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 ϵ-accurate solution is $O(\frac{L}{\mu} \log \frac{1}{\epsilon})$), for μ-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.

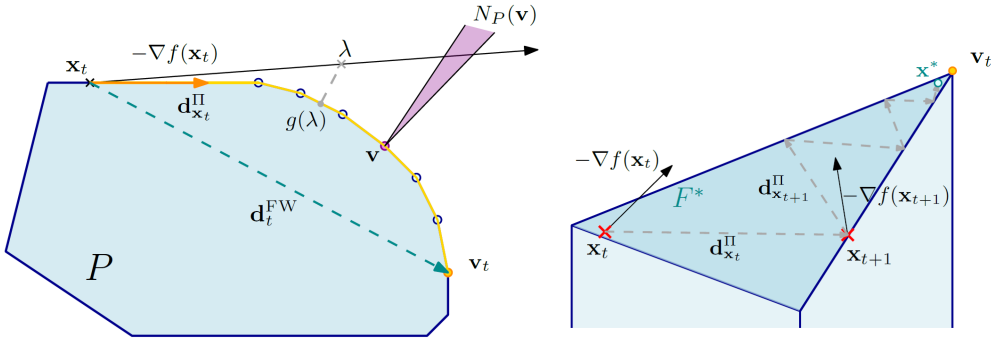

Figure 1: **Left:** Piecewise linear structure of the parametric projection curve $g(\lambda) = \Pi_P(\mathbf{x}_t - \lambda\nabla f(\mathbf{x}_t))$ (yellow line). The end point is the FW vertex $\mathbf{v}_t$ and $\mathbf{d}^{\mathrm{FW}}_t$ the FW direction. Note that $g(\lambda)$ does not change at the same speed as λ, e.g., $g(\lambda) = \mathbf{v}$ for each λ 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 δ 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 γ_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 ℓ_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 δ 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 λ 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, λ_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 λ_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 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 $i = 1$. 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 λ 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 λ_{i-1}^+ and λ_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 ε.

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 μ-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 μ-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*

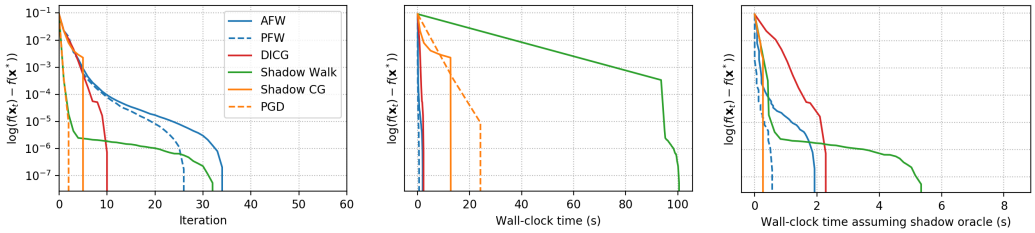

Figure 2: Comparing the performance of away-step FW (AFW) [10], pairwise FW (PFW) [10], decomposition-invariant CG (DICG) [16], SHADOW-WALK (Alg. 1), and SHADOW-CG (Alg. 2). Left plot compares iteration count, middle plot compares wall-clock time (including shadow computation and line search), right 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 ϵ-accurate solution is $O\left(\beta \frac{L}{\mu} \log(\frac{1}{\epsilon})\right)$, where β 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 β. Although the constant β 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 μ-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 ϵ-accurate solution is $O\left((D^2 + \beta) \frac{L}{\mu} \log(\frac{1}{\epsilon})\right)$, where β 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.