# Walking in the Shadow: A New Perspective on Descent Directions for Constrained Minimization

**Hassan Mortagy**
Georgia Institute of Technology
hmortagy@gatech.edu

**Swati Gupta**
Georgia Institute of Technology
swatig@gatech.edu

**Sebastian Pokutta**
Zuse Institute Berlin and Technische Universität Berlin
pokutta@zib.de

## Abstract

Descent directions such as movement towards Frank-Wolfe vertices, away steps, in-face away steps and pairwise directions have been an important design consideration in conditional gradient descent (CGD) variants. In this work, we attempt to demystify the impact of movement in these directions towards attaining constrained minimizers. The best local direction of descent is the directional derivative of the projection of the gradient, which we refer to as the *shadow* of the gradient. We show that the continuous-time dynamics of moving in the shadow are equivalent to those of PGD however non-trivial to discretize. By projecting gradients in PGD, one not only ensures feasibility but also is able to "wrap" around the convex region. We show that Frank-Wolfe (FW) vertices in fact recover the maximal wrap one can obtain by projecting gradients, thus providing a new perspective to these steps. We also claim that the shadow steps give the best direction of descent emanating from the convex hull of all possible away-vertices. Opening up the PGD movements in terms of shadow steps gives linear convergence, dependent on the number of faces. We combine these insights into a novel SHADOW-CG method that uses FW steps (i.e., wrap around the polytope) and shadow steps (i.e., optimal local descent direction), while enjoying linear convergence. Our analysis develops properties of directional derivatives of projections (which may be of independent interest), while providing a unifying view of various descent directions in the CGD literature.

## 1 Introduction

We consider the problem $\min_{\mathbf{x}\in P} f(\mathbf{x})$, where $P \subseteq \mathbb{R}^n$ is a polytope with vertex set vert$(P)$, and $f : P \to \mathbb{R}$ is a smooth and strongly convex function. Smooth convex optimization problems over polytopes are an important class of problems that appear in many settings, such as low-rank matrix completion [1], structured supervised learning [2, 3], electrical flows over graphs [4], video co-localization in computer vision [5], traffic assignment problems [6], and submodular function minimization [7]. First-order methods in convex optimization rely on movement in the best local direction for descent (e.g., negative gradient), and this is enough to obtain linear convergence for unconstrained optimization. In constrained settings however, the gradient may no longer be a feasible direction of descent, and there are two broad classes of methods traditionally: projection-based methods (i.e., move in direction of negative gradient, but project to ensure feasibility), and conditional gradient methods (i.e., move in feasible directions that approximate the gradient). Projection-based methods such as projected gradient descent or mirror descent [8] enjoy dimension independent linear rates of convergence (assuming no acceleration), i.e., $(1 - \frac{\mu}{L})$ contraction in the objective per iteration (so that the number of iterations to get an $\epsilon$-accurate solution is $O(\frac{L}{\mu} \log \frac{1}{\epsilon})$), for $\mu$-strongly convex and $L$-smooth functions, but need to compute an expensive projection step (another constrained convex optimization) in (almost) every iteration. On the other hand, conditional gradient methods

34th Conference on Neural Information Processing Systems (NeurIPS 2020), Vancouver, Canada.

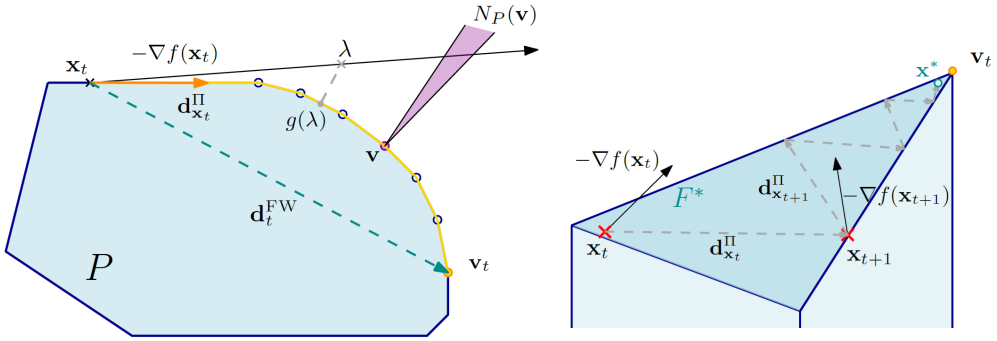


Figure 1: **Left:** Piecewise linear structure of the parametric projection curve $g(\lambda) = \Pi_P(\mathbf{x}_t - \lambda\nabla f(\mathbf{x}_t))$ (yellow line). The end point is the FW vertex $\mathbf{v}_t$ and $\mathbf{d}^{\mathrm{FW}}_t$ the FW direction. Note that $g(\lambda)$ does not change at the same speed as $\lambda$, e.g., $g(\lambda) = \mathbf{v}$ for each $\lambda$ such that $\mathbf{x}_t - \lambda\nabla f(\mathbf{x}_t) - \mathbf{v} \in N_P(\mathbf{v})$ (purple normal cone). **Right:** Moving along the shadow might lead to arbitrarily small progress even once we reach the optimal face $F^* \ni \mathbf{x}^*$. On the contrary, the away-steps FW does not leave $F^*$ after a polytope-dependent iteration [11].

(such as the Frank-Wolfe algorithm [9]) need to solve linear optimization (LO) problems in every iteration and the rates of convergence become dimension-dependent, for e.g., the away-step Frank-Wolfe algorithm has a linear rate of $(1 - \frac{\mu\delta^2}{LD^2})$, where $\delta$ is a geometric constant (polytope dependent) and $D$ is the diameter of the polytope [10].

The vanilla Conditional Gradient method (CG) or the Frank-Wolfe algorithm (FW) [9, 12] has received a lot of interest from the ML community mainly because of its iteration complexity, tractability and sparsity of iterates. In each iteration, the CG algorithm computes the *Frank-Wolfe* vertex $\mathbf{v}_t$ with respect to the current iterate and moves towards the vertex:

$$\mathbf{v}_t = \arg\min_{\mathbf{v}\in \mathrm{vert}(P)} \langle \nabla f(\mathbf{x}_t), \mathbf{v}\rangle\,, \quad \mathbf{x}_{t+1} = \mathbf{x}_t + \gamma_t(\mathbf{v}_t - \mathbf{x}_t), \gamma_t \in [0,1]. \tag{1}$$

CG's primary direction of descent is $\mathbf{v}_t - \mathbf{x}_t$ ($\mathbf{d}^{\mathrm{FW}}_t$ in Figure 1) and its step-size $\gamma_t$ can be selected, e.g., using line-search; this ensures feasibility of $\mathbf{x}_{t+1}$. This algorithm however, can only guarantee a sub-linear rate of $O(1/t)$ for smooth and strongly convex optimization on a compact domain [9, 2], moreover, this rate is tight [13, 14]. An active area of research, therefore, has been to find other descent directions that can enable linear convergence. One reason for vanilla CG's $O(1/t)$ rate is the fact that the algorithm might zig-zag as it approaches the optimal face, slowing down progress [10, 13]. The key idea for obtaining linear convergence was to use the so-called *away-steps* that help push iterates quickly to the optimal face:

$$\mathbf{a}_t = \arg\max_{\mathbf{v}\in \mathrm{vert}(F)} \langle \nabla f(\mathbf{x}_t), \mathbf{v}\rangle\,, \text{ for } F \subseteq P, \tag{2}$$

$$\mathbf{x}_{t+1} = \mathbf{x}_t + \gamma_t(\mathbf{x}_t - \mathbf{a}_t), \text{ where } \gamma_t \in \mathbb{R}_+ \text{ such that } \mathbf{x}_{t+1} \in P, \tag{3}$$

thus, augmenting the potential directions of descent using directions of the form $\mathbf{x}_t - \mathbf{a}_t$, for some $\mathbf{a}_t \in F$, where the precise choice of $F$ in (2) has evolved in CG variants. As early as 1986, Guélat and Marcotte showed that by adding away-steps (with $F =$ minimal face of the current iterate[1]) to vanilla CG, their algorithm has an asymptotic linear convergence rate [11]. In 2015, Lacoste-Julien and Jaggi [10] showed linear convergence results for CG with away-steps[2] (over $F =$ the current active set, i.e., a specific convex decomposition of the current iterate). They also showed linear rate for CG with pairwise-steps (i.e., $\mathbf{v}_t - \mathbf{a}_t$), another direction of descent. In 2015, Freund et. al [1] showed a $O(1/t)$ convergence for convex functions, with $F$ as the minimal face of the current iterate. In 2016, Garber and Meshi [16] showed that pairwise-steps (over 0/1 polytopes) with respect to non-zero components of the gradient are enough for linear convergence, i.e., they also set $F$ to be the minimal face with respect to $\mathbf{x}_t$. In 2017, Bashiri and Zhang [3] generalized this result to show linear convergence for the same $F$ for general polytopes (however at the cost of two expensive oracles). Other CG variants have explored movement towards either the convex or affine minimizer over current active set [10], constraining the Frank-Wolfe vertex to a norm ball around the current iterate ([14], [15]), and mixing FW with gradient descent steps (with the aim of better computational performance) while enjoying linear convergence [17], [18]. Although these variants obtain linear convergence, their rates depend on polytope-dependent geometric, affine-variant constants (that can be arbitrarily small for

[1]The minimal face $F$ with respect to $\mathbf{x}_t$ is a face of the polytope that contains $\mathbf{x}_t$ in its relative interior, i.e., all active constraints at $\mathbf{x}_t$ are tight.

[2]To the best of our knowledge, Garber and Hazan [15] were the first to present a CG variant with global linear convergence for polytopes.

non-polyhedral sets like the $\ell_2$-ball) such as the pyramidal width [10], vertex-facet distance [19], eccentricity of the polytope [10] or sparsity-dependent constants [3], which have been shown to be essentially equivalent[3] [20]. The iterates in these are (basically) affine-invariant, which is the reason why a dimension-dependent factor is unavoidable in the current arguments. We include more details on related work (and a summary in Table 1) in Appendix A, with updated references to recent results that appeared after this work [21, 22].

A natural question at this point is why are these different descent directions useful and which of these are necessary for linear convergence. If one had oracle access to the "best" local direction of descent for constrained minimization, what would it be and is it enough to get linear convergence (as in unconstrained optimization)? Moreover, can we avoid rates of convergence that are dependent on the geometry of the polytope? We partially answer these questions below.

**Contributions.** We show that the "best" local feasible direction of descent, that gives the maximum function value decrease in the diminishing neighborhood of the current iterate $\mathbf{x}_t$, is the *directional derivative* $\mathbf{d}^{\Pi}_{\mathbf{x}_t}$ of the projection of the gradient, which we refer to as the *shadow* of the gradient:

$$\mathbf{d}^{\Pi}_{\mathbf{x}_t} := \lim_{\epsilon \downarrow 0} \frac{\Pi_P(\mathbf{x}_t - \epsilon \nabla f(\mathbf{x}_t)) - \mathbf{x}_t}{\epsilon},$$

where $\Pi_P(\mathbf{y}) = \arg\min_{\mathbf{x} \in P} \|\mathbf{x} - \mathbf{y}\|^2$ is the Euclidean projection operator. A continuous time dynamical system can be defined using descent in the shadow direction at the current point: $\dot{X}(t) = \mathbf{d}^{\Pi}_{X(t)}$, for $X(0) = \mathbf{x}_0 \in P$. We show that this ODE is equivalent to that of projected gradient descent (Theorem 9), however, it is non-trivial to discretize due to non-differentiability of the curve.

Second, we explore structural properties of shadow steps. For any $\mathbf{x} \in P$, we characterize the curve $g(\lambda) = \Pi_P(\mathbf{x} - \lambda \nabla f(\mathbf{x}))$ as a piecewise linear curve, where the breakpoints of the curve typically occur at points where there is a change in the normal cone (Theorem 1) and show how to compute this curve for all $\lambda \geq 0$ (Theorem 3). Moreover, we show the following properties for descent directions:

(i) **Shadow Steps ($\mathbf{d}^{\Pi}_{\mathbf{x}_t}$)**: These are the best "normalized" feasible directions of descent (Lemma 3). Moreover, we show that $\|\mathbf{d}^{\Pi}_{\mathbf{x}_t}\| = 0$ if and only if $\mathbf{x}_t = \arg\min_{\mathbf{x} \in P} f(\mathbf{x})$ (Lemma 12). Hence, $\|\mathbf{d}^{\Pi}_{\mathbf{x}_t}\|$ is a natural quantity to use for bounding primal gaps without any dependence on geometric constants like those used in other CG variants. We show that multiple shadow steps approximate a single projected gradient descent step (Theorem 3). The rate of linear convergence using shadow steps is dependent on number of facets (independent of geometric constants but dimension dependent due to number of facets), and *interpolate smoothly* between projected gradient and conditional gradient methods (Theorem 6).

(ii) **FW Steps ($\mathbf{v}_t - \mathbf{x}_t$)**: Projected gradient steps provide a contraction in the objective independent of the geometric constants or facets of the polytope; they are also able to "wrap" around the polytope by taking unconstrained gradient steps and then projecting. Under mild technical conditions (of uniqueness of $\mathbf{v}_t$), the Frank-Wolfe vertices are in fact the projection of an infinite descent in the negative gradient direction (Theorem 4). This allows the CG methods to wrap around the polytope maximally, compared to PGD methods, thereby giving FW steps a new perspective.

(iii) **Away Steps ($\mathbf{x}_t - \mathbf{a}_t$)**: Shadow steps are the *best normalized away-direction* in the following sense: let $F$ be the minimal face containing the current iterate $\mathbf{x}_t$ (similar to [16, 3]); then, $\mathbf{x}_t - \gamma \mathbf{d}^{\Pi}_{\mathbf{x}_t} \in \text{conv}(F)$ (i.e., the backward extension from $\mathbf{x}_t$ in the shadow direction), and the resultant direction ($\mathbf{d}^{\Pi}_{\mathbf{x}_t}$) is indeed the most aligned with $-\nabla f(\mathbf{x}_t)$ (Lemma 3). Shadow-steps are, however, in general convex combinations of potential active vertices minus the current iterate (Lemma 4) and therefore loose combinatorial properties such as dimension drop in active sets. They can bounce off faces (and add facets back) unlike away-steps that use vertices and have a monotone decrease in dimension when they are consecutive (see Figure 1 (right)).

(iv) **Pairwise Steps ($\mathbf{v}_t - \mathbf{a}_t$)**: The progress in CG variants is bounded crucially using the inner product of the descent direction with the negative gradient. In this sense, pairwise steps are simply the *sum of the FW step and away directions*, and a simple algorithm that uses these steps only does converge linearly (with geometric constants) [10, 3]. Moreover, for feasibility of the descent direction, one requires $\mathbf{a}_t$ to be in an active set (shown in [3], and Lemma 13, Appendix C.4).

[3] Eccentricity $= D/\delta$, where $D$ and $\delta$ are the diameter and pyramidal width of the domain respectively [10].

Armed with these structural properties, we consider a descent algorithm SHADOW-WALK: trace the projections curve by moving in the shadow (or in-face directional derivative) with respect to a fixed iterate until sufficient progress, then update the shadow based on the current iterate. Using properties of normal cones, we can show that once the projections curve at a fixed iterate leaves a face, it can never visit the face again (Theorem 8). We are thus able to break a single PGD step into descent steps, and show linear convergence with rate dependent on the number of facets, but independent of geometric constants like the pyramidal width. Finally, we combine these insights into a novel SHADOW-CG method which uses FW steps (i.e., wrap around the polytope) and shadow steps (i.e., optimal local descent direction), while enjoying linear convergence. This method prioritizes FW steps that achieve maximal "coarse" progress in earlier iterations and shadow steps avoid zig-zagging in the latter iterations. Garber and Meshi [16] and Bashiri and Zhang [3] both compute the best away vertex in the minimal face containing the current iterate, whereas the shadow step recovers the best convex combination of such vertices aligned with the negative gradient. Therefore, these previously mentioned CG methods can *both* be viewed as approximations of SHADOW-CG. Moreover, Garber and Hazan [15] emulate a shadow computation by constraining the FW vertex to a ball around the current iterate. Therefore, their algorithm can be interpreted as an approximation of SHADOW-WALK.

**Outline** We next review preliminaries in Section 2. In Section 3, we derive theoretical properties of the directional derivative and the piecewise-linear curve parameterized by projections. This allows us to dig deeper into properties of descent directions in Section 4. We defer equivalence of continuous time dynamics for movement along the shadow and PGD, as well as SHADOW-WALK algorithm to Section D in the appendix. We next propose a novel SHADOW-CG algorithm that combines FW and shadow steps to obtain linear convergence in Section 6. Finally, preliminary experiments demonstrate that SHADOW-CG outperforms classical and state of the art methods, when assuming oracle access to the shadow. Without oracle access, it interpolates lower iteration count than CG variants (i.e., close to PGD) and higher speed than PGD (i.e., close to CG), thus obtaining the best of both worlds.

## 2 Preliminaries

Let $\|\cdot\|$ denote the Euclidean norm. Denote $[m] = \{1, \dots, m\}$ and let $P$ be defined in the form

$$P = \{\mathbf{x} \in \mathbb{R}^n : \langle \mathbf{a}_i, \mathbf{x} \rangle \leq b_i \; \forall \; i \in [m]\}. \tag{4}$$

We use $\text{vert}(P)$ to denote the vertices of $P$. A function $f : \mathcal{D} \to \mathbb{R}$ (for $\mathcal{D} \subseteq \mathbb{R}^n$ and $P \subseteq \mathcal{D}$) is said to be $L-$smooth if $f(\mathbf{y}) \leq f(\mathbf{x}) + \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle + \frac{L}{2}\|\mathbf{y} - \mathbf{x}\|^2$ for all $\mathbf{x}, \mathbf{y} \in \mathcal{D}$. Furthermore, $f : \mathcal{D} \to \mathbb{R}$ is said to be $\mu-$strongly-convex if $f(\mathbf{y}) \geq f(\mathbf{x}) + \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle + \frac{\mu}{2}\|\mathbf{y} - \mathbf{x}\|^2$ for all $\mathbf{x}, \mathbf{y} \in \mathcal{D}$. Let $D := \sup_{\mathbf{x},\mathbf{y} \in P} \|\mathbf{x} - \mathbf{y}\|$ be the diameter of $P$ and $\mathbf{x}^* = \arg\min_{\mathbf{x} \in P} f(\mathbf{x})$, where uniqueness follows from the strong convexity of the $f$. For any $\mathbf{x} \in P$, let $I(\mathbf{x}) = \{i \in [m] : \langle \mathbf{a}_i, \mathbf{x} \rangle = b_i\}$ be the index set of active constraints at $\mathbf{x}$. Similarly, let $J(\mathbf{x})$ be the index set of inactive constraints at $\mathbf{x}$. Denote by $\mathbf{A}_{I(\mathbf{x})} = [\mathbf{a}_i]_{i \in I(\mathbf{x})}$ the sub-matrix of active constraints at $\mathbf{x}$ and $\mathbf{b}_{I(\mathbf{x})} = [b_i]_{i \in I(\mathbf{x})}$ the corresponding right-hand side. The normal cone at a point $\mathbf{x} \in P$ is defined as

$$N_P(\mathbf{x}) := \{\mathbf{y} \in \mathbb{R}^n : \langle \mathbf{y}, \mathbf{z} - \mathbf{x} \rangle \leq 0 \; \forall \mathbf{z} \in P\} = \{\mathbf{y} \in \mathbb{R}^n : \exists \boldsymbol{\mu} : \mathbf{y} = (A_{I(\mathbf{x})})^T \boldsymbol{\mu}, \; \boldsymbol{\mu} \geq 0\}, \tag{5}$$

which is essentially the the cone of the normals of constraints tight at $\mathbf{x}$. Let $\Pi_P(\mathbf{y}) = \arg\min_{\mathbf{x} \in P} \frac{1}{2}\|\mathbf{x} - \mathbf{y}\|^2$ be the Euclidean projection operator. Using first-order optimality,

$$\langle \mathbf{y} - \mathbf{x}, \mathbf{z} - \mathbf{x} \rangle \leq 0 \quad \forall \mathbf{z} \in P \quad \iff \quad (\mathbf{y} - \mathbf{x}) \in N_P(\mathbf{x}), \tag{6}$$

which implies that $\mathbf{x} = \Pi_P(\mathbf{y})$ if and only if $(\mathbf{y} - \mathbf{x}) \in N_P(\mathbf{x})$, i.e., moving any closer to $\mathbf{y}$ from $\mathbf{x}$ will violate feasibility in $P$. Finally, it is well known that the Euclidean projection operator over convex sets is non-expansive (see for example [23]): $\|\Pi_P(\mathbf{y}) - \Pi_P(\mathbf{x})\| \leq \|\mathbf{y} - \mathbf{x}\|$ for all $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$. Given any point $\mathbf{x} \in P$ and $\mathbf{w} \in \mathbb{R}^n$, let the directional derivative of $\mathbf{w}$ at $\mathbf{x}$ be:

$$\mathbf{d}^{\Pi}_{\mathbf{x}}(\mathbf{w}) := \lim_{\epsilon \downarrow 0} \frac{\Pi_P(\mathbf{x} - \epsilon \mathbf{w}) - \mathbf{x}}{\epsilon}. \tag{7}$$

When $\mathbf{w} = \nabla f(\mathbf{x})$, then we call $\mathbf{d}^{\Pi}_{\mathbf{x}}(\nabla f(\mathbf{x}))$ the *shadow* of the gradient at $\mathbf{x}$, and use notation $\mathbf{d}^{\Pi}_{\mathbf{x}}$ for brevity. In [24], Tapia et. al show that $\mathbf{d}^{\Pi}_{\mathbf{x}}$ is the projection of $-\nabla f(\mathbf{x})$ onto the tangent cone at $\mathbf{x}$ (i.e. the set of feasible directions at $\mathbf{x}$), that is $\mathbf{d}^{\Pi}_{\mathbf{x}} = \arg\min_{\mathbf{d}}\{\| - \nabla f(\mathbf{x}) - \mathbf{d}\|^2 : \mathbf{A}_{I(\mathbf{x})}\mathbf{d} \leq \mathbf{0}\}$, where the uniqueness of the solution follows from strong convexity of the objective. Further, let $\hat{\mathbf{d}}^{\Pi}_{\mathbf{x}}(\nabla f(\mathbf{x})) := \arg\min_{\mathbf{d}}\{\| - \nabla f(\mathbf{x}) - \mathbf{d}\|^2 : \mathbf{A}_{I(\mathbf{x})}\mathbf{d} = \mathbf{0}\} = (\mathbf{I} - \mathbf{A}^{\dagger}_{I(\mathbf{x})}\mathbf{A}_{I(\mathbf{x})})(-\nabla f(\mathbf{x}))$ be the

projection of $-\nabla f(\mathbf{x})$ onto the minimal face of $\mathbf{x}$, where $\mathbf{I} \in \mathbb{R}^{n\times n}$ is the identity matrix, and $\mathbf{A}^{\dagger}_{I(\mathbf{x})}$ is the Moore-Penrose inverse of $\mathbf{A}_{I(\mathbf{x})}$ (see Section 5.13 in [25] for example).

We assume access to (i) a *linear optimization* (LO) oracle where we can compute $\mathbf{v} = \arg\min_{\mathbf{x}\in P} \langle \mathbf{c}, \mathbf{x}\rangle$ for any $\mathbf{c} \in \mathbb{R}^n$, (ii) a *shadow oracle*: given any $\mathbf{x} \in P$ we can compute $\mathbf{d}^{\Pi}_{\mathbf{x}}$, and (iii) *line-search* oracle: given any $\mathbf{x} \in P$ and direction $\mathbf{d} \in \mathbb{R}^n$, we can evaluate $\gamma^{\max} = \max\{\delta : \mathbf{x} + \delta\mathbf{d} \in P\}$. This helps us focus on properties of descent directions and studying their necessity for linear convergence.

## 3 Structure of the Parametric Projections Curve

In this section, we characterize properties of the directional derivative at any $\mathbf{x} \in P$ and the structure of the parametric projections curve $g_{\mathbf{x},\mathbf{w}}(\lambda) = \Pi_P(\mathbf{x} - \lambda\mathbf{w})$, for $\lambda \in \mathbb{R}$, under Euclidean projections. For brevity, we use $g(\cdot)$ when $\mathbf{x}$ and $\mathbf{w}$ are clear from context. The following theorem summarizes our results on characterization and is crucial to our analysis of descent directions:

**Theorem 1** (Structure of Parametric Projection Curve)**.** *Let $P \subseteq \mathbb{R}^n$ be a polytope, with $m$ facet inequalities (e.g., as in* (4)*). For any $\mathbf{x}_0 \in P, \mathbf{w} \in \mathbb{R}^n$, let $g(\lambda) = \Pi_P(\mathbf{x}_0 - \lambda\mathbf{w})$ be the projections curve at $\mathbf{x}_0$ with respect to $\mathbf{w}$ parametrized by $\lambda \in \mathbb{R}$. Then, this curve is piecewise linear starting at $\mathbf{x}_0$: there exist $k$ breakpoints $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_k \in P$, corresponding to projections with $\lambda$ equal to $0 = \lambda_0^- \le \lambda_0^+ < \lambda_1^- \le \lambda_1^+ < \lambda_2^- \le \lambda_2^+ \ldots < \lambda_k^- \le \lambda_k^+$, where*

*(a)* $\lambda_i^- := \min\{\lambda \ge 0 \mid g(\lambda) = \mathbf{x}_i\}$, *and* $\lambda_i^+ := \max\{\lambda \ge 0 \mid g(\lambda) = \mathbf{x}_i\}$, *for* $i \ge 0$,

*(b)* $g(\lambda) = \mathbf{x}_{i-1} + \frac{\mathbf{x}_i - \mathbf{x}_{i-1}}{\lambda_i^- - \lambda_{i-1}^+}(\lambda - \lambda_{i-1}^+)$, *for* $\lambda \in [\lambda_{i-1}^+, \lambda_i^-]$ *for all* $i \ge 1$.

*Moreover, we show the following properties for each $i \ge 1$, and all $\lambda, \lambda' \in (\lambda_{i-1}^+, \lambda_i^-)$:*

*(i)* ***Potentially drop tight constraints on leaving breakpoints:*** $N_P(\mathbf{x}_{i-1}) = N_P(g(\lambda_{i-1}^+)) \supseteq N_P(g(\lambda))$ *for $i \ge 1$. Moreover, if $\lambda_{i-1}^- < \lambda_{i-1}^+$, then the containment is strict.*

*(ii)* ***Constant normal cone between breakpoints:*** $N_P(g(\lambda)) = N_P(g(\lambda'))$,

*(iii)* ***Potentially add tight constraints on reaching breakpoint:*** $N_P(g(\lambda)) \subseteq N_P(g(\lambda_i^-)) = N_P(\mathbf{x}_i)$.

*Further, the following properties also hold:*

*(iv)* ***Equivalence of constant normal cones with linearity:*** *If $N_P(g(\lambda)) = N_P(g(\lambda'))$ for some $\lambda < \lambda'$, then the curve between $g(\lambda)$ and $g(\lambda')$ is linear (Lemma 2).*

*(v)* ***Bound on breakpoints:*** *The number of breakpoints of $g(\cdot)$ is at most the number of faces of the polytope (Theorem 8, Appendix B.5).*

*(vi)* ***Limit of*** $g(\cdot)$***:*** *The end point of the curve $g(\lambda)$ is $\lim_{\lambda\to\infty} g(\lambda) = \mathbf{x}_k \in \arg\min_{\mathbf{x}\in P}\langle \mathbf{x}, \mathbf{w}\rangle$. In fact, $\mathbf{x}_k$ minimizes $\|\mathbf{y} - \mathbf{x}_0\|$ over $\mathbf{y} \in \arg\min_{\mathbf{x}\in P}\langle \mathbf{x}, \mathbf{w}\rangle$ (Theorem 4, Section 4).*

To show the above theorem, we need to develop the properties of the projection curve. Even though our results hold for any $\mathbf{w} \in \mathbb{R}^n$, we will prove the statements for $\mathbf{w} = \nabla f(\mathbf{x}_0)$ for readability in the context of the paper, in Appendix B. We first show that if the direction $\mathbf{w}$ is in the normal cone at the starting point, then the parametric curve reduces to a single point $\mathbf{x}_0$.

**Lemma 1.** *If $-\nabla f(\mathbf{x}_0) \in N_P(\mathbf{x}_0)$, then $g(\lambda) = \Pi_P(\mathbf{x}_0 - \lambda\nabla f(\mathbf{x}_0)) = \mathbf{x}_0$ for all $\lambda \in \mathbb{R}_+$.*

This means, in the notation of Theorem 1, $\lambda_0^+$ is either infinity (when $\mathbf{w} \in N_P(\mathbf{x}_0)$) or it is zero. In the former case, Theorem 1 hold trivially with $g(\lambda) = \mathbf{x}_0$ for all $\lambda \in \mathbb{R}$. We will therefore assume henceforth that $\lambda_0^+ = 0$, without loss of generality. We next prove property (iv) of Theorem 1 about equivalence of constant normal cones with linearity of the parametric projections between two points.

**Lemma 2** (Linearity of projections)**.** *Let $P \subseteq \mathbb{R}^n$ be a polytope defined using $m$ facet inequalities (e.g., as in* (4)*). Let $\mathbf{x}_0 \in P$ and we are given $\nabla f(\mathbf{x}_0) \in \mathbb{R}^n$. Let $g(\lambda) = \Pi_P(x_0 - \lambda\nabla f(\mathbf{x}_0))$ be the parametric projections curve. Then, if $N_P(g(\lambda)) = N_P(g(\lambda'))$ for some $\lambda < \lambda'$, then the curve between $g(\lambda)$ and $g(\lambda')$ is linear, i.e., $g(\delta\lambda + (1-\delta)\lambda') = \delta g(\lambda) + (1-\delta)g(\lambda')$, where $\delta \in [0, 1]$.*

We next show that the normal cones do not change in the *strict* neighborhood of $\mathbf{x}_0$, i.e., there exists a ball $B(\mathbf{x}_0, \delta)$ around $\mathbf{x}_0$ of radius $\delta > 0$ such that the normal cone $N_P(g(\lambda)) = N_P(g(\lambda'))$ for all $g(\lambda), g(\lambda') \in B(\mathbf{x}_0, \delta) \setminus \{\mathbf{x}_0\}$. Using Lemma 2, we get that the first piece of $g(\lambda)$ is linear until the normal cone changes. Moreover, some inequalities tight at $\mathbf{x}_0$ might become inactive for $\lambda > 0$:

**Theorem 2.** *Let $P \subseteq \mathbb{R}^n$ be a polytope defined using $m$ facet inequalities (e.g., as in (4)). Let $\mathbf{x}_0 \in P$ and we are given $\nabla f(\mathbf{x}_0) \in \mathbb{R}^n$. Let $g(\lambda) = \Pi_P(\mathbf{x}_0 - \lambda \nabla f(\mathbf{x}_0))$ be the parametric projections curve. Let $\lambda_1^- = \max\{\lambda \mid \mathbf{x}_0 + \lambda \mathbf{d}^{\Pi}_{\mathbf{x}_0} \in P\}$ be finite and let $\mathbf{x}_1 = g(\lambda_1^-)$. We claim that*

*(i) $N_P(g(\lambda)) = N_P(g(\lambda')) \subseteq N_P(\mathbf{x}_0)$, for all $0 < \lambda < \lambda' < \lambda_1^-$, and*

*(ii) $N_P(\mathbf{x}_1) = N_P(g(\lambda_1^-)) \supset N_P(g(\lambda))$, for all $\lambda \in (0, \lambda_1^-)$.*

*Moreover, the projections curve is given by $g(\lambda) = \mathbf{x}_0 + \lambda \mathbf{d}^{\Pi}_{\mathbf{x}_0}$, for all $\lambda \in [0, \lambda_1^-]$.*

The proof of the above theorem uses the first-order optimality of projections given in (6) and the structure of normal cones for polytopes (5). Theorem 2 characterizes the first linear piece in the parametric projections trajectory. This means that the direction $\mathbf{d} = (\mathbf{x}_1 - \mathbf{x}_0)/\lambda_1^-$ is the directional derivative at $\mathbf{x}_0$, since by definition of the directional derivative at $\mathbf{x}_0$, we get:

$$\mathbf{d}^{\Pi}_{\mathbf{x}_0} := \lim_{\epsilon \downarrow 0} \frac{\Pi_P(\mathbf{x}_0 - \epsilon \nabla f(\mathbf{x}_0)) - \mathbf{x}_0}{\epsilon} = \lim_{\epsilon \downarrow 0} \frac{g(\epsilon) - \mathbf{x}_0}{\epsilon} = \frac{\mathbf{x}_1 - \mathbf{x}_0}{\lambda_1^-}, \tag{8}$$

where the limit exists since $g(\lambda)$ forms a line on the interval $\lambda \in [0, \lambda_1^-)$ (and hence is a continuous function on that interval).[4] This theorem also gives a way of computing the directional derivative $\mathbf{d}^{\Pi}_{\mathbf{x}}$ using a single projection (when we know the breakpoint $\lambda_1^-$).

We now show that $g(\lambda) = \Pi_P(\mathbf{x}_0 - \lambda \nabla f(\mathbf{x}_0))$ can be constructed for all $\lambda \geq 0$ iteratively as follows: given a breakpoint $\mathbf{x}_{i-1}$, the next segment and breakpoint $\mathbf{x}_i$ of the curve can be obtained (a) by either projecting $\nabla f(\mathbf{x}_0)$ onto the minimal face of $\mathbf{x}_{i-1}$ (i.e., in-face movement, using a linear program, (see Appendix B.5 for more details)); or (b) by projecting $\nabla f(\mathbf{x}_0)$ onto the tangent cone at $\mathbf{x}_{i-1}$, and computing this using line search in the directional derivative at $\mathbf{x}_{i-1}$ with respect to $\nabla f(\mathbf{x}_0)$). This proves Theorem 1 (i), (ii), and (iii) by induction.

**Theorem 3** (Tracing the projections curve)**.** *Let $P \subseteq \mathbb{R}^n$ be a polytope defined using $m$ facet inequalities (e.g., as in (4)). Let $\mathbf{x}_{i-1} \in P$ be the $i$th breakpoint in the projections curve $g(\lambda) = \Pi_P(\mathbf{x}_0 - \lambda \nabla f(\mathbf{x}_0))$, with $\mathbf{x}_{i-1} = \mathbf{x}_0$ for $i = 1$. Suppose we are given $\lambda_{i-1}^-, \lambda_{i-1}^+ \in \mathbb{R}$ so that they are respectively the minimum and the maximum step-sizes $\lambda$ such that $g(\lambda) = \mathbf{x}_{i-1}$. Let $\hat{\lambda}_{i-1} := \sup\{\lambda \mid N_P(g(\lambda')) = N_P(\mathbf{x}_{i-1}) \; \forall \lambda' \in [\lambda_{i-1}^-, \lambda)\}$. Then, we show that:*

1. *If $\lambda_{i-1}^- < \lambda_{i-1}^+$, then $\lambda_{i-1}^+ = \hat{\lambda}_{i-1}$. Otherwise, $\lambda_{i-1}^- = \lambda_{i-1}^+ \leq \hat{\lambda}_{i-1}$.*
2. *Linearity of the curve between $g(\lambda_{i-1}^-)$ and $g(\hat{\lambda}_{i-1})$: i.e., $g(\lambda_{i-1}^- + (1-\delta)\hat{\lambda}_{i-1}) = \delta g(\lambda_{i-1}) + (1-\delta) g(\hat{\lambda}_{i-1})$, where $\delta \in [0, 1]$. In particular, $g(\lambda) = \mathbf{x}_{i-1}$ for all $\lambda \in [\lambda_{i-1}^-, \lambda_{i-1}^+]$.*
3. *If $\mathbf{d}^{\Pi}_{\mathbf{x}_{i-1}}(\nabla f(\mathbf{x}_0)) = \mathbf{0}$, then $\lim_{\lambda \to \infty} g(\lambda) = \mathbf{x}_{i-1}$ is the end point of the projections curve $g(\lambda)$.*
4. *Otherwise $\mathbf{d}^{\Pi}_{\mathbf{x}_{i-1}}(\nabla f(\mathbf{x}_0)) \neq \mathbf{0}$, we get $\lambda_{i-1}^+ \leq \hat{\lambda}_{i-1} < \infty$ (from (1)). We then claim:*
   - *(a)* **In-face movements:** *If $\hat{\lambda}_{i-1} > \lambda_{i-1}^+$, then the next breakpoint in the curve occurs by walking in-face up to $\hat{\lambda}_{i-1}$, i.e., $\mathbf{x}_i := g(\hat{\lambda}_{i-1}) = \mathbf{x}_{i-1} + (\hat{\lambda}_{i-1} - \lambda_{i-1}^+)\hat{\mathbf{d}}^{\Pi}_{\mathbf{x}_{i-1}}(\nabla f(\mathbf{x}_0))$ and $\lambda_i^- := \hat{\lambda}_{i-1}$. Moreover, $N_P(\mathbf{x}_{i-1}) \subseteq N_P(g(\hat{\lambda}_{i-1}))$, with strict containment only when the maximum movement along in-face direction takes place, i.e., $\hat{\lambda}_{i-1} = \lambda_{i-1}^+ + \max\{\delta : \mathbf{x}_{i-1} + \delta \hat{\mathbf{d}}^{\Pi}_{\mathbf{x}_{i-1}}(\nabla f(\mathbf{x}_0)) \in P\}$.*
   - *(b)* **Shadow movements:** *Otherwise if $\hat{\lambda}_{i-1} = \lambda_{i-1}^+$, then the movement is in the shadow direction, i.e., $\mathbf{x}_i := g(\lambda_i^-) = \mathbf{x}_{i-1} + (\lambda_i^- - \lambda_{i-1}^+)\mathbf{d}^{\Pi}_{\mathbf{x}_{i-1}}(\nabla f(\mathbf{x}_0))$ where $\lambda_i^- := \lambda_{i-1}^+ + \max\{\delta : \mathbf{x}_{i-1} + \delta \mathbf{d}^{\Pi}_{\mathbf{x}_{i-1}}(\nabla f(\mathbf{x}_0)) \in P\}$.*

*In particular, the projections curve is linear between $\lambda_{i-1}^+$ and $\lambda_i^-$. Further, we show that properties (i), (ii) and (iii) in Theorem 1 hold for their respective normal cones for $\lambda, \lambda' \in (\lambda_{i-1}^+, \lambda_i^-)$, where the containments in (i) and (iii) are strict for case (b).*

Assuming oracle access to compute $\mathbf{d}^{\Pi}_{\mathbf{x}}(\mathbf{w})$ and $\hat{\lambda}_{i-1}$ for any $\mathbf{x} \in P$, Theorem 3 gives a constructive method for tracing the whole piecewise linear curve of $g_{\mathbf{x},\mathbf{w}}(\cdot)$. We include this as an algorithm, TRACE$(\mathbf{x}, \mathbf{w})$ and discuss more details on its implementation in Appendix B.5. We defer the proof on the number of breakpoints (Theorem 1 (v)) in the parametric projections curve to Appendix B.5 (Theorem 8), which crucially uses Lemma 2. Using Theorem 1, it is easy to see that multiple line searches in *shadow directions* with respect to $\mathbf{x}_0$ are equivalent to computing a single projected gradient descent step from $\mathbf{x}_0$. This will be useful in our analysis of SHADOW-CG in Section 6.

[4] This gives a different proof for existence of $\mathbf{d}^{\Pi}_{\mathbf{x}}$ for polytopes, compared to Tapia et. al [24].

## 4 Descent Directions

Having characterized the properties of the parametric projections curve, we highlight connections with descent directions in conditional gradient variants. We first claim that the shadow is the best local feasible direction of descent in the following sense - it has the highest inner product with the negative gradient at $\mathbf{x}$ compared to any other normalized feasible direction (proof in Appendix C.1):

**Lemma 3** (Local Optimality of Shadow Steps)**.** *Let $P$ be a polytope defined as in* (4) *and let $\mathbf{x} \in P$ with gradient $\nabla f(\mathbf{x})$. Let $\mathbf{y}$ be any feasible direction at $\mathbf{x}$, i.e., $\exists \gamma > 0$ s.t. $\mathbf{x} + \gamma \mathbf{y} \in P$. Then*

$$\left\langle -\nabla f(\mathbf{x}), \frac{\mathbf{d}_{\mathbf{x}}^{\Pi}}{\|\mathbf{d}_{\mathbf{x}}^{\Pi}\|} \right\rangle^2 = \|\mathbf{d}_{\mathbf{x}}^{\Pi}\|^2 \geq \left\langle \mathbf{d}_{\mathbf{x}}^{\Pi}, \frac{\mathbf{y}}{\|\mathbf{y}\|} \right\rangle^2 \geq \left\langle -\nabla f(\mathbf{x}), \frac{\mathbf{y}}{\|\mathbf{y}\|} \right\rangle^2. \tag{9}$$

The above lemma will be useful in convergence proof for our novel SHADOW-CG method (Theorem 7). We also show that the shadow steps give a true estimate of convergence to optimal[5], in the sense that $\|\mathbf{d}_{\mathbf{x}_t}^{\Pi}\| = 0$ if and only if $\mathbf{x}_t = \arg\min_{\mathbf{x} \in P} f(\mathbf{x})$ (Lemma 12). On the other hand, note that $\|\nabla f(\mathbf{x}_t)\|$ does not satisfy this property and can be strictly positive at the constrained optimal solution [12]. We next show that the end point of the projections curve is in fact the FW vertex under mild technical conditions. FW vertices are therefore able to wrap around the polytope maximally compared to any projected gradient method and serve as an anchor point in the projections curve.

**Theorem 4** (Optimism in Frank-Wolfe Vertices)**.** *Let $P \subseteq R^n$ be a polytope and let $\mathbf{x} \in P$. Let $g(\lambda) = \Pi_P(\mathbf{x} - \lambda \nabla f(\mathbf{x}))$ for $\lambda \geq 0$. Then, the end point of this curve is: $\lim_{\lambda \to \infty} g(\lambda) = \mathbf{v}^* = \arg\min_{\mathbf{v} \in F} \|\mathbf{x} - \mathbf{v}\|^2$, where $F = \arg\min_{\mathbf{v} \in P} \langle \nabla f(\mathbf{x}), \mathbf{v} \rangle$, i.e., the face of $P$ that minimizes the gradient $\nabla f(\mathbf{x})$. In particular, if $F$ is a vertex, then $\lim_{\lambda \to \infty} g(\lambda) = \mathbf{v}^*$ is the Frank-Wolfe vertex.*

To give a quick proof sketch, using the proximal definition of the projection (see e.g., [23]) we have:

$$g(\lambda) = \arg\min_{\mathbf{y} \in P} \{\|\mathbf{x} - \lambda \nabla f(\mathbf{x}) - \mathbf{y}\|^2\} = \arg\min_{\mathbf{y} \in P} \left\{ f(\mathbf{x}) + \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle + \frac{\|\mathbf{x} - \mathbf{y}\|^2}{2\lambda} \right\}.$$

Assuming that the FW vertex $\arg\min_{\mathbf{y} \in P} \{\langle \nabla f(\mathbf{x}), \mathbf{y} \rangle\}$ is unique and we show that one can interchange the limit and $\arg\min$ operator, we get $\lim_{\lambda \to \infty} g(\lambda) = \arg\min_{\mathbf{y} \in P} \{f(\mathbf{x}) + \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle\}$, thus recovering the FW vertex. The complete analysis is technical and included in Appendix C.3.

Next, we show that the shadow-steps also give the best away direction emanating from away-vertices in the minimal face at any $x \in P$ (which is precisely the set of *possible* away vertices (see Appendix C.4)), using Lemma 3 and the following result:

**Lemma 4** (Away-Steps)**.** *Let $P$ be a polytope defined as in* (4) *and fix $\mathbf{x} \in P$. Let $F = \{\mathbf{z} \in P : \mathbf{A}_{I(\mathbf{x})}\mathbf{z} = \mathbf{b}_{I(\mathbf{x})}\}$ be the minimal face containing $\mathbf{x}$. Further, choose $\delta_{\max} = \max\{\delta : \mathbf{x} - \delta \mathbf{d}_{\mathbf{x}}^{\Pi} \in P\}$ and consider the maximal backward away point $\mathbf{a}_{\mathbf{x}} = \mathbf{x} - \delta_{\max} \mathbf{d}_{\mathbf{x}}^{\Pi}$. Then, $\mathbf{a}_{\mathbf{x}}$ lies in $F$ and the corresponding away-direction is simply $\mathbf{x} - \mathbf{a}_{\mathbf{x}} = \delta_{\max} \mathbf{d}_{\mathbf{x}}^{\Pi}$.*

Lemma 4 states that the backward extension from $\mathbf{x}$ in the shadow direction, $\mathbf{a}_{\mathbf{x}}$, lies in the convex hull of $A := \{\mathbf{v} \in \text{vert}(P) \cap F\}$. The set $A$ is precisely the set of all possible away vertices (see Appendix C.4). Thus, the shadow gives the best direction of descent emanating from the convex hull of all possible away-vertices. We include a proof of this lemma in Appendix C.4.

## 5 Shadow-Walk and Continuous-time Dynamics

We established in the last section that the shadow of the negative gradient $\mathbf{d}_{\mathbf{x}_t}^{\Pi}$ is indeed the best "local" direction of descent (Lemma 3), and a true measure of primal gaps since convergence in $\|\mathbf{d}_{\mathbf{x}_t}^{\Pi}\|$ implies optimality (Lemma 12). Having characterized the parametric projections curve, the natural question is if a shadow-descent algorithm that walks along the directional derivative with respect to negative gradient at iterate $\mathbf{x}_t$ (using say line search), converge linearly? We start by answering that question positively for continuous-time dynamics.

### 5.1 ODE for moving in the shadow of gradient

We now present the continuous-time dynamics for moving along the shadow of the gradient in the polytope. Let $X(t)$ denote the continuous-time trajectory of our dynamics and $\dot{X}$ denote the time-derivative of $X(t)$, i.e., $\dot{X}(t) = \frac{d}{dt} X(t)$. The continuous time dynamics of tracing the shadow are

[5] Lemma 3 with $\mathbf{y} = \mathbf{x}^* - \mathbf{x}$ can be used to estimate the primal gap: $\|\mathbf{d}_{\mathbf{x}}^{\Pi}\|^2 \geq 2\mu(f(\mathbf{x}) - f(\mathbf{x}^*))$ (see (63))

**Algorithm 1** SHADOW-WALK Algorithm

**Input:** Polytope $P \subseteq \mathbb{R}^n$, function $f : P \to \mathbb{R}$ and initialization $\mathbf{x}_0 \in P$.

1: **for** $t = 0, \ldots .T$ **do**
2: Update $\mathbf{x}_{t+1} := \text{TRACE}(\mathbf{x}_t, \nabla f(\mathbf{x}_t))$ . ▷ `trace projections curve`
3: **end for**

**Return:** $\mathbf{x}_{T+1}$

**Algorithm 2** Shadow Conditional Gradient (SHADOW-CG)

**Input:** Polytope $P \subseteq \mathbb{R}^n$, function $f : P \to \mathbb{R}$, initialization $\mathbf{x}_0 \in P$ and accuracy parameter $\varepsilon$.

1: **for** $t = 0, \ldots .T$ **do**
2: Let $\mathbf{v}_t := \arg\min_{\mathbf{v} \in P} \langle \nabla f(\mathbf{x}_t), \mathbf{v} \rangle$ and $\mathbf{d}_t^{\text{FW}} := \mathbf{v}_t - \mathbf{x}_t$. ▷ `FW direction`
3: **if** $\left\langle -\nabla f(\mathbf{x}_t), \mathbf{d}_t^{\text{FW}} \right\rangle \leq \varepsilon$ **then return** $\mathbf{x}_t$ ▷ `primal gap is small enough`
4: Compute the derivative of projection of the gradient $\mathbf{d}_{\mathbf{x}_t}^{\Pi}$
5: **if** $\left\langle -\nabla f(\mathbf{x}_t), \mathbf{d}_{\mathbf{x}_t}^{\Pi} / \|\mathbf{d}_{\mathbf{x}_t}^{\Pi}\| \right\rangle \leq \left\langle -\nabla f(\mathbf{x}_t), \mathbf{d}_t^{\text{FW}} \right\rangle$
6: $\mathbf{d}_t := \mathbf{d}_t^{\text{FW}}$ and $\mathbf{x}_{t+1} := \mathbf{x}_t + \gamma_t \mathbf{d}_t$ ($\gamma_t \in [0, 1]$). ▷ `use line-search towards FW vertex`
7: **else** $\mathbf{d}_t := \mathbf{d}_{\mathbf{x}_t}^{\Pi}$ and $\mathbf{x}_{t+1} := \text{TRACE}(\mathbf{x}_t, \nabla f(\mathbf{x}_t))$ . ▷ `trace projection curve`
8: **end for**

**Return:** $\mathbf{x}_{T+1}$

simply $\dot{X}(t) = \mathbf{d}_{X(t)}^{\Pi}$, $X(0) = \mathbf{x}_0 \in P$. We show that those continuous time dynamics of movement in the shadow, are equivalent to those of projected gradient descent (Theorem 9 in Appendix D). Moreover, we also show the following convergence result of those dynamics (proof in Appendix D):

**Theorem 5.** *Let $P \subseteq \mathbb{R}^n$ be a polytope and suppose that $f : P \to \mathbb{R}$ is differentiable and $\mu$-strongly convex over $P$. Consider the shadow dynamics $\dot{X}(t) = \mathbf{d}_{X(t)}^{\Pi}$ with initial conditions $X(0) = \mathbf{x}_0 \in P$. Then for each $t \geq 0$, we have $X(t) \in P$. Moreover, the primal gap $h(X(t)) := f(X(t)) - f(\mathbf{x}^*)$ associated with the shadow dynamics decreases as: $h(X(t)) \leq e^{-2\mu t} h(\mathbf{x}_0)$.*

### 5.2 Shadow-Walk Method

Although the continuous-dynamics of moving along the shadow are the same as those of PGD and achieve linear convergence, it is unclear how to discretize this continuous-time process and obtain a linearly convergent algorithm. To ensure feasibility we may have arbitrarily small step-sizes, and therefore, cannot show sufficient progress in such cases. This is a phenomenon similar to that in the Away-Step and Pairwise CG variants, where the maximum step-size that one can take might not be big enough to show sufficient progress. In [10], the authors overcome this problem by bounding the number of such 'bad' steps using dimension reduction arguments crucially relying on the fact that these algorithms maintain their iterates as a convex combination of vertices. However, unlike away-steps in CG variants, we consider $\mathbf{d}_{\mathbf{x}}^{\Pi}$ as direction for descent, which is independent from the vertices of $P$ and thus eliminating the need to maintain active sets for the iterates of the algorithm. In general, the shadow ODE might revisit a fixed facet a large number times (see Figure 1) with decreasing step-sizes. This problem does not occur when discretizing PGD's continuous time dynamics since we can take *unconstrained* gradient steps and then the projections ensure feasibility.

Inspired by PGD's discretization and the structure of the parametric projections curve, we propose a SHADOW-WALK algorithm (Algorithm 1) with a slight twist: trace the projections curve by walking along the shadow at an iterate $\mathbf{x}_t$ using line search or the in-face condition, until the maximum step size is not selected. To do this, we use the TRACE (Algorithm 3 in Appendix B.5) process to trace the projections curve, which chains consecutive short descent steps until it ensures enough progress as a single PGD step with fixed $1/L$ step size. One important property of TRACE is that it only requires one gradient oracle call. Also, if we know the smoothness constant $L$, then TRACE can be terminated early once we have traced the projections curve until we reach the PGD step. This results in linear convergence, as long as the number of steps by TRACE are bounded polynomially, i.e., the number of "bad" boundary cases. Using fundamental properties of normal cones attained in the projections curve, we are able bound these steps to be at most the number of faces of the polytope (Theorem 8):

**Theorem 6.** *Let $P \subseteq \mathbb{R}^n$ be a polytope and suppose that $f : P \to \mathbb{R}$ is $L$-smooth and $\mu$-strongly convex over $P$. Then the primal gap $h(\mathbf{x}_t) := f(\mathbf{x}_t) - f(\mathbf{x}^*)$ of the* SHADOW WALK *algorithm decreases geometrically: $h(\mathbf{x}_{t+1}) \leq \left(1 - \frac{\mu}{L}\right) h(\mathbf{x}_t)$ with each iteration of the* SHADOW WALK *algorithm (assuming* TRACE *is a single step). Moreover, the number of oracle calls to shadow, in-face*

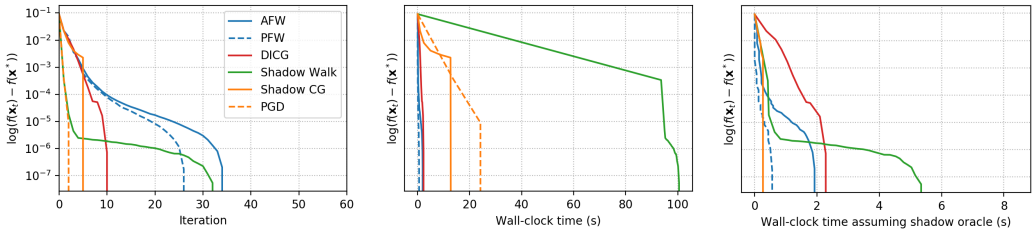


Figure 2: Comparing the performance of away-step FW (AFW) [10], pairwise FW (PFW) [10], decomposition-invariant CG (DICG) [16], SHADOW-WALK (Alg. 1), and SHADOW-CG (Alg. 2). Left plot compares iteration count, middle plot compares wall-clock time (including shadow computation and line search), right plot compares wall-clock time assuming oracle access to shadow. The right plot does not include PGD for a fair comparison.

*direction and line-search oracles to obtain an $\epsilon$-accurate solution is $O\left(\beta \frac{L}{\mu} \log(\frac{1}{\epsilon})\right)$, where $\beta$ is the maximum number of breakpoints of the parametric projections curve that the* TRACE *method visits.*

This result is the key interpolation between PGD and CGD methods, attaining geometric constant independent rates. Comparing this convergence rate with the one in Theorem 5, we see that we pay for discretization of the ODE with the constants $L$ and $\beta$. Although the constant $\beta$ depends on the number of facets $m$ and in fact the combinatorial structure of the face-lattice of the polytope, it is invariant under any deformations of the actual geometry of the polytope preserving the face-lattice (in contrast to vertex-facet distance and pyramidal width); See for example Figure 4's discussion in Appendix D. Although we show $\beta \leq O(2^m)$, we believe that it can be much smaller (i.e., $O(nm)$) for structured polytopes. Moreover, computationally we see much fewer oracles than $O(2^m)$.

## 6 Shadow Conditional Gradient Method

Using our insights on descent directions, we propose the SHADOW-CG algorithm (Algorithm 2), which uses Frank-Wolfe steps earlier in the algorithm, and uses shadow steps more frequently towards the end of the algorithm. Frank-Wolfe steps allow us to greedily skip a lot of facets by wrapping maximally over the polytope (Lemma 4). Shadow steps operate as "optimal" away-steps (Lemma 4) thus reducing zig-zagging phenomenon [10] close to the optimal solution. As the algorithm progresses, one can expect Frank-Wolfe directions to become close to orthogonal to negative gradient. However, in this case the norm of the shadow also starts diminishing. Therefore, we choose FW direction whenever $\left\langle -\nabla f(\mathbf{x}_t), \mathbf{d}_t^{\text{FW}} \right\rangle \geq \left\langle -\nabla f(\mathbf{x}_t), \mathbf{d}_{\mathbf{x}_t}^{\Pi} / \|\mathbf{d}_{\mathbf{x}_t}^{\Pi}\| \right\rangle = \|\mathbf{d}_{\mathbf{x}_t}^{\Pi}\|$, and shadow direction otherwise. This is sufficient to give us linear convergence (proof in Appendix E):

**Theorem 7.** *Let $P \subseteq \mathbb{R}^n$ be a polytope with diameter $D$ and suppose that $f : P \to \mathbb{R}$ is $L$-smooth and $\mu$-strongly convex over $P$. Then, the primal gap $h(\mathbf{x}_t) := f(\mathbf{x}_t) - f(\mathbf{x}^*)$ of* SHADOW-CG *decreases geometrically: $h(\mathbf{x}_{t+1}) \leq \left(1 - \frac{\mu}{LD^2}\right) h(\mathbf{x}_t)$, with each iteration of the* SHADOW-CG *algorithm (assuming* TRACE *is a single step). Moreover, the number of shadow, in-face directions and line oracle calls for an $\epsilon$-accurate solution is $O\left((D^2 + \beta) \frac{L}{\mu} \log(\frac{1}{\epsilon})\right)$, where $\beta$ is the number of breakpoints of the parametric projections curve that the* TRACE *method visits.*

The theoretical bound on iteration complexity for a given fixed accuracy is better for SHADOW-WALK compared to SHADOW-CG. However, the computational complexity for SHADOW-CG is better since FW steps are cheaper to compute compared to the shadow and we can avoid the potentially expensive computation via the TRACE-routine. This is also observed in the experiments next (and Appendix F).

## 7 Computations

We consider the video co-localization problem from computer vision, where the goal is to track an object across different video frames. We used the YouTube-Objects dataset [10] and the problem formulation of Joulin et. al [5]. This consists of minimizing a quadratic function $f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T A\mathbf{x} + \mathbf{b}^T\mathbf{x}$, where $\mathbf{x} \in \mathbb{R}^{660}$, $A \in \mathbb{R}^{660 \times 660}$ and $\mathbf{b} \in \mathbb{R}^{660}$, over a flow polytope, the convex hull of paths in a network. For preliminary computations, we utilize an approximate TRACE procedure that excludes the in-face trace steps (algorithm 7 in Appendix F). We observe that SHADOW-CG has lower iteration count than CG variants (slightly higher than PGD), while also improving on wall-clock time compared to PGD (i.e., close to CG) without assuming any oracle access. Moreover, when assuming access to shadow oracle, SHADOW-CG outperforms the CG variants both in iteration count and wall-clock time. Finally, we observe that the number of iterations spent in TRACE is much smaller (bounded by 10 for SHADOW-WALK and by 4 for Shadow-CG) than the number of faces of the polytope. SHADOW CG spends much fewer iterations in TRACE than SHADOW-WALK due to the addition of FW steps. We refer the reader to Appendix F for additional computational results, with qualitatively similar findings.

## 8 Broader Impact

We believe that this work does not have any foreseeable negative ethical or societal impact.

## 9 Acknowledgements

The research presented in this paper was partially supported by the NSF grant CRII-1850182, the Research Campus MODAL funded by the German Federal Ministry of Education and Research (grant number 05M14ZAM), and the Georgia Institute of Technology ARC TRIAD fellowship. We would also like to thank Damiano Zeffiro for pointing out a missing case in the statement of Theorem 3 in an earlier version of this paper, which is now corrected.

## References

[1] R. Freund, P. Grigas, and R. Mazumder, "An extended Frank–Wolfe method with "in-face" directions, and its application to low-rank matrix completion," *SIAM Journal on Optimization*, vol. 27, no. 1, p. 319–346, 2015.

[2] M. Jaggi, "Revisiting Frank-Wolfe: Projection-free sparse convex optimization," in *Proceedings of the 30th international conference on machine learning*, 2013, pp. 427–435.

[3] M. A. Bashiri and X. Zhang, "Decomposition-invariant conditional gradient for general polytopes with line search," in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, 2017, p. 2687–2697.

[4] R. Lyons and Y. Peres, *Probability on trees and networks*. Cambridge University Press, New York, 2005.

[5] A. Joulin, K. D. Tang, and F. Li, "Efficient image and video co-localization with frank-wolfe algorithm," in *Computer Vision - ECCV 2014 - 13th European Conference*, 2014, pp. 253–268.

[6] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin, *Network Flows: Theory, Algorithms, and Applications*. Prentice-Hall, Inc., 1993.

[7] S. Fujishige and S. Isotani, "A submodular function minimization algorithm based on the minimum-norm base," *Pacific Journal of Optimization*, vol. 7, 2009.

[8] A. S. Nemirovski and D. B. Yudin, "Problem complexity and method efficiency in optimization," *Wiley-Interscience, New York*, 1983.

[9] M. Frank and P. Wolfe, "An algorithm for quadratic programming," *Naval Research Logistics Quarterly*, vol. 3, no. 1-2, pp. 95–110, 1956.

[10] S. Lacoste-Julien and M. Jaggi, "On the global linear convergence of Frank-Wolfe optimization variants," in *Advances in Neural Information Processing Systems (NIPS)*, 2015, pp. 496–504.

[11] J. GuéLat and P. Marcotte, "Some comments on wolfe's 'away step'," *Mathematical Programming*, vol. 35, pp. 110–119, 1986.

[12] E. Levitin and B. Polyak, "Constrained minimization methods," *USSR Computational Mathematics and Mathematical Physics*, vol. 6, p. 1–50, 1966.

[13] M. D. Canon and C. Cullum, "A tight upper bound on the rate of convergence of Frank-Wolfe algorithm," *SIAM Journal on Control*, vol. 6, no. 4, p. 509–516, 1968.

[14] G. Lan, "The complexity of large-scale convex programming under a linear optimization oracle," *arXiv preprint arXiv:1512.06142*, 2013.

[15] D. Garber and E. Hazan, "A linearly convergent variant of the conditional gradient algorithm under strong convexity, with applications to online and stochastic optimization," *SIAM Journal on Optimization*, vol. 26, no. 3, p. 1493–1528, 2016.

[16] D. Garber and O. Meshi, "Linear-memory and decomposition-invariant linearly convergent conditional gradient algorithm for structured polytopes," in *Proceedings of the 30th International Conference on Neural Information Processing Systems*, 2016, p. 1009–1017.

[17] G. Lan and Y. Zhou, "Conditional gradient sliding for convex optimization," *SIAM Journal on Optimization*, vol. 26, no. 2, pp. 1379—-1409, 2016.

[18] G. Braun, S. Pokutta, D. Tu, and S. Wright, "Blended conditional gradients: the unconditioning of conditional gradients," *arXiv preprint arXiv:1805.07311*, 2018.

[19] A. Beck and S. Shtern, "Linearly convergent away-step conditional gradient for non-strongly convex functions," *Mathematical Programming*, vol. 164, pp. 1–27, 2017.

[20] J. Penã and D. Rodríguez, "Polytope conditioning and linear convergence of the frank-wolfe algorithm," *arXiv preprint arXiv:1512.06142*, 2015.

[21] F. Rinaldi and D. Zeffiro, "A unifying framework for the analysis of projection-free first-order methods under a sufficient slope condition," *arXiv preprint arXiv:2008.09781*, 2020.

[22] ——, "Avoiding bad steps in frank wolfe variants," *arXiv preprint arXiv:2012.12737*, 2020.

[23] D. P. Bertsekas, *Nonlinear programming*. Athena Scientific, 1997.

[24] G. P. McCormick and R. A. Tapia, "The gradient projection method under mild differentiability conditions," *SIAM Journal on Control*, vol. 10, no. 1, pp. 93–98, 1972.

[25] C. D. Meyer, *Matrix analysis and applied linear algebra*. Siam, 2000, vol. 71.

[26] J. C. Dunn, "Rates of convergence for conditional gradient algorithms near singular and nonsingular extremals," *SIAM Journal on Control and Optimization*, vol. 17, no. 2, pp. 187–211, 1979.

[27] 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.

[28] C. W. Combettes and S. Pokutta, "Boosting frank-wolfe by chasing gradients," *arXiv preprint arXiv:2003.06369*, 2020.

[29] D. Bertsekas, A. Nedic, and O. AE, *Convex Analysis and Optimization*. Athena Scientific, 2003.

[30] W. Krichene, A. Bayen, and P. L. Bartlett, "Accelerated mirror descent in continuous and discrete time," in *Advances in Neural Information Processing Systems 28*, 2015, pp. 2845–2853.

[31] R. T. Rockafellar, *Convex analysis*. Princeton University Press, 1970.

[32] T. H. Gronwall, "Note on the derivatives with respect to a parameter of the solutions of a system of differential equations," *Annals of Mathematics*, pp. 292–296, 1919.

[33] G. Söderlind, *Numerical Methods for Differential Equations*. Springer, 2017.

[34] H. Karimi, J. Nutini, and M. Schmidt, "Linear convergence of gradient and proximal-gradient methods under the polyak-lojasiewicz condition," in *European Conference on Machine Learning and Knowledge Discovery in Databases - Volume 9851*, ser. ECML PKDD 2016. Springer-Verlag, 2016, p. 795–811.

[35] G. Optimization, "Gurobi optimizer reference manual version 7.5," 2017, uRL: https://www.gurobi.com/documentation/7.5/refman.

# A Related Work

| Paper | Algorithm | Steps to get $\epsilon$-error |
|---|---|---|
| Dunn (1979) [26] | Geometric analysis for vanilla CG. | $O(LD^2/\epsilon)$ |
| Guélat and Marcotte (1986) [11] | Vanilla FW with $\mathbf{x}^*$ having distance $\Delta > 0$ from the boundary. | $O\left(\kappa \left(\frac{D}{\Delta}\right)^2 \log \frac{1}{\epsilon}\right)$ |
| Jaggi [2] (2013) | Vanilla FW with the uniform step-size rule $\gamma_t = \frac{2}{t+2}$. | $O\left(\frac{LD^2}{\epsilon}\right)$ |
| Lan [14] (2013) | Constraining FW vertex to a ball around the current iterate. | $O\left(\kappa \log \frac{D\mu}{\epsilon}\right)$ |
| Freund et. al (2015) [27] | FW with in-face directions (promoting sparsity) as a generalization to away-steps. | $O\left(\frac{LD^2}{\epsilon}\right)$ |
| Lacoste-Julien and Jaggi (2015) [10] | FW & Away-steps (over current active set) for general polytopes (AFW & PFW). | $O\left(\kappa \left(\frac{D}{\delta}\right)^2 \log \frac{1}{\epsilon}\right)$ |
| Garber and Hazan (2016) [15] | Constraining FW vertex to a ball around the current iterate with a focus on polytopes. | $O\left(\kappa n \rho \log \frac{1}{\epsilon}\right)$ |
| Garber and Meshi (2016) [16] | Pairwise steps for structured 0/1 polytopes[6] using best away vertex in minimal face of iterate (DICG). | $O\left(\kappa \|\mathbf{x}^*\|_0 D^2 \log \frac{1}{\epsilon}\right)$ |
| Beck and Shtern (2017) [3] | FW & Away steps using best away vertex in current active set for specific non-strongly convex objective functions[7]. | $O\left(Ln \left(\frac{D}{\Phi}\right)^2 \log \frac{1}{\epsilon}\right)$ |
| Bashiri and Zhang (2017) [3] | FW & Away steps using best away vertex in current active set. | $O\left(n\kappa D^2 H_s \log \frac{1}{\epsilon}\right)$ |
| Braun et. al (2018) [18] | Lazy FW & gradient descent steps over simplex formed by current active set. | $O\left(K\kappa \left(\frac{D}{\delta}\right)^2 \log \frac{1}{\epsilon}\right)$ |
| Combettes and Pokutta (2020) [28] | FW with descent directions better aligned with the negative gradients. | $O\left(\frac{\kappa}{\alpha^2 \omega} \log \frac{1}{\epsilon}\right)$ |
| **This paper** | Moving along the 'shadow' of gradient (SHADOW-WALK). | $O\left(\kappa \beta \log \frac{1}{\epsilon}\right)$ |
| **This paper** | Moving in the 'shadow' of gradient with FW steps. (SHADOW-CG) | $O\left(\kappa (D^2 + \beta) \log \frac{1}{\epsilon}\right)$ |

Table 1: Summary of different descent techniques used in CG variants and their linear convergence rates. The factor $\kappa := L/\mu$ is the condition number of the function and $D$ is the diameter of the domain. Also, $\delta$ is the pyramidal width, $\rho$ and $\Phi$ are notions of vertex-facet distance and $H_s$ is a sparsity-dependent geometric constant. Moreover, $K$ is a parameter for finding approximate FW vertices. The constants $\alpha$ and $\omega$ arise from the gradient alignment procedure. Finally, $\beta$ is the number of breakpoints when walking along the shadow of a direction (within TRACE), which is a function of the number of facets of the polytope.

**Other Related Work:** In 1966, Levitin and Polyak [12] showed that the conditional gradient method can obtain linear convergence for strongly-convex domains when the gradient at any point in the domain is lower-bounded by a constant. In order to emulate strongly convex set domains, Lan [14] showed that constraining the Frank-Wolfe vertex to a ball (instead of entire polytope) around the current iterate is sufficient for linear convergence. In 2016, Garber and Hazan [15] generalized their result to polytopes and showed that this 'constrained' Frank-Wolfe vertex could be computed by a single linear optimization (i.e. without additional computational complexity compared to vanilla

[6]These include: the path polytope of a graph (aka the unit flow polytope), the perfect matching polytope of a bipartite graph, and the base polyhedron of a matroid, for which we have highly efficient combinatorial algorithms for linear optimization.

[7]They consider objective functions of the form $f(\mathbf{x}) := g(\mathbf{E}\mathbf{x}) + \langle \mathbf{b}, \mathbf{x} \rangle$, where $g$ is a strongly convex function and $E$ is a matrix. Note that for a general matrix $\mathbf{E}$, the function $f$ is not necessarily strongly convex.

CG), and accordingly prove the first global linear convergence result for CG variants. These results essentially translate the regularization in mirror-descent variants as a norm-ball in CG variants. The idea is that this restriction obtains a good approximation to the gradient descent direction, not scaled by the length of the FW vector $\mathbf{v}_t - \mathbf{x}_t$. There has also been extensive work on mixing FW and gradient descent steps with the aim of better computational performance while enjoying linear convergence. For instance, in 2014, Lan and Zhou solve projection subproblems approximately by invoking an internal CG subroutine [17]. In 2018, Braun et. al [18] show linear convergence for a CG variant when projected gradient descent steps are used to solve convex subproblems over carefully maintained active sets. Combettes and Pokutta [28] recently explored employing a FW subroutine to compute an approximate shadow direction, and then they consider that approximate shadow as their descent direction. Although they show theoretically that in the worst-case their algorithm has global sublinear convergence due to the 'bad' steps where a maximal step size is chosen and cannot show sufficient progress in this case, they prove linear convergence for 'good' steps and demonstrated significant speed-ups computationally. Following our work, there have been recent results on extensions using TRACE-like procedures to avoid "bad" steps in CG variants and accordingly obtain linear convergence rates that depend on a slope condition rather than geometric constants [21, 22]. Our goal in this work is to put these CG variants in perspective and understand desired properties of feasible directions of descent.

## B Missing Proofs and Results for Section 3

Before, we delve deeper into the analysis of results presented in Section 3, we first give an explanation of the structure of the parametric projections curve through the following figure:

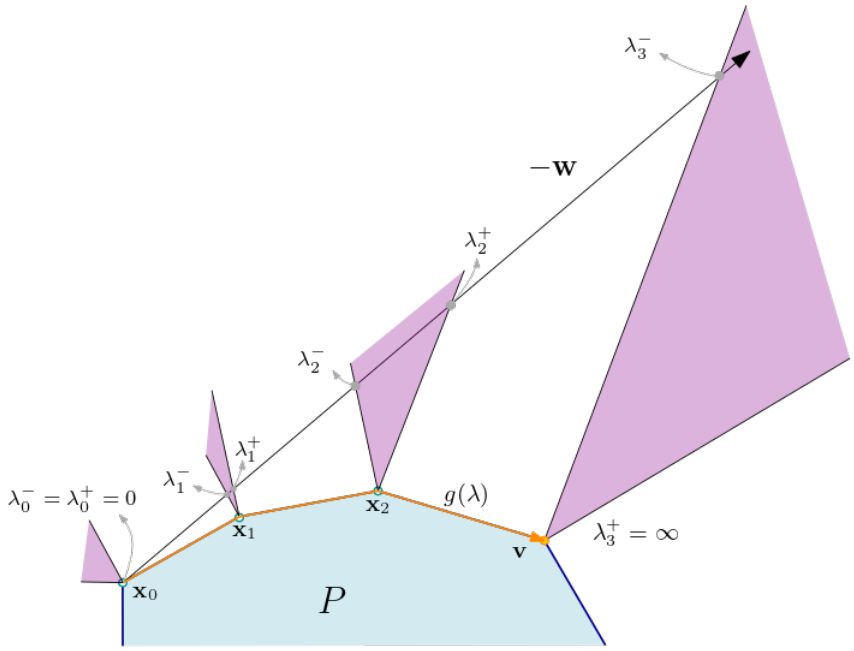


Figure 3: Figure showing the structure of the parametric projections curve $g(\mathbf{x}_0 - \lambda\mathbf{w})$ for $\lambda \geq 0$, which is depicted by the orange line. Breakpoints in the curve correspond to $\mathbf{x}_1, \mathbf{x}_2$ and $\mathbf{x}_3 = \mathbf{v}$ with $g(\lambda_i^-) = g(\lambda_i^+) = \mathbf{x}_i$, and $\lambda_3^+ = \infty$ since $\lim_{\lambda\to\infty} g(\lambda) = \mathbf{v} = \arg\min_{\mathbf{y}\in P} \langle \mathbf{y}, \mathbf{w}\rangle$.

In the above figure, the curve $g(\mathbf{x}_0 - \lambda\mathbf{w})$ is depicted by the orange line and is piecewise linear. First, $g(\lambda) = \mathbf{x}_0 + \lambda \mathbf{d}_{\mathbf{x}}^{\Pi}(\mathbf{w})$ for $\lambda \in [0, \lambda_1^-]$, where $\mathbf{d}_{\mathbf{x}}^{\Pi}(\mathbf{w}) = \frac{\mathbf{x}_1 - \mathbf{x}_0}{\lambda_1^-}$ is the directional-derivative with respect to $-\mathbf{w}$. At that point, we see that $\mathbf{x}_0 - \lambda\mathbf{w} - \mathbf{x}_1 \in N_P(\mathbf{x}_1)$ for all $\lambda \in (\lambda_1^-, \lambda_1^+]$. Hence, $g(\lambda) = \mathbf{x}_1$ for all $\lambda \in (\lambda_1^-, \lambda_1^+]$, i.e. we will keep projecting back to the same point $\mathbf{x}_1$ in that interval. Thus, $g(\lambda)$ does not change at the same speed with respect to $\lambda$. Moreover, we have $N_P(g(\lambda)) = N_P(g(\lambda')) \subset N_P(\mathbf{x}_0)$ for all $\lambda, \lambda' \in (0, \lambda_1^-)$. Then, another constraint becomes tight

at the end point of the first segment $\mathbf{x}_1$, and thus we have $N_P(g(\lambda)) = N_P(g(\lambda'))\subset N_P(\mathbf{x}_1)$ for all $\lambda, \lambda' \in (0, \lambda_1^-)$.

In other words, in the notation of Theorem 3, since $\hat{\lambda}_1 = \max\{\lambda \mid N_P(g(\lambda')) = N_P(\mathbf{x}_1)\ \forall \lambda' \in [\lambda_1^-, \lambda),\ N_P(\mathbf{x}_1) \subseteq N_P(g(\lambda))\} > \lambda_1^-$ and $\hat{\mathbf{d}}^{\Pi}_{\mathbf{x}_1}(\mathbf{w}) = 0$, it follows that $\lambda_1^+ = \hat{\lambda}_1$ and $g(\lambda) = \mathbf{x}_1$ for all $\lambda \in (\lambda_1^-, \lambda_1^+]$, i.e. we will keep projecting back to the same point $\mathbf{x}_1$ in that interval. Now, after that point, we have $\hat{\lambda}_1 = \lambda_1^+$ and again using Theorem 3, know that the next breakpoint could be computed using a line search and shadow computation. In particular, $\lambda_2^- = \lambda_{i-1}^+ + \max\{\delta : \mathbf{x}_{i-1} + \delta \mathbf{d}^{\Pi}_{\mathbf{x}_{i-1}}(\nabla f(\mathbf{x}_0)) \in P\}$ and $\mathbf{x}_2 = \mathbf{x}_1 + (\lambda_2^- - \lambda_1^+)\mathbf{d}^{\Pi}_{\mathbf{x}_1}(\nabla f(\mathbf{x}_0))$. Furthermore, this process of adding and dropping constraints continues until we reach $\lambda_3^-$. We show that once the parametric projections curve (given by the orange line in the figure) leaves a face, it never returns to it again (Theorem 8). At this point, $\mathbf{x}_0 - \lambda\mathbf{w} - \mathbf{v} \in N_P(\mathbf{v})$ for $\lambda \geq \lambda_3^-$, i.e $g(\lambda) = \mathbf{v}$ and we will keep projecting back to $\mathbf{v}$. This is consistent with the characterization of the end point of $g(\lambda)$ as the FW vertex: $-\mathbf{w} \in N_P(\mathbf{v})$ if and only if $\mathbf{v} = \arg\min_{\mathbf{x}\in P} \langle \mathbf{w}, \mathbf{x}\rangle$.

Even though the results in this section hold for any direction $\mathbf{w} \in \mathbb{R}^n$, we will prove the statements for $\mathbf{w} = \nabla f(\mathbf{x}_0)$ for readability in the context of the paper.

### B.1 Proof of Lemma 1

**Lemma 1.** *If $-\nabla f(\mathbf{x}_0) \in N_P(\mathbf{x}_0)$, then $g(\lambda) = \Pi_P(\mathbf{x}_0 - \lambda\nabla f(\mathbf{x}_0)) = \mathbf{x}_0$ for all $\lambda \in \mathbb{R}_+$.*

*Proof.* Note that by definition $g(\lambda) = \arg\min_{\mathbf{y}\in P}\left\{\frac{\|\mathbf{x}_0-\mathbf{y}\|^2}{2\lambda} + \langle \nabla f(\mathbf{x}_0), \mathbf{y}\rangle\right\}$ for any $\lambda > 0$. Then, by optimality of $g(\lambda)$ we have

$$\frac{\|\mathbf{x}_0 - g(\lambda)\|^2}{2\lambda} + \langle \nabla f(\mathbf{x}_0), g(\lambda)\rangle \leq \frac{\|\mathbf{x}_0 - \mathbf{z}\|^2}{2\lambda} + \langle \nabla f(\mathbf{x}_0), \mathbf{z}\rangle \tag{10}$$

for all $\mathbf{z} \in P$.

The condition $-\nabla f(\mathbf{x}_0) \in N_P(\mathbf{x}_0)$ is equivalent to $\langle \nabla f(\mathbf{x}_0), \mathbf{z} - \mathbf{x}_0\rangle \geq 0$ for all $\mathbf{z} \in P$. Plugging $\mathbf{x}_0$ for $\mathbf{z}$ on the right-hand side of (10), we have that for any $\lambda > 0$

$$0 \leq \frac{\|\mathbf{x}_0 - g(\lambda)\|^2}{2\lambda} \leq \langle \nabla f(\mathbf{x}_0), \mathbf{x}_0 - g(\lambda)\rangle \leq 0.$$

This implies $g(\lambda) = \mathbf{x}_0$ for all $\lambda > 0$. □

### B.2 Proof of Lemma 2

**Lemma 2** (Linearity of projections)**.** *Let $P \subseteq \mathbb{R}^n$ be a polytope defined using $m$ facet inequalities (e.g., as in (4)). Let $\mathbf{x}_0 \in P$ and we are given $\nabla f(\mathbf{x}_0) \in \mathbb{R}^n$. Let $g(\lambda) = \Pi_P(x_0 - \lambda\nabla f(\mathbf{x}_0))$ be the parametric projections curve. Then, if $N_P(g(\lambda)) = N_P(g(\lambda'))$ for some $\lambda < \lambda'$, then the curve between $g(\lambda)$ and $g(\lambda')$ is linear, i.e., $g(\delta\lambda + (1-\delta)\lambda') = \delta g(\lambda) + (1-\delta)g(\lambda')$, where $\delta \in [0, 1]$.*

*Proof.* Recall from Section 2 that $\mathbf{y}^* = \Pi_P(\mathbf{x}_0 - \lambda\nabla f(\mathbf{x}_0))$ if and only if $(\mathbf{x}_0 - \lambda\nabla f(\mathbf{x}_0) - \mathbf{y}^*) \in N_P(\mathbf{y}^*)$. Thus, the optimality of $g(\lambda)$ implies

$$\mathbf{x}_0 - \lambda\nabla f(\mathbf{x}_0) - g(\lambda) \in N_P(g(\lambda)). \tag{11}$$

Similarly, using the optimality of $g(\lambda')$ we have

$$\mathbf{x}_0 - \lambda'\nabla f(\mathbf{x}_0) - g(\lambda') \in N_P(g(\lambda')). \tag{12}$$

Aggregate equations (11) and (12) with weights $\delta$ and $(1-\delta)$ respectively to obtain:

$$\mathbf{x}_0 - (\delta\lambda + (1-\delta)\lambda')\nabla f(\mathbf{x}_0) - (\delta g(\lambda) + (1-\delta)g(\lambda')) \in \delta N_P(g(\lambda)) + (1-\delta)N_P(g(\lambda')). \tag{13}$$

Now we claim that

$$\delta N_P(g(\lambda)) + (1-\delta)N_P(g(\lambda')) = N_P(g(\lambda')) = N_P(g(\lambda)) = N_P(\delta g(\lambda) + (1-\delta)g(\lambda')). \tag{14}$$

The first two equalities follow from that fact that $\delta \in [0,1]$ and $N_P(g(\lambda)) = N_P(g(\lambda'))$. To show the third equality, note from (5) that $N_P(g(\lambda)) = N_P(g(\lambda'))$ implies that $I(g(\lambda)) = I(g(\lambda'))$ and hence $J(g(\lambda)) = J(g(\lambda'))$. Therefore,

$$\begin{aligned}\mathbf{A}_{I(g(\lambda))}(\delta g(\lambda) + (1-\delta)g(\lambda')) &= \delta \mathbf{A}_{I(g(\lambda))} g(\lambda) + (1-\delta)\mathbf{A}_{I(g(\lambda'))} g(\lambda') \\ &= \delta \mathbf{b}_{I(g(\lambda))} + (1-\delta)\mathbf{b}_{I(g(\lambda'))} \\ &= \mathbf{b}_{I(g(\lambda))}.\end{aligned}$$

Similarly,

$$\begin{aligned}\mathbf{A}_{J(g(\lambda))}(\delta g(\lambda) + (1-\delta)g(\lambda')) &= \delta \mathbf{A}_{J(g(\lambda))} g(\lambda) + (1-\delta)\mathbf{A}_{J(g(\lambda'))} g(\lambda') \\ &< \delta \mathbf{b}_{J(g(\lambda))} + (1-\delta)\mathbf{b}_{J(g(\lambda'))} \\ &= \mathbf{b}_{J(g(\lambda))}.\end{aligned}$$

Thus, we have shown that $I(g(\lambda)) = I(\delta g(\lambda) + (1-\delta)g(\lambda')) = I(g(\lambda'))$ and $J(g(\lambda)) = J(\delta g(\lambda) + (1-\delta)g(\lambda')) = J(g(\lambda'))$, which completes the proof of (14).

Now using (14), we can equivalently write (13) as follows:

$$\mathbf{x}_0 - (\delta\lambda + (1-\delta)\lambda')\nabla f(\mathbf{x}_0) - (\delta g(\lambda) + (1-\delta)g(\lambda')) \in N_P(\delta g(\lambda) + (1-\delta)g(\lambda')).$$

This shows that $\delta g(\lambda) + (1-\delta)g(\lambda')$ satisfies the optimality condition for $g(\delta\lambda + (1-\delta)\lambda')$, which concludes the proof. □

### B.3 Proof of Theorem 2

We prove this theorem in a sequence of steps given by the next couple of lemmas. We first show that the normal cones do not change in the *strict* neighborhood of $\mathbf{x}_0$, i.e., there exists a ball $B(\mathbf{x}_0, \delta)$ around $\mathbf{x}_0$ of radius $\delta > 0$ such that the normal cone $N_P(g(\lambda)) = N_P(g(\lambda'))$ for all $g(\lambda), g(\lambda') \in B(\mathbf{x}_0, \delta) \setminus \{\mathbf{x}_0\}$.

**Lemma 5.** *Let $P \subseteq \mathbb{R}^n$ be a polytope defined using $m$ facet inequalities (e.g., as in (4)). Let $\mathbf{x}_0 \in P$ and we are given $\nabla f(\mathbf{x}_0) \in \mathbb{R}^n$. Let $g(\lambda) = \Pi_P(\mathbf{x}_0 - \lambda \nabla f(\mathbf{x}_0))$ be the parametric projections curve. Then there exists a scalar $\delta > 0$ such that $N_P(g(\lambda)) = N_P(g(\lambda')) \subseteq N_P(\mathbf{x}_0)$, for all $0 < \lambda < \lambda' < \delta$.*

To prove this lemma we will need a few properties about orthogonal projections. The first property is a simple fact, which follows from the fact that the Euclidean projection operator is non-expansive (see Section 2):

$$\|g(\lambda) - g(\lambda+\epsilon)\| \leq \|(\mathbf{x}_0 - \lambda\nabla f(\mathbf{x}_0)) - (\mathbf{x}_0 - (\lambda+\epsilon)\nabla f(\mathbf{x}_0))\| = |\epsilon|\|\nabla f(\mathbf{x}_0)\|. \tag{15}$$

The second property we need is that if for some $\lambda' > \lambda$ the point $\mathbf{z} := \left(1 - \frac{\lambda'}{\lambda}\right)\mathbf{x}_0 + \frac{\lambda'}{\lambda} g(\lambda)$ in the affine hull of $g(\lambda)$ and $\mathbf{x}_0$ is feasible, then it indeed coincides with the projection $g(\lambda')$:

**Lemma 6** (Affine hull expansion of projections)**.** *Let $\mathcal{X} \subseteq \mathbb{R}^n$ be a compact and convex set. Let $\mathbf{x}_0 \in \mathcal{X}$ and we are given $\nabla f(\mathbf{x}_0) \in \mathbb{R}^n$. Further, let $g(\lambda) = \Pi_{\mathcal{X}}(\mathbf{x}_0 - \lambda \nabla f(\mathbf{x}_0))$ be the parametric projections curve. Then, if $\mathbf{z} := \left(1 - \frac{\lambda'}{\lambda}\right)\mathbf{x}_0 + \frac{\lambda'}{\lambda} g(\lambda) \in \mathcal{X}$ for some $\lambda < \lambda'$, then $g(\lambda') = \mathbf{z}$.*

*Proof.* We will show that $\mathbf{z} := \left(1 - \frac{\lambda'}{\lambda}\right)\mathbf{x}_0 + \frac{\lambda'}{\lambda} g(\lambda) = \mathbf{x}_0 + \frac{\lambda'}{\lambda}(g(\lambda) - \mathbf{x}_0) \in \mathcal{X}$ satisfies first-order optimality for the projection at $\lambda'$. Suppose for a contradiction, there exists some $\mathbf{y} \in \mathcal{X}$ with

$$\langle \mathbf{x}_0 - \lambda'\nabla f(\mathbf{x}_0) - \mathbf{z}, \mathbf{y} - \mathbf{z}\rangle > 0 \tag{16}$$

$$= \left\langle \mathbf{x}_0 - \lambda'\nabla f(\mathbf{x}_0) - \left(\mathbf{x}_0 + \frac{\lambda'}{\lambda}(g(\lambda) - \mathbf{x}_0)\right), \mathbf{y} - \left(\mathbf{x}_0 + \frac{\lambda'}{\lambda}(g(\lambda) - \mathbf{x}_0)\right)\right\rangle > 0 \tag{17}$$

$$\Leftrightarrow \frac{\lambda'}{\lambda}\left\langle \mathbf{x}_0 - \lambda\nabla f(\mathbf{x}_0) - g(\lambda), \mathbf{y} + \left(\frac{\lambda'}{\lambda} - 1\right)\mathbf{x}_0 - \frac{\lambda'}{\lambda} g(\lambda)\right\rangle > 0 \tag{18}$$

$$\Leftrightarrow \left(\frac{\lambda'}{\lambda}\right)^2 \left\langle \mathbf{x}_0 - \lambda\nabla f(\mathbf{x}_0) - g(\lambda), \frac{\lambda}{\lambda'}\mathbf{y} + \left(1 - \frac{\lambda}{\lambda'}\right)\mathbf{x}_0 - g(\lambda)\right\rangle > 0. \tag{19}$$

Observe that $\frac{\lambda}{\lambda'}\mathbf{y} + \left(1 - \frac{\lambda}{\lambda'}\right)\mathbf{x}_0 \in \mathcal{X}$ since $\frac{\lambda}{\lambda'} \in (0,1)$ and $\mathcal{X}$ is a convex set. This contradicts the first-order optimality condition of $g(\lambda)$:

$$\langle \mathbf{x}_0 - \lambda \nabla f(\mathbf{x}_0) - g(\lambda), \mathbf{x} - g(\lambda) \rangle \leq 0 \quad \forall \mathbf{x} \in \mathcal{X}.$$

□

We are now ready to prove the lemma:

*Proof of Lemma 5.* Let $I$ and $J$ denote the index-set of active and inactive constraints at $\mathbf{x}_0$ respectively. We will prove that any $\delta$ satisfying

$$0 < \delta \leq \min_{i \in J} \frac{b_i - \langle \mathbf{a}_i, \mathbf{x}_0 \rangle}{\|\mathbf{a}_i\| \|\nabla f(\mathbf{x}_0)\|}, \tag{20}$$

satisfies the condition stated in the lemma ($J$ is non-empty since otherwise the polytope contains only one point and the lemma follows trivially).

We first show that $N_P(g(\lambda)) \subseteq N_P(\mathbf{x}_0)$ for any $\lambda \in (0, \delta)$. Indeed, for any $j \in J$ (so that $\langle \mathbf{a}_j, \mathbf{x}_0 \rangle < b_j$), we have

$$\begin{aligned}
\langle \mathbf{a}_j, g(\lambda) \rangle &= \langle \mathbf{a}_j, \mathbf{x}_0 + g(\lambda) - \mathbf{x}_0 \rangle \\
&= \langle \mathbf{a}_j, \mathbf{x}_0 + g(\lambda) - g(0) \rangle \\
&\leq \langle \mathbf{a}_j, \mathbf{x}_0 \rangle + \|\mathbf{a}_j\| \|g(\lambda) - \mathbf{x}_0\| && \text{by Cauchy-Schwartz inequality} && (21) \\
&\leq \langle \mathbf{a}_j, \mathbf{x}_0 \rangle + \lambda \|\mathbf{a}_j\| \|\nabla f(\mathbf{x}_0)\| && \text{by non-expansivity of projections (15)} && (22) \\
&< \langle \mathbf{a}_j, \mathbf{x}_0 \rangle + \delta \|\mathbf{a}_j\| \|\nabla f(\mathbf{x}_0)\| && \text{since } \lambda \in (0, \delta) && (23) \\
&\leq b_j && \text{choice of } \delta \text{ in (20).} && (24)
\end{aligned}$$

This shows that the inactive constraints at $\mathbf{x}_0$ remain inactive at $g(\lambda)$ for any $0 < \lambda < \delta$, i.e., $N_P(g(\lambda)) \subseteq N_P(\mathbf{x}_0)$. What remains to show is that active constraints at $g(\lambda)$ are the same as active constraints at $g(\lambda')$, i.e., for any $i \in I$, we have $\langle \mathbf{a}_i, g(\lambda) \rangle = b_i$ if and only if $\langle \mathbf{a}_i, g(\lambda') \rangle = b_i$ for $0 < \lambda < \lambda' < \delta$. To show that, we only need to show colinearity of $g(\lambda), g(\lambda'), \mathbf{x}_0$, i.e.,

$$g(\lambda') = \mathbf{x}_0 + \frac{\lambda'}{\lambda}(g(\lambda) - \mathbf{x}_0) := \mathbf{z}, \tag{25}$$

since this implies

$$\langle \mathbf{a}_i, \mathbf{x}_0 - g(\lambda) \rangle = 0 \iff \langle \mathbf{a}_i, \mathbf{x}_0 - g(\lambda') \rangle = 0 \iff \langle \mathbf{a}_i, g(\lambda') \rangle = b_i \iff \langle \mathbf{a}_i, g(\lambda) \rangle = b_i.$$

Let us now prove colinearity of $g(\lambda), g(\lambda'), \mathbf{x}_0$ (25). Using Lemma 6, we know that $g(\lambda') = \mathbf{z}$ as long as $\mathbf{z} \in P$. Hence, it suffices to show feasibility of $\mathbf{z}$:

> **Feasibility**: We claim $\mathbf{x}_0 + \frac{\lambda'}{\lambda}(g(\lambda) - \mathbf{x}_0) \in P$.
>
> *Proof.* Any inactive constraint $j \in J$ remains inactive, since:
>
> $$\left\langle \mathbf{a}_j, \mathbf{x}_0 + \frac{\lambda'}{\lambda}(g(\lambda) - \mathbf{x}_0) \right\rangle \leq \langle \mathbf{a}_j, \mathbf{x}_0 \rangle + \frac{\lambda'}{\lambda} \|\mathbf{a}_j\| \|g(\lambda) - \mathbf{x}_0\| \leq \langle \mathbf{a}_j, \mathbf{x}_0 \rangle + \lambda' \|\mathbf{a}_j\| \|\nabla f(\mathbf{x}_0)\| < b_j$$
>
> where the last inequality uses $\lambda' < \delta$. Each active constraint $i \in I$ also remains feasible, since
>
> $$\langle \mathbf{a}_i, \mathbf{x}_0 + (g(\lambda) - \mathbf{x}_0) \rangle = \langle \mathbf{a}_i, g(\lambda) \rangle \leq b_i \implies \langle \mathbf{a}_i, g(\lambda) - \mathbf{x}_0 \rangle \leq 0.$$
>
> Multiplying the last inequality with $\frac{\lambda'}{\lambda} > 0$, and adding $\langle \mathbf{a}_i, \mathbf{x}_0 \rangle = b_i$, we get:
>
> $$\left\langle \mathbf{a}_i, \mathbf{x}_0 + \frac{\lambda'}{\lambda}(g(\lambda) - \mathbf{x}_0) \right\rangle \leq b_i.$$

The result now follows from colinearity of $g(\lambda), g(\lambda')$ and $\mathbf{x}_0$. □

Since $\lambda, \lambda' \in (0, \delta)$ were arbitrary in the above proof, we get the following corollary:

**Corollary 1.** *Let $P \subseteq \mathbb{R}^n$ be a polytope defined using $m$ facet inequalities (e.g., as in (4)). Let $\mathbf{x}_0 \in P$ and we are given $\nabla f(\mathbf{x}_0) \in \mathbb{R}^n$. Let $g(\lambda) = \Pi_P(\mathbf{x}_0 - \lambda \nabla f(\mathbf{x}_0))$ be the parametric projections curve. Then, there exists a scalar $\delta > 0$ such that*

$$g(\lambda) = \Pi_P(\mathbf{x}_0 - \lambda \nabla f(\mathbf{x}_0)) = \mathbf{x}_0 + \lambda \mathbf{d}^{\Pi}_{\mathbf{x}_0} \quad \textit{for all } \lambda \in [0, \delta).$$

*Proof.* We have shown in Lemma 5 there exists a scalar $\delta > 0$ such that $g(\lambda)$, $g(\lambda')$ and $\mathbf{x}_0$ are co-linear for all $0 < \lambda < \lambda' < \delta$. In other words, this is equivalent to saying that $g(\lambda) = \mathbf{x}_0 + \lambda \mathbf{d}$ for all $\lambda \in [0, \delta)$, where $\mathbf{d} = \frac{g(\lambda') - \mathbf{x}_0}{\lambda'} \in \mathbb{R}^n$ for an arbitrary $\lambda' \in (0, \delta)$ (see (25)). Now, the result follows by definition of the directional derivative:

$$\mathbf{d}^{\Pi}_{\mathbf{x}_0} = \lim_{\epsilon \downarrow 0} \frac{\Pi_P(\mathbf{x}_0 - \epsilon \nabla f(\mathbf{x}_0)) - \mathbf{x}_0}{\epsilon} = \lim_{\epsilon \downarrow 0} \frac{g(\epsilon) - \mathbf{x}_0}{\epsilon} = \frac{g(\lambda') - \mathbf{x}_0}{\lambda'} = \mathbf{d}. \tag{26}$$

□

We will also need the following lemma to prove Theorem 2. So far, we have shown that equal normal cones at $g(\lambda)$ and $g(\lambda')$ imply linear curve between these (Lemma 2). There exists some $\delta > 0$ such that normal cones up to $g(\lambda)$ ($\lambda < \delta$) do not change around $\mathbf{x}_0$ (Lemma 5), and projections form a line from $\mathbf{x}_0$ to $g(\delta)$ (Corollary 1). We next show the converse: if projections do form a line emanating from $\mathbf{x}_0$ up to some $g(\theta)$, then the normal cones up to $g(\lambda)$ ($\lambda < \theta$) must also be the same. (This means that $\theta \geq \delta$.)

**Lemma 7.** *Let $P \subseteq \mathbb{R}^n$ be a polytope defined using $m$ facet inequalities (e.g., as in (4)). Let $\mathbf{x}_0 \in P$ and we are given $\nabla f(\mathbf{x}_0) \in \mathbb{R}^n$. Let $g(\lambda) = \Pi_P(\mathbf{x}_0 - \lambda \nabla f(\mathbf{x}_0))$ be the parametric projections curve. Suppose the projections curve is linear from $\mathbf{x}_0$ up to $g(\delta)$ for some $\delta \geq 0$, i.e.:*

$$g(\lambda) = \Pi_P(\mathbf{x}_0 - \lambda \nabla f(\mathbf{x}_0)) = \mathbf{x}_0 + \lambda \mathbf{d} \quad \textit{for all } \lambda \in [0, \delta],$$

*for some direction $\mathbf{d} \in \mathbb{R}^n$. Then, $N_P(g(\lambda)) = N_P(g(\lambda')) \subseteq N_P(\mathbf{x}_0)$, for all $0 < \lambda < \lambda' < \delta$.*

*Proof.* Let $I$ and $J$ denote the index-set of active and inactive constraints at $\mathbf{x}_0$ respectively. We will show that $N_P(g(\lambda)) = N_P(g(\lambda')) \subseteq N_P(\mathbf{x}_0)$ for all $0 < \lambda < \lambda' < \delta$. Our first claim is that since $\lambda' < \delta$, inactive constraints at $\mathbf{x}_0$ must remain inactive (claim (a)), and second, we use the fact that the projection is linear to show that the active constraints are maintained at all $\lambda, \lambda' \in (0, \delta)$ (claim (b)). Now, fix $\lambda, \lambda' \in (0, \delta)$ arbitrarily such that $\lambda' > \lambda$.

(a) **Inactive constraints remain inactive:** We show that $\mathbf{A}_J g(\lambda) < \mathbf{b}_J$ and $\mathbf{A}_J g(\lambda') < \mathbf{b}_J$ (component-wise). Since $\mathbf{x}_0 + \delta \mathbf{d} \in P$,

$$\mathbf{A}_J g(\delta) = \mathbf{A}_J(\mathbf{x}_0 + \delta \mathbf{d}) \leq \mathbf{b}_J$$

which implies that $\mathbf{A}_J \mathbf{d} \leq \frac{\mathbf{b}_J - \mathbf{A}_J \mathbf{x}_0}{\delta}$. Now, for any $\lambda < \delta$, we have

$$\mathbf{A}_J g(\lambda) = \mathbf{A}_J(\mathbf{x}_0 + \lambda \mathbf{d}) \leq \mathbf{A}_J \mathbf{x}_0 + \lambda \frac{\mathbf{b}_J - \mathbf{A}_J \mathbf{x}_0}{\delta} = \left(1 - \frac{\lambda}{\delta}\right) \mathbf{A}_J \mathbf{x}_0 + \mathbf{b}_J \frac{\lambda}{\delta} < \mathbf{b}_J.$$

where the last (strict) inequality follows from the fact that we are taking a convex combination. This shows $\mathbf{A}_J g(\lambda) < \mathbf{b}_J$ (component-wise) for all $\lambda \in (0, \delta)$. This implies that $N_P(g(\lambda)) \subseteq N_P(\mathbf{x}_0)$ for all $\lambda \in (0, \delta)$.

(b) **Active constraints are maintained:** We show that active constraints at $g(\lambda)$ and $g(\lambda')$ are the same. Since we know $A_J g(\lambda) < b_J$, $A_J g(\lambda') < b_J$, we need to check the constraints in the index set $I$, the set of active constraints at $\mathbf{x}_0$. Consider any $i \in I$. Since $\langle \mathbf{a}_i, \mathbf{x}_0 \rangle = b_i$ and $\mathbf{x}_0 + \delta \mathbf{d} \in P$ ($\delta > 0$), we know that $\langle \mathbf{a}_i, \mathbf{d} \rangle \leq 0$. If $\langle \mathbf{a}_i, \mathbf{d} \rangle = 0$, then since $g(\lambda) = \mathbf{x}_0 + \lambda \mathbf{d}$, we see that $\langle \mathbf{a}_i, g(\lambda) \rangle = b_i$ for all $\lambda \in [0, \delta]$. So the constraint corresponding to $\mathbf{a}_i$ is active at both $g(\lambda)$ and $g(\lambda')$. On the other hand, if $\langle \mathbf{a}_i, \mathbf{d} \rangle < 0$, then this constraint must become inactive at $g(\lambda)$, for any $\lambda > 0$, i.e., we have

$$\langle \mathbf{a}_i, g(\lambda) \rangle = \langle \mathbf{a}_i, \mathbf{x}_0 + \lambda \mathbf{d} \rangle < b_i,$$

and therefore, any constraint in $I$ inactive at $g(\lambda)$ must also be inactive at $g(\lambda')$.

We have thus shown that the set of active constraints are the same, i.e. $N_P(g(\lambda)) = N_P(g(\lambda')) \subseteq N_P(\mathbf{x}_0)$, for all $\lambda, \lambda' \in (0, \delta)$. $\square$

So far, we have shown that equal normal cones at $g(\lambda)$ and $g(\lambda')$ imply linear curve between these (Lemma 2). There exists some $\delta > 0$ such that normal cones up to $g(\lambda)$ $(\lambda < \delta)$ do not change around $\mathbf{x}_0$ (Lemma 5), and projections form a line from $\mathbf{x}_0$ to $g(\delta)$ (Corollary 1). We have also shown the converse: if projections do form a line emanating from $\mathbf{x}_0$ up to some $g(\theta)$, then the normal cones up to $g(\lambda)$ $(0 < \lambda < \theta)$ must also be the same and a subset of normal cone at $\mathbf{x}_0$. We now show that the maximum value of $\theta$ and $\delta$ is the same, and corresponds to the maximum step-size in the directional derivative of $\mathbf{w} = \nabla f(\mathbf{x}_0)$ at $\mathbf{x}_0$. These properties together give us Theorem 2.

We are now ready to prove the following:

**Theorem 2.** *Let $P \subseteq \mathbb{R}^n$ be a polytope defined using $m$ facet inequalities (e.g., as in (4)). Let $\mathbf{x}_0 \in P$ and we are given $\nabla f(\mathbf{x}_0) \in \mathbb{R}^n$. Let $g(\lambda) = \Pi_P(\mathbf{x}_0 - \lambda \nabla f(\mathbf{x}_0))$ be the parametric projections curve. Let $\lambda_1^- = \max\{\lambda \mid \mathbf{x}_0 + \lambda \mathbf{d}_{\mathbf{x}_0}^{\Pi} \in P\}$ be finite and let $\mathbf{x}_1 = g(\lambda_1^-)$. We claim that*

*(i) $N_P(g(\lambda)) = N_P(g(\lambda')) \subseteq N_P(\mathbf{x}_0)$, for all $0 < \lambda < \lambda' < \lambda_1^-$, and*

*(ii) $N_P(\mathbf{x}_1) = N_P(g(\lambda_1^-)) \supset N_P(g(\lambda))$, for all $\lambda \in (0, \lambda_1^-)$.*

*Moreover, the projections curve is given by $g(\lambda) = \mathbf{x}_0 + \lambda \mathbf{d}_{\mathbf{x}_0}^{\Pi}$, for all $\lambda \in [0, \lambda_1^-]$.*

*Proof.* We now put everything together to complete the proof of Theorem 2.

*Claim.* We first claim that:

$$g(\lambda) = \mathbf{x}_0 + \lambda \mathbf{d}_{\mathbf{x}_0}^{\Pi} \quad \text{for all } \lambda \in [0, \lambda_1^-], \tag{27}$$

which, in particular, means that $\mathbf{x}_1 = g(\lambda_1^-) = \mathbf{x}_0 + \lambda_1^- \mathbf{d}_{\mathbf{x}_0}^{\Pi}$, using the definition of $\mathbf{x}_1$.

*Pf.* We know using Corollary 1 that $\exists \delta > 0$ such that $g(\lambda) = \mathbf{x}_0 + \lambda \mathbf{d}_{\mathbf{x}_0}^{\Pi}$ for all $\lambda \in [0, \delta)$. Hence, to prove the claim, we have to show $g(\lambda') = \mathbf{x}_0 + \lambda' \mathbf{d}_{\mathbf{x}_0}^{\Pi}$ for all $\lambda' \in [\delta, \lambda_1^-]$. Using (26), we know that

$$\mathbf{x}_0 + \lambda' \mathbf{d}_{\mathbf{x}_0}^{\Pi} = \mathbf{x}_0 + \frac{\lambda'}{\lambda}(g(\lambda) - \mathbf{x}_0) \quad \text{for any } \lambda \in (0, \delta).$$

Then, since $\mathbf{x}_0 + \frac{\lambda'}{\lambda}(g(\lambda) - \mathbf{x}_0) \in P$ by definition of $\lambda_1^-$ and $\lambda' > \lambda$, using Lemma 6 we have $g(\lambda') = \mathbf{x}_0 + \frac{\lambda'}{\lambda}(g(\lambda) - \mathbf{x}_0) = \mathbf{x}_0 + \lambda' \mathbf{d}_{\mathbf{x}_0}^{\Pi}$. The claim now follows since $\lambda'$ was arbitrary.

We can now complete the proof of the theorem as follows.

Case $(i)$ Since, $g(\lambda) = \mathbf{x}_0 + \lambda \mathbf{d}_{\mathbf{x}_0}^{\Pi}$ for all $\lambda \in [0, \lambda_1^-]$, it follows from Lemma 7 that $N_P(g(\lambda)) = N_P(g(\lambda')) \subseteq N_P(\mathbf{x}_0)$ for all $0 < \lambda < \lambda' < \lambda_1^-$. This shows that $(i)$ in Theorem 2 holds.

Case $(ii)$ Note that $I(g(\lambda)) \subseteq I(\mathbf{x}_0)$ for any $\lambda \in [0, \lambda_1^-)$ by property $(i)$. Therefore, for any $i \in I(g(\lambda))$, we have

$$\begin{aligned} \langle \mathbf{a}_i, g(\lambda) \rangle = \left\langle \mathbf{a}_i, \mathbf{x}_0 + \lambda \mathbf{d}_{\mathbf{x}_0}^{\Pi} \right\rangle = b_i \implies \left\langle \mathbf{a}_i, \mathbf{d}_{\mathbf{x}_0}^{\Pi} \right\rangle = 0 \\ \implies \langle \mathbf{a}_i, \mathbf{x}_1 \rangle = \left\langle \mathbf{a}_i, \mathbf{x}_0 + \lambda_1^- \mathbf{d}_{\mathbf{x}_0}^{\Pi} \right\rangle = b_i, \end{aligned}$$

and so the constraint corresponding to $\mathbf{a}_i$ (which is active at $g(\lambda)$) is also active at $\mathbf{x}_1$. We have thus shown that $N_P(g(\lambda)) \subseteq N_P(\mathbf{x}_1)$ for all $\lambda \in (0, \lambda_1^-)$.

We will now show that this containment is strict, i.e. there is at least one constraint that is active at $\mathbf{x}_1$ but is not active at $g(\lambda)$ for all $\lambda \in (0, \lambda_1^-)$. Note that since $\mathbf{d}_{\mathbf{x}_0}^{\Pi}$ is a feasible direction at $\mathbf{x}_0$, we have $\left\langle \mathbf{a}_i, \mathbf{d}_{\mathbf{x}_0}^{\Pi} \right\rangle \leq 0$ for all $i \in I(\mathbf{x}_0)$. Thus, it follows that the maximum step size in which we can move along $\mathbf{d}_{\mathbf{x}_0}^{\Pi}$ is given by

$$\lambda_1^- = \min_{\substack{j \in J: \\ \left\langle \mathbf{a}_j, \mathbf{d}_{\mathbf{x}_0}^{\Pi} \right\rangle > 0}} \frac{b_j - \langle \mathbf{a}_j, \mathbf{x}_0 \rangle}{\left\langle \mathbf{a}_j, \mathbf{d}_{\mathbf{x}_0}^{\Pi} \right\rangle}, \tag{28}$$

where the feasible set of the above problem is non-empty, since otherwise this would imply that $\mathbf{d}^{\Pi}_{\mathbf{x}_0}$ is a recessive direction (i.e. direction of unboundedness), contradicting the fact that $P$ is a polytope. Let $j^*$ be any optimal index to the optimization problem in (28), where $\langle \mathbf{a}_{j^*}, \mathbf{x}_1 \rangle = b_{j^*}$. Now, for any $\lambda \in (0, \lambda_1^-)$

$$\begin{aligned}\langle \mathbf{a}_{j^*}, g(\lambda) \rangle &= \left\langle \mathbf{a}_{j^*}, \mathbf{x}_0 + \lambda \mathbf{d}^{\Pi}_{\mathbf{x}_0} \right\rangle \\ &< \left\langle \mathbf{a}_{j^*}, \mathbf{x}_0 + \lambda_1^- \mathbf{d}^{\Pi}_{\mathbf{x}_0} \right\rangle \qquad \text{(choice of } \lambda \text{ and } \left\langle \mathbf{a}_{j^*}, \mathbf{d}^{\Pi}_{\mathbf{x}_0} \right\rangle > 0) \\ &= b_{j^*},\end{aligned}$$

implying that the constraint $\mathbf{a}_{j^*}$ is not active at $g(\lambda)$. Thus, we have $N_P(g(\lambda)) \subset N_P(\mathbf{x}_1)$ for all $\lambda \in (0, \lambda_1^-)$, which shows property $(ii)$ in Theorem 2.

□

### B.4 Proof of Theorem 3

To prove this theorem we first claim that under some conditions, the first linear segment of the projection curve starting at the $i-1^{\text{th}}$ breakpoint $\mathbf{x}_{i-1}$ with respect to any vector $\mathbf{w}$ is the same as the $i^{\text{th}}$ piecewise linear segment of the projection curve starting from $\mathbf{x}_0$ with respect to $\mathbf{w}$. Suppose $\mathbf{w} = \nabla f(\mathbf{x}_0)$, then the invariance property we would like to specifically show is:

**Lemma 8** (Invariance of projections)**.** *Let $P \subseteq \mathbb{R}^n$ be a polytope. Let $\mathbf{x}_0 \in P$ and $\nabla f(\mathbf{x}_0) \in \mathbb{R}^n$ be given. Further, let $\mathbf{x}_{i-1} \in P$ be the ith breakpoint in the projections curve $g(\lambda) = \Pi_P(\mathbf{x}_0 - \lambda \nabla f(\mathbf{x}_0))$, with $\mathbf{x}_{i-1} = \mathbf{x}_0$ for $i = 1$. Define $\lambda^-_{i-1} := \min\{\lambda \mid g(\lambda) = \mathbf{x}_{i-1}\}$, $\lambda^+_{i-1} := \max\{\lambda \mid g(\lambda) = \mathbf{x}_{i-1}\}$, $\hat{\lambda}_{i-1} := \sup\{\lambda \mid N_P(g(\lambda')) = N_P(\mathbf{x}_{i-1}) \ \forall \lambda' \in [\lambda^-_{i-1}, \lambda)\}$, and suppose that $\lambda^+_{i-1} = \hat{\lambda}_{i-1}$. Then we claim the following invariance property of orthogonal projections:*

$$\tilde{g}(\lambda) := \Pi_P(\mathbf{x}_{i-1} - (\lambda - \lambda^+_{i-1}) \nabla f(\mathbf{x}_0)) = g(\lambda) \quad \textit{for } \lambda \in [\lambda^+_{i-1}, \lambda^-_i]. \tag{29}$$

To prove this lemma, we need some technical results. The first result we have is a structural one about minimizing strongly functions over *polyhedrons*, and states that if we know the optimal (minimal) face then we can restrict the optimization to that optimal face and ignore the remaining faces of the polyhedron:

**Lemma 9** (Reduction of optimization problem to optimal face)**.** *Let $\mathcal{P} = \{\mathbf{x} \in \mathbb{R}^n : \langle \mathbf{a}_i, \mathbf{x} \rangle \leq b_i \ \forall \ i \in [m]\}$ be a polyhedron and suppose that $f : \mathbb{R}^n \to \mathbb{R}$ be $\mu$-strongly convex over $\mathcal{P}$. Let $\mathbf{x}^* = \arg\min_{\mathbf{x} \in \mathcal{P}} f(\mathbf{x})$, where uniqueness and existence of the optimal solution follow from the strong convexity of $f$. Further, let*

$$\tilde{\mathbf{x}} = \arg\min_{\mathbf{x} \in \mathbb{R}^n} \{f(\mathbf{x}) \mid \mathbf{A}_{I(\mathbf{x}^*)} \mathbf{x} = \mathbf{b}_{I(\mathbf{x}^*)}\}. \tag{30}$$

*Then, we claim that $\mathbf{x}^* = \tilde{\mathbf{x}}$.*

*Proof.* Let $J(\mathbf{x}^*)$ denote the index set of inactive constraints at $\mathbf{x}^*$. We assume that $J(\mathbf{x}^*) \neq \emptyset$, since otherwise the result follows trivially. Now, suppose for a contradiction that $\mathbf{x}^* \neq \tilde{\mathbf{x}}$. Due to uniqueness of the minimizer of the strongly convex function over $\mathcal{P}$, we have that $\tilde{\mathbf{x}} \notin \mathcal{P}$ (otherwise it contradicts optimality of $\mathbf{x}^*$ over $\mathcal{P}$). Define

$$\gamma := \min_{\substack{j \in J(\mathbf{x}^*): \\ \langle \mathbf{a}_j, \tilde{\mathbf{x}} - \mathbf{x}^* \rangle > 0}} \frac{b_j - \langle \mathbf{a}_j, \mathbf{x}^* \rangle}{\langle \mathbf{a}_j, \tilde{\mathbf{x}} - \mathbf{x}^* \rangle} > 0, \tag{31}$$

with the convention that $\gamma = \infty$ if the feasible set of (31) is empty, i.e. $\langle \mathbf{a}_j, \tilde{\mathbf{x}} - \mathbf{x}^* \rangle \leq 0$ for all $j \in J(\mathbf{x}^*)$. Select $\tilde{\theta} \in (0, \min\{\gamma, 1\})$. Further, define $\mathbf{y} := \mathbf{x}^* + \tilde{\theta}(\tilde{\mathbf{x}} - \mathbf{x}^*) \neq \mathbf{x}^*$ to be a strict convex combination of $\mathbf{x}^*$ and $\tilde{\mathbf{x}}$. We claim that that $(i)$ $\mathbf{y} \in \mathcal{P}$ and $(ii)$ $f(\mathbf{y}) < f(\mathbf{x}^*)$, which contradicts the optimality of $\mathbf{x}^*$.

Claim $(i)$: $\mathbf{y} \in \mathcal{P}$. Any inequality satisfied by $\tilde{\mathbf{x}}$ is also satisfied by $\mathbf{x}^*$ and therefore by $\mathbf{y}$. Consider $j \in J(\mathbf{x}^*)$ such that $\langle \mathbf{a}_j, \tilde{\mathbf{x}} \rangle > b_j \geq \langle \mathbf{a_j}, \mathbf{x}^* \rangle$. Then, we have

$$\begin{aligned}\langle \mathbf{a}_j, \mathbf{y} \rangle &= \langle \mathbf{a}_j, \mathbf{x}^* \rangle + \tilde{\theta} \langle \mathbf{a}_j, \tilde{\mathbf{x}} - \mathbf{x}^* \rangle \leq \langle \mathbf{a}_j, \mathbf{x}^* \rangle + \gamma \langle \mathbf{a}_j, \tilde{\mathbf{x}} - \mathbf{x}^* \rangle \\ &\leq \langle \mathbf{a}_j, \mathbf{x}^* \rangle + b_j - \langle \mathbf{a}_j, \mathbf{x}^* \rangle = b_j,\end{aligned}$$

where we used the fact that $\tilde{\theta} \leq \gamma$ in the first inequality, and the definition of $\gamma$ (31) in the second inequality. This establishes the feasibility of $\mathbf{y} \in \mathcal{P}$.

Claim $(ii)$: $f(\mathbf{y}) < f(\mathbf{x}^*)$. Observe that $f(\tilde{\mathbf{x}}) \leq f(\mathbf{x}^*)$ since $\mathbf{x}^*$ is feasible for (30). We can now complete the proof of this claim as follows:

$$f(\mathbf{y}) = f((1-\tilde{\theta})\mathbf{x}^* + \tilde{\theta}\tilde{\mathbf{x}}) \tag{32}$$

$$\leq (1-\tilde{\theta})f(\mathbf{x}^*) + \tilde{\theta}f(\tilde{\mathbf{x}}) - \frac{\tilde{\theta}(1-\tilde{\theta})\mu}{2}\|\mathbf{x}^* - \tilde{\mathbf{x}}\|^2 \tag{33}$$

$$< (1-\tilde{\theta})f(\mathbf{x}^*) + \tilde{\theta}f(\tilde{\mathbf{x}}) \tag{34}$$

$$\leq f(\mathbf{x}^*), \tag{35}$$

where we used the fact $\tilde{\theta} \in (0,1)$ and the strong convexity of $f$ in (33), the fact that $\mathbf{x}^* \neq \tilde{\mathbf{x}}$ in (34), and finally the fact that $f(\tilde{\mathbf{x}}) \leq f(\mathbf{x}^*)$ in (35).

This completes the proof. □

The second technical result we need for the proof of Lemma 8 is a continuity property of the projections curve, and states that for any point on the projections curve $g(\lambda)$, any inactive constraint at $g(\lambda)$ is also inactive at all points $g(\lambda \pm \epsilon)$ for $\epsilon \geq 0$ sufficiently small:

**Lemma 10** (A continuity property of the projections curve)**.** *Let $P \subseteq \mathbb{R}^n$ be a polytope. Let $\mathbf{x}_0 \in P$ and $\nabla f(\mathbf{x}_0) \in \mathbb{R}^n$ be given. Further, let $g(\lambda) = \Pi_P(\mathbf{x}_0 - \lambda\nabla f(\mathbf{x}_0))$ denote the projections curve. Fix $\lambda > 0$ and define $\mathbf{y} := g(\lambda)$. Let $J(\mathbf{y})$ be the index set of inactive constraints at $\mathbf{y}$. Then, there exists a scalar $\delta > 0$ such that $J(\mathbf{y}) = J(g(\lambda \pm \epsilon))$ for all $\epsilon \in (-\delta, \delta)$, that is for all $j \in J(\mathbf{y})$ we have*

$$\langle \mathbf{a}_j, g(\lambda + \epsilon)\rangle < b_j \quad \textit{for all } \epsilon \in (-\delta, \delta).$$

*In particular, if $J(\mathbf{y}) = \emptyset$, then the polytope only contains a single point and $\delta = \infty$, otherwise $\delta < \infty$.*

*Proof.* First, if $J(\mathbf{y}) = \emptyset$, then the polytope $P$ contains only one point $\mathbf{y}$, in which case the result trivially follows with $\delta = \infty$. So, we now assume that $J(\mathbf{y}) \neq \emptyset$. Consider any $\delta$ satisfying

$$0 < \delta \leq \min_{j \in J(\mathbf{y})} \frac{b_j - \langle \mathbf{a}_j, \mathbf{y}\rangle}{\|\mathbf{a}_j\|\|\nabla f(\mathbf{x}_0)\|}. \tag{36}$$

We will now show that $\delta$ satisfies the conditions stated in the lemma. For all $j \in J(\mathbf{y})$ (so that $\langle \mathbf{a}_j, \mathbf{y}\rangle < b_j$) and $\epsilon \in (-\delta, \delta)$, we have

$$\begin{aligned}
\langle \mathbf{a}_j, g(\lambda+\epsilon)\rangle &= \langle \mathbf{a}_j, \mathbf{y} + g(\lambda+\epsilon) - \mathbf{y}\rangle \\
&\leq \langle \mathbf{a}_j, \mathbf{y}\rangle + \|\mathbf{a}_j\|\|g(\lambda+\epsilon) - \mathbf{y}\| && (37)\\
&= \langle \mathbf{a}_j, \mathbf{y}\rangle + \|\mathbf{a}_j\|\|g(\lambda+\epsilon) - g(\lambda)\| && (38)\\
&\leq \langle \mathbf{a}_j, \mathbf{y}\rangle + |\epsilon|\|\mathbf{a}_j\|\|\nabla f(\mathbf{x}_0)\| && (39)\\
&< \langle \mathbf{a}_j, \mathbf{y}\rangle + \delta\|\mathbf{a}_j\|\|\nabla f(\mathbf{x}_0)\| && (40)\\
&\leq b_j, && (41)
\end{aligned}$$

where (37) follows from Cauchy-Schwartz, (39) from non-expansiveness of the projection operator (15), (40) from the choice of $\epsilon < \delta$, and (41) from the choice of $\delta$ in (36). □

The final technical result we need for the proof of Lemma 8 states that if $N_P(\mathbf{x}_{i-1}) \supset N_P(g(\lambda_{i-1}^+ + \epsilon))$ for $\epsilon$ sufficiently small at a breakpoint $\mathbf{x}_{i-1}$ (i.e., a constraint is dropped at $\mathbf{x}_{i-1}$), then the normal vector of the Euclidean projection of $\mathbf{x}_0 - \lambda_{i-1}^+\nabla f(\mathbf{x}_0)$ is orthogonal to the shadow at $\mathbf{x}_{i-1}$ with respect to $\nabla f(\mathbf{x}_0)$, that is $\left\langle \mathbf{x}_0 - \lambda_{i-1}^+\nabla f(\mathbf{x}_0) - \mathbf{x}_{i-1}, \mathbf{d}^{\Pi}_{\mathbf{x}_{i-1}}(\nabla f(\mathbf{x}_0)\right\rangle = 0$.

**Lemma 11.** *Let $P \subseteq \mathbb{R}^n$ be a polytope. Let $\mathbf{x}_0 \in P$ and $\nabla f(\mathbf{x}_0) \in \mathbb{R}^n$ be given. Further, let $\mathbf{x}_{i-1} \in P$ be the $i$th breakpoint in the projections curve $g(\lambda) = \Pi_P(\mathbf{x}_0 - \lambda\nabla f(\mathbf{x}_0))$, with $\mathbf{x}_{i-1} = \mathbf{x}_0$ for $i = 1$. Define $\lambda_{i-1}^- := \min\{\lambda \mid g(\lambda) = \mathbf{x}_{i-1}\}$, $\lambda_{i-1}^+ := \max\{\lambda \mid g(\lambda) = \mathbf{x}_{i-1}\}$, $\hat{\lambda}_{i-1} := \sup\{\lambda \mid N_P(g(\lambda')) = N_P(\mathbf{x}_{i-1})\ \forall \lambda' \in [\lambda_{i-1}^-, \lambda)\}$, and suppose that $\lambda_{i-1}^+ = \hat{\lambda}_{i-1}$. Then,*

$$\left\langle \mathbf{x}_0 - \lambda_{i-1}^+\nabla f(\mathbf{x}_0) - \mathbf{x}_{i-1}, \mathbf{d}^{\Pi}_{\mathbf{x}_{i-1}}(\nabla f(\mathbf{x}_0)\right\rangle = 0. \tag{42}$$

*Proof.* Assume $\mathbf{d}^{\Pi}_{\mathbf{x}_{i-1}}(\nabla f(\mathbf{x}_0)) \neq \mathbf{0}$ otherwise the statement follows trivially. For notational brevity we let $I$ and $J$ denote the index-set of active and inactive constraints at $\mathbf{x}_{i-1}$ respectively. Recall that (see Section 2)

$$\mathbf{d}^{\Pi}_{\mathbf{x}_{i-1}}(\nabla f(\mathbf{x}_0)) = \arg\min_{\mathbf{d}}\{\| - \nabla f(\mathbf{x}_0) - \mathbf{d}\|^2 \mid \mathbf{A}_I \mathbf{d} \leq \mathbf{0}\}. \tag{43}$$

Let $\hat{I} \subseteq I$ be the subset of constraints that satisfy $\mathbf{A}_{\hat{I}}\mathbf{d}^{\Pi}_{\mathbf{x}_{i-1}}(\nabla f(\mathbf{x}_0)) = \mathbf{0}$. By reducing the above optimization problem to the optimal face (9), we can rewrite the optimization problem in (43) as

$$\mathbf{d}^{\Pi}_{\mathbf{x}_{i-1}}(\nabla f(\mathbf{x}_0)) = \arg\min_{\mathbf{d}}\{\| - \nabla f(\mathbf{x}_0) - \mathbf{d}\|^2 \mid \mathbf{A}_{\hat{I}} \mathbf{d} = \mathbf{0}\}, \tag{44}$$

where its solution is given $\mathbf{d}^{\Pi}_{\mathbf{x}_{i-1}}(\nabla f(\mathbf{x}_0)) = (\mathbf{I} - \mathbf{A}^{\dagger}_{\hat{I}}\mathbf{A}_{\hat{I}})(-\nabla f(\mathbf{x}_0))$.

Denote the normal vector $\boldsymbol{p} := \mathbf{x}_0 - \lambda^+_{i-1}\nabla f(\mathbf{x}_0) - \mathbf{x}_{i-1}$ and assume that $\boldsymbol{p} \neq \mathbf{0}$, since otherwise the statement follows trivially. Thus, since $\mathbf{d}^{\Pi}_{\mathbf{x}_{i-1}}(\nabla f(\mathbf{x}_0))$ is the projection of $-\nabla f(\mathbf{x}_0)$ onto the nullspace of $\mathbf{A}_{\hat{I}}$ and $\mathbf{d}^{\Pi}_{\mathbf{x}_{i-1}}(\nabla f(\mathbf{x}_0)) \neq \mathbf{0}$, it follows that $\left\langle \boldsymbol{p}, \mathbf{d}^{\Pi}_{\mathbf{x}_{i-1}}(\nabla f(\mathbf{x}_0)) \right\rangle = 0$ if and only if $\boldsymbol{p}$ is in the rowspace of $\mathbf{A}_{\hat{I}}$, that is

$$\left\langle \boldsymbol{p}, \mathbf{d}^{\Pi}_{\mathbf{x}_{i-1}}(\nabla f(\mathbf{x}_0)) \right\rangle = 0 \Leftrightarrow \boldsymbol{p} = \mathbf{A}^{\top}_{\hat{I}}\boldsymbol{v} \quad \text{for some } \boldsymbol{v} \in \mathbb{R}^{|\hat{I}|} \tag{45}$$

Since $\mathbf{x}_{i-1} = g(\lambda^+_{i-1})$, we have $\boldsymbol{p} \in N_P(\mathbf{x}_{i-1})$, that is $\boldsymbol{p} = \mathbf{A}^{\top}_{I}\boldsymbol{\mu}$ for some $\boldsymbol{\mu} \in \mathbb{R}^{|I|}_+$. Thus, using (45), to complete the proof we will show that $\boldsymbol{p}$ can be written as a conic combination using only the subset of constraints $\hat{I}$, i.e $\boldsymbol{\mu}_{\boldsymbol{a}_i} = 0$ for $i \in I \setminus \hat{I}$, in the case that $\hat{I} \subset I$. The rest of the proof is devoted to showing this fact.

By the continuity of the projections curve (Lemma 10 with $\mathbf{y} := g(\lambda^+_{i-1})$), we know that there exists some $\delta > 0$ such that $\left\langle \mathbf{a}_j, g(\lambda^+_{i-1} + \epsilon) \right\rangle < b_j$ for all $0 < \epsilon < \delta$. This shows that the inactive constraints at $\mathbf{x}_{i-1}$ remain inactive at $g(\lambda^+_{i-1} + \epsilon))$ for any $0 < \epsilon < \delta$, i.e., $N_P(g(\lambda^+_{i-1} + \epsilon))) \subseteq N_P(\mathbf{x}_{i-1})$. However, we are also given that $\lambda^+_{i-1} = \hat{\lambda}_{i-1}$, which together with our previous claim implies that this containment is strict in a small neighborhood around $\mathbf{x}_{i-1}$, i.e. $N_P(g(\lambda^+_{i-1} + \epsilon)) \subset N_P(\mathbf{x}_{i-1})$ for any $0 < \epsilon < \delta$. So, some of the constraints at the breakpoint $\mathbf{x}_{i-1}$ are relaxed.

Furthermore, using the definition of $\mathbf{d}^{\Pi}_{\mathbf{x}_{i-1}}(\nabla f(\mathbf{x}_0))$ (43), it follows that for any $\epsilon$ sufficiently small, the set of active constraints at $g(\lambda^+_{i-1}+\epsilon)$ will also be $\hat{I}$, i.e., the constraints relaxed by the projections curve at $\mathbf{x}_{i-1}$ coincide with those relaxed by the shadow computation in (43). Denoting the normal vector of the projection $g(\lambda^+_{i-1} + \epsilon)$ by

$$\boldsymbol{p}_{\epsilon} := \mathbf{x}_0 - (\lambda^+_{i-1} + \epsilon)\nabla f(\mathbf{x}_0) - g(\lambda^+_{i-1} + \epsilon) \in N_P(g(\lambda^+_{i-1} + \epsilon)), \tag{46}$$

this implies that we can write $\boldsymbol{p}_{\epsilon} = \mathbf{A}^{\top}_{\hat{I}}\boldsymbol{\mu}$ for some $\boldsymbol{\mu} \in \mathbb{R}^{|\hat{I}|}_+$. Since this is true for any $\epsilon > 0$ arbitrarily small and the curve $g(\lambda)$ is continuous, the result follows by letting $\epsilon \downarrow 0$. That is, we can write the vector $\boldsymbol{p} \in N_P(\mathbf{x}_{i-1})$ as $\boldsymbol{p} = \mathbf{A}^{\top}_{\hat{I}}\boldsymbol{\mu}$ for some $\boldsymbol{\mu} \in \mathbb{R}^{|\hat{I}|}_+$, i.e. as a conic combination using only the subset of constraints indexed by $\hat{I}$. □

We are now ready to prove the lemma:

*Proof of Lemma 8.* Fix a non-negative scalar $\lambda \in [\lambda^+_{i-1}, \lambda^-_i]$ and suppose for a contradiction that $\tilde{g}(\lambda) = \mathbf{z}$ and $g(\lambda) = \mathbf{y}$ where $\mathbf{z} \neq \mathbf{y}$. Then, since $\mathbf{z}$ is not the projection $g(\lambda) = \Pi_P(\mathbf{x}_0 - \lambda\nabla f(\mathbf{x}_0))$, we know there exists $\tilde{\mathbf{z}} \in P$:

$$\begin{aligned}
&\langle \mathbf{x}_0 - \lambda\nabla f(\mathbf{x}_0) - \mathbf{z}, \tilde{\mathbf{z}} - \mathbf{z}\rangle > 0 && (47)\\
\Rightarrow &\left\langle \mathbf{x}_0 - \lambda^+_{i-1}\nabla f(\mathbf{x}_0) - \mathbf{x}_{i-1}, \tilde{\mathbf{z}} - \mathbf{z}\right\rangle + \left\langle \mathbf{x}_{i-1} - (\lambda - \lambda^+_{i-1})\nabla f(\mathbf{x}_0) - \mathbf{z}, \tilde{\mathbf{z}} - \mathbf{z}\right\rangle > 0 \\
\Rightarrow &\left\langle \mathbf{x}_0 - \lambda^+_{i-1}\nabla f(\mathbf{x}_0) - \mathbf{x}_{i-1}, \tilde{\mathbf{z}} - \mathbf{z}\right\rangle > 0 && (48)\\
\Rightarrow &\left\langle \mathbf{x}_0 - \lambda^+_{i-1}\nabla f(\mathbf{x}_0) - \mathbf{x}_{i-1}, \tilde{\mathbf{z}} - \mathbf{x}_{i-1}\right\rangle + \left\langle \mathbf{x}_0 - \lambda^+_{i-1}\nabla f(\mathbf{x}_0) - \mathbf{x}_{i-1}, \mathbf{x}_{i-1} - \mathbf{z}\right\rangle > 0, \\
\Rightarrow &\left\langle \mathbf{x}_0 - \lambda^+_{i-1}\nabla f(\mathbf{x}_0) - \mathbf{x}_{i-1}, \mathbf{x}_{i-1} - \mathbf{z}\right\rangle > 0, && (49)
\end{aligned}$$

where (48) follows by first-order optimality at $\mathbf{z}$, and (49) follows by first-order optimality at $\mathbf{x_{i-1}}$. Using Theorem 2 with the projections curve starting at $\mathbf{x}_{i-1}$ with parameter $\theta = (\lambda - \lambda^+_{i-1})$, we have

$$\tilde{g}(\theta+\lambda^+_{i-1}) := \Pi_P(\mathbf{x}_{i-1}-\theta\nabla f(\mathbf{x}_0)) = \mathbf{x}_{i-1}+\theta\mathbf{d}^{\Pi}_{\mathbf{x}_{i-1}}(\nabla f(\mathbf{x}_0)), \text{ for all } \theta \in [0, \lambda^-_i - \lambda^+_{i-1}], \quad (50)$$

where recall that $\mathbf{d}^{\Pi}_{\mathbf{x}_{i-1}}(\nabla f(\mathbf{x}_0))$ is the directional derivative at $\mathbf{x}_{i-1}$ with respect to the direction given by $\nabla f(\mathbf{x}_0)$:

$$\mathbf{d}^{\Pi}_{\mathbf{x}_{i-1}}(\nabla f(\mathbf{x}_0)) = \lim_{\epsilon \downarrow 0} \frac{\Pi_p(\mathbf{x}_{i-1} - \epsilon\nabla f(\mathbf{x}_0)) - \mathbf{x}_{i-1}}{\epsilon}.$$

Thus, using this fact in (50), we have

$$\begin{aligned}
&\left\langle \mathbf{x}_0 - \lambda^+_{i-1}\nabla f(\mathbf{x}_0) - \mathbf{x}_{i-1}, \mathbf{x}_{i-1} - \mathbf{z} \right\rangle = \left\langle \mathbf{x}_0 - \lambda^+_{i-1}\nabla f(\mathbf{x}_0) - \mathbf{x}_{i-1}, \mathbf{x}_{i-1} - \tilde{g}(\lambda) \right\rangle \\
&\quad = \left\langle \mathbf{x}_0 - \lambda^+_{i-1}\nabla f(\mathbf{x}_0) - \mathbf{x}_{i-1}, -(\lambda - \lambda^+_{i-1})\mathbf{d}^{\Pi}_{\mathbf{x}_{i-1}}(\nabla f(\mathbf{x}_0)) \right\rangle > 0 \text{ for } \lambda > \lambda^+_{i-1} \\
&\quad \Rightarrow \left\langle \mathbf{x}_0 - \lambda^+_{i-1}\nabla f(\mathbf{x}_0) - \mathbf{x}_{i-1}, \mathbf{d}^{\Pi}_{\mathbf{x}_{i-1}}(\nabla f(\mathbf{x}_0)) \right\rangle < 0.
\end{aligned}$$

However, using Lemma 11, we know that $\left\langle \mathbf{x}_0 - \lambda^+_{i-1}\nabla f(\mathbf{x}_0) - \mathbf{x}_{i-1}, \mathbf{d}^{\Pi}_{\mathbf{x}_{i-1}}(\nabla f(\mathbf{x}_0)) \right\rangle = 0$, which gives us the desired contradiction. □

**Theorem 3** (Tracing the projections curve)**.** *Let $P \subseteq \mathbb{R}^n$ be a polytope defined using $m$ facet inequalities (e.g., as in* (4)*). Let $\mathbf{x}_{i-1} \in P$ be the $i$th breakpoint in the projections curve $g(\lambda) = \Pi_P(\mathbf{x}_0 - \lambda\nabla f(\mathbf{x}_0))$, with $\mathbf{x}_{i-1} = \mathbf{x}_0$ for $i = 1$. Suppose we are given $\lambda^-_{i-1}, \lambda^+_{i-1} \in \mathbb{R}$ so that they are respectively the minimum and the maximum step-sizes $\lambda$ such that $g(\lambda) = \mathbf{x}_{i-1}$. Let $\hat{\lambda}_{i-1} := \sup\{\lambda \mid N_P(g(\lambda')) = N_P(\mathbf{x}_{i-1}) \; \forall \lambda' \in [\lambda^-_{i-1}, \lambda)\}$. Then, we show that:*

1. *If $\lambda^-_{i-1} < \lambda^+_{i-1}$, then $\lambda^+_{i-1} = \hat{\lambda}_{i-1}$. Otherwise, $\lambda^-_{i-1} = \lambda^+_{i-1} \leq \hat{\lambda}_{i-1}$.*
2. *Linearity of the curve between $g(\lambda^-_{i-1})$ and $g(\hat{\lambda}_{i-1})$: i.e., $g(\lambda^-_{i-1} + (1-\delta)\hat{\lambda}_{i-1}) = \delta g(\lambda_{i-1}) + (1-\delta)g(\hat{\lambda}_{i-1})$, where $\delta \in [0,1]$. In particular, $g(\lambda) = \mathbf{x}_{i-1}$ for all $\lambda \in [\lambda^-_{i-1}, \lambda^+_{i-1}]$.*
3. *If $\mathbf{d}^{\Pi}_{\mathbf{x}_{i-1}}(\nabla f(\mathbf{x}_0)) = \mathbf{0}$, then $\lim_{\lambda\to\infty} g(\lambda) = \mathbf{x}_{i-1}$ is the end point of the projections curve $g(\lambda)$.*
4. *Otherwise $\mathbf{d}^{\Pi}_{\mathbf{x}_{i-1}}(\nabla f(\mathbf{x}_0)) \neq \mathbf{0}$, we get $\lambda^+_{i-1} \leq \hat{\lambda}_{i-1} < \infty$ (from (1)). We then claim:*
   - *(a)* **In-face movements:** *If $\hat{\lambda}_{i-1} > \lambda^+_{i-1}$, then the next breakpoint in the curve occurs by walking in-face up to $\hat{\lambda}_{i-1}$, i.e., $\mathbf{x}_i := g(\hat{\lambda}_{i-1}) = \mathbf{x}_{i-1} + (\hat{\lambda}_{i-1} - \lambda^+_{i-1})\hat{\mathbf{d}}^{\Pi}_{\mathbf{x}_{i-1}}(\nabla f(\mathbf{x}_0))$ and $\lambda^-_i := \hat{\lambda}_{i-1}$. Moreover, $N_P(\mathbf{x}_{i-1}) \subseteq N_P(g(\hat{\lambda}_{i-1}))$, with strict containment only when the maximum movement along in-face direction takes place, i.e., $\hat{\lambda}_{i-1} = \lambda^+_{i-1} + \max\{\delta : \mathbf{x}_{i-1} + \delta\hat{\mathbf{d}}^{\Pi}_{\mathbf{x}_{i-1}}(\nabla f(\mathbf{x}_0)) \in P\}$.*
   - *(b)* **Shadow movements:** *Otherwise if $\hat{\lambda}_{i-1} = \lambda^+_{i-1}$, then the movement is in the shadow direction, i.e., $\mathbf{x}_i := g(\lambda^-_i) = \mathbf{x}_{i-1} + (\lambda^-_i - \lambda^+_{i-1})\mathbf{d}^{\Pi}_{\mathbf{x}_{i-1}}(\nabla f(\mathbf{x}_0))$ where $\lambda^-_i := \lambda^+_{i-1} + \max\{\delta : \mathbf{x}_{i-1} + \delta\mathbf{d}^{\Pi}_{\mathbf{x}_{i-1}}(\nabla f(\mathbf{x}_0)) \in P\}$.*

*In particular, the projections curve is linear between $\lambda^+_{i-1}$ and $\lambda^-_i$. Further, we show that properties (i), (ii) and (iii) in Theorem 1 hold for their respective normal cones for $\lambda, \lambda' \in (\lambda^+_{i-1}, \lambda^-_i)$, where the containments in (i) and (iii) are strict for case (b).*

*Proof.* We now prove the different cases in the theorem statement:

1. If $\mathbf{x}_{i-1}$ is the endpoint of the projections curve, then $\lambda^+_{i-1} = \infty$ and the result trivially follows. So, consider the case when $\mathbf{x}_{i-1}$ is not the endpoint of the projections curve. We will show that there is a drop in the normal cone at $\mathbf{x}_{i-1}$ beyond the stepsize $\lambda^+_{i-1}$: if $\lambda^-_{i-1} < \lambda^+_{i-1}$, then there exists some $\delta > 0$ such that $N_P(g(\lambda^+_{i-1} + \epsilon)) \subset N_P(\mathbf{x}_{i-1})$ for all $\epsilon \in (0, \delta)$, which implies that $\hat{\lambda}_{i-1} = \lambda^+_{i-1}$ as claimed.

   First, by the continuity of the projections curve (Lemma 10 with $\mathbf{y} := g(\lambda^+_{i-1})$), we know that there exists some $\delta \in (0, \infty)$ such that $N_P(g(\lambda^+_{i-1} + \epsilon)) \subseteq N_P(\mathbf{x}_{i-1})$ for any $0 < \epsilon < \delta$. We further claim that this containment is strict. To see this, suppose for a contradiction that

$N_P(g(\lambda)) = N_P(\mathbf{x}_{i-1})$ for some $\lambda^+_{i-1} < \tilde{\lambda} < \lambda^+_{i-1} + \delta$. Since $\lambda^+_{i-1} := \max\{\lambda \mid g(\lambda) = \mathbf{x}_{i-1}\}$, it follows that $g(\tilde{\lambda}) \neq \mathbf{x}_{i-1}$. Using Lemma 2 (linearity of projections), we know that $g(\lambda^+_{i-1}) = \delta g(\lambda^-_{i-1}) + (1-\theta)g(\tilde{\lambda})$ for $\theta = \frac{\lambda^+_{i-1} - \lambda^-_{i-1}}{\tilde{\lambda} - \lambda^-_{i-1}} \in (0,1)$. But this is a contradiction since $g(\lambda^+_{i-1}) = g(\lambda^-_{i-1}) = \mathbf{x}_{i-1}$ but $g(\tilde{\lambda}) \neq \mathbf{x}_{i-1}$.

2. As stated in the previous claim, there are two cases: $\lambda^-_{i-1} < \lambda^+_{i-1} = \hat{\lambda}_{i-1}$ or $\lambda^-_{i-1} = \lambda^+_{i-1} \leq \hat{\lambda}_{i-1}$. In either case, the normal cones at $\lambda^-_{i-1}$, $\lambda^+_{i-1}$, and $\hat{\lambda}_{i-1}$ are the same. Therefore, the projections curve is linear between $\lambda^-_{i-1}$ and $\hat{\lambda}_{i-1}$ using Lemma 2. In particular, when $\lambda^-_{i-1} < \lambda^+_{i-1}$, then $g(\lambda) = \mathbf{x}_{i-1}$ for all $\lambda \in [\lambda^-_{i-1}, \lambda^+_{i-1}]$. Otherwise, when $\lambda^-_{i-1} = \lambda^+_{i-1} \leq \hat{\lambda}_{i-1}$, then $g(\delta\lambda^-_{i-1} + (1-\delta)\hat{\lambda}_{i-1}) = \delta g(\lambda^-_{i-1}) + (1-\delta)g(\hat{\lambda}_{i-1})$ for $\delta \in (0,1)$.

3. Since $\mathbf{d}^{\Pi}_{\mathbf{x}_{i-1}}(\nabla f(\mathbf{x}_0)) = \mathbf{0}$, it follows that $-\nabla f(\mathbf{x}_0) \in N_P(\mathbf{x}_{i-1})$, i.e. $\langle -\nabla f(\mathbf{x}_0), \mathbf{z} - \mathbf{x}_{i-1}\rangle \leq 0$ for all $\mathbf{z} \in P$. Using the first-order optimality condition of $g(\lambda^+_{i-1}) = \mathbf{x}_{i-1}$, we have
$$\left\langle \mathbf{x}_0 - \lambda^+_{i-1}\nabla f(\mathbf{x}_0) - \mathbf{x}_{i-1}, \mathbf{z} - \mathbf{x}_{i-1}\right\rangle \leq 0 \quad \forall \mathbf{z} \in P.$$
Since $\langle -\nabla f(\mathbf{x}_0), \mathbf{z} - \mathbf{x}_{i-1}\rangle \leq 0$ for all $\mathbf{z} \in P$ and $\lambda \geq \lambda^+_{i-1}$, we get
$$\langle \mathbf{x}_0 - \lambda\nabla f(\mathbf{x}_0) - \mathbf{x}_{i-1}, \mathbf{z} - \mathbf{x}_{i-1}\rangle \leq 0 \quad \forall \mathbf{z} \in P$$
so that $\mathbf{x}_{i-1}$ satisfies the first-order optimality condition for $g(\lambda)$ when $\lambda \geq \lambda^+_{i-1}$.

4. We now consider the case when $\mathbf{d}^{\Pi}_{\mathbf{x}_{i-1}}(\nabla f(\mathbf{x}_0)) \neq \mathbf{0}$ so that $\mathbf{x}_{i-1}$ is not the endpoint of the projections curve. Then, in this case $\hat{\lambda}_{i-1} < \infty$ as $\mathbf{x}_{i-1}$ is not the endpoint of $g(\lambda)$, and $\hat{\lambda}_{i-1} \geq \lambda^-_{i-1}$ by the definition of $\hat{\lambda}_{i-1}$. We now prove the different sub-cases in the $(a)$ and $(b)$ claimed in the theorem statement:

   $(a)$ **In-face movements:** *Recall that* $\hat{\lambda}_{i-1} := \sup\{\lambda \mid N_P(g(\lambda')) = N_P(\mathbf{x}_{i-1})\ \forall \lambda' \in [\lambda^-_{i-1}, \lambda)\}$ *and let* $\hat{\lambda}_{i-1} > \lambda^+_{i-1}$. *We will now show that* $\mathbf{x}_i := g(\hat{\lambda}_{i-1})$ *is in fact the next breakpoint. Also, that* $\mathbf{x}_i$ *is obtained by movement along the in-face direction, i.e.,* $\mathbf{x}_i = \mathbf{x}_{i-1} + (\hat{\lambda}_{i-1} - \lambda^+_{i-1})\hat{\mathbf{d}}^{\Pi}_{\mathbf{x}_{i-1}}(\nabla f(\mathbf{x}_0))$. *Moreover,* $N_P(\mathbf{x}_{i-1}) \subseteq N_P(g(\hat{\lambda}_{i-1}))$, *with strict containment only when the maximum movement along in-face direction takes place, i.e.,* $\hat{\lambda}_{i-1} = \lambda^+_{i-1} + \max\{\delta : \mathbf{x}_{i-1} + \delta\hat{\mathbf{d}}^{\Pi}_{\mathbf{x}_{i-1}}(\nabla f(\mathbf{x}_0)) \in P\}$. *This shows Theorem 1(iii).*

   Note that, the projections curve is linear between $\lambda^+_{i-1}$ and $\lambda^-_i$, since the normal cone is unchanged by definition of $\hat{\lambda}_{i-1}$, which shows Theorem 1(ii). Further, since $\hat{\lambda}_{i-1} > \lambda^+_{i-1}$, we know that when leaving the breakpoint $\mathbf{x}_{i-1}$ no constraint is dropped (hence, Theorem 1(i) holds), i.e., $N_P(\mathbf{x}_{i-1}) = N_P(g(\lambda))$, for $\lambda \in [\lambda^+_{i-1}, \hat{\lambda}_{i-1})$.

   **Proof of case** $(a)$**.** For notational brevity we will use $I$ to denote $I(\mathbf{x_{i-1}})$. We first claim that since the projections curve is continuous, $N_P(\mathbf{x}_{i-1}) \subseteq N_P(g(\hat{\lambda}_{i-1}))$:

   **Claim 4.1.** *Growth in normal cone* $\hat{\lambda}_{i-1}$: $N_P(\mathbf{x}_{i-1}) \subseteq N_P(g(\hat{\lambda}_{i-1}))$.

   *Pf.* We are given that $\hat{\lambda}_{i-1} > \lambda^+_{i-1}$. Now, suppose for a contradiction that $N_P(\mathbf{x}_{i-1}) \not\subseteq N_P(g(\hat{\lambda}_{i-1}))$. Then, there exists some $i \in I$ such that $\left\langle \mathbf{a}_i, g(\hat{\lambda}_{i-1})\right\rangle < b_i$. By the continuity of the projections curve (Lemma 10 with $\mathbf{y} := g(\hat{\lambda}_{i-1})$), we know that there exists some $\delta < \infty$ such that $\left\langle \mathbf{a}_i, g(\hat{\lambda}_{i-1} - \epsilon)\right\rangle < b_i$ for all $\epsilon \in (0, \delta)$. Consider any $\epsilon' \in (0, \min\{\hat{\lambda}_{i-1} - \lambda^+_{i-1}, \delta\})$ and define $\lambda' := \hat{\lambda}_{i-1} - \epsilon'$. Then $\lambda' \in (\lambda^+_{i-1}, \hat{\lambda}_{i-1})$ but $\langle \mathbf{a}_i, g(\lambda')\rangle < b_i$, i.e. $N_P(g(\lambda')) \neq N_P(\mathbf{x}_{i-1})$, which is a contradiction.

   **Claim 4.2.** *We will show that the projections curve in-face is given by* $\hat{\mathbf{d}}^{\Pi}_{\mathbf{x}_{i-1}}(\nabla f(\mathbf{x}_0))$ (58) *upto* $\hat{\lambda}_{i-1}$, *that is*
   $$g(\lambda) = \mathbf{x}_{i-1} + (\lambda - \lambda^-_{i-1})\hat{\mathbf{d}}^{\Pi}_{\mathbf{x}_{i-1}}(\nabla f(\mathbf{x}_0)) \quad \textit{for all } \lambda \in [\lambda^-_{i-1}, \hat{\lambda}_{i-1}]. \tag{51}$$

*Pf.* To show (51), note that we have $N_P(g(\lambda)) = N_P(\mathbf{x}_{i-1})$ for all $\lambda \in [\lambda_{i-1}^-, \hat{\lambda}_{i-1})$ and $N_P(g(\hat{\lambda}_{i-1})) \supseteq N_P(\mathbf{x}_{i-1})$ at the limit point using Claim 4.1 above. In other words, $g(\lambda)$ lies on the face defined by $\mathbf{A}_{I(\mathbf{x_{i-1}})} g(\lambda) = \boldsymbol{b}_{I(\mathbf{x_{i-1}})}$ for $\lambda \in [\lambda_{i-1}^-, \hat{\lambda}_{i-1}]$ until the projections curve is about to leave the minimal face. Thus, by reducing the projection optimization problem (51) to the optimal face (Lemma 9) and using the definition of $\hat{\lambda}_{i-1}$, we can write

$$g(\lambda) = \arg\min_{\mathbf{y}} \{\|\mathbf{x}_0 - \lambda \nabla f(\mathbf{x}_0) - \mathbf{y}\|^2 \mid \mathbf{A}_{I(\mathbf{x_{i-1}})}\mathbf{y} = \boldsymbol{b}_{I(\mathbf{x_{i-1}})}\} \quad \forall \lambda \in [\lambda_{i-1}^-, \hat{\lambda}_{i-1}]. \tag{52}$$

The least-squares optimization problem in (52) could be computed in closed form as follows (see Section 5.13 in [25] for example):

$$\begin{aligned}
g(\lambda) &= (\mathbf{I} - \mathbf{A}_I^\dagger \mathbf{A}_I)(\mathbf{x}_0 - \lambda \nabla f(\mathbf{x}_0)) + \mathbf{A}_I^\dagger b_I && (53)\\
&= \mathbf{x}_0 - \lambda \nabla f(\mathbf{x}_0) - \mathbf{A}_I^\dagger(\mathbf{A}_I(\mathbf{x}_0 - \lambda \nabla f(\mathbf{x}_0)) - \boldsymbol{b}_I) && (54)\\
&= \mathbf{x}_0 - \lambda_{i-1}^+ \nabla f(\mathbf{x}_0) - \mathbf{A}_I^\dagger(\mathbf{A}_I(\mathbf{x}_0 - \lambda_{i-1}^+ \nabla f(\mathbf{x}_0)) - \boldsymbol{b}_I) \\
&\qquad + (\mathbf{I} - \mathbf{A}_I^\dagger \mathbf{A}_I)((\lambda - \lambda_{i-1}^+) - \nabla f(\mathbf{x}_0)) && (55)\\
&= (\mathbf{I} - \mathbf{A}_I^\dagger \mathbf{A}_I)(\mathbf{x}_0 - \lambda_{i-1}^+ \nabla f(\mathbf{x}_0)) + \mathbf{A}_I^\dagger b_I \\
&\qquad + (\mathbf{I} - \mathbf{A}_I^\dagger \mathbf{A}_I)((\lambda - \lambda_{i-1}^+) - \nabla f(\mathbf{x}_0)) && (56)\\
&= \Pi_P(\mathbf{x}_0 - \lambda_{i-1}^+ \nabla f(\mathbf{x}_0)) + (\lambda - \lambda_{i-1}^+)(\mathbf{I} - \mathbf{A}_I^\dagger \mathbf{A}_I)(-\nabla f(\mathbf{x}_0)) && (57)\\
&= \mathbf{x}_{i-1} + (\lambda - \lambda_{i-1}^+)\hat{\mathbf{d}}_{\mathbf{x}_{i-1}}^{\Pi}(\nabla f(\mathbf{x}_0)) \quad \forall \lambda \in [\lambda_{i-1}^-, \hat{\lambda}_{i-1}], && (58)
\end{aligned}$$

where we used the equivalence of projection on the whole polytope $P$ and the projection onto the optimal face (Lemma 9) in (57).

**Claim 4.3** We claim that $\mathbf{x}_i := g(\hat{\lambda}_{i-1})$ is the next breakpoint, since the projections curve leaves the minimal face by definition of $\hat{\lambda}_{i-1}$, after this point, and therefore a direction change in the projections curve must happen (by Lemma 2).

Finally, to complete the proof of this case we show the following claim.

**Claim 4.4.** *We know show that $N_P(\mathbf{x}_{i-1}) \subset N_P(g(\hat{\lambda}_{i-1}))$ if and only if $\lambda_i^- = \lambda_{i-1}^+ + \max\{\delta : \mathbf{x}_{i-1} + \delta \hat{\mathbf{d}}_{\mathbf{x}_{i-1}}^{\Pi}(\nabla f(\mathbf{x}_0)) \in P\}$.*

*Pf.* First, note that $\hat{\mathbf{d}}_{\mathbf{x}_{i-1}}^{\Pi}(\nabla f(\mathbf{x}_0)) \neq \mathbf{0}$ from Claim 4.2.

Suppose $\lambda_i^- = \hat{\lambda}_{i-1} = \lambda_{i-1}^+ + \max\{\delta : \mathbf{x}_{i-1} + \delta \hat{\mathbf{d}}_{\mathbf{x}_{i-1}}^{\Pi}(\nabla f(\mathbf{x}_0)) \in P\}$. Since we cannot suddenly drop constraints at the limit point (Claim 4.1) we have $N_P(\mathbf{x}_{i-1}) \subseteq N_P(g(\hat{\lambda}_{i-1}))$. However this containment must be strict since $\hat{\lambda}_{i-1}^-$ corresponds to the maximum movement along $\hat{\mathbf{d}}_{\mathbf{x}_{i-1}}^{\Pi}(\nabla f(\mathbf{x}_0)) \neq \mathbf{0}$, and thus it follows that we add at least one tight constraint at $g(\hat{\lambda}_{i-1})$), i.e. $N_P(\mathbf{x}_{i-1}) \subset N_P(g(\hat{\lambda}_{i-1}))$.

Conversely, suppose that $N_P(\mathbf{x}_{i-1}) \subset N_P(g(\hat{\lambda}_{i-1}))$. Let $\delta^* := \max\{\delta : \mathbf{x}_{i-1} + \delta \hat{\mathbf{d}}_{\mathbf{x}_{i-1}}^{\Pi}(\nabla f(\mathbf{x}_0)) \in P\}$. Thus, we need to show that $\hat{\lambda}_{i-1} - \lambda_{i-1}^+ = \delta^*$.

First, using (58) we know that $g(\hat{\lambda}_{i-1}) = \mathbf{x}_{i-1} + (\hat{\lambda}_{i-1} - \lambda_{i-1}^+)\hat{\mathbf{d}}_{\mathbf{x}_{i-1}}^{\Pi}(\nabla f(\mathbf{x}_0)) \in P$, which implies by the definition of $\delta^*$ that $\delta^* \geq \lambda_{i-1}^+ - \hat{\lambda}_{i-1}$. We now show that $\delta^* \leq \hat{\lambda}_{i-1} - \lambda_{i-1}^+$, which proves the claim.

Since $N_P(\mathbf{x}_{i-1}) \subset N_P(g(\hat{\lambda}_{i-1}))$, we add at least one tight constraint $\mathbf{a}_j$ for some $j \in J$ (so that $\langle \mathbf{a}_j, \mathbf{x}_{i-1} \rangle < b_j$)) at the point $g(\hat{\lambda}_{i-1})$, i.e. $\left\langle \mathbf{a}_j, g(\hat{\lambda}_{i-1}) \right\rangle = b_j$. Since the movement to $g(\hat{\lambda}_{i-1})$ is in the direction $\hat{\mathbf{d}}_{\mathbf{x}_{i-1}}^{\Pi}(\nabla f(\mathbf{x}_0))$ (using (58)), it follows that $\left\langle \mathbf{a}_j, \hat{\mathbf{d}}_{\mathbf{x}_{i-1}}^{\Pi}(\nabla f(\mathbf{x}_0))) \right\rangle > 0$. This implies that $\mathbf{x}_{i-1} + (\lambda - \lambda_{i-1}^+)\hat{\mathbf{d}}_{\mathbf{x}_{i-1}}^{\Pi}(\nabla f(\mathbf{x}_0)) \notin P$ for $\lambda > \hat{\lambda}_{i-1}$, which yields $\delta^* \leq \hat{\lambda}_{i-1} - \lambda_{i-1}^+$.

**(b) Shadow movements:** $\hat{\lambda}_{i-1} = \lambda_{i-1}^+$**.** *if $\hat{\lambda}_{i-1} = \lambda_{i-1}^+$, then the movement is in the shadow direction, i.e., $\mathbf{x}_i := g(\lambda_i^-) = \mathbf{x}_{i-1} + (\lambda_i^- - \lambda_{i-1}^+)\mathbf{d}_{\mathbf{x}_{i-1}}^{\Pi}(\nabla f(\mathbf{x}_0))$ where $\lambda_i^- :=$*

$\lambda_{i-1}^+ + \max\{\delta : \mathbf{x}_{i-1} + \delta \mathbf{d}^{\Pi}_{\mathbf{x}_{i-1}}(\nabla f(\mathbf{x}_0)) \in P\}$. *In particular, the projections curve is linear between $\lambda_{i-1}^+$ and $\lambda_i^-$. Further, we show that properties (i), (ii) and (iii) in Theorem 1 hold for their respective normal cones for $\lambda, \lambda' \in (\lambda_{i-1}^+, \lambda_i^-)$, where the containments in (i) and (iii) are strict for case (b).*

*Proof of case* $(b)$. Finally, suppose that $\hat{\lambda}_{i-1} = \lambda_{i-1}^+$. Thus, we satisfy the conditions of the invariance property of orthogonal projections (Lemma 8) and will now invoke this property. By the invariance property of orthogonal projections (equations (29) and (50) in Lemma 8), we have $\tilde{g}(\lambda) = \mathbf{x}_{i-1} + (\lambda - \lambda_{i-1}^+)\mathbf{d}^{\Pi}_{\mathbf{x}_{i-1}}(\nabla f(\mathbf{x}_0)) = g(\lambda)$ for $\lambda \in [\lambda_{i-1}^+, \lambda_i^-]$. This shows that the projections curve is linear between $\lambda_{i-1}^+$ and $\lambda_i^-$. Moreover, using Theorem 2 with characterization in (50), we immediately have that $\mathbf{x}_i = g(\lambda_i^-) = \mathbf{x}_{i-1} + (\lambda_i^- - \lambda_{i-1}^+)\mathbf{d}^{\Pi}_{\mathbf{x}_{i-1}}(\nabla f(\mathbf{x}_0))$ and $N_P(g(\lambda)) = N_P(g(\lambda')) \subseteq N_P(\mathbf{x}_{i-1})$, for all $\lambda_{i-1}^+ < \lambda < \lambda' < \lambda_i^-$. In particular, this containment holds strictly since, as stated above, $\hat{\lambda}_{i-1} = \lambda_{i-1}^+$. This shows that the the first part of property $(i)$ ($N_P(\mathbf{x}_{i-1}) = N_P(g(\lambda_{i-1}^+)) \supseteq N_P(g(\lambda))$) and property $(ii)$ hold. Moreover, the containment in property $(iii)$ holds strictly by using $(ii)$ in Theorem 2 with $(\lambda - \lambda_{i-1}^+)$ in place $\lambda$ and $g(\lambda_{i-1}^+)$ in place of $\mathbf{x}_0$. Moreover the second part of property $(i)$, the drop in the normal cone, follows from (1).

This concludes the entire proof. □

### B.5 Number of Breakpoints in the Projections Curve

**Theorem 8** (Bound on breakpoints in parametric projection curve)**.** *Let $P \subseteq \mathbb{R}^n$ be a polytope, with $m$ facet inequalities (e.g., as in* (4)*) and fix $\mathbf{x} \in P$. Then, the procedure* TRACE$(\mathbf{x}, \nabla f(\mathbf{x}))$ *is correct and traces the projection piecewise linear curve $g(\lambda) = \Pi_P(\mathbf{x} - \lambda \nabla f(\mathbf{x}))$. Moreover, it terminates in at most $O(2^m)$ steps.*

*Proof.* Fix $\mathbf{x} \in P$ and consider the procedure TRACE$(\mathbf{x}, \nabla f(\mathbf{x}))$. The fact that TRACE$(\mathbf{x}, \nabla f(\mathbf{x}))$ correctly traces the curve $g(\lambda) = \Pi_P(\mathbf{x} - \lambda \nabla f(\mathbf{x}))$ follows from the constructive proof given for Theorem 3. Moreover, upon termination there are two things that can happen: (i) the line-search evaluates to a step-size that is not maximal in one of the segments of $g(\lambda)$, (ii) we reach the end point of the curve $\mathbf{v}^* = \lim_{\lambda \to \infty} g(\lambda)$ (as defined in Theorem 4). At this point, we know that $-\nabla f(\mathbf{x}) \in N_P(\mathbf{v}^*)$. Recall that for any $\mathbf{x} \in P$, we can compute the directional derivative using a single projection:

$$\mathbf{d}^{\Pi}_{\mathbf{x}} = \frac{\Pi_P(\mathbf{x} - \epsilon \nabla f(\mathbf{x})) - \mathbf{x}}{\epsilon},$$

where $\epsilon$ is sufficiently small. Thus, when recomputing the directional derivative in the subsequent iteration:

$$\frac{\Pi_P(\mathbf{v}^* - \epsilon \nabla f(\mathbf{x})) - \mathbf{v}^*}{\epsilon} = \frac{\mathbf{v}^* - \mathbf{v}^*}{\epsilon} = \mathbf{0},$$

where $\epsilon$ is again sufficiently small. The second equality above follows from the definition of $\mathbf{v}^*$ and Lemma 1. This proves the correctness of the termination criterion given in Algorithm 3.

Once the curve leaves the interior of a face, it can no longer visit the face again. This is because equivalence of normal cones at two projections implies the projections curve is linear between the two points (Lemma 2). Therefore, the number of breakpoints can be at most the number of faces, i.e., $O(2^m)$. □

We now discuss the implementation of step 2 in Algorithm 4 in what follows. First, following the discussion after the proof of Theorem 3 we check whether $\langle \mathbf{x}_0 - \lambda_{\mathbf{x}} \mathbf{d} - \mathbf{x}, \mathbf{d}^{\Pi}_{\mathbf{x}} \rangle = 0$. If this turned out to be true then we can skip steps 3-7 in algorithm 3 and take a shadow step, in which case we do need to enter Algorithm 4 to begin with.

Otherwise, if this is not the case, then we check whether we add a tight constraint when moving in-face by doing a line-search for feasibility in the direction $\hat{\mathbf{d}}^{\Pi}_{\mathbf{x}}$ when $\hat{\mathbf{d}}^{\Pi}_{\mathbf{x}} \neq \mathbf{0}$, i.e. we let $\hat{\gamma}^{\max} =$

**Algorithm 3** Tracing Parametric Projections Curve: TRACE$(\mathbf{x}, \nabla f(\mathbf{x}))$

**Input:** Polytope $P \subseteq \mathbb{R}^n$, function $f : P \to \mathbb{R}$ and initialization $\mathbf{x} \in P$.

Let $\mathbf{d} = \nabla f(\mathbf{x})$, $\mathbf{x}_0 = \mathbf{x}$ and $\gamma^{\text{total}} = 0$. ▷ `fix gradient and starting point`
**while** True **do**
  $\hat{\mathbf{d}}^{\Pi}_{\mathbf{x}}, \hat{\lambda}$ = TRACE-IN-FACE$(\mathbf{x}_0, \mathbf{x}, \mathbf{d}, \gamma^{\text{total}})$.
  **if** $\hat{\lambda} > \gamma^{\text{total}}$ **then** ▷ `case (a.ii) in Theorem 3`
    Let $\gamma^{\max} = \hat{\lambda}$ and $\gamma^* \in \arg\min_{\gamma \in [0, \gamma^{\max}]} f(\mathbf{x} + \gamma \hat{\mathbf{d}}^{\Pi}_{\mathbf{x}})$. ▷ `check line-search solution`
    Update $\mathbf{x} = \mathbf{x} + \gamma^* \hat{\mathbf{d}}^{\Pi}_{\mathbf{x}}$. ▷ `projections curve moves in-face`
  **else**
    Compute $\mathbf{d}^{\Pi}_{\mathbf{x}} := \lim_{\epsilon \downarrow 0} \frac{\Pi_P(\mathbf{x} - \epsilon \nabla f(\mathbf{x})) - \mathbf{x}}{\epsilon}$ and let $\gamma^{\max} = \max\{\delta \mid \mathbf{x} + \delta \mathbf{d}^{\Pi}_{\mathbf{x}} \in P\}$.
    $\gamma^* \in \arg\min_{\gamma \in [0, \gamma^{\max}]} f(\mathbf{x} + \gamma \mathbf{d}^{\Pi}_{\mathbf{x}})$. ▷ `check optimality of line-search solution`
    Update $\mathbf{x} = \mathbf{x} + \gamma^* \mathbf{d}^{\Pi}_{\mathbf{x}}$. ▷ `invariance of projections gives next curve segment`
  **end if**
  Recompute $\mathbf{d}^{\Pi}_{\mathbf{x}} := \lim_{\epsilon \downarrow 0} \frac{\Pi_P(\mathbf{x} - \epsilon \mathbf{d}) - \mathbf{x}}{\epsilon}$.
  Update $\gamma^{\text{total}} = \gamma^{\text{total}} + \gamma^*$. ▷ `keeping track of total step-size accrued`
  **if** $\gamma^* \leq \gamma^{\max}$ **or** $\mathbf{d}^{\Pi}_{\mathbf{x}} = 0$ **then** ▷ `either sufficient progress or we reached endpoint`
    **break** ▷ `suffices to also terminate when` $\gamma^{\text{total}} \geq 1/L$
  **end if**
**end while**

**Return:** $\mathbf{x}$

**Algorithm 4** Tracing Parametric Projections Curve In-face: TRACE-IN-FACE$(\mathbf{x}_0, \mathbf{x}, \mathbf{d}, \lambda_{\mathbf{x}})$

**Input:** Polytope $P \subseteq \mathbb{R}^n$, starting point of projections curve $\mathbf{x}_0 \in P$, current breakpoint $\mathbf{x} \in P$ and $\lambda_{\mathbf{x}}$ satisfying $g(\lambda_{\mathbf{x}}) := \Pi_P(\mathbf{x}_0 - \lambda_{\mathbf{x}} \mathbf{d}) = \mathbf{x}$.

Compute $\hat{\mathbf{d}}^{\Pi}_{\mathbf{x}} = (\mathbf{I} - \mathbf{A}^{\top}_{I(\mathbf{x})}(\mathbf{A}_{I(\mathbf{x})}\mathbf{A}^{\top}_{I(\mathbf{x})})^{\dagger}\mathbf{A}_{I(\mathbf{x})})(-\mathbf{d})$ ▷ `project` $-\mathbf{d}$ `onto minimal face of` $\mathbf{x}$
Evaluate $\hat{\lambda} = \sup\{\lambda \mid N_P(g(\lambda')) = N_P(\mathbf{x}_{i-1}) \ \forall \lambda' \in [\lambda_{\mathbf{x}}, \lambda)\} = \max\{\lambda \mid g(\lambda) = \mathbf{x} + (\lambda - \lambda_{\mathbf{x}})\hat{\mathbf{d}}^{\Pi}_{\mathbf{x}}\}$ ▷ `This gives` $\hat{\lambda}$ `same as the one defined in Theorem 3. See text for an implementation of this step`

**Return:** $\hat{\mathbf{d}}^{\Pi}_{\mathbf{x}}, \hat{\lambda}$

$\max\{\delta \mid \mathbf{x} + \delta \hat{\mathbf{d}}^{\Pi}_{\mathbf{x}} \in P\}$. Then we can check whether $g(\lambda_{\mathbf{x}} + \hat{\gamma}^{\max}) = \mathbf{x} + \hat{\gamma}^{\max} \hat{\mathbf{d}}^{\Pi}_{\mathbf{x}}$ or not. This amounts to checking the first-order optimality condition:

$$\left\langle \mathbf{x}_0 - (\lambda_{\mathbf{x}} + \hat{\gamma}^{\max})\mathbf{d} - (\mathbf{x} + \hat{\gamma}^{\max}\hat{\mathbf{d}}^{\Pi}_{\mathbf{x}}), \mathbf{z} - (\mathbf{x} + \hat{\gamma}^{\max}\hat{\mathbf{d}}^{\Pi}_{\mathbf{x}}) \right\rangle \leq 0 \quad \forall \mathbf{z} \in P, \tag{59}$$

which could be done by solving a linear program that maximizes the above inner product over all $\mathbf{z} \in P$, and then checking whether the objective value is non-positive. If this turned out to be true, then we are done.

Otherwise, we know that $\hat{\lambda} < \hat{\gamma}^{\max}$ and thus $\hat{\lambda} = \max\{\lambda \mid N_P(g(\lambda')) = N_P(\mathbf{x}) \ \forall \lambda' \in [\lambda_{\mathbf{x}}, \lambda]\}$. Furthermore, using Theorem 3, we know that $g(\lambda) = \mathbf{x} + (\lambda - \lambda_{\mathbf{x}})\hat{\mathbf{d}}^{\Pi}_{\mathbf{x}}$ for all $\lambda \in [\lambda_{\mathbf{x}}, \hat{\lambda}]$. We can thus do a binary search until this condition is satisfied.

# C Missing Proofs for Section 4 on Descent Directions

## C.1 Proof of Lemma 3

**Lemma 3** (Local Optimality of Shadow Steps)**.** *Let $P$ be a polytope defined as in* (4) *and let $\mathbf{x} \in P$ with gradient $\nabla f(\mathbf{x})$. Let $\mathbf{y}$ be any feasible direction at $\mathbf{x}$, i.e., $\exists \gamma > 0$ s.t. $\mathbf{x} + \gamma \mathbf{y} \in P$. Then*

$$\left\langle -\nabla f(\mathbf{x}), \frac{\mathbf{d}^{\Pi}_{\mathbf{x}}}{\|\mathbf{d}^{\Pi}_{\mathbf{x}}\|} \right\rangle^2 = \|\mathbf{d}^{\Pi}_{\mathbf{x}}\|^2 \geq \left\langle \mathbf{d}^{\Pi}_{\mathbf{x}}, \frac{\mathbf{y}}{\|\mathbf{y}\|} \right\rangle^2 \geq \left\langle -\nabla f(\mathbf{x}), \frac{\mathbf{y}}{\|\mathbf{y}\|} \right\rangle^2. \tag{9}$$

*Proof.* We prove the result using first-order optimality of projections. First, observe that we can uniquely decompose $-\nabla f(\mathbf{x}) = \mathbf{p} - \nabla f(\mathbf{x})_P$ such that $\langle -\nabla f(\mathbf{x})_P, \mathbf{p} \rangle = 0$, where $\nabla f(\mathbf{x})_P$ is the component of $\nabla f(\mathbf{x})$ projected onto the cone of feasible directions at $\mathbf{x}$, and $\mathbf{p}$ is the orthogonal component. Recall from Section 2 that $\mathbf{d}^{\Pi}_{\mathbf{x}} = \arg\min_{\mathbf{d}}\{\| - \nabla f(\mathbf{x}) - \mathbf{d}\|^2 : A_{I(\mathbf{x})}\mathbf{d} \leq 0\}$, and so by definition we have $-\nabla f(\mathbf{x})_P = \mathbf{d}^{\Pi}_{\mathbf{x}}$. Therefore, $\langle -\nabla f(\mathbf{x}), \mathbf{d}^{\Pi}_{\mathbf{x}} \rangle = \|\mathbf{d}^{\Pi}_{\mathbf{x}}\|^2$. This gives the first equality in (9).

We will now show that

$$\left\langle \mathbf{d}_{\mathbf{x}}^{\Pi}, \mathbf{y} \right\rangle \geq \left\langle -\nabla f(\mathbf{x}), \mathbf{y} \right\rangle. \tag{60}$$

To do that, we recall the first-order optimality condition for $g(\lambda) = \Pi_P(\mathbf{x} - \lambda \nabla f(\mathbf{x}))$ for $\lambda > 0$:

$$\langle g(\lambda) - \mathbf{x} + \lambda \nabla f(\mathbf{x}), \mathbf{z} - g(\lambda) \rangle \geq 0 \quad \forall\, \mathbf{z} \in P.$$

Using Theorem 2, we know that there exists some scalar $\lambda^-$ such that $g(\lambda) = \mathbf{x} + \lambda \mathbf{d}_{\mathbf{x}}^{\Pi}$ for any $0 < \lambda < \lambda^-$. Hence, for any such $\lambda \in (0, \lambda^-)$, the first-order optimality condition becomes:

$$\left\langle \mathbf{x} + \lambda \mathbf{d}_{\mathbf{x}}^{\Pi} - \mathbf{x} + \lambda \nabla f(\mathbf{x}), \mathbf{z} - \mathbf{x} - \lambda \mathbf{d}_{\mathbf{x}}^{\Pi} \right\rangle = \lambda \left\langle \mathbf{d}_{\mathbf{x}}^{\Pi} + \nabla f(\mathbf{x}), \mathbf{z} - \mathbf{x} - \lambda \mathbf{d}_{\mathbf{x}}^{\Pi} \right\rangle \geq 0, \tag{61}$$

for all $\mathbf{z} \in P$. Note that the above equation holds for any $\mathbf{z} \in P$ and $\lambda \in (0, \lambda^-)$.

Since, $\mathbf{x} + \gamma \mathbf{y} \in P$, it follows that $\mathbf{x} + \bar{\lambda}\mathbf{y}$ is also in $P$, where $\bar{\lambda} = \min\{\lambda^-/2, \gamma\}$. Thus, since $\bar{\lambda} \in (0, \lambda^-)$ and $\mathbf{x} + \bar{\lambda}\mathbf{y} \in P$, we can plug in $\bar{\lambda}$ for $\lambda$ and $\mathbf{x} + \bar{\lambda}\mathbf{y}$ for $\mathbf{z}$ in (61) to obtain $\bar{\lambda}^2 \left\langle \mathbf{d}_{\mathbf{x}}^{\Pi} + \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{d}_{\mathbf{x}}^{\Pi} \right\rangle \geq 0$. Thus, using the fact that $\left\langle -\nabla f(\mathbf{x}), \mathbf{d}_{\mathbf{x}}^{\Pi} \right\rangle = \|\mathbf{d}_{\mathbf{x}}^{\Pi}\|^2$, this implies

$$\left\langle \mathbf{d}_{\mathbf{x}}^{\Pi}, \mathbf{y} \right\rangle \geq \|\mathbf{d}_{\mathbf{x}}^{\Pi}\|^2 + \left\langle -\nabla f(\mathbf{x}), \mathbf{y} - \mathbf{d}_{\mathbf{x}}^{\Pi} \right\rangle = \left\langle -\nabla f(\mathbf{x}), \mathbf{y} \right\rangle$$

as claimed in (60).

We can now complete the proof using (60) as follows

$$\begin{aligned}
\left\langle -\nabla f(\mathbf{x}), \frac{\mathbf{d}_{\mathbf{x}}^{\Pi}}{\|\mathbf{d}_{\mathbf{x}}^{\Pi}\|} \right\rangle^2 &= \|\mathbf{d}_{\mathbf{x}}^{\Pi}\|^2 && \text{(definition of } \mathbf{d}_{\mathbf{x}}^{\Pi}\text{)} \\
&\geq \left\langle \mathbf{d}_{\mathbf{x}}^{\Pi}, \frac{\mathbf{y}}{\|\mathbf{y}\|} \right\rangle^2 && \text{(Cauchy-Schwartz)} \\
&\geq \left\langle -\nabla f(\mathbf{x}), \frac{\mathbf{y}}{\|\mathbf{y}\|} \right\rangle^2, && \text{(using (60))}
\end{aligned}$$

which concludes the proof. □

### C.2 Using the derivative of the projection operator to estimate primal gaps

We will now show that $\|\mathbf{d}_{\mathbf{x}_t}^{\Pi}\| = 0$ if and only if $\mathbf{x}_t = \mathbf{x}^*$. On the other hand, note that, e.g., $\|\nabla f(\mathbf{x}_t)\|$ does not satisfy this property and can be strictly positive at the constrained optimal solution. Hence, $\|\mathbf{d}_{\mathbf{x}_t}^{\Pi}\|$ is a natural quantity to use for estimating primal gaps without any dependence on geometric constants like those used in CG variants.

**Lemma 12** (Primal Gap Estimate)**.** *Let $P$ be a polytope defined as in* (4) *and fix $\mathbf{x} \in P$. Let $g(\lambda) = \Pi_P(\mathbf{x} - \lambda \nabla f(\mathbf{x}))$ be the curve parameterized by the step-size $\lambda$. Then, $\|\mathbf{d}_{\mathbf{x}}^{\Pi}\| = 0$ if and only if $\mathbf{x} = \mathbf{x}^*$, where $\mathbf{x}^* = \arg\min_{\mathbf{x} \in P} f(\mathbf{x})$.*

*Proof.* First assume that $\|\mathbf{d}_{\mathbf{x}}^{\Pi}\| = 0$ so that $\mathbf{d}_{\mathbf{x}}^{\Pi} = \mathbf{0}$. From (26) in Corollary 1, we know that $\mathbf{d}_{\mathbf{x}}^{\Pi} = \frac{g(\epsilon) - \mathbf{x}}{\epsilon}$ for $\epsilon > 0$ sufficiently small. Hence, the assumption that $\mathbf{d}_{\mathbf{x}}^{\Pi} = \mathbf{0}$ implies that $g(\epsilon) = \mathbf{x}$. Using the first-order optimality of $g(\epsilon)$ we have

$$\langle \mathbf{x} - \epsilon \nabla f(\mathbf{x}) - g(\epsilon), \mathbf{z} - g(\epsilon) \rangle \leq 0 \quad \forall \mathbf{z} \in P.$$

However, since $g(\epsilon) = \mathbf{x}$, this becomes

$$\langle -\epsilon \nabla f(\mathbf{x}), \mathbf{z} - \mathbf{x} \rangle \leq 0 \quad \forall \mathbf{z} \in P.$$

In other words, this is equivalent to saying $-\nabla f(\mathbf{x}) \in N_P(\mathbf{x})$, so that $\mathbf{x} = \mathbf{x}^*$.

Conversely suppose that $\mathbf{x} = \mathbf{x}^*$. Then, it follows that $-\nabla f(\mathbf{x}) \in N_P(\mathbf{x})$. Using Lemma 1, this implies that $g(\lambda) = \mathbf{x}$ for all $\lambda > 0$. Since from (8) we know that $\mathbf{d}_{\mathbf{x}}^{\Pi} = \frac{g(\epsilon) - \mathbf{x}}{\epsilon}$ for $\epsilon > 0$ sufficiently small, it follows that $\mathbf{d}_{\mathbf{x}}^{\Pi} = \mathbf{0}$. Thus, $\|\mathbf{d}_{\mathbf{x}}^{\Pi}\| = 0$ as desired. □

To prove convergence results for our algorithms, we additionally need dual gap bound on $h(\mathbf{x}_t) := f(\mathbf{x}_t) - f(\mathbf{x}^*)$ using $\mathbf{d}_{\mathbf{x}_t}^{\Pi}$. To do this, consider the strong convexity inequality given in Section 2 with $\mathbf{y} \leftarrow \mathbf{x}_t + \gamma(\mathbf{x}^* - \mathbf{x}_t)$ and $\mathbf{x} \leftarrow \mathbf{x}_t$:

$$f(\mathbf{x}_t + \gamma(\mathbf{x}^* - \mathbf{x}_t)) - f(\mathbf{x}_t) \geq \gamma \left\langle \nabla f(\mathbf{x}_t), \mathbf{x}^* - \mathbf{x}_t \right\rangle + \frac{\mu \gamma^2 \|\mathbf{x}^* - \mathbf{x}_t\|^2}{2}.$$

The RHS is convex in $\gamma$ and is minimized when $\gamma^* = \frac{\langle -\nabla f(\mathbf{x}_t), \mathbf{x}^* - \mathbf{x}_t \rangle}{\mu \|\mathbf{x}^* - \mathbf{x}_t\|^2}$. Plugging $\gamma^*$ in the above expression and re-arranging we obtain

$$f(\mathbf{x}_t + \gamma(\mathbf{x}^* - \mathbf{x}_t)) - f(\mathbf{x}^*) \leq \frac{\langle -\nabla f(\mathbf{x}_t), \mathbf{x}^* - \mathbf{x}_t \rangle^2}{2\mu \|\mathbf{x}^* - \mathbf{x}_t\|^2}.$$

As the LHS is independent of $\gamma$, we can set $\gamma = 1$, which gives

$$h(\mathbf{x}_t) := f(\mathbf{x}_t) - f(\mathbf{x}^*) \leq \frac{\langle -\nabla f(\mathbf{x}_t), \mathbf{x}^* - \mathbf{x}_t \rangle^2}{2\mu \|\mathbf{x}^* - \mathbf{x}_t\|^2}. \tag{62}$$

Now using Lemma 3 with $\mathbf{x}^* - \mathbf{x}_t$ substituted for $\mathbf{y}$ in (9) we obtain

$$\left\langle -\nabla f(\mathbf{x}_t), \frac{\mathbf{d}_{\mathbf{x}_t}^{\Pi}}{\|\mathbf{d}_{\mathbf{x}_t}^{\Pi}\|} \right\rangle^2 \geq \left\langle -\nabla f(\mathbf{x}_t), \frac{\mathbf{x}^* - \mathbf{x}_t}{\|\mathbf{x}^* - \mathbf{x}_t\|} \right\rangle^2 \geq 2\mu h(\mathbf{x}_t), \tag{63}$$

where the last inequality follows from (62).

### C.3 Relating projections to FW vertices

**Theorem 4** (Optimism in Frank-Wolfe Vertices)**.** *Let $P \subseteq R^n$ be a polytope and let $\mathbf{x} \in P$. Let $g(\lambda) = \Pi_P(\mathbf{x} - \lambda \nabla f(\mathbf{x}))$ for $\lambda \geq 0$. Then, the end point of this curve is: $\lim_{\lambda \to \infty} g(\lambda) = \mathbf{v}^* = \arg\min_{\mathbf{v} \in F} \|\mathbf{x} - \mathbf{v}\|^2$, where $F = \arg\min_{\mathbf{v} \in P} \langle \nabla f(\mathbf{x}), \mathbf{v} \rangle$, i.e., the face of $P$ that minimizes the gradient $\nabla f(\mathbf{x})$. In particular, if $F$ is a vertex, then $\lim_{\lambda \to \infty} g(\lambda) = \mathbf{v}^*$ is the Frank-Wolfe vertex.*

*Proof.* If $\nabla f(\mathbf{x}) = 0$, then $g(\lambda) = \mathbf{x}$ for all $\lambda \in \mathbb{R}^n$, and the theorem holds trivially. We therefore assume that $\nabla f(\mathbf{x}) \neq 0$. Let $\mathbf{x}_i \in P$ be the $i$th breakpoint in the projections curve $g(\lambda) = \Pi_P(\mathbf{x}_0 - \lambda \nabla f(\mathbf{x}_0))$, with $\mathbf{x}_i = \mathbf{x}$ for $i = 0$. Using Theorem 8, we know that the number of breakpoints curve $k \leq 2^m$. Consider the last breakpoint $\mathbf{x}_k$ in the curve and let $\lambda_k^- = \min\{\lambda \geq 0 \mid g(\lambda) = \mathbf{x}_k\}$. We will now show that $\mathbf{x}_k = \mathbf{v}^*$.

(i) We first show that $\mathbf{x}_k \in F$, i.e. $-\nabla f(\mathbf{x}) \in N_P(\mathbf{x}_k)$. Suppose for a contradiction that this not true. Then there exists some $\mathbf{z} \in P$ such that $\langle -\nabla f(\mathbf{x}), \mathbf{z} - \mathbf{x}_k \rangle > 0$. Consider any scalar $\bar{\lambda}$ satisfying $\bar{\lambda} > \max\{-\frac{\langle \mathbf{x} - \mathbf{x}_k, \mathbf{z} - \mathbf{x}_k \rangle}{\langle -\nabla f(x), \mathbf{z} - \mathbf{x}_k \rangle}, \lambda_k^-\}$. Then, using the choice of $\bar{\lambda}$ we have

$$\begin{aligned} &\langle \mathbf{x} - \mathbf{x}_k, \mathbf{z} - \mathbf{x}_k \rangle + \bar{\lambda} \langle -\nabla f(\mathbf{x}), \mathbf{z} - \mathbf{x}_k \rangle > 0 \\ &\implies \left\langle \mathbf{x} - \mathbf{x}_k - \bar{\lambda} \nabla f(\mathbf{x}), \mathbf{z} - \mathbf{x}_k \right\rangle > 0. \end{aligned}$$

Now, since $g(\lambda) = \mathbf{x}_k$ for $\lambda \geq \lambda_k^-$, we know that $g(\bar{\lambda}) = \mathbf{x}_k$. Thus, the above equation could be written as

$$\left\langle \mathbf{x} - \bar{\lambda} \nabla f(\mathbf{x}) - g(\bar{\lambda}), \mathbf{z} - g(\bar{\lambda}) \right\rangle > 0,$$

which contradicts the first-order optimality for $g(\bar{\lambda})$.

(ii) We will now show that $\mathbf{x}_k$ is additionally the closest point to $\mathbf{x}$ in $\ell_2$ norm. Again, suppose for contradiction that this not true. Let $\epsilon := \|\mathbf{x}_k - \mathbf{v}^*\| > 0$. First, note that by definition, $g(\lambda) = \arg\min_{\mathbf{y} \in P} \left\{ \frac{\|\mathbf{x} - \mathbf{y}\|^2}{2\lambda} + \langle \nabla f(\mathbf{x}), \mathbf{y} \rangle \right\}$ for any $\lambda > 0$. Then, since $g(\lambda_k^-) = \mathbf{x}_k$ we have

$$\frac{\|\mathbf{x} - \mathbf{x}_k\|^2}{2\lambda_k^-} + \langle \nabla f(\mathbf{x}), \mathbf{x}_k \rangle \leq \frac{\|\mathbf{x} - \mathbf{z}\|^2}{2\lambda_k^-} + \langle \nabla f(\mathbf{x}), \mathbf{z} \rangle, \ \ \forall \mathbf{z} \in P. \tag{64}$$

Moreover, the first-order optimality condition for $\mathbf{v}^*$ (for minimizing $\|\mathbf{x} - \mathbf{y}\|^2$ over $\mathbf{y} \in F$) implies $\langle \mathbf{v}^* - \mathbf{x}, \mathbf{z} - \mathbf{v}^* \rangle \geq 0$ for all $\mathbf{z} \in F$. In particular, $(\mathbf{v}^* - \mathbf{x})^T (\mathbf{x}_k - \mathbf{v}^*) \geq 0$ since $\mathbf{x}_k \in F$. Therefore,

$$\begin{aligned} \|\mathbf{x} - \mathbf{v}^*\|^2 + \|\mathbf{x}_k - \mathbf{v}^*\|^2 &= \|\mathbf{x}\|^2 + 2\|\mathbf{v}^*\|^2 - 2\mathbf{x}^T \mathbf{v}^* + \|\mathbf{x}_k\|^2 - 2\mathbf{x}_k^T \mathbf{v}^* && (65) \\ &= \|\mathbf{x}_k - \mathbf{x}\|^2 - 2(\mathbf{v}^* - \mathbf{x})^T (\mathbf{x}_k - \mathbf{v}^*) && (66) \\ &\leq \|\mathbf{x}_k - \mathbf{x}\|^2. && (67) \end{aligned}$$

But then, since $\mathbf{x}_k \in F$, we know that $\langle \nabla f(\mathbf{x}), \mathbf{x}_k \rangle = \langle \nabla f(\mathbf{x}), \mathbf{v}^* \rangle$, which implies

$$\begin{aligned}
\frac{\|\mathbf{x} - \mathbf{v}^*\|^2}{2\lambda_k^-} + \langle \nabla f(\mathbf{x}), \mathbf{v}^* \rangle &\leq \frac{\|\mathbf{x}_k - \mathbf{x}\|^2 - \|\mathbf{x}_k - \mathbf{v}^*\|^2}{2\lambda_k^-} + \langle \nabla f(\mathbf{x}), \mathbf{v}^* \rangle \quad \text{(using (67))} \\
&= \frac{\|\mathbf{x}_k - \mathbf{x}\|^2 - \epsilon}{2\lambda_k^-} + \langle \nabla f(\mathbf{x}), \mathbf{v}^* \rangle \qquad (\|\mathbf{x}_k - \mathbf{v}^*\| = \epsilon) \\
&< \frac{\|\mathbf{x}_k - \mathbf{x}\|^2}{2\lambda_k^-} + \langle \nabla f(\mathbf{x}), \mathbf{x}_k \rangle , \qquad (\epsilon > 0)
\end{aligned}$$

contradicting optimality of $\mathbf{x}_k$ (64).

□

### C.4 Connecting Shadow-steps to Away-steps

**Lemma 4** (Away-Steps)**.** *Let $P$ be a polytope defined as in* (4) *and fix $\mathbf{x} \in P$. Let $F = \{\mathbf{z} \in P : \mathbf{A}_{I(\mathbf{x})}\mathbf{z} = \mathbf{b}_{I(\mathbf{x})}\}$ be the minimal face containing $\mathbf{x}$. Further, choose $\delta_{\max} = \max\{\delta : \mathbf{x} - \delta \mathbf{d}_{\mathbf{x}}^{\Pi} \in P\}$ and consider the maximal backward away point $\mathbf{a}_{\mathbf{x}} = \mathbf{x} - \delta_{\max}\mathbf{d}_{\mathbf{x}}^{\Pi}$. Then, $\mathbf{a}_{\mathbf{x}}$ lies in $F$ and the corresponding away-direction is simply $\mathbf{x} - \mathbf{a}_{\mathbf{x}} = \delta_{\max}\mathbf{d}_{\mathbf{x}}^{\Pi}$.*

We first recall this result from Bashiri and Zhang [3]:

**Lemma 13** (Best away vertex, [3])**.** *Let $P$ be a polytope defined as in* (4) *and fix $\mathbf{x} \in P$. Let $F = \{\mathbf{z} \in P : \mathbf{A}_{I(\mathbf{x})}\mathbf{z} = \mathbf{b}_{I(\mathbf{x})}\}$ be the minimal face containing $\mathbf{x}$ and define $A := \{\mathbf{v} \in \operatorname{vert}(P) : \mathbf{v} \in F\}$ to be the set of vertices in $F$. Also, let $\mathcal{S}(\mathbf{x}) := \{S : S \subseteq \operatorname{vert}(P) \mid \mathbf{x}$ is a proper convex combination of all the elements in $S\}$ be the set of all possible active sets for $\mathbf{x}$. Then,*

$$\max_{\mathbf{v} \in A} \langle \nabla f(\mathbf{x}), \mathbf{v} \rangle = \max_{S \in \mathcal{S}(\mathbf{x})} \max_{\mathbf{v} \in S} \langle \nabla f(\mathbf{x}), \mathbf{v} \rangle .$$

*Proof.* For the first direction, we claim that any $S \in \mathcal{S}(\mathbf{x})$ must be contained in $A = \operatorname{vert}(F)$. Let $S \in \mathcal{S}(\mathbf{x})$. Then, we can write $\mathbf{x} = \sum_{\mathbf{v} \in S} \alpha_{\mathbf{v}} \mathbf{v}$, where $\alpha_{\mathbf{v}} \in (0, 1)$ and $\sum_{\mathbf{v} \in S} \alpha_{\mathbf{v}} = 1$. Fix $\mathbf{y} \in S$ and let $\mathbf{z} := \frac{1}{1-\alpha_{\mathbf{y}}} \sum_{\mathbf{v} \in S \setminus \{\mathbf{y}\}} \alpha_{\mathbf{v}} \mathbf{v} \in P$. Then, $\mathbf{x} = \alpha_{\mathbf{y}}\mathbf{y} + (1 - \alpha_{\mathbf{y}})\mathbf{z}$. Now, if $\langle \mathbf{a}_i, \mathbf{x} \rangle = b_i$, then the fact that $\langle \mathbf{a}_i, \mathbf{z} \rangle \leq b_i$ implies that $\langle \mathbf{a}_i, \mathbf{y} \rangle = b_i$, so that $\mathbf{y} \in A$.

Conversely, we claim that any $\mathbf{v} \in A$ lies in some $S \in \mathcal{S}(\mathbf{x})$. Let $\mathbf{v} \in A$. Consider $\mathbf{z}_\alpha = \frac{1}{1-\alpha}(\mathbf{x} - \alpha\mathbf{v})$ for $\alpha \in (0, 1)$. First, if $\langle \mathbf{a}_i, \mathbf{x} \rangle = b_i$ (i.e. $i \in I(\mathbf{x})$), since we have $\langle \mathbf{a}_i, \mathbf{v} \rangle = b_i$ by choice of $\mathbf{v}$, it follows that $\langle \mathbf{a}_i, \mathbf{z}_\alpha \rangle = b_i$. Otherwise, if $\langle \mathbf{a}_i, \mathbf{x} \rangle < b_i$ (i.e. $i \in J(\mathbf{x})$) then $\lim_{\alpha \downarrow 0} \langle \mathbf{a}_i, \mathbf{z}_\alpha \rangle = \langle \mathbf{a}_i, \mathbf{x} \rangle < b_i$. Thus, since we have a finite number of constraints, we can ensure that $\langle \mathbf{a}_i, \mathbf{z}_{\alpha^*} \rangle \leq b_i$ for all $i \in J(\mathbf{x})$, where $\alpha^*$ is sufficiently small. Thus, we have shown we can write $\mathbf{x}_t = (1 - \alpha^*)\mathbf{z}_{\alpha^*} + \alpha^*\mathbf{v}$, where $\mathbf{z}_{\alpha^*} \in P$. Therefore, there exists some active $S \in \mathcal{S}(\mathbf{x}_t)$ containing $\mathbf{v}$.

□

**Proof of Lemma 4.** First, if $\delta_{\max} = 0$, then $\mathbf{a}_t = \mathbf{x}_t$, and the result holds trivially. Now assume that $\delta_{\max} > 0$. By definition of $\mathbf{d}_{\mathbf{x}}^{\Pi}$, we know that $\mathbf{A}_{I(\mathbf{x})}\mathbf{d}_{\mathbf{x}}^{\Pi} \leq \mathbf{0}$. Hence, since $-\mathbf{d}_{\mathbf{x}}^{\Pi}$ is also feasible, it follows that we must have $\mathbf{A}_{I(\mathbf{x})}\mathbf{d}_{\mathbf{x}}^{\Pi} = 0$. This then implies that $\mathbf{A}_{I(\mathbf{x})}\mathbf{a}_{\mathbf{x}} = \mathbf{A}_{I(\mathbf{x})}(\mathbf{x} - \delta_{\max}\mathbf{d}_{\mathbf{x}}^{\Pi}) = \mathbf{A}_{I(\mathbf{x})}\mathbf{x} = \mathbf{b}_{I(\mathbf{x})}$. Thus, we have $\mathbf{a}_{\mathbf{x}} \in F$. Moreover, in the proof of the previous lemma (Lemma 13), we show that the vertices of $F$ in fact form all possible away steps. The result then follows. □

## D Continuous-time Dynamics and SHADOW-WALK Algorithm

We now present the continuous-time dynamics for moving along the shadow of the gradient in the polytope. In this section we let $\mathcal{D}^*$ be the dual space of $\mathcal{D}$ (in our case since $\mathcal{D} \subseteq \mathbb{R}^n$, $\mathcal{D}^*$ can also be identified with $\mathbb{R}^n$). Let $\phi : \mathcal{D} \to \mathbb{R}$ be a strongly convex and differentiable function. This function will be used as the *mirror-map* in a generalization of projected gradient descent algorithm, known as *mirror descent* [8]. Let $\phi^*$ be the Fenchel-conjugate of $\phi$ with effective domain

$P$, that is $\phi^*(\mathbf{y}) = \max_{\mathbf{x}\in P}\{\langle \mathbf{y}, \mathbf{x}\rangle - \phi(\mathbf{x})\}$. From Danskin's theorem (see e.g., [29]), we know that $\nabla\phi^*(\mathbf{y}) = \arg\max_{\mathbf{x}\in P}\{\langle \mathbf{y}, \mathbf{x}\rangle - \phi(\mathbf{x})\}$, so that $\nabla\phi^* : \mathcal{D}^* \to P$ is the mirror-map operator mapping from $\mathcal{D}^*$ to $\mathcal{D}$. We use $\nabla^2\phi^*(\cdot)$ to denote the Hessian of $\phi^*$.

### D.1 ODE for moving in the shadow of gradient

Let $X(t)$ denote the continuous-time trajectory of our dynamics and $\dot{X}$ denote the time-derivative of $X(t)$, i.e., $\dot{X}(t) = \frac{d}{dt}X(t)$. In [30], Krichene et. al propose the following coupled dynamics $(X(t), Z(t))$ for mirror descent, where $X(t)$ evolves in the primal space $\mathcal{D}$, and $Z(t)$ evolves in the dual space $\mathcal{D}^*$, as follows, initialized with $Z(0) = \mathbf{z}_0$ with $\nabla\phi^*(\mathbf{z}_0) = \mathbf{x}_0 \in P$:

$$\dot{Z}(t) = -\nabla f(X(t)), \quad X(t) = \nabla\phi^*(Z(t)). \tag{68}$$

This ODE corresponds to continuous time dynamics of projected gradient descent when $\phi = \frac{1}{2}\|\mathbf{x}\|^2$ (and $\nabla\phi(\mathbf{x}) = \mathbf{x}$). Let $\mathbf{d}^{\phi}_{X(t)}$ be the directional derivative with respect to the Bregman projections in the mirror descent algorithm, i.e.,

$$\mathbf{d}^{\phi}_{X(t)} = \lim_{\epsilon\downarrow 0} \frac{\nabla\phi^*(\nabla\phi(X(t)) - \epsilon\nabla f(X(t))) - X(t)}{\epsilon}.$$

The continuous time dynamics of tracing the shadow are simply

$$\dot{X}(t) = \mathbf{d}^{\phi}_{X(t)}. \tag{69}$$

They solely operate in the primal space and one can initialize these with $X(0) = \mathbf{x}_0 \in P$ and show that they are equivalent to (68) under mild technical conditions:

**Theorem 9.** *Let $\phi : \mathcal{D} \to \mathbb{R}$ be a mirror map that is strongly convex and differentiable, and assume that the directional derivative $\mathbf{d}^{\phi}_{X(t)}$ exists for all $t \geq 0$. Then, the dynamics for mirror descent* (68) *are equivalent to the shadow dynamics $\dot{X}(t) = \mathbf{d}^{\phi}_{X(t)}$ with the same initial conditions $X(0) = \mathbf{x}_0 \in P$.*

*Proof.* Consider the dynamics given in (68). Using the chain rule we know that

$$\dot{X}(t) = \frac{d}{dt}\nabla\phi^*(Z(t)) = \left\langle \nabla^2\phi^*(Z(t)), \dot{Z}(t)\right\rangle = \left\langle \nabla^2\phi^*(Z(t)), -\nabla f(X(t))\right\rangle.$$

By definition, the directional derivative of $\nabla\phi^*$ with respect to the direction $-\nabla f(X(t))$ is given by (see for example [23])

$$\nabla^2_{-\nabla f(X(t))}\phi(Z(t)) := \lim_{\epsilon\downarrow 0} \frac{\nabla\phi^*(Z(t) - \epsilon\nabla f(X(t))) - \nabla\phi^*(Z(t))}{\epsilon} = \left\langle \nabla^2\phi^*(Z(t)), -\nabla f(X(t))\right\rangle$$

Hence, using this fact we have

$$\begin{aligned}
\dot{X}(t) &= \left\langle \nabla^2\phi^*(Z(t)), -\nabla f(X(t))\right\rangle \\
&= \lim_{\epsilon\downarrow 0} \frac{\nabla\phi^*(Z(t) - \epsilon\nabla f(X(t))) - \nabla\phi^*(Z(t))}{\epsilon} \\
&= \lim_{\epsilon\downarrow 0} \frac{\nabla\phi^*(Z(t) - \epsilon\nabla f(X(t))) - X(t)}{\epsilon} \qquad \text{(using ODE definition in (68))}
\end{aligned}$$

Since $\phi$ is differentiable on the image of $\nabla\phi^*$, it is known that $\nabla\phi = (\nabla\phi^*)^{-1}$ (in particular, from the duality of $\phi$ and $\phi^*$ we know that $\mathbf{x} = \nabla\phi^*(\tilde{\mathbf{x}})$ if any only $\tilde{\mathbf{x}} = \nabla\phi(\mathbf{x})$; see Theorem 23.5 in [31]). Moreover, by definition of the mirror descent ODE given in (68), we have $X(t) = \nabla\phi^*(Z(t))$. Using these facts we get $Z(t) = (\nabla\phi^*)^{-1}(X(t)) = \nabla\phi(X(t))$. Thus,

$$\dot{X}(t) = \lim_{\epsilon\downarrow 0} \frac{\nabla\phi^*(\nabla\phi(X(t)) - \epsilon\nabla f(X(t))) - X(t)}{\epsilon} = \mathbf{d}^{\phi}_{X(t)}$$

which coincides with dynamics for moving in the shadow of the gradient given in (69). □

Although the results of Theorem 9 hold for general mirror-maps, in this work we focus on the case when $\phi = \frac{1}{2}\|\cdot\|^2$ to exploit the piecewise linear structure of the shadow of the gradient proved in Theorem 1. Note that when the mirror map $\phi = \frac{1}{2}\|\cdot\|^2$, we have

$$\begin{aligned}\nabla\phi^*(\mathbf{y}) &= \arg\max_{\mathbf{x}\in P}\{\langle \mathbf{y}, \mathbf{x}\rangle - \phi(\mathbf{x})\} = \arg\min_{\mathbf{x}\in P}\left\{\frac{1}{2}\|\mathbf{x}\| - \langle \mathbf{y}, \mathbf{x}\rangle\right\} \\ &= \arg\min_{\mathbf{x}\in P}\left\{\frac{1}{2}\|\mathbf{x}\|^2 + \frac{1}{2}\|\mathbf{y}\|^2 - \langle \mathbf{y}, \mathbf{x}\rangle\right\} = \arg\min_{\mathbf{x}\in P}\frac{1}{2}\|\mathbf{y}-\mathbf{x}\|^2.\end{aligned}$$

This implies

$$\begin{aligned}\mathbf{d}^{\phi}_{X(t)} &= \lim_{\epsilon\downarrow 0}\frac{\nabla\phi^*(X(t) - \epsilon\nabla f(X(t))) - X(t)}{\epsilon} && (\nabla\phi(X(t)) = X(t)) \\ &= \lim_{\epsilon\downarrow 0}\frac{\arg\min_{\mathbf{x}\in P}\frac{1}{2}\|X(t) - \epsilon\nabla f(X(t)) - \mathbf{x}\|^2 - X(t)}{\epsilon} && (\text{using definition of } \nabla\phi^*) \\ &= \mathbf{d}^{\Pi}_{X(t)}.\end{aligned}$$

Therefore, Theorem 9 shows that the continuous-time dynamics of moving in the (Euclidean) shadow of the gradient are equivalent to those of PGD.

### D.2 Proof of Theorem 5

We now analyze the convergence rate continuous-time dynamics of moving in the Euclidean shadow of the gradient:

**Theorem 5.** *Let $P \subseteq \mathbb{R}^n$ be a polytope and suppose that $f : P \to \mathbb{R}$ is differentiable and $\mu$-strongly convex over $P$. Consider the shadow dynamics $\dot{X}(t) = \mathbf{d}^{\Pi}_{X(t)}$ with initial conditions $X(0) = \mathbf{x}_0 \in P$. Then for each $t \geq 0$, we have $X(t) \in P$. Moreover, the primal gap $h(X(t)) := f(X(t)) - f(\mathbf{x}^*)$ associated with the shadow dynamics decreases as: $h(X(t)) \leq e^{-2\mu t}h(\mathbf{x}_0)$.*

*Proof.* First, the fact that $X(t) \in P$ for all $t \geq 0$ is guaranteed by the equivalence between the dynamics of PGD (68) and shadow dynamics asserted in Theorem 5, which by construction satisfy $X(t) \in P$ for all $t \geq 0$.

Now the proof for the convergence rate uses a Lyapunov argument, where we let $h(X(t))$ be our Lyapunov potential function. Using the chain rule we have

$$\begin{aligned}\frac{dh(X(t))}{dt} &= \left\langle \nabla f(X(t)), \dot{X}(t)\right\rangle && (70)\\ &= \left\langle \nabla f(X(t)), \mathbf{d}^{\Pi}_{X(t)}\right\rangle && (71)\\ &= -\|\mathbf{d}^{\Pi}_{X(t)}\|^2 && (72)\\ &\leq -2\mu h(X(t)), && (73)\end{aligned}$$

where we used the fact that $\dot{X}(t) = \mathbf{d}^{\Pi}_{X(t)}$ in (71), the fact that $\left\langle -\nabla f(X(t)), \mathbf{d}^{\Pi}_{X(t)}\right\rangle = \|\mathbf{d}^{\Pi}_{X(t)}\|^2$ in (72), and finally the primal gap estimate (63) in (73).

Using Grönwall's inequality [32] to integrate both sides of the above inequality we have

$$\int_0^t \frac{dh(X(t))}{h(X(t))} \leq \int_0^t -\mu dt \implies \ln\left(\frac{h(X(t))}{h(\mathbf{x}_0)}\right) \leq -2\mu t,$$

which further implies $h(X(t)) \leq e^{-2\mu t}h(\mathbf{x}_0)$ as claimed. □

### D.3 Discretization and Analysis

Suppose we try to discretize the dynamics in (69) using a forward Euler approach (see for example chapter 2 in [33]): $\mathbf{x}_{k+1} = \mathbf{x}_k + \lambda_k \mathbf{d}^{\Pi}_{\mathbf{x}_k}$, where $\lambda_k$ is the discretization parameter chosen in a way

**Algorithm 5** SHADOW-WALK Algorithm (detailed)

**Input:** Polytope $P \subseteq \mathbb{R}^n$, function $f : P \to \mathbb{R}$ and initialization $\mathbf{x}_0 \in P$.

**for** $t = 0, \ldots .T$ **do**
  Compute $\mathbf{d}^{\Pi}_{\mathbf{x}_t}$ and let $\gamma_t^{\max} = \max\{\delta \mid \mathbf{x} + \delta \mathbf{d}^{\Pi}_{\mathbf{x}_t} \in P\}$ ▷ `derivative of projection`
  Evaluate $\gamma_t \in \arg\min_{\gamma \in [0, \gamma_t^{\max}]} f(\mathbf{x}_t + \gamma \mathbf{d}^{\Pi}_{\mathbf{x}_t})$. ▷ `line-search along shadow`
  **if** $\gamma_t = \gamma_t^{\max}$ **then** ▷ `boundary case`
    Update $\mathbf{x}_{t+1} := \text{TRACE}(\mathbf{x}_t, \nabla f(\mathbf{x}_t))$ . ▷ `trace projection piecewise linear curve`
  **else**
    Update $\mathbf{x}_{t+1} := \mathbf{x}_t + \gamma_t \mathbf{d}^{\Pi}_{\mathbf{x}_t}$. ▷ `update using line-search`
  **end if**
**end for**

**Return:** $\mathbf{x}_{T+1}$

ensuring feasibility. We know from (8) that there exists a scalar $\lambda^- > 0$ such that $\mathbf{x}_k + \lambda_k \mathbf{d}^{\Pi}_{\mathbf{x}_k}$ is feasible whenever $\lambda_k \leq \lambda^-$. The problem is that $\lambda^-$ can be arbitrarily small, thus making it impossible to show sufficient progress per iteration to obtain linear convergence. However, this is not a problem when discretizing (68), since we can take *unconstrained* gradient descent steps in $\mathcal{D}^*$ and then map these unconstrained steps back to a feasible point in $P$ by computing a Bregman projection. The PGD algorithm is thus able to 'wrap' around the polytope, and avoid this phenomenon of being restricted to small step sizes that feasible descent methods like CG variants inevitably run into.

This is a phenomenon similar to that in the Away-Step and Pairwise CG variants, where the maximum step-size that one can take might not be big enough to show sufficient progress. In [10], the authors overcome this problem by bounding the number of such 'bad' steps using dimension reduction arguments crucially relying on the fact that these algorithms maintain their iterates as a convex combination of vertices. However, unlike away-steps in CG variants, we consider $\mathbf{d}^{\Pi}_{\mathbf{x}}$ as direction for descent, which is independent from the vertices of $P$ and thus eliminating the need to maintain active sets for the iterates of the algorithm. We overcome these problematic cases by *tracing* the piecewise linear curve of $g(\lambda)$, which is guaranteed to be at least as good as PGD step, essentially using the structure of projections to wrap around the polytope and ensure sufficient progress. We give a more detailed algorithmic description (Algorithm 5) than the one included in the main body of the paper for SHADOW-WALK.

We established the following guarantee in the main body of the paper:

**Theorem 6.** *Let $P \subseteq \mathbb{R}^n$ be a polytope and suppose that $f : P \to \mathbb{R}$ is $L$-smooth and $\mu$-strongly convex over $P$. Then the primal gap $h(\mathbf{x}_t) := f(\mathbf{x}_t) - f(\mathbf{x}^*)$ of the* SHADOW WALK *algorithm decreases geometrically: $h(\mathbf{x}_{t+1}) \leq \left(1 - \frac{\mu}{L}\right) h(\mathbf{x}_t)$ with each iteration of the* SHADOW WALK *algorithm (assuming* TRACE *is a single step). Moreover, the number of oracle calls to shadow, in-face direction and line-search oracles to obtain an $\epsilon$-accurate solution is $O\left(\beta \frac{L}{\mu} \log(\frac{1}{\epsilon})\right)$, where $\beta$ is the maximum number of breakpoints of the parametric projections curve that the* TRACE *method visits.*

To analyze the algorithm and prove the above theorem we need some preliminary results, which we first state in the following subsection.

### D.3.1 Preliminaries needed for the proof

Recall that $\mathbf{x}^* = \arg\min_{\mathbf{x} \in P} f(\mathbf{x})$ denotes the constrained optimal solution. Consider an iterative descent scheme of the form $\mathbf{x}_{t+1} = \mathbf{x}_t + \gamma_t \mathbf{d}_t$. First, to obtain a measure of progress, consider the smoothness inequality given in Section 2 with $\mathbf{y} \leftarrow \mathbf{x}_{t+1}$ and $\mathbf{x} \leftarrow \mathbf{x}_t$:

$$f(\mathbf{x}_{t+1}) \leq f(\mathbf{x}_t) + \langle \nabla f(\mathbf{x}_t), \mathbf{x}_{t+1} - \mathbf{x}_t \rangle + \frac{L}{2} \|\mathbf{x}_{t+1} - \mathbf{x}_t\|^2 \tag{74}$$

$$= f(\mathbf{x}_t) + \gamma_t \langle \nabla f(\mathbf{x}_t), \mathbf{d}_t \rangle + \frac{L \gamma_t^2}{2} \|\mathbf{d}_t\|^2 \tag{75}$$

Let $\gamma_t^{\max} = \max\{\delta \mid \mathbf{x} + \delta \mathbf{d}_t \in P\}$. Now consider the step-size $\gamma_{\mathbf{d}_t} := \frac{\langle -\nabla f(\mathbf{x}_t), \mathbf{d}_t \rangle}{L \|\mathbf{d}_{\mathbf{x}_t}\|^2}$ minimizing the RHS of the inequality above and suppose that $\gamma_{\mathbf{d}_t} \leq \gamma_t^{\max}$. Then, plugging in $\gamma_{\mathbf{d}_t}$ in (75) and rearranging we have

$$h(\mathbf{x}_t) - h(\mathbf{x}_{t+1}) = f(\mathbf{x}_t) - f(\mathbf{x}_{t+1}) \geq \frac{\langle -\nabla f(\mathbf{x}_t), \mathbf{d}_t \rangle^2}{2L \|\mathbf{d}_t\|^2}. \tag{76}$$

It is important to note that $\gamma_{\mathbf{d}_t}$ is not the step-size we obtain from line-search. It is just used as means to lower bound the progress obtained from the line-search step.

Another measure of optimality that we will use is the *Wolfe Gap*:

$$h(\mathbf{x}_t) := f(\mathbf{x}_t) - f(\mathbf{x}^*) \leq \langle -\nabla f(\mathbf{x}_t), \mathbf{x}^* - \mathbf{x}_t \rangle \leq \max_{\mathbf{v} \in P} \langle -\nabla f(\mathbf{x}_t), \mathbf{v} - \mathbf{x}_t \rangle . \tag{77}$$

where the first inequality uses the convexity of $f$.

Finally, we will invoke the following theorem in the global linear convergence proof of Theorem 7 and Theorem 6:

**Theorem 10** (Theorem 5 in [34])**.** *Consider the problem $\min_{x \in \mathcal{X}} f(\mathbf{x})$, where $\mathcal{X} \subseteq \mathbb{R}^n$ is a convex and compact domain, and $f : \mathcal{X} \to \mathbb{R}$ is $L$-smooth and $\mu$-strongly convex over $\mathcal{X}$. Further, consider the projected gradient descent (PGD) algorithm with a fixed step-size of $1/L$:*

$$\mathbf{x}_{t+1} := \Pi_{\mathcal{X}}(\mathbf{x}_t - \nabla f(\mathbf{x}_t)/L). \tag{78}$$

*Then the primal gap $h(\mathbf{x}_t) := f(\mathbf{x}_t) - f(\mathbf{x}^*)$ of the PGD algorithm decreases geometrically:*

$$h(\mathbf{x}_{t+1}) \leq \left(1 - \frac{\mu}{L}\right) h(\mathbf{x}_t) \tag{79}$$

*with each iteration of the PGD algorithm.*

We now give a proof of this result for completeness. First, we need the following lemma for the proof:

**Lemma 14** (Lemma 1 in [34])**.** *Let $\mathcal{X} \subseteq \mathbb{R}^n$ be a convex and compact domain and suppose that $f : \mathcal{X} \to \mathbb{R}$ is $L$-smooth and $\mu$-strongly convex over $\mathcal{X}$. For any $\mathbf{x} \in \mathcal{X}$ and $c \in \mathbb{R}$, define*

$$D(\mathbf{x}, c) := -2c \min_{\mathbf{y} \in \mathcal{X}} \left\{ \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle + \frac{c}{2} \|\mathbf{y} - \mathbf{x}\|^2 \right\}.$$

*Then, we have $D(\mathbf{x}, L) \geq D(\mathbf{x}, \mu)$ for all $\mathbf{x} \in \mathcal{X}$.*

*Proof.* Fix any $\mathbf{x} \in \mathcal{X}$. Therefore, by completing the square we have

$$\begin{aligned}
D(\mathbf{x}, c) &= -\min_{\mathbf{y} \in \mathcal{X}} \left\{ 2c \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle + c^2 \|\mathbf{y} - \mathbf{x}\|^2 \right\} \\
&= \min_{\mathbf{y} \in \mathcal{X}} \left\{ \|\nabla f(\mathbf{x})\|^2 - \|\nabla f(\mathbf{x})\|^2 - 2c \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle - c^2 \|\mathbf{y} - \mathbf{x}\|^2 \right\} \\
&= \|\nabla f(\mathbf{x})\|^2 - \min_{\mathbf{y} \in \mathcal{X}} \|c(\mathbf{y} - \mathbf{x}) + \nabla f(\mathbf{x})\|^2 \\
&= \|\nabla f(\mathbf{x})\|^2 - \min_{\bar{\mathbf{y}} \in c(\mathcal{X} - \mathbf{x})} \|\bar{\mathbf{y}} + \nabla f(\mathbf{x})\|^2,
\end{aligned}$$

where in the last equality we used the change of variables $\bar{\mathbf{y}} = c(\mathbf{y} - \mathbf{x})$.

We claim that, since by definition $\mu \leq L$, we have $\mu(\mathcal{X} - \mathbf{x}) \subseteq L(\mathcal{X} - \mathbf{x})$. Indeed, let $\mathbf{z} \in \mu(\mathcal{X} - \mathbf{x})$. Then, $\mathbf{z} = \mu(\mathbf{y} - \mathbf{x}) = L(\frac{\mu}{L}(\mathbf{y} - \mathbf{x}))$ for some $\mathbf{y} \in \mathcal{X}$. Since $\mathbf{y} - \mathbf{x} \in \mathcal{X} - \mathbf{x}$ and $\mathbf{0} = \mathbf{x} - \mathbf{x} \in \mathcal{X} - \mathbf{x}$, it follows that $\frac{\mu}{L}(\mathbf{y} - \mathbf{x}) = \frac{\mu}{L}(\mathbf{y} - \mathbf{x}) + (1 - \frac{\mu}{L})\mathbf{0} \in \mathcal{X} - \mathbf{x}$ by the convexity of $\mathcal{X} - \mathbf{x}$ and the fact that $\mu \leq L$. Thus, we have $\mathbf{z} \in L(\mathcal{X} - \mathbf{x})$ and the claim follows.

Now using this claim we have

$$\begin{aligned}
D(\mathbf{x}, L) &= \|\nabla f(\mathbf{x})\|^2 - \min_{\bar{\mathbf{y}} \in L(\mathcal{X} - \mathbf{x})} \|\bar{\mathbf{y}} + \nabla f(\mathbf{x})\|^2 \\
&\geq \|\nabla f(\mathbf{x})\|^2 - \min_{\bar{\mathbf{y}} \in \mu(\mathcal{X} - \mathbf{x})} \|\bar{\mathbf{y}} + \nabla f(\mathbf{x})\|^2 \qquad \text{(using } \mu(\mathcal{X} - \mathbf{x}) \subseteq L(\mathcal{X} - \mathbf{x})\text{)} \\
&= D(\mathbf{x}, \mu)
\end{aligned}$$

as desired. □

**Proof of Theorem 10.** Let $g(\lambda) = \Pi_P(\mathbf{x} - \lambda \nabla f(\mathbf{x}))$ be the curve parameterized by the step-size $\lambda$. Recall that by the proximal definition of the projection (see e.g., [23]) we have

$$g(1/L) = \Pi_{\mathcal{X}}(\mathbf{x}_t - \nabla f(\mathbf{x}_t)/L) = \arg\min_{\mathbf{y} \in \mathcal{X}} \left\{ \langle \nabla f(\mathbf{x}_t), \mathbf{y} - \mathbf{x}_t \rangle + \frac{L}{2} \|\mathbf{y} - \mathbf{x}_t\|^2 \right\}. \tag{80}$$

We can now show the $(1 - \frac{\mu}{L})$ rate of decrease as follows:

$$\begin{aligned}
h(\mathbf{x}_t) - h(\mathbf{x}_{t+1}) &= f(\mathbf{x}_t) - f(\mathbf{x}_{t+1}) && (81)\\
&= f(\mathbf{x}_t) - f(g(1/L)) && (82)\\
&\geq -\left( \langle \nabla f(\mathbf{x}_t), g(1/L) - \mathbf{x}_t \rangle + \frac{L}{2} \|g(1/L) - \mathbf{x}_t\|^2 \right) && (83)\\
&= -\min_{\mathbf{y} \in \mathcal{X}} \left\{ \langle \nabla f(\mathbf{x}_t), \mathbf{y} - \mathbf{x}_t \rangle + \frac{L}{2} \|\mathbf{y} - \mathbf{x}_t\|^2 \right\} && (84)\\
&= \frac{1}{2L} \left( -2L \min_{\mathbf{y} \in \mathcal{X}} \left\{ \langle \nabla f(\mathbf{x}_t), \mathbf{y} - \mathbf{x}_t \rangle + \frac{L}{2} \|\mathbf{y} - \mathbf{x}_t\|^2 \right\} \right) && (85)\\
&\geq \frac{\mu}{L} \left( -\min_{\mathbf{y} \in \mathcal{X}} \left\{ \langle \nabla f(\mathbf{x}_t), \mathbf{y} - \mathbf{x}_t \rangle + \frac{\mu}{2} \|\mathbf{y} - \mathbf{x}_t\|^2 \right\} \right) && (86)\\
&= \frac{\mu}{L} \left( \max_{\mathbf{y} \in \mathcal{X}} \left\{ \langle -\nabla f(\mathbf{x}_t), \mathbf{y} - \mathbf{x}_t \rangle - \frac{\mu}{2} \|\mathbf{y} - \mathbf{x}_t\|^2 \right\} \right) && (87)\\
&\geq \frac{\mu}{L} \left( \langle -\nabla f(\mathbf{x}_t), \mathbf{x}^* - \mathbf{x}_t \rangle - \frac{\mu}{2} \|\mathbf{x}^* - \mathbf{x}_t\|^2 \right) && (88)\\
&\geq \frac{\mu}{L} h(\mathbf{x}_t), && (89)
\end{aligned}$$

where (83) follows from the smoothness inequality given in Section 2 applied with $\mathbf{y} \leftarrow g(1/L)$ and $\mathbf{x} \leftarrow \mathbf{x}_t$, (84) follows from the definition of $g(1/L)$ given in (80), (86) follows from Lemma 14, (88) follows from the fact that $\mathbf{x}^* \in \mathcal{X}$, and finally (89) follows from the strong convexity inequality given in Section 2 applied with $\mathbf{y} \leftarrow \mathbf{x}^*$ and $\mathbf{x} \leftarrow \mathbf{x}_t$. □

### D.3.2 Proof of Theorem 6

Let $\gamma_t$ be the step size chosen by line-search and $\gamma_t^{\max}$ be the maximum step size that one can move along our chosen direction $\mathbf{d}_t$ while maintaining feasibility. In other words, $\gamma_t^{\max} = \max\{\delta \mid \mathbf{x}_t + \delta \mathbf{d}_t \in P\}$. Finally, we also note from the previous section that $\gamma_{\mathbf{d}_t} = \frac{\langle -\nabla f(\mathbf{x}_t), \mathbf{d}_t \rangle}{L \|\mathbf{d}_{\mathbf{x}_t}\|^2}$ is the step-size obtained from optimizing the smoothness inequality so that we get (76). In the algorithm, we have $\mathbf{d}_t = \mathbf{d}_{\mathbf{x}_t}^{\Pi}$. We split the proof of convergence into two cases depending on whether $\gamma_t < \gamma_t^{\max}$ or not.

(a) **Case 1:** We have $\gamma_t < \gamma_t^{\max}$. In this case we can use the step-size $\gamma_{\mathbf{d}_t}$ to lower bound the progress even if $\gamma_{\mathbf{d}_t}$ is not a feasible step size. To see this, note that the optimal solution of the line-search step is in the interior of the interval $[0, \gamma_t^{\max}]$. Define $\mathbf{x}_\gamma := \mathbf{x}_t + \gamma \mathbf{d}_t$. Then, because $f(\mathbf{x}_\gamma)$ is convex in $\gamma$, we know that $\min_{\gamma \in [0, \gamma_t^{\max}]} f(\mathbf{x}_\gamma) = \min_{\gamma \geq 0} f(\mathbf{x}_\gamma)$ and thus $\min_{\gamma \in [0, \gamma_t^{\max}]} f(\mathbf{x}_\gamma) = f(\mathbf{x}_{t+1}) \leq f(\mathbf{x}_\gamma)$ for all $\gamma \geq 0$. In particular, $f(\mathbf{x}_{t+1}) \leq f(\mathbf{x}_{\gamma_{\mathbf{d}_t}})$. Hence, we can use (76) to bound the progress per iteration as follows:

$$\begin{aligned}
h(\mathbf{x}_t) - h(\mathbf{x}_{t+1}) &\geq \frac{\left\langle -\nabla f(\mathbf{x}_t), \mathbf{d}_{\mathbf{x}_t}^{\Pi} \right\rangle^2}{2L \|\mathbf{d}_{\mathbf{x}_t}^{\Pi}\|^2} && \text{(using (76))}\\
&\geq \frac{\mu}{L} h(\mathbf{x}_t) && \text{(using (63))}
\end{aligned}$$

(b) **Case 2:** We have a boundary case: $\gamma_t = \gamma_t^{\max}$. This is a step where we run the $\textsc{Trace}(\mathbf{x}_t, \nabla f(\mathbf{x}_t))$ procedure. Now, by Theorem 8, we know that $\textsc{Trace}(\mathbf{x}_t, \nabla f(\mathbf{x}_t))$ traces the whole curve of $g(\lambda) = \Pi_P(\mathbf{x}_t - \lambda \nabla f(\mathbf{x}_t))$. Since we are doing exact line-search, we know that at the point $\mathbf{x}_{t+1} := \textsc{Trace}(\mathbf{x}_t, \nabla f(\mathbf{x}_t))$ we have $f(\mathbf{x}_{t+1}) \leq f(g(\lambda))$ for all $\lambda > 0$. In particular, $f(\mathbf{x}_{t+1}) \leq f(g(1/L))$. Thus,

$$h(\mathbf{x}_t) - h(\mathbf{x}_{t+1}) = f(\mathbf{x}_t) - f(\mathbf{x}_{t+1}) \geq f(\mathbf{x}_t) - f(g(1/L)),$$

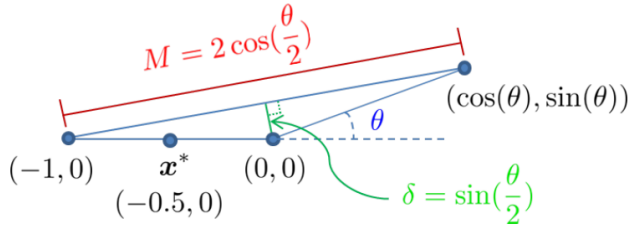


Figure 4: Figure from Lacoste-Julien and Jaggi's work [10] showing the pyramidal width $\delta$ of a simple triangle domain as a function of the angle $\theta$.

and hence we get the same rate $(1 - \frac{\mu}{L})$ of decrease as PGD with fixed step size $1/L$ using Theorem 10.

The iteration complexity of the number of oracle calls stated in the theorem, now follows using the above rate of decrease in the primal gap. □

### D.4 A note on the linear convergence rate

Although our linear convergence rate depends on the number of facet inequalities $m$, it eliminates the dependence on the geometry of the domain that is needed in CG variants. For example, Jaggi and Lacoste-Julien [10] prove a linear rate of $\left(1 - \frac{\mu}{L}\left(\frac{\delta}{D}\right)^2\right)$ to get an $\epsilon-$accurate solution for Away FW, where $\delta$ is the pyramidal width of the domain. Now, consider the example in Figure 4 showing how the pyramidal width $\delta$ of a simple triangle domain changes as the angle $\theta$ changes. In particular, the pyramidal width will be arbitrarily small for small $\theta$. However, note that the number of facets for this triangle domain is $m = 3$, and the number of breakpoints of the projections curve $\beta$ is not dependent on the angle $\theta$. Therefore, we smoothly interpolate between the $\left(1 - \frac{\mu}{L}\right)$ rate for PGD and the rates for CG variants (see Table 1 for a summary of these rates).

## E Missing Proofs in Section 6

We give a more detailed algorithmic description (Algorithm 6) than the one included in the main body of the paper for SHADOW-CG. Using our insights on descent directions, we propose using Frank-Wolfe steps earlier in the algorithm, and use shadow steps more frequently towards the end of the algorithm. Frank-Wolfe steps allow us to greedily skip a lot of facets by wrapping maximally over the polytope (Lemma 4). Shadow steps operate as "optimal" away-steps (Lemma 4) thus reducing zig-zagging phenomenon [10] close to the optimal solution. As the algorithm progresses, one can expect Frank-Wolfe directions to become close to orthogonal to the negative gradient. However, in this case the norm of the shadow also starts diminishing. Therefore, we make the choice of Frank-Wolfe direction versus shadow steps by comparing the inner product of negative gradient with *normalized* shadow direction and the Frank-Wolfe direction.

In line 7 of Algorithm 6, we choose the FW direction whenever $\left\langle -\nabla f(\mathbf{x}_t), \mathbf{d}^{\Pi}_{\mathbf{x}_t}/\|\mathbf{d}^{\Pi}_{\mathbf{x}_t}\|\right\rangle \leq \left\langle -\nabla f(\mathbf{x}_t), \mathbf{d}^{\text{FW}}_t\right\rangle$. Recall from Lemma 3 that $\left\langle -\nabla f(\mathbf{x}), \mathbf{d}^{\Pi}_{\mathbf{x}}/\|\mathbf{d}^{\Pi}_{\mathbf{x}}\|\right\rangle^2 \geq \langle -\nabla f(\mathbf{x}), \mathbf{y}/\|\mathbf{y}\|\rangle^2$ for any direction $\mathbf{y}$ that is feasible at $\mathbf{x}$. With this choice criterion, we choose the FW direction whenever it is sufficiently aligned with the gradient, which allows us to wrap around the polytope and also obtain sufficient descent progress. This choice criterion will not be satisfied once FW starts to zig-zag, at which point the algorithm will take shadow steps. We also note that we need to enter the $\text{TRACE}(\mathbf{x}_t, \nabla f(\mathbf{x}_t))$ procedure only when we hit a boundary case when taking a shadow step, since at this point we cannot guarantee sufficient progress as explained in Figure 1. Since FW steps allow us to greedily skip a lot of facets by wrapping maximally over the polytope, we are able reduce the number of iterations spent in the $\text{TRACE}(\mathbf{x}_t, \nabla f(\mathbf{x}_t))$ procedure.

**Algorithm 6** Shadow Conditional Gradients (SHADOW-CG-detailed)

**Input:** Polytope $P \subseteq \mathbb{R}^n$, function $f : P \to \mathbb{R}$,initialization $\mathbf{x} \in P$ and accuracy parameter $\varepsilon$.

**for** $t = 0, \ldots .T$ **do**
  Let $\mathbf{v}_t := \arg\min_{\mathbf{v} \in P} \langle \nabla f(\mathbf{x}_t), \mathbf{v} \rangle$ and $\mathbf{d}_t^{\text{FW}} := \mathbf{v}_t - \mathbf{x}_t$. ▷ `FW direction`
  **if** $\left\langle -\nabla f(\mathbf{x}_t), \mathbf{d}_t^{\text{FW}} \right\rangle \leq \varepsilon$ **then**
    **return** $\mathbf{x}_t$. ▷ `primal gap is small enough`
  **end if**
  Compute $\mathbf{d}_{\mathbf{x}_t}^{\Pi} := \lim_{\epsilon \downarrow 0} \frac{\Pi_P(\mathbf{x}_t - \epsilon \nabla f(\mathbf{x}_t)) - \mathbf{x}_t}{\epsilon}$. ▷ `derivative of projection`
  **if** $\left\langle -\nabla f(\mathbf{x}_t), \frac{\mathbf{d}_{\mathbf{x}_t}^{\Pi}}{\|\mathbf{d}_{\mathbf{x}_t}^{\Pi}\|} \right\rangle \leq \left\langle -\nabla f(\mathbf{x}_t), \mathbf{d}_t^{\text{FW}} \right\rangle$ **then** ▷ `wrap-around using FW`
    $\mathbf{d}_t := \mathbf{d}_t^{\text{FW}}$ and $\gamma_t^{\max} = 1$. ▷ `choose optimistic step`
  **else**
    $\mathbf{d}_t := \mathbf{d}_{\mathbf{x}_t}^{\Pi}$ and $\gamma_t^{\max} = \max\{\delta \mid \mathbf{x}_t + \delta \mathbf{d}_t \in P\}$. ▷ `choose pessimistic step`
  **end if**
  $\gamma_t \in \arg\min_{\gamma \in [0, \gamma_t^{\max}]} f(\mathbf{x}_t + \gamma \mathbf{d}_t)$. ▷ `line-search along chosen descent direction`
  **if** $\mathbf{d}_t = \mathbf{d}_{\mathbf{x}_t}^{\Pi}$ **and** $\gamma_t = \gamma_t^{\max}$ **then**
    Update $\mathbf{x}_{t+1} := \text{TRACE}(\mathbf{x}_t, \nabla f(\mathbf{x}_t))$ . ▷ `trace projection piecewise linear curve`
  **else**
    Update $\mathbf{x}_{t+1} := \mathbf{x}_t + \gamma_t \mathbf{d}_t$. ▷ `update using line search`
  **end if**
**end for**

**Return:** $\mathbf{x}_{T+1}$

### E.1 Proof of Theorem 7

**Theorem 7.** *Let $P \subseteq \mathbb{R}^n$ be a polytope with diameter $D$ and suppose that $f : P \to \mathbb{R}$ is $L$-smooth and $\mu$-strongly convex over $P$. Then, the primal gap $h(\mathbf{x}_t) := f(\mathbf{x}_t) - f(\mathbf{x}^*)$ of* SHADOW-CG *decreases geometrically: $h(\mathbf{x}_{t+1}) \leq \left(1 - \frac{\mu}{LD^2}\right) h(\mathbf{x}_t)$, with each iteration of the* SHADOW-CG *algorithm (assuming* TRACE *is a single step). Moreover, the number of shadow, in-face directions and line oracle calls for an $\epsilon$-accurate solution is $O\left((D^2 + \beta)\frac{L}{\mu} \log(\frac{1}{\epsilon})\right)$, where $\beta$ is the number of breakpoints of the parametric projections curve that the* TRACE *method visits.*

*Proof.* Let $\gamma_t$ be the step size chosen by line-search and $\gamma_t^{\max}$ be the maximum step size that one can move along our chosen direction $\mathbf{d}_t$ while maintaining feasibility. In other words, $\gamma_t^{\max} = \max\{\delta \mid x_t + \delta \mathbf{d}_t \in P\}$. Finally, we also note from the previous section that $\gamma_{\mathbf{d}_t} = \frac{\langle -\nabla f(\mathbf{x}_t), \mathbf{d}_t \rangle}{L \|\mathbf{d}_{\mathbf{x}_t}\|^2}$ is the step size obtained from optimizing the smoothness inequality so that we get (76). In the algorithm, we have either have $\mathbf{d}_t = \mathbf{d}_{\mathbf{x}_t}^{\Pi}$ or $\mathbf{d}_t = \mathbf{d}_t^{\text{FW}}$. We split the proof of convergence into two cases depending on whether $\gamma_t < \gamma_t^{\max}$ or not.

(a) **Case 1:** We have $\gamma_t < \gamma_t^{\max}$. In this case we can use the step size from $\gamma_{\mathbf{d}_t}$ to lower bound the progress even if $\gamma_{\mathbf{d}_t}$ is not a feasible step size. To see this, note that the optimal solution of the line-search step is in the interior of the interval $[0, \gamma_t^{\max}]$. Define $\mathbf{x}_\gamma := \mathbf{x}_t + \gamma \mathbf{d}_t$. Then, because $f(\mathbf{x}_\gamma)$ is convex in $\gamma$, we know that $\min_{\gamma \in [0, \gamma_t^{\max}]} f(\mathbf{x}_\gamma) = \min_{\gamma \geq 0} f(\mathbf{x}_\gamma)$ and thus $\min_{\gamma \in [0, \gamma_t^{\max}]} f(\mathbf{x}_\gamma) = f(\mathbf{x}_{t+1}) \leq f(\mathbf{x}_\gamma)$ for all $\gamma \geq 0$. In particular, $f(\mathbf{x}_{t+1}) \leq f(\mathbf{x}_{\gamma_{\mathbf{d}_t}})$.

Hence, we can use (76) to bound the progress per iteration. We split this into two further sub-cases depending on whether we take a FW step or a shadow step:

(i) First suppose that we take a shadow step so that $\mathbf{d}_t = \mathbf{d}_{\mathbf{x}_t}^{\Pi}$. Then we have

$$h(\mathbf{x}_t) - h(\mathbf{x}_{t+1}) \geq \frac{\left\langle -\nabla f(\mathbf{x}_t), \mathbf{d}_{\mathbf{x}_t}^{\Pi} \right\rangle^2}{2L \|\mathbf{d}_{\mathbf{x}_t}^{\Pi}\|^2} \qquad \text{(using (76))}$$

$$\geq \frac{\mu}{L} h(\mathbf{x}_t) \qquad \text{(using (63)).}$$

(ii) Now suppose that $\mathbf{d}_t = \mathbf{d}_t^{\text{FW}}$. Then we can bound the progress as follows:

$$h(\mathbf{x}_t) - h(\mathbf{x}_{t+1}) \geq \frac{\left\langle -\nabla f(\mathbf{x}_t), \mathbf{d}_t^{\text{FW}} \right\rangle^2}{2L \|\mathbf{d}_t^{\text{FW}}\|^2} \qquad \text{(using (76))}$$

$$
\begin{aligned}
&\geq \frac{\left\langle -\nabla f(\mathbf{x}_t), \mathbf{d}_t^{\mathrm{FW}} \right\rangle^2}{2LD^2} \\
&\geq \frac{\left\langle -\nabla f(\mathbf{x}_t), \frac{\mathbf{d}_{\mathbf{x}_t}^{\Pi}}{\|\mathbf{d}_{\mathbf{x}_t}^{\Pi}\|} \right\rangle^2}{2LD^2} && \text{(choice of descent step)} \\
&\geq \frac{\mu}{LD^2} h(\mathbf{x}_t) && \text{(using (63)).}
\end{aligned}
$$

This shows the rate stated in the theorem.

(b) **Case 2:** We have a boundary case: $\gamma_t = \gamma_t^{\max}$. We further divide this case into two sub-cases:

(i) First assume that $\gamma_{\mathbf{d}_t} \leq \gamma_t^{\max}$ so that the step size from smoothness is feasible. Then, using the same argument as above we again have a worst-case geometric rate of decrease of $(1 - \frac{\mu}{LD^2})$.

(ii) Now assume $\gamma_{\mathbf{d}_t} > \gamma_t^{\max}$. First suppose that we take a shadow step, i.e. $\mathbf{d}_t = \mathbf{d}_{\mathbf{x}_t}^{\Pi}$. Then, in this step we run the $\textsc{Trace}(\mathbf{x}_t, \nabla f(\mathbf{x}_t))$. Now, by Theorem 8, we know that $\textsc{Trace}(\mathbf{x}_t, \nabla f(\mathbf{x}_t))$ traces the whole curve of $g(\lambda) = \Pi_P(\mathbf{x}_t - \lambda \nabla f(\mathbf{x}_t))$. Since we are doing exact line-search, we know that at the point $\mathbf{x}_{t+1} := \textsc{Trace}(\mathbf{x}_t, \nabla f(\mathbf{x}_t))$ we have $f(\mathbf{x}_{t+1}) \leq f(g(\lambda))$ for all $\lambda > 0$. In particular, $f(\mathbf{x}_{t+1}) \leq f(g(1/L))$. Thus,

$$
h(\mathbf{x}_t) - h(\mathbf{x}_{t+1}) = f(\mathbf{x}_t) - f(\mathbf{x}_{t+1}) \geq f(\mathbf{x}_t) - f(g(1/L)),
$$

and hence we get the same rate $(1 - \frac{\mu}{L})$ of decrease as PGD with fixed step size $1/L$ using Theorem 10.

(iii) Finally assume that $\gamma_{\mathbf{d}_t} > \gamma_t^{\max}$ and $\mathbf{d}_t = \mathbf{d}_t^{\mathrm{FW}}$. Observe that $\gamma_{\mathbf{d}_t} = \frac{\left\langle -\nabla f(\mathbf{x}_t), \mathbf{d}_t^{\mathrm{FW}} \right\rangle}{L\|\mathbf{d}_t^{\mathrm{FW}}\|^2} > \gamma_t^{\max} = 1$ implies that $\left\langle -\nabla f(\mathbf{x}_t), \mathbf{d}_t^{\mathrm{FW}} \right\rangle \geq L\|\mathbf{d}_t^{\mathrm{FW}}\|_2^2$. Hence, using the fact that $\gamma_t = \gamma_t^{\max}$ in the smoothness inequality in (75), we have

$$
\begin{aligned}
h(\mathbf{x}_t) - h(\mathbf{x}_{t+1}) &\geq \left\langle -\nabla f(\mathbf{x}_t), \mathbf{d}_t^{\mathrm{FW}} \right\rangle - \frac{L}{2}\|\mathbf{d}_t^{\mathrm{FW}}\|_2^2 \\
&\geq \frac{\left\langle -\nabla f(\mathbf{x}_t), \mathbf{d}_t^{\mathrm{FW}} \right\rangle}{2} && \text{(using } \gamma_{\mathbf{d}_t} > \gamma_t^{\max}\text{)} \\
&\geq \frac{h(\mathbf{x}_t)}{2} && \text{(using Wolfe gap (77))}
\end{aligned}
$$

Hence, we get a geometric rate of decrease of 1/2.

The iteration complexity of the number of oracle calls stated in the theorem now follows using the above rate of decrease in the primal gap. □

# F Computations

We implemented all algorithms in `Python` 3.5, utilizing `numpy` and `scipy` for some of our functions. We used these packages from the Anaconda 4.7.12 distribution as well as `Gurobi` 9 [35] as a black box solver for some of the oracles assumed in the paper. All experiments were performed on a 16-core machine with Intel Core i7-6600U 2.6-GHz CPUs and 256GB of main memory.

For the computations, we need to solve the following subproblems:

(i) **Linear optimization (LO):** Compute $\mathbf{v} = \arg\min_{\mathbf{x} \in P} \langle \mathbf{c}, \mathbf{x} \rangle$ for any $\mathbf{c} \in \mathbb{R}^n$.

(ii) **Shadow computation:** Given any point $\mathbf{x} \in P$ and direction $\mathbf{w} \in \mathbb{R}^n$, compute $\mathbf{d}_{\mathbf{x}}^{\Pi}(\mathbf{w})$.

(iii) **Feasibility:** Given any point $\mathbf{x} \in P$ and direction $\mathbf{d} \in \mathbb{R}^n$, evaluate $\gamma^{\max} = \max\{\delta : \mathbf{x} + \delta \mathbf{d} \in P\}$.

(iv) **Line-search:** Given any point $\mathbf{x} \in P$ and direction $\mathbf{d} \in \mathbb{R}^n$, solve the one-dimensional problem $\min_{\gamma \in [0, \gamma^{\max}]} f(\mathbf{x} + \gamma \mathbf{d})$.

**Algorithm 7** Tracing Parametric Projections Curve Approximately: TRACE-APP$(\mathbf{x}, \nabla f(\mathbf{x}))$

**Input:** Polytope $P \subseteq \mathbb{R}^n$, function $f : P \to \mathbb{R}$ and initialization $\mathbf{x} \in P$.

Compute $\mathbf{d}_{\mathbf{x}}^{\Pi} := \lim_{\epsilon \downarrow 0} \frac{\Pi_P(\mathbf{x} - \epsilon \nabla f(\mathbf{x})) - \mathbf{x}}{\epsilon}$ and let $\gamma^{\max} = \max\{\delta \mid \mathbf{x} + \delta \mathbf{d}_{\mathbf{x}}^{\Pi} \in P\}$.
Let $\mathbf{d} = \nabla f(\mathbf{x})$ and $\gamma^* \in \arg\min_{\gamma \in [0, \gamma^{\max}]} f(\mathbf{x} + \gamma \mathbf{d}_{\mathbf{x}}^{\Pi})$. ▷ `line-search along derivative`
**while** $\gamma^* = \gamma^{\max}$ **do**
  $\mathbf{x} = \mathbf{x} + \gamma^{\max} \mathbf{d}_{\mathbf{x}}^{\Pi}$ ▷ `Approximately obtain next segment in PW curve`
  Recompute $\mathbf{d}_{\mathbf{x}}^{\Pi} := \lim_{\epsilon \downarrow 0} \frac{\Pi_P(\mathbf{x} - \epsilon \mathbf{d}) - \mathbf{x}}{\epsilon}$ and let $\gamma^{\max} = \max\{\delta \mid \mathbf{x} + \delta \mathbf{d}_{\mathbf{x}}^{\Pi} \in P\}$.
  **if** $\mathbf{d}_{\mathbf{x}}^{\Pi} = 0$ **then**
    **break** and **return** $\mathbf{x}$ ▷ `we reached endpoint of projections curve`
  **end if**
  $\gamma^* \in \arg\min_{\gamma \in [0, \gamma^{\max}]} f(\mathbf{x} + \gamma \mathbf{d}_{\mathbf{x}}^{\Pi})$. ▷ `check optimality of line-search solution`
**end while**

**Return:** $\mathbf{x} + \gamma^* \mathbf{d}_{\mathbf{x}}^{\Pi}$

We elaborate on the implementation of the LO subproblems later on as it is dependent on the application. For the shadow oracle, given any point $\mathbf{x} \in P$ and direction $\mathbf{w} \in \mathbb{R}^n$ we solve the problem $\mathbf{d}_{\mathbf{x}}^{\Pi}(\mathbf{w}) = \arg\min_{\mathbf{d}} \{\| - \nabla f(\mathbf{x}) - \mathbf{d}\|^2 : A_{I(\mathbf{x})} \mathbf{d} \leq 0\}$ using `Gurobi`. Moreover, for the feasibility problem, given $\mathbf{x} \in P$ and descent direction $\mathbf{d} \in \mathbb{R}^n$, we compute the maximum step-size ensuring feasibility as follows:

$$\gamma^{\max} = \min_{\substack{j \in J(\mathbf{x}): \\ \langle \mathbf{a}_j, \mathbf{d} \rangle > 0}} \frac{b_j - \langle \mathbf{a}_j, \mathbf{x} \rangle}{\langle \mathbf{a}_j, \mathbf{d} \rangle}, \tag{90}$$

where the feasible set of the above problem is non-empty, since otherwise this would imply that $\mathbf{d}$ is a recessive direction (i.e. direction of unboundedness), contradicting the fact that $P$ is a polytope. We consider polytopes with a polynomial number of constraints, and hence (90) can be efficiently solved. For the line-search sub-problem, we utilize a bracketing method[8] for line search (see, for example [23]). Finally, regarding the TRACE procedure used in the computations, we consider a preliminary approximate TRACE procedure TRACE-APP$(\mathbf{x}, \nabla f(\mathbf{x}))$ that excludes the in-face trace steps. The exact implementation we use is given in Algorithm 7.

### F.1 Video Co-localization

The first application we consider is the video co-localization problem from computer vision, where the goal is to track an object across different video frames. We used the YouTube-Objects dataset[9] and the problem formulation of Joulin et. al [5]. This consists of minimizing a quadratic function $f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T A \mathbf{x} + \mathbf{b}^T \mathbf{x}$, where $\mathbf{x} \in \mathbb{R}^{660}$, $A \in \mathbb{R}^{660 \times 660}$ and $\mathbf{b} \in \mathbb{R}^{660}$, over a flow polytope, the convex hull of paths in a network. Our linear minimization oracle over the flow polytope amounts to computing a shortest path in the corresponding directed acyclic graph. We now present the computational results in Figure 5.

We find that SHADOW-CG has a lower iteration count than other CG variants DICG, AFW and PFW (slightly higher than PGD) for this experiment. Without assuming oracle access, SHADOW-CG improves on the wall-clock time compared to PGD (i.e., close to CG). Moreover, we also find that assuming access to shadow oracle, the SHADOW-CG algorithm outperforms the CG variants both in iteration count and wall-clock time. For completeness, we also compare these different algorithms with respect to the duality gap $\langle -\nabla f(\mathbf{x}_t), \mathbf{v}_t - \mathbf{x}_t \rangle$ (77) in Figure 6.

[8] We specifically use golden-section search that iteratively reduces the interval locating the minimum.

[9] We obtained the data from `https://github.com/Simon-Lacoste-Julien/linearFW`.

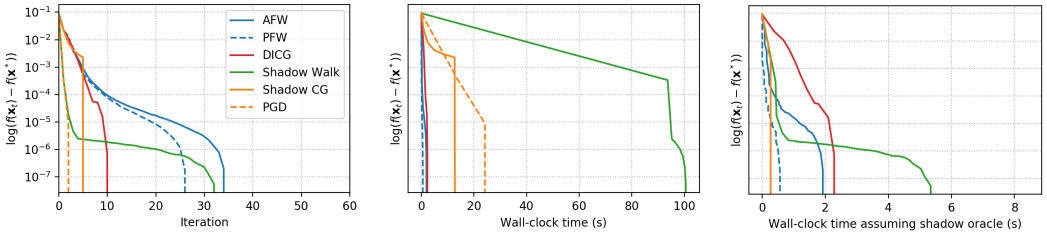


Figure 5: Optimality gap for the video co-localization problem: Away-step FW (AFW) [10], pairwise FW (PFW) [10], decomposition-invariant CG (DICG) [16], SHADOW-WALK (Algorithm 1), and SHADOW-CG (Algorithm 2). Left plot compares iteration count, middle and right plots compare wall-clock time with and without access to shadow oracle. We removed the PGD from the rightmost plot for a better comparison of other algorithms as it takes significantly more time due to the projection step and thus skews the plot.

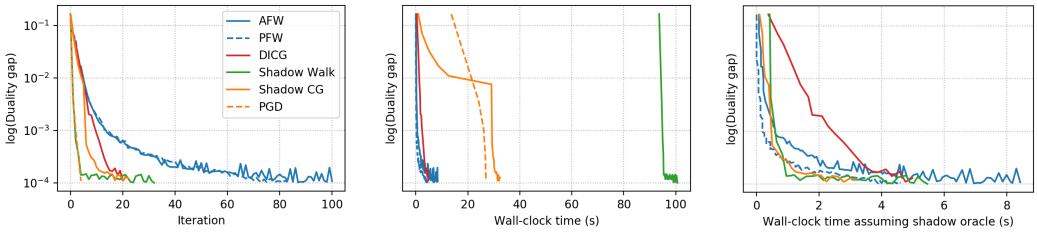


Figure 6: Duality gap for the video co-localization problem: Away-step FW (AFW) [10], pairwise FW (PFW) [10], decomposition-invariant CG (DICG) [16], SHADOW-WALK (Algorithm 1), and SHADOW-CG (Algorithm 2). Left plot compares iteration count, middle and right plots compare wall-clock time with and without access to shadow oracle. We removed the PGD from the rightmost plot for a better comparison of other algorithms as it takes significantly more time due to the projection step and thus skews the plot.

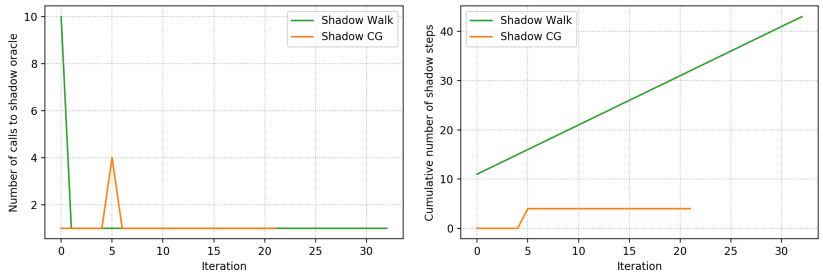


Figure 7: Left: Comparing the number of shadow oracles calls made per iteration in the video co-localization problem, by SHADOW-WALK (goes upto 10) and SHADOW-CG (goes upto 4 iterations) in the Lasso regression problem. Right: Comparing the cumulative number of shadow steps taken, where the FW steps in SHADOW-CG cause a significant reduction in the number of shadow steps taken compared to the SHADOW-WALK algorithm. Each iteration of SHADOW-CG requires a single computation of the shadow to evaluation condition for selecting FW or shadow step. Instead, we only plot the shadow steps actually taken by each of the algorithms.

### F.2 Lasso Regression

The second application we consider is the Lasso regression problem, i.e. $\ell_1-$regularized least squares regression. This consists of minimizing a quadratic function $f(\mathbf{x}) = \|\mathbf{A}\mathbf{x} - \mathbf{b}\|$ over a scaled $\ell_1-$ ball. We considered a random Gaussian matrix $\mathbf{A} \in \mathbb{R}^{50\times 100}$ and a noisy measurement $\mathbf{b} = \mathbf{A}\mathbf{x}^*$ with $\mathbf{x}^*$ being a sparse vector with 25 entries $\pm 1$, and some additive noise. Linear minimization over

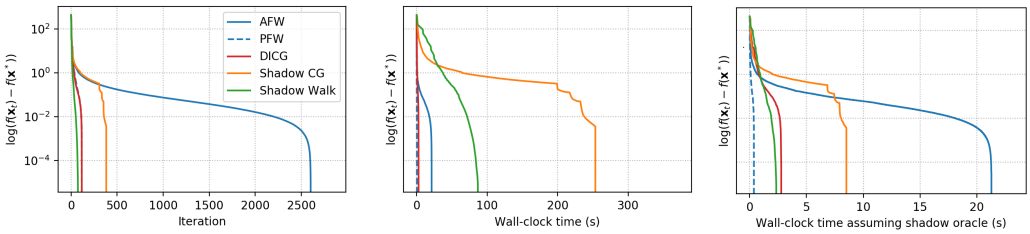


Figure 8: Optimality gaps for the Lasso regression problem: Away-step FW (AFW) [10], pairwise FW (PFW) [10], decomposition-invariant CG (DICG) [16], SHADOW-WALK (Algorithm 1), and SHADOW-CG (Algorithm 2). Left plot compares iteration count, middle and right plots compare wall-clock time with and without access to shadow oracle.

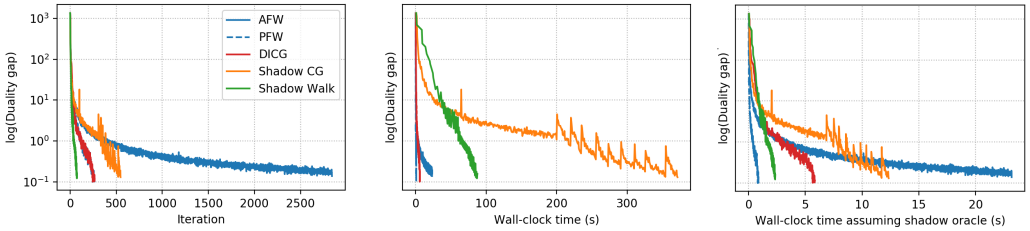


Figure 9: Duality gaps for the Lasso regression problem: Away-step FW (AFW) [10], pairwise FW (PFW) [10], decomposition-invariant CG (DICG) [16], SHADOW-WALK (Algorithm 1), and SHADOW-CG (Algorithm 2). Left plot compares iteration count, middle and right plots compare wall-clock time with and without access to shadow oracle.

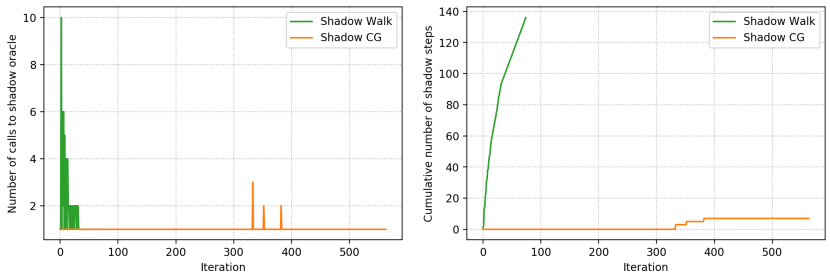


Figure 10: Left: Comparing the number of shadow oracles calls made per iteration in the Lasso regression problem by SHADOW-WALK (goes upto 10) and SHADOW-CG (goes upto 3 iterations) in the Lasso regression problem. Right: Comparing the cummulative number of shadow steps taken, where the FW steps in SHADOW-CG cause a significant reduction in the number of shadow steps taken compared to the SHADOW-WALK algorithm. Each iteration of SHADOW-CG requires a single computation of the shadow to evaluation condition for selecting FW or shadow step. Instead, we only plot the shadow steps actually taken by each of the algorithms.

the $\ell_1-$ball, simply amounts to selecting the column of $\mathbf{A}$ with best inner product with the residual vector $\mathbf{A}\mathbf{x} - \mathbf{b}$.

We present its computational results in Figure 8. In these experiments, we observe that SHADOW-WALK algorithm is superior in iteration count and outperforms all other CG variants. Moreover, SHADOW-CG, SHADOW-CG has a significantly lower iteration count than AFW as expected. In addition, assuming access to a shadow oracle, both the SHADOW-WALK and SHADOW-CG algorithm have improvements over CG variants both in iteration count and wall-clock time. For completeness, we also compare these different algorithms with respect to the duality gap $\langle -\nabla f(\mathbf{x}_t), \mathbf{v}_t - \mathbf{x}_t \rangle$ (77) in Figure 9.

Finally, we demonstrate computationally that the number of iterations spent in the TRACE procedure is a lot better than the worst-case bound we prove in Theorem 8 by looking at the number of oracles calls made by the SHADOW CG and SHADOW-WALK algorithms per iteration. In particular, we find

that the number of shadow oracle calls made per iteration by the SHADOW CG algorithm is smaller than the number of calls made by the SHADOW-WALK algorithm, which in turn implies that SHADOW CG spends a smaller number of iterations in the TRACE procedure as expected. Moreover, we also find that the addition of FW steps causes the SHADOW CG algorithm to take a significantly smaller number of shadow steps than SHADOW-WALK does. This behavior is demonstrated in Figures 7 and 10 corresponding to the two experiments. Note that both algorithms have to make at least one call to a shadow oracle every iteration, however the SHADOW CG algorithm has the flexibility of not actually taking a shadow step and choosing to take a FW step instead, in which case the orange curve in the right plot of Figures 7 and 10 remains flat, and hence the step-wise structure of the curve.

### F.3 A Smaller Lasso Regression Instance

To distinguish between the algorithms further and highlight the theoretical aspects presented in this paper, we consider a smaller instance of the same Lasso regression problem given in previous section: we now consider a smaller random Gaussian matrix $\mathbf{A} \in \mathbb{R}^{40\times 60}$ and a noisy measurement $\mathbf{b} = \mathbf{A}\mathbf{x}^*$ with $\mathbf{x}^*$ being a sparse vector with 15 entries $\pm 1$, and some additive noise. We present the results in Figures 11 and 12.

Our findings regarding the iteration count and wall-clock time of the different algorithms are the same as the previous Lasso regression instance. We now focus on the difference between the SHADOW CG and SHADOW-WALK algorithms in terms of the number of shadow steps taken and number of calls made to shadow oracles in more details. First, we find that the SHADOW CG algorithms takes FW steps only at the end when the FW directions start to become more orthogonal to the gradient. This behavior is now more pronounced in Figure 13. Moreover, we again find that the number of shadow oracle calls made per iteration by the SHADOW CG algorithm is smaller than the number of calls made by the SHADOW-WALK algorithm. Finally, upon comparing the cumulative number of shadow steps taken, we see that the curve for the SHADOW-WALK algorithm given in the right plot of Figure 13 is concave-like. This implies that the SHADOW-WALK algorithm spends a bigger number of iterations in the TRACE procedure in the beginning as it wants to wrap around the polytope. On the other hand, we see that the curve for the SHADOW-CG algorithm in Figure 13 is a step-wise curve where we only take shadow steps in the end; these steps essentially serve as optimal away-steps that help us overcome zig-zagging.

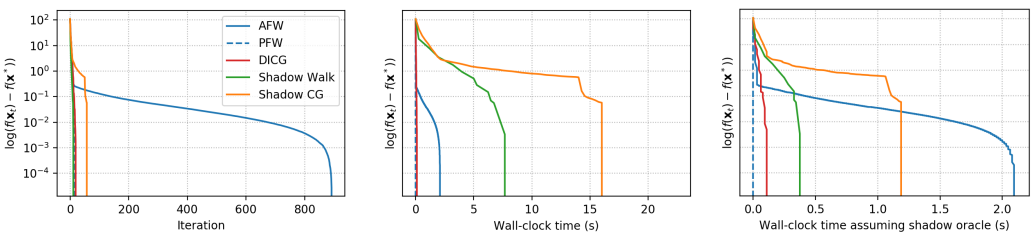


Figure 11: Optimality gaps for the smaller Lasso regression instance: Away-step FW (AFW) [10], pairwise FW (PFW) [10], decomposition-invariant CG (DICG) [16], SHADOW-WALK (Algorithm 1), and SHADOW-CG (Algorithm 2) Left plot compares iteration count, middle and right plots compare wall-clock time with and without access to shadow oracle.

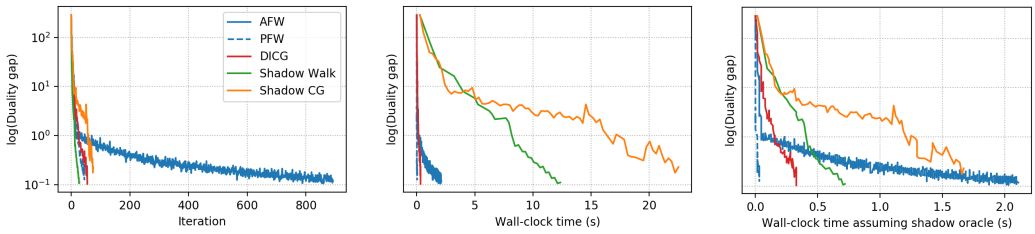


Figure 12: Duality gaps for the smaller Lasso regression instance: Away-step FW (AFW) [10], pairwise FW (PFW) [10], decomposition-invariant CG (DICG) [16], SHADOW-WALK (Algorithm 1), and SHADOW-CG (Algorithm 2). Left plot compares iteration count, middle and right plots compare wall-clock time with and without access to shadow oracle.

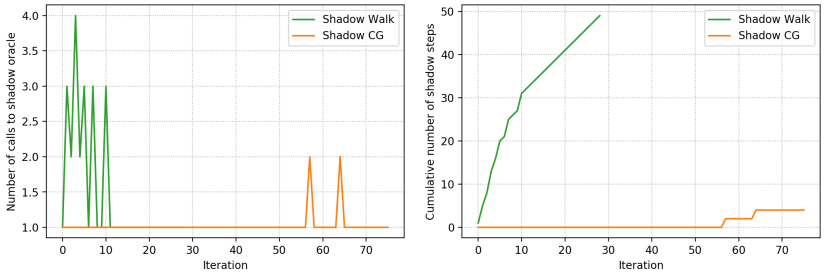


Figure 13: Left: Comparing the number of shadow oracles calls made per iteration in the smaller Lasso regression instance by SHADOW-WALK (goes upto 4) and SHADOW-CG (goes upto 2 iterations) in the the smaller Lasso regression instance. Right: Comparing the cummulative number of shadow steps taken, where the FW steps in SHADOW-CG cause a significant reduction in the number of shadow steps taken compared to the SHADOW-WALK algorithm. Each iteration of SHADOW-CG requires a single computation of the shadow to evaluation condition for selecting FW or shadow step. Instead, we only plot the shadow steps actually taken by each of the algorithms.