c|c} I & -I \end{array}\right]$, where $I \in \mathbb{R}^{n\times n}$ is the identity matrix. Let $x \in \mathbb{R}^n$ be any feasible point of problem (20). We can write

$$Mw = x, \quad e^T w = 1, \quad w \geq 0, \quad w \in \mathbb{R}^{2n}.$$

Now, starting from problem (20), we consider a new matrix $\tilde{M} \in \mathbb{R}^{(n+1)\times(2n+1)}$ that enables us to extend the variable space:

$$\tilde{M} = \tau \left[\begin{array}{c|c|c} I & -I & \begin{array}{c} 0 \\ \vdots \\ 0 \end{array} \\ \hline 0 \ \ldots \ 0 & 0 \ \ldots \ 0 & 1 \end{array}\right],$$

and the following feasible set:

$$\begin{aligned} P &= \left\{(x,z) \in \mathbb{R}^{n+1} : \begin{pmatrix} x \\ z \end{pmatrix} = \tilde{M}y,\, e^T y = 1,\, y \geq 0\right\} \\ &= \operatorname{conv}\{\pm\tau e_1, \ldots, \pm\tau e_n, \tau e_{n+1}\}. \end{aligned} \tag{21}$$

We further define the new function $\bar{h}\colon \mathbb{R}^{n+1} \to \mathbb{R}$ as follows: $\bar{h}(x,\, z) = h(x)$, for every $(x, z) \in \mathbb{R}^{n+1}$. We thus get the new equivalent problem:

$$\begin{array}{ll} \min\limits_{y\in\mathbb{R}^{2n+1}} & f(y) = \bar{h}(\tilde{M}y) \\ \text{s.t.} & e^T y = 1 \\ & y \geq 0. \end{array} \tag{22}$$

This specific choice allows to describe any feasible point of the original problem using a *"minimal" representation* (i.e., a representation with the smallest number of nonzero components $y_1, \ldots,\ y_{2n}$), that is:

$$\begin{aligned} y_i &= \frac{1}{\tau}\max\{0, x_i\}, \quad i = 1, \ldots, n, \\ y_{n+i} &= \frac{1}{\tau}\max\{0, -x_i\}, \quad i = 1, \ldots, n, \\ y_{2n+1} &= \frac{\tau - \|x\|_1}{\tau}\,. \end{aligned} \tag{23}$$

We have $\nabla f(y) = \tilde{M}^T \nabla\bar{h}(x) = \tau\Big(\nabla_1 h(x), \ldots, \nabla_n h(x), -\nabla_1 h(x), \ldots, -\nabla_n h(x), 0\Big)^T$, so that

$$\nabla f(y)^T y = \nabla\bar{h}(x)^T \tilde{M}y = \begin{bmatrix}\nabla h(x)^T & 0\end{bmatrix} \tilde{M}y = \nabla h(x)^T x. \tag{24}$$

For every feasible point x of problem (20), we can consider the following sets:

$A_{\ell_1}(x)$, the set of indices of the estimated active variables;

$N_{\ell_1}(x)$, the set of indices of the estimated nonactive variables.

We show that there exists a correspondence between the variables x_i estimated active for problem (20) (i.e., those variables that are estimated to be zero at the stationary point) and the variables y_i estimated active for problem (22).

From (21), we can write

$$x_i = \tau\,(y_i - y_{n+i}), \quad i = 1, \ldots, n.$$

Consequently, if both y_i and y_{n+i} are estimated active for problem (22), we can estimate x_i active for problem (20). So, we define $A_{\ell_1}(x)$ and $N_{\ell_1}(x)$ as follows:

$$A_{\ell_1}(x) = \big\{i \in \{1, \ldots, n\} \colon i \in A(y) \ \text{ and } \ (n+i) \in A(y)\big\}, \tag{25}$$

$$N_{\ell_1}(x) = \big\{i \in \{1, \ldots, n\} \colon i \in N(y) \ \text{ or } \ (n+i) \in N(y)\big\}. \tag{26}$$

Using (23) and (24), for each index $i \in \{1, \ldots, n\}$ we can distinguish two cases:

(i) $x_i \geq 0$. Recalling (9)–(10), we have that $i \in A(y)$ if and only if

$$\begin{aligned} 0 \leq \frac{1}{\tau} x_i = y_i &\leq \epsilon \nabla f(y)^T (e_i - y) = \epsilon(\nabla_i f(y) - \nabla f(y)^T y) \\ &= \epsilon(\tau \nabla_i h(x) - \nabla h(x)^T x) = \epsilon \nabla h(x)^T (\tau e_i - x) \end{aligned} \tag{27}$$

and $(n+i) \in A(y)$ if and only if

$$\begin{aligned} -\frac{1}{\tau} x_i \leq 0 = y_{n+i} &\leq \epsilon \nabla f(y)^T (e_{n+i} - y) = \epsilon(\nabla_{n+i} f(y) - \nabla f(y)^T y) \\ &= \epsilon(-\tau \nabla_i h(x) - \nabla h(x)^T x) = -\epsilon \nabla h(x)^T (\tau e_i + x). \end{aligned} \tag{28}$$

(ii) $x_i < 0$. Similarly to the previous case, we have that $i \in A(y)$ if and only if

$$\begin{aligned} \frac{1}{\tau} x_i < 0 = y_i &\leq \epsilon \nabla f(y)^T (e_i - y) = \epsilon(\nabla_i f(y) - \nabla f(y)^T y) \\ &= \epsilon(\tau \nabla_i h(x) - \nabla h(x)^T x) = \epsilon \nabla h(x)^T (\tau e_i - x) \end{aligned} \tag{29}$$

and $(n+i) \in A(y)$ if and only if

$$\begin{aligned} 0 < -\frac{1}{\tau} x_i = y_{n+i} &\leq \epsilon \nabla f(y)^T (e_{n+i} - y) = \epsilon(\nabla_{n+i} f(y) - \nabla f(y)^T y) \\ &= \epsilon(-\tau \nabla_i h(x) - \nabla h(x)^T x) = -\epsilon \nabla h(x)^T (\tau e_i + x). \end{aligned} \tag{30}$$

From (27)–(30), we obtain

$$\begin{aligned} A_{\ell_1}(x) = \{i \colon \epsilon\,\tau \nabla h(x)^T (\tau e_i + x) \leq 0 \leq x_i \leq \epsilon\,\tau \nabla h(x)^T (\tau e_i - x) \ \text{or} \\ \epsilon\,\tau \nabla h(x)^T (\tau e_i + x) \leq x_i \leq 0 \leq \epsilon\,\tau \nabla h(x)^T (\tau e_i - x)\}, \end{aligned} \tag{31}$$

$$N_{\ell_1}(x) = \{1, \ldots, n\} \setminus A_{\ell_1}(x). \tag{32}$$

Now, we show how the algorithmic framework described in the Section 4 can be adapted to solve problem (20), using the active and nonactive set estimates (31)–(32).

Proposition 4. *Let $J_{\ell_1}(x)$ be the set*

$$J_{\ell_1}(x) = \Big\{ j \in \{1, \dots, n\} \colon j \in \operatorname*{Argmax}_{i=1,\dots,n} \big\{ |\nabla_i h(x)| \big\} \Big\}.$$

Let x be a feasible point of problem (20) *and assume that x is non-stationary. Then,*

$$N_{\ell_1}(x) \cap J_{\ell_1}(x) \neq \emptyset.$$

Proof. See Appendix D. □

Assumption 3. *Assume that the parameter ϵ appearing in the estimates* (31)–(32) *satisfies the following conditions:*

$$0 < \epsilon \leq \frac{1}{2\tau^2 L n}, \tag{33}$$

where L is the Lipschitz constant of $\nabla h(x)$ over the feasible set of (20).

Proposition 5. *Let Assumption 3 hold. Given a feasible point x of problem* (20)*, let us assume that x is non-stationary. Let $j \in N_{\ell_1}(x) \cap J_{\ell_1}(x)$ and $I = \{1, \dots, n\} \setminus \{j\}$. Let $\hat{A}_{\ell_1}(x)$ be a set of indices such that*

$$\hat{A}_{\ell_1}(x) \subseteq A_{\ell_1}(x).$$

Let $\tilde{x}$ be the feasible point defined as follows:

$$\tilde{x}_{\hat{A}_{\ell_1}(x)} = 0; \quad \tilde{x}_{I \setminus \hat{A}_{\ell_1}(x)} = x_{I \setminus \hat{A}_{\ell_1}(x)}; \quad \tilde{x}_j = x_j - \operatorname{sgn}(\nabla_j h(x)) \sum_{h \in \hat{A}_{\ell_1}(x)} |x_h|. \tag{34}$$

Then,

$$h(\tilde{x}) - h(x) \leq -L \|\tilde{x} - x\|^2.$$

Proof. See Appendix D. □

We report in Algorithm 3 the algorithmic framework to solve problem (20) that extends Algorithm 1 to the case of minimization problems over the ℓ_1-ball.

Algorithm 3 `Active-Set algorithmic framework for minimization over the` ℓ_1`-ball (AS-`ℓ_1`)`

... as in Algorithm 1 except for

sets A^k, N^k and J^k respectively replaced by

$$A^k_{\ell_1} = A_{\ell_1}(x^k),\ N^k_{\ell_1} = N_{\ell_1}(x^k) \text{ and } J^k_{\ell_1} = J_{\ell_1}(x^k),$$

and line:

6 Set $\tilde{x}^k_{A^k_{\ell_1}} = 0$, $\tilde{x}^k_{\tilde{N}^k_{\ell_1}} = x^k_{\tilde{N}^k_{\ell_1}}$, $\tilde{x}^k_j = x^k_j - \operatorname{sgn}(\nabla_j h(x^k)) \sum_{h \in A^k_{\ell_1}} |x^k_h|$

Also in this case, we choose to compute $d^k_{N^k_{\ell_1}}$ by means of the standard Frank-Wolfe direction, or one of its variants. In particular, exploiting the relations between problem (20) and (22), we can easily compute, in the subspace $N^k_{\ell_1}$, every variant of the Frank-Wolfe direction that has been considered in Subsection 4.1.

For the sake of completeness, we report the way we compute such directions. At every iteration k, we distinguish whether $\tilde{x}^k$ lies on the boundary or in the interior of the feasible set. In the first case, (23) is the unique representation of $\tilde{x}^k$ in the y space. Then, we simply compute the search direction as explained in Subsection 4.1.

Vice versa, if $\|\tilde{x}^k\|_1 < \tau$, there exist infinite representations of $\tilde{x}^k$ in the y space. In particular, for each index $h \in \{1, \dots, n\}$, we can compute a point y that satisfies conditions in (21) with $y_h > 0$, $y_{n+h} > 0$ and $y_{2n+1} = 0$, by setting

$$y_i = \begin{cases} \frac{1}{\tau}\max\{0, \tilde{x}^k_i\} + \frac{1}{2\tau}(\tau - \|\tilde{x}^k\|_1), & i = h, \\ \frac{1}{\tau}\max\{0, \tilde{x}^k_i\}, & i \in \{1, \dots, n\} \setminus \{h\}, \end{cases}$$

$$y_{n+i} = \begin{cases} \frac{1}{\tau}\max\{0, -\tilde{x}^k_i\} + \frac{1}{2\tau}(\tau - \|\tilde{x}^k\|_1), & i = h, \\ \frac{1}{\tau}\max\{0, -\tilde{x}^k_i\}, & i \in \{1, \dots, n\} \setminus \{h\}. \end{cases}$$

This trick allows us to consider all vertices of the ℓ_1-ball in the computation of the away-step direction when $\tilde{x}^k$ is in the interior of the feasible set.

More specifically, at every iteration k, two feasible search directions can be computed (in the subspace $N^k_{\ell_1}$):

- the Frank-Wolfe direction:

$$d^{\mathrm{FW}}_{N^k_{\ell_1}} = -\tau \operatorname{sgn}(\nabla_{\hat{\imath}} h(\tilde{x}^k))\, e_{\hat{\imath}} - \tilde{x}^k_{N^k_{\ell_1}}, \quad \hat{\imath} \in \operatorname*{Argmax}_{i \in N^k_{\ell_1}} \big\{|\nabla_i h(\tilde{x}^k)|\big\};$$

- the away-step direction:

$$d^{\mathrm{A}}_{N^k_{\ell_1}} = \begin{cases} \tilde{x}^k_{N^k_{\ell_1}} - \tau \operatorname{sgn}(\tilde{x}^k_{\hat{\jmath}})\, e_{\hat{\jmath}}, & \text{if } \|\tilde{x}^k\|_1 = \tau, \\ \tilde{x}^k_{N^k_{\ell_1}} - \tau \operatorname{sgn}(\nabla_{\hat{\imath}} h(\tilde{x}^k))\, e_{\hat{\imath}}, & \text{otherwise,} \end{cases}$$

  where $\hat{\imath} \in \operatorname*{Argmax}_{i \in N^k_{\ell_1}} \big\{|\nabla_i h(\tilde{x}^k)|\big\}$ and $\hat{\jmath} \in \operatorname*{Argmax}_{j \in N^k_{\ell_1} :\, \tilde{x}^k_j \neq 0} \big\{\nabla_j h(\tilde{x}^k)\, \operatorname{sgn}(\tilde{x}^k_j)\big\}$.

The search direction d^k can thus be computed according to the rules we defined before (i.e., (FW), (AFW) and (PFW)).

The maximum stepsize $\alpha^k_{\max}$ can be set again by distinguishing whether $\tilde{x}^k$ lies on the boundary or in the interior of the feasible set, following the same reasoning made before for the computation of the search direction:

(FW) Frank-Wolfe direction: $\alpha^k_{\max} = 1$;

(AFW) away-step Frank-Wolfe direction:

$$\text{if } d^k_{N^k_{\ell_1}} = d^{\mathrm{FW}}_{N^k_{\ell_1}}, \text{ then } \alpha^k_{\max} = 1;$$

$$\text{if } d^k_{N^k_{\ell_1}} = d^{\mathrm{A}}_{N^k_{\ell_1}}, \text{ then } \alpha^k_{\max} = \frac{\sigma^k}{1-\sigma^k}, \text{ where}$$

$$\sigma^k = \begin{cases} |\tilde{x}^k_{\hat{\jmath}}|/\tau, & \text{if } \|\tilde{x}^k\|_1 = \tau, \\ \dfrac{2\max\{0, \operatorname{sgn}(\nabla_{\hat{\imath}} h(\tilde{x}^k))\, \tilde{x}^k_{\hat{\imath}}\} + \tau - \|\tilde{x}^k\|_1}{2\tau}, & \text{otherwise,} \end{cases} \tag{35}$$

and $\hat{\imath}$, $\hat{\jmath}$ are the indices calculated in the away-step direction;

(PFW) pairwise Frank-Wolfe direction: $\alpha^k_{\max} = \sigma^k$, with σ^k as in (35).

From the relations between problem (20) and problem (22), recalling Proposition 5 and taking into account how we compute d^k, the convergence of `AS-`ℓ_1 for every considered variant of the Frank-Wolfe direction follows from the convergence results of `AS-SIMPLEX`.

Before ending the section, we would like to highlight again that the proposed active-set framework works in the original problem space and no transformation of the variables is needed in practice to handle the ℓ_1-ball.

7 Numerical Results

In this section, we report the numerical experience related to our active-set algorithmic framework. We first analyze the benefits of embedding the active-set strategy in the Frank-Wolfe algorithm by starting with an illustrative example. Then, we analyze the performance of the three active-set Frank-Wolfe variants described in the previous sections on two different classes of problems:

- convex quadratic instances that satisfy strict complementarity at the optimal solution;
- lasso problems.

For each instance, we first ran the Frank-Wolfe variants without using the active-set estimate, namely, the Frank-Wolfe, the away-step Frank-Wolfe and the pairwise Frank-Wolfe method (all of them implemented according to the schemes described in [18]). Then, we ran the corresponding active-set versions.

In the following, we denote by `FW`, `AFW` and `PFW` the Frank-Wolfe, the away-step Frank-Wolfe and the pairwise Frank-Wolfe method, respectively. We further denote by `AS-FW`, `AS-AFW` and `AS-PFW` the methods we have from our algorithmic framework, where the search direction d^k is computed according to (FW), (AFW) and (PFW) rule, respectively.

In order to calculate the estimates at each iteration, we need to set the ϵ parameter to a proper value. In general, the value of this parameter cannot be a priori computed. Following [6, 9], we employ this simple updating rule: at every iteration k, we compute $\tilde{x}^k$ as indicated at Step 6 of Algorithm 1 and 3 and, if a sufficient decrease in the objective function is obtained, then we accept $\tilde{x}^k$ and we do not change the value of ϵ. Otherwise,

we do not accept $\tilde{x}^k$, we reduce ϵ and we estimate the active set again, continuing until we get a sufficient decrease in the objective function. The starting value for the ϵ parameter is 10^{-1}.

All the codes used in the tests were implemented in Matlab R2014b and the experiments were ran on an Intel Xeon(R), CPU E5-1650 v2 3.50 GHz.

7.1 Benefits of the active-set strategy for the standard Frank-Wolfe

It is well known that, when dealing with minimization problems over polytopes, the convergence rate of the Frank-Wolfe method is linear in case the optimal solution lies in the relative interior of the feasible set and the objective function is strongly convex (see, e.g., [15]). This fact suggests an interesting methodological effect of the active-set strategy we have proposed: as long as the optimal solution of problem (1) satisfies the strict complementarity condition, according to Theorem 1, we are able to identify the optimal active set in a finite number of iterations. This implies that, after a finite number of iterations, Algorithm 1 with the Frank-Wolfe direction behaves like the Frank-Wolfe method applied to a problem whose solution lies in the relative interior of the feasible set, so that a linear convergence rate is guaranteed (see proof of Theorem 3). Hence, we have cases where the convergence rate of Frank-Wolfe is sublinear, but one can still get a linear convergence by using our active-set strategy.

In Figure 1, we report the performance of the standard Frank-Wolfe and its active-set version (namely, Algorithm 1 with $d^k = d^{\mathrm{FW}}$) on a 3 dimensional example:

$$\begin{array}{ll} \min\limits_{x\in\mathbb{R}^n} & \frac{1}{2}x^TQx \\ \text{s.t.} & e^Tx = 1 \\ & x \geq 0, \end{array}$$

where $Q \in \mathbb{R}^{3\times 3}$ is the following symmetric and positive definite matrix

$$Q = \begin{pmatrix} 3 & 0 & 3 \\ 0 & 3/2 & 3/2 \\ 3 & 3/2 & 5 \end{pmatrix}.$$

It is easy to verify that $x^* = (1/3, 2/3, 0)^T$ is the optimal solution of the problem and together with $\mu^* = (0, 0, 1)^T$ and $\lambda^* = 1$ satisfies the strict complementarity conditions. The starting point for both algorithms was $x^0 = (0.1, 0.3, 0.6)^T$ and we asked for a tolerance of 10^{-5} (in other words, we stopped each algorithm as soon as an iteration k such that $\nabla f(x^k)^T d^k \geq -10^{-5}$ was reached). We used an Armijo line search for both algorithms. After 10^5 iterations, the standard Frank-Wolfe method is not able to stop, while its active-set version stops after 12 iterations. The optimal active set, namely $\bar{A}(x^*) = \{3\}$, is identified by Algorithm 1 after the first iteration.

7.2 Comparison on convex quadratic instances

In the following, we report our numerical experience on convex quadratic instances whose solutions satisfy the strict complementarity conditions. Taking inspiration from [21], we

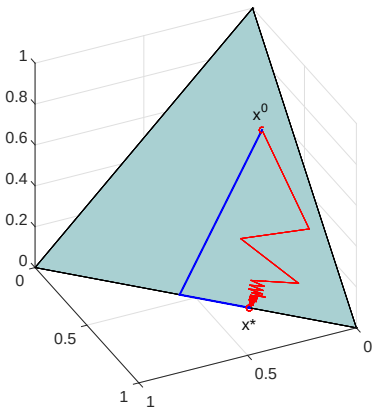

Figure 1: Performance of AS-FW (blue line) and FW (red line) on a 3D example. Strict complementarity holds at the solution.

built instances of the following problem

$$\begin{array}{ll} \min\limits_{x\in\mathbb{R}^n} & \frac{1}{2}x^TQx - c^Tx \\ \text{s.t.} & e^Tx = 1 \\ & x \geq 0, \end{array} \tag{36}$$

where $Q \succ 0$ is randomly generated and $c \in \mathbb{R}^n$ is chosen so that the randomly generated solution x^* satisfies the strict complementarity condition.

More specifically, we generated artificial problems with

- dimension $n = 2^{13}$;
- number of nonzero components in the optimal solution $T = \text{round}(\rho\, n)$, with $\rho \in \{0.01, 0.03, 0.05, 0.07, 0.1\}$.

We then defined $c = Qx^* - r$, where $r \in \mathbb{R}^n$ is such that $r_i = 1$ if $x_i^* > 0$ and $r_i > 1$ if $x_i^* = 0$. In this way, we ensured the satisfaction of the strict complementarity conditions at x^*.

All the considered algorithms employed the vector e_1 as starting point and were stopped at the fist iteration k satisfying

$$\nabla h(x^k)^T(x - x^k) \geq -10^{-6},$$

for all x in the feasible set, where h is the objective function of (36). Moreover, we arrested an algorithm when the number of iterations exceeded $200\,T$. For every ρ, the results have been averaged over 10 runs.

In Figure 2, we plot the objective function error $E^k = f(x^k) - f(x^*)$ versus the computational time. We can easily see that the use of the active-set estimate significantly improves the performance of the algorithms for every considered sparsity level ρ. In particular, when using `AS-AFW` and `AS-PFW`, we notice a pretty fast reduction that enables those algorithms to stop much earlier than the original `AFW` and `PFW`.

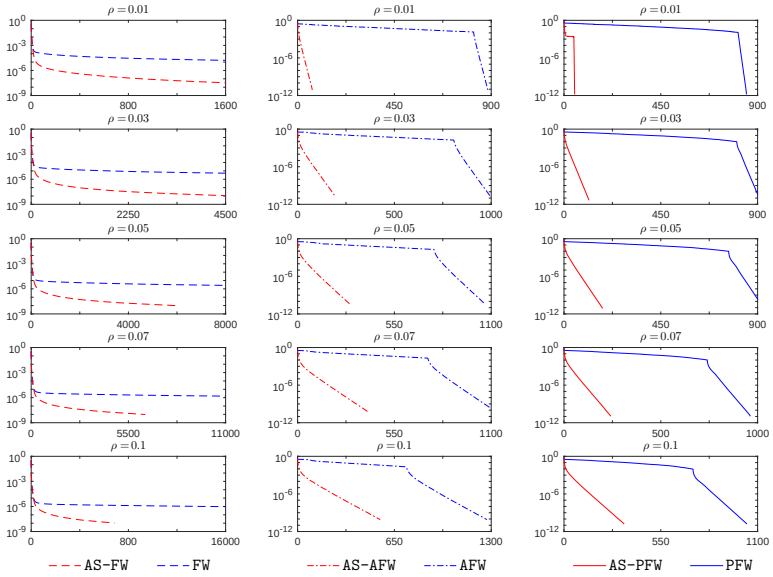

Figure 2: Objective function error vs CPU time (in seconds). Comparison between original and active-set Frank-Wolfe variants. Strict complementarity holds at the solution. The y axis is in logarithmic scale.

7.3 Comparison on lasso problems

We further tested our active-set variants on problems of the form:

$$\begin{aligned} \min_{x\in\mathbb{R}^n} \ & \frac{1}{2}\|Ax-b\|^2 \\ & \|x\|_1 \leq \tau, \end{aligned} \tag{37}$$

with $A \in \mathbb{R}^{m\times n}$, $b \in \mathbb{R}^m$ and $\tau > 0$.

In order to generate testing problems of the form (37), we followed [9, 12]. More specifically, we generated artificial signals with

- dimension $n \in \{2^{11}, 2^{12}, 2^{13}, 2^{14}\}$;
- number of observations $m = n/4$;
- number of nonzero components in the optimal solution $T = \text{round}(\rho\, m)$, with $\rho \in \{0.01, 0.03, 0.05, 0.07, 0.1\}$.

Matrix A was obtained by generating $m\times n$ independent and identically distributed elements from the Normal distribution $N(0,1)$, and then normalizing the columns. Once matrix A was built, the "true" signal $x^* \in \mathbb{R}^n$ was generated as a vector with all components equal to 0, except for T randomly placed ± 1 spikes. Vector b was build as $Ax^* + \eta$, with η drawn from a normal distribution with mean 0 and variance 10^{-3}. Finally, we set $\tau = 0.99\|x^*\|_1$.

All the considered algorithms employed the origin as starting point and were stopped at the fist iteration k satisfying

$$\nabla h(x^k)^T(x - x^k) \geq -10^{-6}, \quad \forall x\colon \|x\|_1 \leq \tau,$$

where h is the objective function of (37). Moreover, we arrested an algorithm when the number of iterations exceeded $10\,T$. For every fixed n and ρ, the results have been averaged over 10 runs.

For what concerns `FW`, we actually did not observe significant differences when the active-set estimate is employed. Namely, `AS-FW` and `FW` perform quite similarly. This might be due to the fact that lasso instances do not usually satisfy the assumptions needed to get the linear rate.

On the other hand, for both `AFW` and `PFW`, the use of the active-set estimate leads to remarkable improvements. In particular, in Figure 3, we compare `AS-AFW` with `AFW`, and in Figure 4 we compare `AS-PFW` with `PFW`. In both figures, we plot the objective function versus the computational time (function error is not considered in this case, since the value $f(x^*)$ is not available). It is clear that the objective function decreases much faster when the active-set estimate is employed, for every considered dimension n and sparsity level ρ.

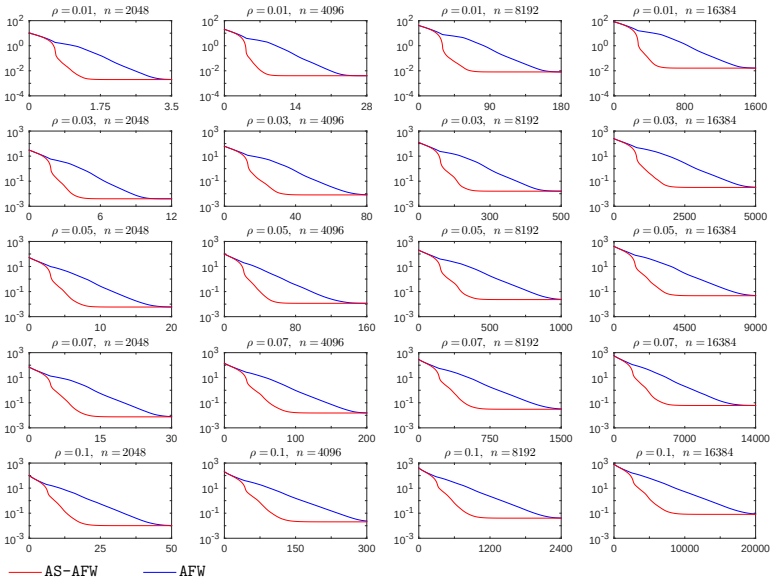

Figure 3: Objective function vs CPU time (in seconds). Comparison between `AS-AFW` and `AFW`. The y axis is in logarithmic scale.

8 Conclusions

In this paper, we focused on minimization problems over the simplex and the ℓ_1-ball, and described an active-set algorithmic framework that encompasses different active-set Frank-Wolfe variants. The active-set strategy embedded in the framework does not only focus on the zero variables and keep them fixed, but rather tries to quickly identify as many active variables as possible (including nonzero variables) at a given point. Furthermore, it suitably reduces the objective function (when setting to zero those variables estimated active), while guaranteeing feasibility. This last feature enable us to make our framework globally convergent. In particular, we proved convergence for three different active-set

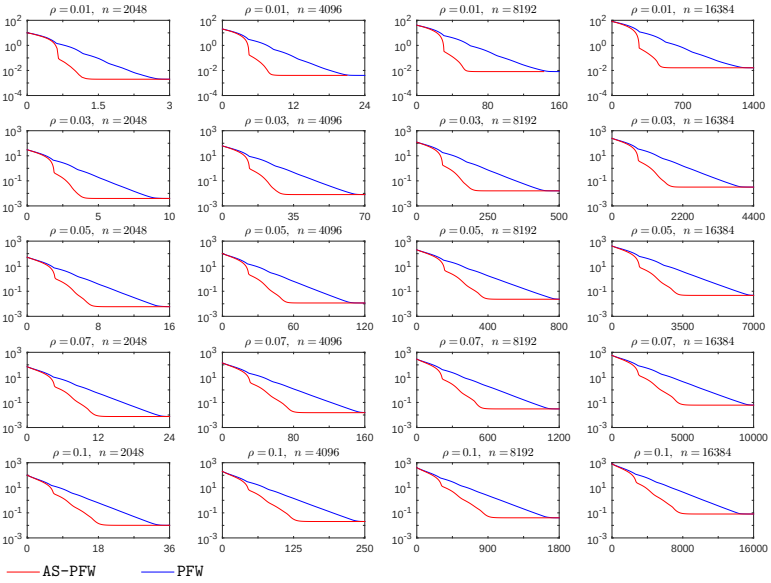

Figure 4: Objective function vs CPU time (in seconds). Comparison between `AS-PFW` and `PFW`. The y axis is in logarithmic scale.

Frank-Wolfe variants. We also showed that all active-set variants converge at a linear rate (convex case) under weaker assumptions than the classical counterparts. The numerical results highlighted that our active-set strategy gives a significant speedup when embedded into a Frank-Wolfe like algorithm.

A Theoretical results related to the property of the active-set estimate

In this appendix, we analyze the theoretical results related to the main property of the active set estimate.

Proof of Proposition 1. We distinguish two different cases. First, we consider the case $|J(x)| = n$. For every $j \in J(x)$, we have

$$\nabla f(x)^T x = \nabla_j f(x) e^T x = \nabla_j f(x), \tag{38}$$

Exploiting the feasibility of x, we can choose an index $\nu \in J(x)$ such that $x_\nu > 0$ and, recalling definition of multipliers (7) and equation (38), we can write

$$\mu_\nu(x) = \nabla_\nu f(x) - \lambda(x) = \nabla_\nu f(x) - \nabla f(x)^T x = \nabla_\nu f(x) - \nabla_\nu f(x) = 0 < x_\nu.$$

Since $x_\nu > 0$ and $\mu_\nu(x) = 0$, we have that $x_\nu > \epsilon\mu_\nu(x)$ and then $\nu \in N(x)$.

Now, let us assume that $|J(x)| < n$. We consider two subcases. We first assume that for every h such that $\nabla_h f(x) > \nabla_j f(x)$, $j \in J(x)$, we have $x_h = 0$. It follows that

$$\sum_{j \in J(x)} x_j = 1$$

and, reasoning as in the previous case, we get that (38) holds for all $j \in J(x)$. Using the fact that x is a feasible solution for problem (1), we can choose an index $\nu \in J(x)$ such that $x_\nu > 0$ and, recalling definition of multipliers (7) and equation (38), we can write

$$\mu_\nu(x) = \nabla_\nu f(x) - \lambda(x) = \nabla_\nu f(x) - \nabla f(x)^T x = \nabla_\nu f(x) - \nabla_\nu f(x) = 0 < x_\nu.$$

Since $x_\nu > 0$ and $\mu(x) = 0$, we have that $x_\nu > \epsilon\mu_\nu(x)$ and then $\nu \in N(x)$.

Now, we consider the case when there exists h such that $\nabla_h f(x) > \nabla_j f(x)$, $j \in J(x)$, and $x_h > 0$. It follows that

$$\nabla f(x)^T x > \nabla_j f(x) e^T x = \nabla_j f(x).$$

Choosing $\nu = j$, for any $j \in J(x)$, and reasoning as before, we can write

$$\mu_\nu(x) = \nabla_\nu f(x) - \lambda(x) = \nabla_\nu f(x) - \nabla f(x)^T x < \nabla_\nu f(x) - \nabla_\nu f(x) = 0 \leq x_\nu.$$

Since $x_\nu \geq 0$ and $\mu_\nu(x) < 0$, we have that $x_\nu > \epsilon\mu_\nu(x)$ and then $\nu \in N(x)$. □

Proof of Proposition 2. Define $\hat{A} = \hat{A}(x)$. Using the mean value theorem, we can write:

$$f(\tilde{x}) = f(x) + \nabla f(w)^T(\tilde{x} - x),$$

where $w = x + \xi(\tilde{x} - x)$, $\xi \in (0, 1)$. From the Lipschitz continuity of the gradient, we have that

$$\begin{aligned} f(\tilde{x}) &= f(x) + \nabla f(x)^T(\tilde{x} - x) + \big[\nabla f(w) - \nabla f(x)\big]^T(\tilde{x} - x) \\ &\leq f(x) + \nabla f(x)^T(\tilde{x} - x) + \big\|\nabla f(w) - \nabla f(x)\big\|\big\|\tilde{x} - x\big\| \\ &\leq f(x) + \nabla f(x)^T(\tilde{x} - x) + L\|\tilde{x} - x\|^2 \end{aligned}$$

and, by adding and removing $L\|\tilde{x} - x\|^2$, we get

$$f(\tilde{x}) \leq f(x) + \nabla f(x)^T(\tilde{x} - x) + 2L\|\tilde{x} - x\|^2 - L\|\tilde{x} - x\|^2. \tag{39}$$

In order to prove the proposition, we need to show that

$$\nabla f(x)^T(\tilde{x} - x) + 2L\|\tilde{x} - x\|^2 \leq 0. \tag{40}$$

From the definition of $\tilde{x}$, we get

$$\|\tilde{x} - x\|^2 = \sum_{i \in \hat{A}} (x_i)^2 + \left(\sum_{i \in \hat{A}} x_i\right)^2 \leq \sum_{i \in \hat{A}} (x_i)^2 + |\hat{A}| \sum_{i \in \hat{A}} (x_i)^2 = (|\hat{A}| + 1) x_{\hat{A}}^T x_{\hat{A}} \tag{41}$$

and

$$\nabla f(x)^T(\tilde{x} - x) = -\nabla_{\hat{A}} f(x)^T x_{\hat{A}} + \nabla_j f(x) \sum_{i \in \hat{A}} x_i = x_{\hat{A}}^T \Big(\nabla_j f(x) e_{\hat{A}} - \nabla_{\hat{A}} f(x)\Big). \tag{42}$$

From the definition of the index j, we have that $\nabla_i f(x) \geq \nabla_j f(x)$ for all $i \in \{1, \ldots, n\}$. Therefore, we can write

$$\sum_{i=1}^{n} \nabla_i f(x) x_i \geq \sum_{i=1}^{n} \nabla_j f(x) x_i = \nabla_j f(x) \sum_{i=1}^{n} x_i = \nabla_j f(x). \tag{43}$$

Recalling the active-set estimate and using (43), we have that

$$x_i \leq \epsilon\Big(\nabla_i f(x) - \sum_{i=1}^{n} \nabla_i f(x) x_i\Big) \leq \epsilon\big(\nabla_i f(x) - \nabla_j f(x)\big), \quad \forall i \in \hat{A},$$

so that, by (41), we can write

$$\|\tilde{x} - x\|^2 \leq \epsilon(|A| + 1)\, x_{\hat{A}}^T \Big(\nabla_{\hat{A}} f(x) - \nabla_j f(x) e_{\hat{A}} \Big). \tag{44}$$

From (42) and (44), we get

$$\begin{aligned}
\nabla f(x)^T(\tilde{x} - x) + 2L\|\tilde{x} - x\|^2 &\leq x_{\hat{A}}^T \big[\nabla_j f(x) e_{\hat{A}} - \nabla_{\hat{A}} f(x) \big] + \\
&\quad + 2L(|A| + 1)\epsilon\, x_{\hat{A}}^T \Big(\nabla_{\hat{A}} f(x) - \nabla_j f(x) e_{\hat{A}} \Big) \\
&= \big[2L(|A| + 1)\epsilon - 1 \big]\, x_{\hat{A}}^T \Big(\nabla_{\hat{A}} f(x) - \nabla_j f(x) e_{\hat{A}} \Big) \\
&\leq (2Ln\epsilon - 1)\, x_{\hat{A}}^T \Big(\nabla_{\hat{A}} f(x) - \nabla_j f(x) e_{\hat{A}} \Big),
\end{aligned}$$

where the last inequality follows from the non-negativity of $x_{\hat{A}}^T \Big(\nabla_{\hat{A}} f(x) - \nabla_j f(x) e_{\hat{A}} \Big)$ (implied by (44)) and from the fact that $|A| \leq n - 1$ (implied by Proposition 1). Then, inequality (40) follows from the assumption we made on ϵ. □

B Theoretical results related to the convergence analysis

Here, we report the proofs of the theoretical results related to the convergence analysis. We first start with the two lemmas describing the properties of the descent directions.

Proof of Lemma 1. First, we consider $d^k = d^{\mathrm{FW}}$. We have that

$$\nabla f(\tilde{x}^k)^T d^k \leq -\nabla_{\hat{\imath}} f(\tilde{x}^k) \sum_{h \in N^k} \tilde{x}_h^k + \nabla_{\hat{\imath}} f(\tilde{x}^k) = 0,$$

where the first inequality follows from the definition of $\hat{\imath}$ in (13) and the feasibility of $\tilde{x}^k$ and the last equality follows from the fact that $\tilde{x}_{A^k} = 0$.

Now, we consider $d^k = d^{\mathrm{AFW}}$. As we have already shown that the assertion holds when $d_{N^k}^k = d_{N^k}^{\mathrm{FW}}$, we only have to prove that $\nabla f(\tilde{x}^k)^T d^k \leq 0$ when $d_{N^k}^k = d_{N^k}^{\mathrm{A}}$. In this case, we have

$$\nabla f(\tilde{x}^k)^T d^k \leq \nabla_{\hat{\jmath}} f(\tilde{x}^k) \sum_{h \in N^k} \tilde{x}_h^k - \nabla_{\hat{\jmath}} f(\tilde{x}^k) = 0,$$

where the first inequality follows from the definition of $\hat{\jmath}$ in (14) and the feasibility of $\tilde{x}^k$ and the last equality follows from the fact that $\tilde{x}_{A^k} = 0$.

Finally, it is easy to see that the assertion is true when $d^k = d^{\mathrm{PFW}}$ as well, since $d^{\mathrm{PFW}} = d^{\mathrm{FW}} + d^{\mathrm{AFW}}$. □

Proof of Lemma 2. In the following, we indicate with $\hat{\imath}$ and $\hat{\jmath}$ the indices defined as in (13) and (14), respectively. From the feasibility of $\tilde{x}^k$, and the fact that $\tilde{x}_{A^k} = 0$, we can write

$$\begin{aligned}
\nabla f(\tilde{x}^k)^T d^{\mathrm{FW}} &= \nabla_{\hat{\imath}} f(\tilde{x}^k) - \nabla f(\tilde{x}^k)^T \tilde{x}^k = \nabla_{\hat{\imath}} f(\tilde{x}^k) - \textstyle\sum_{h \in N^k} \nabla_h f(\tilde{x}^k)\, \tilde{x}_h^k \\
&\geq \nabla_{\hat{\imath}} f(\tilde{x}^k) - \nabla_{\hat{\jmath}} f(\tilde{x}^k) \textstyle\sum_{h \in N^k} \tilde{x}_h^k = \nabla_{\hat{\imath}} f(\tilde{x}^k) - \nabla_{\hat{\jmath}} f(\tilde{x}^k) \\
&= \nabla f(\tilde{x}^k)^T d^{\mathrm{PFW}}.
\end{aligned}$$

Similarly, we have that

$$\begin{aligned}
\nabla f(\tilde{x}^k)^T d^{\mathrm{A}} &= \nabla f(\tilde{x}^k)^T \tilde{x}^k - \nabla_{\hat{\jmath}} f(\tilde{x}^k) = \textstyle\sum_{h \in N^k} \nabla_h f(\tilde{x}^k)\, \tilde{x}_h^k - \nabla_{\hat{\jmath}} f(\tilde{x}^k) \\
&\geq \nabla_{\hat{\imath}} f(\tilde{x}^k) \textstyle\sum_{h \in N^k} \tilde{x}_h^k - \nabla_{\hat{\jmath}} f(\tilde{x}^k) = \nabla_{\hat{\imath}} f(\tilde{x}^k) - \nabla_{\hat{\jmath}} f(\tilde{x}^k) \\
&= \nabla f(\tilde{x}^k)^T d^{\mathrm{PFW}}.
\end{aligned}$$

From the above relations, we get

$$\nabla f(\tilde{x}^k)^T d^{\mathrm{PFW}} \leq \min\{\nabla f(\tilde{x}^k)^T d^{\mathrm{FW}}, \nabla f(\tilde{x}^k)^T d^{\mathrm{A}}\} = \nabla f(\tilde{x}^k)^T d^{\mathrm{AFW}}$$

that proves the result. □

Now, we prove the result related to the Armijo line search.

Proof of Proposition 3. We first prove (15). From the instructions of Algorithm 2 and Proposition 2, we can write

$$f(x^{k+1}) \leq f(\tilde{x}^k) \leq f(x^k) - L\|\tilde{x}^k - x^k\|^2. \tag{45}$$

From the continuity of the objective function and the compactness of the feasible set, it follows that

$$\lim_{k\to\infty} [f(x^{k+1}) - f(x^k)] = 0. \tag{46}$$

The above relation, combined with (45), proves (15).

Now, we recall that, by Lemma 1, $\nabla f(\tilde{x}^k)^T d^k \leq 0$. To prove (16), we consider separately the iterations in which $\nabla f(\tilde{x}^k)^T d^k < 0$ from those in which $\nabla f(\tilde{x}^k)^T d^k = 0$. More specifically, we identify two iteration index subsets $H, K \subseteq \{1, 2, \dots\}$, such that:

- $\nabla f(\tilde{x}^k)^T d^k < 0$, for all $k \in K$;
- $H = \{1, 2, \dots\} \setminus K$.

By assumption, Algorithm 1 does not terminate in a finite number of iterations, so that at least one of the above sets is infinite. We assume without loss of generality that both H and K are infinite sets.

From the instructions of the algorithm, it is straightforward to verify that

$$\lim_{k\to\infty,\, k\in H} \nabla f(\tilde{x}^k)^T d^k = 0.$$

Therefore, we limit our analysis to consider the subsequence $\{x^k\}_K$. For all $k \in K$, since $\nabla f(\tilde{x}^k)^T d^k < 0$, the Algorithm 2 computes a value $\alpha^k \in (0, 1]$ in a finite number of iterations, such that

$$f(x^{k+1}) \leq f(\tilde{x}^k) + \gamma\, \alpha^k\, \nabla f(\tilde{x}^k)^T d^k, \quad \forall k \in K,$$

or equivalently,

$$f(\tilde{x}^k) - f(x^{k+1}) \geq \gamma\, \alpha^k\, |\nabla f(\tilde{x}^k)^T d^k|, \quad \forall k \in K.$$

From (15) and (46), we get that the left-hand side of the above inequality converges to zero for $k \to \infty$, hence

$$\lim_{k\to\infty} \alpha^k\, |\nabla f(\tilde{x}^k)^T d^k| = 0. \tag{47}$$

Now, proceeding by contradiction, we assume that (16) does not hold. From the compactness of the feasible set, $\{x^k\}_K$ attains limit points. Let $\bar{x}$ be any limit point of $\{x^k\}_K$. Using (15), since $\{x^k\}$, $\{\tilde{x}^k\}$ and $\{d^k\}$ are bounded, and taking into account that A^k and N^k are subsets of a finite set of indices, without loss of generality we redefine $\{x^k\}_K$ the subsequence such that

$$\lim_{k\to\infty,\, k\in K} x^k = \lim_{k\to\infty,\, k\in K} \tilde{x}^k = \bar{x} \tag{48}$$

and

$$A^k = \hat{A}, \quad N^k = \hat{N}, \quad \forall k \in K, \tag{49}$$

$$\lim_{k\to\infty,\, k\in K} d^k = \bar{d}. \tag{50}$$

As we have assumed that (16) does not hold, then the above relations, combined with the continuity of the gradient, imply that

$$\lim_{k\to\infty,\, k\in K} \nabla f(\tilde{x}^k)^T d^k = \nabla f(\bar{x})^T \bar{d} = -\eta < 0. \tag{51}$$

We first prove that, if (51) holds, then $M > 0$ exists such that

$$\alpha^k_{\max} \geq M, \quad \forall k \in K. \tag{52}$$

By contradiction, let us assume that an infinite subset of K (that we denote with K for simplicity) exists such that

$$\lim_{k\to\infty,\, k\in K} \alpha^k_{\max} = 0. \tag{53}$$

We distinguish three different cases, depending on the strategy used for computing the direction d^k at Step 7–8 in Algorithm 1:

- Case (FW): it is easy to see that we get a contradiction since $\alpha^k_{\max}$ has a constant value equal to 1.
- Case (AFW): recalling the definition of d^{AFW}, the case we need to analyze is the one where we get an infinite subsequence of away-step directions in N^k. So, we assume that an infinite subset $\tilde{K} \subseteq K$ exists such that

  $$d^k_{N^k} = d^{\mathrm{A}}_{N^k}, \quad \forall k \in \tilde{K}.$$

  We have that $\alpha^k_{\max} = \dfrac{\tilde{x}^k_{\hat{\jmath}}}{1 - \tilde{x}^k_{\hat{\jmath}}}$, for all $k \in \tilde{K}$, where $\hat{\jmath}$ is the index computed according to (14).

  Since the number of indices in $\hat{N}$ is finite, we can consider a further subsequence (that we denote with $\tilde{K}$ for simplicity), where the index $\hat{\jmath}$ is fixed. Taking into account (53), it is easy to see that

  $$\lim_{k\to\infty,\, k\in\tilde{K}} \tilde{x}^k_{\hat{\jmath}} = 0. \tag{54}$$

  Now, from (51), (54) and the continuity of $\nabla f(x)$, it follows that an index $\tilde{k} \in \tilde{K}$ exists such that, for all $k \geq \tilde{k}$, $k \in \tilde{K}$, we have that

  $$\nabla f(\tilde{x}^k)^T d^k = \nabla f(\tilde{x}^k)^T (\tilde{x}^k - e_{\hat{\jmath}}) \leq -\frac{\eta}{2},$$
  $$\tilde{x}^k_{\hat{\jmath}} \leq \epsilon \frac{\eta}{2}.$$

  Therefore, we obtain

  $$\tilde{x}^k_{\hat{\jmath}} \leq \epsilon \nabla f(x^k)^T (e_{\hat{\jmath}} - \tilde{x}^k), \quad \forall k \geq \tilde{k}, k \in \tilde{K}.$$

  Recalling (9), this implies that $\hat{\jmath} \in \hat{A}$ and, considering the definition of $\hat{\jmath}$ in (14), we get a contradiction.
- Case (PFW): in this case $\alpha^k_{\max} = \tilde{x}^k_{\hat{\jmath}}$ and the contradiction follows from the same reasoning done above for (AFW).

So, (52) holds. Now, from (47) and (51), we get

$$\lim_{k\to\infty,\, k\in K} \alpha^k = 0. \tag{55}$$

Taking into account (52), it follows that a value $\bar{k} \in K$ exists such that

$$\alpha^k < \alpha^k_{\max}, \quad \forall k \geq \bar{k}, k \in K.$$

In other words, for $k \geq \bar{k}$, $k \in K$, the stepsize α^k cannot be set equal to the maximum stepsize and, taking into account the line search procedure, we can write

$$f\Big(\tilde{x}^k + \frac{\alpha^k}{\delta} d^k\Big) > f(\tilde{x}^k) + \gamma \frac{\alpha^k}{\delta} \nabla f(\tilde{x}^k)^T d^k, \quad \forall k \geq \bar{k},\, k \in K. \tag{56}$$

We can apply the mean value theorem and we have that $\xi_k \in (0,1)$ exists such that

$$f\Big(\tilde{x}^k + \frac{\alpha^k}{\delta} d^k\Big) = f(\tilde{x}^k) + \frac{\alpha^k}{\delta} \nabla f\Big(\tilde{x}^k + \xi_k \frac{\alpha^k}{\delta} d^k\Big)^T d^k, \quad \forall k \geq \bar{k},\, k \in K. \tag{57}$$

By substituting (57) within (56), we have

$$\nabla f\Big(\tilde{x}^k + \xi_k \frac{\alpha^k}{\delta} d^k\Big)^T d^k > \gamma \nabla f(\tilde{x}^k)^T d^k, \quad \forall k \geq \bar{k},\, k \in K. \tag{58}$$

From (55), (48), and exploiting the fact that $\{\xi_k\}$ and $\{d^k\}$ are bounded, we also get

$$\lim_{k \to \infty,\, k \in K} \tilde{x}^k + \xi_k \frac{\alpha^k}{\delta} d^k = \lim_{k \to \infty,\, k \in K} \tilde{x}^k = \bar{x}. \tag{59}$$

Finally, from (51), (58) and (59), we obtain

$$-\eta = \nabla f(\bar{x})^T \bar{d} \geq \gamma \nabla f(\bar{x})^T \bar{d} = -\gamma\eta,$$

which is a contradiction, since we set $\gamma < 1$. □

Finally, we report the proof related to the convergence of our algorithmic framework.

Proof of Theorem 2. First, we consider the case where d^k is computed according to (FW) rule, that is, $d^k = d^{\mathrm{FW}}$. Then, we will prove the remaining two cases.

Let $\{x^k\}$ be the sequence produced by Algorithm 1 and let us assume that a stationary point is not produced in a finite number of iterations. Since the feasible set is compact, then the sequence $\{x^k\}$ attains a limit point x^* and, recalling (15) of Proposition 3, there exists $K \subseteq \mathbb{N}$ such that

$$\lim_{k \to \infty,\, k \in K} x^k = \lim_{k \to \infty,\, k \in K} \tilde{x}^k = x^*. \tag{60}$$

Taking into account the structure of the feasible set, we can characterize a stationary point using the following conditions

$$\nabla f(x)^T (e_i - x) \geq 0, \quad \forall\, i \in \{1, \ldots, n\}.$$

Let $\Phi_i(x)$ be the continuous function defined as

$$\Phi_i(x) = \max\{0, -\nabla f(x)^T (e_i - x)\},$$

that measures the violation of the stationarity conditions for a variable x_i, $i = 1, \ldots, n$.

By contradiction, we assume that x^* is not a stationary point, so that an index $\nu \in \{1, \ldots, n\}$ exists such that

$$|\Phi_\nu(x^*)| > 0. \tag{61}$$

Taking into account that the number of possible different choices of A^k and N^k is finite, we can find a subset of iteration indices $\bar{K} \subseteq K$ such that $A^k = \hat{A}$ and $N^k = \hat{N}$ for all $k \in \bar{K}$.

First, suppose that $\nu \in \hat{A}$. Then, by Definition 3, we can write

$$0 \leq x_\nu^k \leq \epsilon \nabla f(x^k)^T (e_\nu - x^k),$$

so that $\Phi_\nu(x^k) = \max\{0, -\nabla f(x^k)^T (e_\nu - x^k)\} = 0$, for all $k \in \bar{K}$. Therefore, from (60) and the continuity of the function $\Phi_i(\cdot)$, we get a contradiction with (61).

Now, suppose that $\nu \in \hat{N}$. We can choose an index $\bar{\nu} \in \{1, \ldots, n\}$ and a further subset of iteration indices $\hat{K} \subseteq \bar{K}$ such that

$$\Phi_{\bar{\nu}}(\tilde{x}^k) = \max_{i \in \hat{N}} \{\Phi_i(\tilde{x}^k)\}, \quad \forall\ k \in \hat{K}.$$

Hence, for all $k \in \hat{K}$, $\Phi_{\bar{\nu}}(\tilde{x}^k) \geq \Phi_\nu(\tilde{x}^k) \geq 0$, which, by continuity of $\Phi_i(\cdot)$, implies that

$$\Phi_{\bar{\nu}}(x^*) \geq \Phi_\nu(x^*) > 0. \tag{62}$$

From the definition of $\Phi_i(x)$ and $\bar{\nu}$, for all $k \in \hat{K}$ we can write

$$\begin{aligned} \Phi_{\bar{\nu}}(\tilde{x}^k) &= \max\nolimits_{i \in \hat{N}} \big\{\max\{0, -\nabla f(\tilde{x}^k)^T (e_i - \tilde{x}^k)\}\big\} \\ &= -\min\nolimits_{i \in \hat{N}} \{\nabla f(\tilde{x}^k)^T (e_i - \tilde{x}^k)\} = -\nabla f(\tilde{x}^k)^T d^{\mathrm{FW}}. \end{aligned} \tag{63}$$

Since we are considering the case where $d^k = d^{\mathrm{FW}}$, from (60), (16) of Proposition 3 and the continuity of $\Phi_i(\cdot)$, we have that

$$0 = \lim_{\substack{k \to \infty \\ k \in \hat{K}}} \nabla f(\tilde{x}^k)^T d^k = \lim_{\substack{k \to \infty \\ k \in \hat{K}}} -\Phi_{\bar{\nu}}(\tilde{x}^k) = -\Phi_{\bar{\nu}}(x^*),$$

which, combined with (62), implies that $\Phi_\nu(x^*) = 0$, thus contradicting (61). Then, the assertion is proved for $d^k = d^{\mathrm{FW}}$.

Now, we consider together the cases where $d^k = d^{\mathrm{AFW}}$ and $d^k = d^{\mathrm{PFW}}$. In both cases, we can apply the same reasoning made before and we obtain (63) again. Recalling the definition of d^{AFW} and Lemma 2, we can write

$$-\nabla f(\tilde{x}^k)^T d^k \geq -\nabla f(\tilde{x}^k)^T d^{\mathrm{FW}} = \Phi_{\bar{\nu}}(\tilde{x}^k),$$

for both $d^k = d^{\mathrm{AFW}}$ and $d^k = d^{\mathrm{PFW}}$. Consequently,

$$0 = \lim_{\substack{k \to \infty \\ k \in \hat{K}}} \nabla f(\tilde{x}^k)^T d^k \leq \lim_{\substack{k \to \infty \\ k \in \hat{K}}} -\Phi_{\bar{\nu}}(\tilde{x}^k) = -\Phi_{\bar{\nu}}(x^*),$$

which, combined with (62), implies that $\Phi_\nu(x^*) = 0$, thus contradicting (61). Then, the assertion is also proved for $d^k = d^{\mathrm{AFW}}$ and $d^k = d^{\mathrm{PFW}}$. □

C Theoretical results related to the convergence rate analysis

We first prove that active-set Frank-Wolfe converges at a linear rate.

Proof of Theorem 3. From Theorem 1, exploiting the fact that strict complementarity holds at x^*, for sufficiently large k we have that

$$N(x^k) = N(\tilde{x}^k) = \bar{N} \quad \text{and} \quad A(x^k) = A(\tilde{x}^k) = \bar{A}.$$

From the instructions of `AS-SIMPLEX`, it follows that $\tilde{x}^k = x^k$ for sufficiently large k, and then the minimization is restricted to $N^k = \bar{N}$. Since the search direction d^k is computed according to (FW) rule, the rest of the proof follows by repeating the same arguments of the proof given for Theorem 3 in [17], observing that $\mu_f(\bar{N}) > 0$ and $C_f(\bar{N}) < \infty$ under the hypothesis we made. □

Now, we report the proof related to the convergence rate analysis of the other active-set variants.

Proof of Theorem 4. First, we observe that Theorem 1 implies that $A^k \supseteq A^+$ and $N^k \subseteq N^+$ for sufficiently large k. Now, we show that there exists an iteration index $\bar{k}$ such that

(i) $x^k_{A^+} = \tilde{x}^k_{A^+} = 0$, for all $k \geq \tilde{k}$;

(ii) $\nabla f(\tilde{x}^k)^T d^k < 0$, for all $k \geq \bar{k} \geq \tilde{k}$.

Point (i) follows from the instructions of the algorithm and the fact that $A^k \supseteq A^+$, for k sufficiently large. In order to prove point (ii) we proceed by contradiction. We assume that an infinite subsequence $\{\tilde{x}^k\}_K$ exists such that $\tilde{x}^k_{N^k}$ satisfies stationarity conditions over Δ_{N^k} (but $\tilde{x}^k$ does not satisfy stationarity conditions over Δ), for all $k \in K$. Since f is strongly convex on Δ_{N^+}, there exists a unique point satisfying stationarity conditions over Δ_{N^k}. Taking into account that A^k and N^k are subsets of a finite set of indices and $\tilde{x}^k_{A_k} = 0$, we have that, after a finite number of iterations, the algorithm should cycle. This cannot be possible as we guarantee a strict decrease of the objective function at each iteration.

Consequently, recalling that $d^k_{A^k} = 0$, we can repeat the same arguments of the proof given for Theorem 8 in [18], to provide the following bound:

$$f(x^{k+1}) - f(x^*) \leq (1-\rho)\big[f(\tilde{x}^k) - f(x^*)\big] \leq (1-\rho)\big[f(x^k) - f(x^*)\big], \quad \forall k \geq \bar{k},$$

where the last inequality follows from the fact that $f(\tilde{x}^k) \leq f(x^k)$ and ρ is defined as in (19). Moreover, we have that $\mu_f^{\Delta}(N^+) > 0$ and $C_f^{\Delta}(N^+) < \infty$ under the hypothesis we made.

Finally, we are left to bound the number of iterations for which k is not a good step. For what concerns away-step Frank-Wolfe, we need to consider those iterations such that $\alpha^k = \alpha^k_{\max} < 1$, for $k \geq \bar{k}$. The fact that $\alpha^k_{\max} < 1$ implies that $d^k = d^{\mathrm{A}}$. Consequently, when $\alpha^k = \alpha^k_{\max}$, we have that $x^{k+1}_{\hat{\jmath}} = 0$, where $\hat{\jmath}$ is the index computed according to (14). In other words, the number of zero components in x^{k+1} increases by 1. From the instructions of the algorithm, we also have that the number of zero components in $\tilde{x}^{k+1}$ cannot decrease from x^{k+1}. Combining these observations with the fact that $\tilde{x}^k_{A^+} = 0$ for all $k \geq \bar{k}$, we conclude that after at most $|N^+| - 1$ iterations with $\alpha^k = \alpha^k_{\max} < 1$, a point $\tilde{x}^k$ with $n-1$ zero components is produced. Of course, we cannot further increase the number of zero components.

For what concerns pairwise Frank-Wolfe, the claimed bound can simply be obtained by adapting the arguments used in [18], recalling that, for $k \geq \bar{k}$, we have that $x^k_{A^+} = \tilde{x}^k_{A^+} = 0$ and the calculation of the search direction is restricted to N^+. □

D Theoretical results related to minimization problems over the ℓ_1-ball

Here, we report the results related to the active-set estimates for minimization problems over the ℓ_1-ball.

Proof of Proposition 4. Let y be the point given by (23). Considering problem (22), we can compute the active and nonactive-set estimates $A(y)$, $N(y)$. From the expression of $\nabla f(y)$, and exploiting the hypothesis that x is non-stationary, it follows that

$$\min_{i=1,\dots,2n+1} \{\nabla_i f(y)\} < 0.$$

In particular, this implies that

$$(2n+1) \notin \operatorname*{Argmin}_{i=1,\dots,2n+1} \{\nabla_i f(y)\}.$$

From Proposition 1, there exists $\nu \in \{1, \dots, 2n\}$ such that

$$\nu \in \operatorname*{Argmin}_{i=1,\dots,2n} \{\nabla_i f(y)\}, \tag{64}$$

$$\nu \in N(y). \tag{65}$$

Recalling the expression of $\nabla f(y)$, we can rewrite (64) as

$$\nabla_\nu f(y) \leq \min_{i=1,\dots,n} \{\tau \nabla_1 h(x), \dots, \tau \nabla_n h(x), -\tau \nabla_1 h(x), \dots, -\tau \nabla_n h(x)\},$$

that is,

$$-|\nabla_\nu f(y)| \leq -\tau |\nabla_i h(x)|, \quad \forall i = 1, \dots, n. \tag{66}$$

Let $j \in \{1, \dots, n\}$ be the following index:

$$j = \begin{cases} \nu, & \text{if } \nu \in \{1, \dots, n\}, \\ \nu - n, & \text{if } \nu \in \{n+1, \dots, 2n\}. \end{cases} \tag{67}$$

Exploiting again the expression of $\nabla f(y)$, we get $|\nabla_\nu f(y)| = |\nabla_j f(y)| = \tau |\nabla_j h(x)|$. This relation, combined with (66), implies that

$$j \in \operatorname*{Argmax}_{i=1,\dots,n} \big\{|\nabla_i h(x)|\big\}.$$

Finally, using (65) and (67), it follows that at least one index between j and $(n+j)$ belongs to $N(y)$. Recalling (26), we have that $j \in N_{\ell_1}(x)$ and the assertion is proved. □

Proof of Proposition 5. Define $\hat{A} = \hat{A}_{\ell_1}(x)$. As in the proof of Proposition 2, using the mean value theorem, we get

$$h(\tilde{x}) \leq h(x) + \nabla h(x)^T (\tilde{x} - x) + 2L \|\tilde{x} - x\|^2 - L \|\tilde{x} - x\|^2$$

and, in order to prove the proposition, we need to show that

$$\nabla h(x)^T (\tilde{x} - x) + 2L \|\tilde{x} - x\|^2 \leq 0. \tag{68}$$

From the definition of $\tilde{x}$, reasoning as in the proof of Proposition 2, we get

$$\|\tilde{x} - x\|^2 \leq (|\hat{A}| + 1) x_{\hat{A}}^T x_{\hat{A}} = (|\hat{A}| + 1)(sgn(x_{\hat{A}}) \cdot x_{\hat{A}})^T (sgn(x_{\hat{A}}) \cdot x_{\hat{A}}), \tag{69}$$

where the product $sgn(x_{\hat{A}}) \cdot x_{\hat{A}}$ has to be intended componentwise. Furthermore, we have

$$\begin{aligned} &\nabla h(x)^T (\tilde{x} - x) = -\nabla_{\hat{A}} h(x)^T x_{\hat{A}} - |\nabla_j h(x)| \textstyle\sum_{i \in \hat{A}} |x_i| \\ &= x_{\hat{A}}^T \Big(-|\nabla_j h(x)| sgn(x_{\hat{A}}) \cdot e_{\hat{A}} - \nabla_{\hat{A}} h(x) \Big) \\ &= (sgn(x_{\hat{A}}) \cdot x_{\hat{A}})^T \Big(-|\nabla_j h(x)| e_{\hat{A}} - \nabla_{\hat{A}} h(x) \cdot sgn(x_{\hat{A}}) \Big). \end{aligned} \tag{70}$$

From the definition of the index j, we have that $\nabla_i h(x) \geq -|\nabla_j h(x)|$ for all $i \in \{1, \dots, n\}$. Therefore, we can write

$$\begin{aligned} &\textstyle\sum_{i=1}^n \nabla_i h(x) x_i = \sum_{i=1}^n \nabla_i h(x) sgn(x_i) |x_i| \\ &\geq \textstyle\sum_{i=1}^n -|\nabla_j h(x)| |x_i| = -|\nabla_j h(x)| \|x\|_1 \geq -|\nabla_j h(x)| \tau. \end{aligned} \tag{71}$$

Recalling the active-set estimate and using (71), we have that

$$x_i \leq \epsilon \tau \left(\nabla_i h(x) \tau - \sum_{l=1}^n \nabla_l h(x) x_l \right) \leq \epsilon \tau \left(\nabla_i h(x) \tau + |\nabla_j h(x)| \tau \right), \quad \forall i \in \hat{A}$$

$$\text{and} \quad -x_i \leq \epsilon\,\tau\left(-\nabla_i h(x)\tau - \sum_{l=1}^{n} \nabla_l h(x) x_l\right) \leq \epsilon\,\tau\left(-\nabla_i h(x)\,\tau + |\nabla_j h(x)|\,\tau\right), \quad \forall i \in \hat{A},$$

so that, we can majorize $|x_i|$, for all $i \in \hat{A}$, as

$$|x_i| = sgn(x_i)x_i \leq \epsilon\,\tau^2\left(\nabla_i h(x)\,sgn(x_i) + |\nabla_j h(x)|\right).$$

By using this majorization in (69), we can write

$$0 \leq \|\tilde{x} - x\|^2 \leq \epsilon\,\tau^2\,(|\hat{A}| + 1)\,(sgn(x_{\hat{A}}) \cdot x_{\hat{A}})^T\Big(\nabla_{\hat{A}} h(x) \cdot sgn(x_{\hat{A}}) + |\nabla_j h(x)| e_{\hat{A}}\Big). \tag{72}$$

From (70) and (72), we get

$$\begin{aligned}
\nabla h(x)^T(\tilde{x} - x) + 2L\|\tilde{x} - x\|^2 \leq & \\
&\leq \left[\epsilon\,\tau^2\,2L(|\hat{A}| + 1) - 1\right](sgn(x_{\hat{A}}) \cdot x_{\hat{A}})^T\Big(\nabla_{\hat{A}} h(x) \cdot sgn(x_{\hat{A}}) + |\nabla_j h(x)| e_{\hat{A}}\Big) \\
&\leq (\epsilon\,\tau^2\,2Ln - 1)\,(sgn(x_{\hat{A}}) \cdot x_{\hat{A}})^T\Big(\nabla_{\hat{A}} h(x) \cdot sgn(x_{\hat{A}}) + |\nabla_j h(x)| e_{\hat{A}}\Big),
\end{aligned}$$

where the last inequality follows from the non-negativity of $\nabla_{\hat{A}} h(x) \cdot sgn(x_{\hat{A}}) + |\nabla_j h(x)| e_{\hat{A}}$ (implied by (72)) and from the fact that $|\hat{A}| \leq n-1$ (implied by Proposition 4). Then, inequality (68) follows from the assumption we made on ϵ. □

References

[1] A. Beck and S. Shtern. Linearly convergent away-step conditional gradient for non-strongly convex functions. *Math. Program.*, pages 1–27, 2015.

[2] D. P. Bertsekas. *Nonlinear programming*. Athena scientific Belmont, 1999.

[3] C. Buchheim, M. De Santis, S. Lucidi, F. Rinaldi, and L. Trieu. A Feasible Active Set Method with Reoptimization for Convex Quadratic Mixed-Integer Programming. *SIAM J. Optim.*, 26(3):1695–1714, 2016.

[4] C.-C. Chang and C.-J. Lin. LIBSVM: a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2(3), 2011.

[5] K. L. Clarkson. Coresets, sparse greedy approximation, and the Frank-Wolfe algorithm. *ACM Trans. Algorithms*, 6(4), 2010.

[6] A. Cristofari, M. De Santis, S. Lucidi, and F. Rinaldi. A Two-Stage Active-Set Algorithm for Bound-Constrained Optimization. *J. Optim. Theory Appl.*, pages 1–33, 2016.

[7] E. de Klerk. The complexity of optimizing over a simplex, hypercube or sphere: a short survey. *European J. Oper. Res.*, 16(2):111–125, 2008.

[8] M. De Santis, G. Di Pillo, and S. Lucidi. An active set feasible method for large-scale minimization problems with bound constraints. *Comput. Opt. Appl.*, 53(2):395–423, 2012.

[9] M. De Santis, S. Lucidi, and F. Rinaldi. A Fast Active Set Block Coordinate Descent Algorithm for ℓ_1-regularized least squares. *SIAM J. Optim.*, 26(1):781–809, 2016.

[10] G. Di Pillo and L. Grippo. A class of continuously differentiable exact penalty function algorithms for nonlinear programming problems. In *System Modelling and Optimization*, pages 246–256. Springer, 1984.

[11] F. Facchinei and S. Lucidi. Quadratically and superlinearly convergent algorithms for the solution of inequality constrained minimization problems. *J. Optim. Theory Appl.*, 85(2):265–289, 1995.

[12] K. Fountoulakis and J. Gondzio. A second-order method for strongly convex l1-regularization problems. *Math. Program.*, 156:189–219, 2014.

[13] M. Frank and P. Wolfe. An algorithm for quadratic programming. *Naval research logistics quarterly*, 3(1-2):95–110, 1956.

[14] R. M. Freund, P. Grigas, and R. Mazumder. An Extended Frank-Wolfe Method with "In-Face" Directions, and its Application to Low-Rank Matrix Completion. *arXiv preprint arXiv:1511.02204*, 2015.

[15] J. Guélat and P. Marcotte. Some comments on Wolfe's away step. *Math. Program.*, 35(1):110–119, 1986.

[16] M. Jaggi. Revisiting Frank-Wolfe: Projection-Free Sparse Convex Optimization. In *ICML (1)*, pages 427–435, 2013.

[17] S. Lacoste-Julien and M. Jaggi. An affine invariant linear convergence analysis for Frank-Wolfe algorithms. *arXiv preprint arXiv:1312.7864*, 2013.

[18] S. Lacoste-Julien and M. Jaggi. On the global linear convergence of Frank-Wolfe optimization variants. In *NIPS 2015 - Advances in Neural Information Processing Systems*, 2015.

[19] C.-J. Lin. On the convergence of the decomposition method for support vector machines. *IEEE Transactions on Neural Networks*, 12(6):1288–1298, 2001.

[20] B. Mitchell, V. F. Dem'yanov, and V. Malozemov. Finding the point of a polyhedron closest to the origin. *SIAM Journal on Control*, 12(1):19–26, 1974.

[21] J. J. Moré and G. Toraldo. On the Solution of Large Quadratic Programming Problems with Bound Constraints. *SIAM J. Optim.*, 1(1):93–113, 1991.

[22] R. Ñanculef, E. Frandi, C. Sartori, and H. Allende. A novel frank–wolfe algorithm. analysis and applications to large-scale svm training. *Inform. Sci*, 285:66–99, 2014.

[23] J. Nocedal and S. Wright. *Numerical Optimization*. Springer, 2000.

[24] J. M. Ortega and W. C. Rheinboldt. *Iterative solution of nonlinear equations in several variables*. SIAM, 2000.

[25] J. Peña, D. Rodríguez, and N. Soheili. On the von Neumann and Frank–Wolfe Algorithms with Away Steps. *SIAM J. Optim.*, 26(1):499–512, 2016.

[26] J. C. Platt. Fast training of support vector machines using sequential minimal optimization. *Advances in kernel methods*, pages 185–208, 1999.

[27] A. Raj, J. Olbrich, B. Gärtner, B. Schölkopf, and M. Jaggi. Screening Rules for Convex Problems. *arXiv preprint arXiv:1609.07478*, 2016.

[28] P. Wolfe. Convergence theory in nonlinear programming. *Integer and nonlinear programming*, pages 1–36, 1970.