Fast Projection-Free Approach (without Optimization Oracle) for Optimization over Compact Convex Set

Chenghao Liu[1], **Enming Liang**[1,*], **Minghua Chen**[1,2,*]
[1]Department of Data Science, City University of Hong Kong
[2]School of Data Science, The Chinese University of Hong Kong, Shenzhen

Abstract

Projection-free first-order methods, e.g., the celebrated Frank-Wolfe (FW) algorithms, have emerged as powerful tools for optimization over simple convex sets such as polyhedra, because of their scalability, fast convergence, and iteration-wise feasibility without costly projections. However, extending these methods effectively to general compact convex sets remains challenging and largely open, as FW methods rely on expensive linear optimization oracles (LOO), while penalty-based methods often struggle with poor feasibility. We tackle this open challenge by presenting **Hom-PGD**, a novel projection-free method without expensive (optimization) oracles. Our method constructs a homeomorphism between the convex constraint set and a unit ball, transforming the original problem into an equivalent ball-constrained formulation, thus enabling efficient gradient-based optimization while preserving the original problem structure. We prove that Hom-PGD attains *optimal* convergence rates matching gradient descent with constant step-size to find an ϵ-approximate (stationary) solution: $\mathcal{O}(\log(1/\epsilon))$ for strongly convex objectives, $\mathcal{O}(\epsilon^{-1})$ for convex objectives, and $\mathcal{O}(\epsilon^{-2})$ for non-convex objectives. Meanwhile, Hom-PGD enjoys a low per-iteration complexity of $\mathcal{O}(n^2)$, without expensive oracles like LOO or projection, where n is the input size. Our framework further extends to certain non-convex sets, broadening its applicability in practical optimization scenarios with complex constraints. Extensive numerical experiments demonstrate that Hom-PGD achieves comparable convergence rates to state-of-the-art projection-free methods, while significantly reducing per-iteration runtime (up to 5 orders of magnitude faster) and thus the total problem-solving time.

1 Introduction

We consider constrained optimization where the objective is smooth, possibly non-convex, and the constrained set is compact convex. Although popular second-order methods, such as interior-point methods [PW00, Wri97] and cutting plane methods [B$^+$15], achieve linear convergence rates, their per-iteration computational complexity scales super-linearly with the problem size, typically on the order of $\mathcal{O}(n^3)$ due to solving a linear system. Consequently, these methods become impractical for large-scale problems. Alternative approaches, such as projection-based gradient descent (PGD) first-order methods (see, e.g., [Bec17, ZPL22, ZL22]), provide a computational benefit for simple convex sets where orthogonal projections can be performed efficiently, such as Euclidean balls and boxes. Despite their slower convergence rate of $\mathcal{O}(1/\epsilon)$ in the convex smooth setting and $\mathcal{O}(1/\epsilon^2)$ in the non-convex smooth setting, these methods are favorable in practice due to their relatively low per-iteration cost. However, the projection operation, i.e., solving a convex problem with a quadratic objective over constraints, is computationally expensive except for simple constraint sets.

*Corresponding authors: Enming Liang (enming.cityu@gmail.com) and Minghua Chen (minghua@cuhk.edu.cn)

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

Table 1: Summary of existing projection-free methods for solving optimization problems.

Reference	Settings: Obj.	Ctr.	Key Assumption	Algorithm	Step-size[2]	Per-iteration Complexity	Convergence Rate[3]
[LMY23]	NC	Simplex	-	Hadamard Parameterization + Pertubed PGD	Constant	$\mathcal{O}(n)$	$\mathcal{O}(\epsilon^{-2})$
[LG23]	C	C	SC (Obj + Ctr Set) S (Obj + Ctr Set) SC & S (Obj + Ctr Set)	Sub-GD Alg. for the RD[4]	Implicit and Diminishing	QOO	$\mathcal{O}(\epsilon^{-1})$ $\mathcal{O}(\epsilon^{-0.5})$ $\mathcal{O}(\log \epsilon^{-1})$
[Gri24b]	C	C	Upper Radial Obj	Accelerate Sub-GD Alg. for the RD[4]	Implicit or Vanishing	MO	$\mathcal{O}(\epsilon^{-0.5})$
[MHSY25]	DC	C	-	Frank-Wolfe	Diminishing	LOO	$\mathcal{O}(\epsilon^{-2})$
Theorem 1	C	C	ND Minimizer	**Hom-PGD** (Sec. 3)	Constant	MO	$\mathcal{O}(\epsilon^{-1})$
Theorem 2	SC	C	-				$\mathcal{O}(\log \epsilon^{-1})$
Theorem 3	NC	C	-				$\mathcal{O}(\epsilon^{-2})$

[1] **Abbreviations**: C = "convex", NC = "non-convex", DC = "difference of convex", SC = "strongly convex", S = "Smooth", Obj = "objective", Ctr = "constraint", GD = "gradient descent", ND = "non-degenerate", RD= "radial dual", LOO = "linear optimization oracle", QOO = "quadratic optimization oracle", MO = "membership oracle".
[2] **Step-size**: (i) vanishing step-size: depends on ϵ, (ii) diminishing step-size: decreases as poly$(1/K)$ with the number of iterations K, (iii) constant step-size: is independent of both ϵ and K, and (iv) implicit step-size: has implicit parameters such as smoothness and optimal objective.
[3] **Convergence Rate**: number of iterations for finding an ϵ-approximate stationary point for non-convex optimizations or an ϵ-approximate optimum for convex optimizations.
[4] The radial dual (RD) of a convex constrained problem is an unconstrained min-max problem [Gri24a, Gri24b].

To circumvent these issues, projection-free methods based on the Frank-Wolfe (FW) algorithm [FW$^+$56] have been widely studied (e.g., [MZWG16, THZK21, Mha22, MHSY25]). These methods employ a linear optimization oracle (LOO) at each iteration instead of projections, with the former often being performed efficiently [CP21]. However, LOO can still be computationally expensive over complex constrained sets; therefore, FW methods are confined to scenarios where the LOO is efficient. Moreover, they exhibit oscillatory behavior near the solution, resulting in slow convergence [BRZ24, FG16]. Beyond FW methods, penalty-based approaches [Ber76, SCB$^+$97, LMX22] struggle with ill-conditioning as penalty parameters increase and perform poorly, particularly when dealing with complex constraints. Recent advances explore projection-free strategies leveraging techniques such as *reparameterization* [LMY23, TT24, CV25] and *radial dual* reformulation [Gri24a, Gri24b]. These methods highlight recent efforts to address the limitations of classical projection-free methods, but remain restricted to structured constraint sets and may suffer from practical drawbacks, such as impractical step-size choices (see Table 1). Please refer to Appendix A for a detailed discussion of related work to reduce the per-iteration cost and accelerate convergence for solving convex-constrained optimization. Despite the success of existing projection-free methods, the research gap still remains:

Can we design a projection-free approach for optimization over a general compact convex set with desirable properties, including fast convergence and cheap per-iteration cost?

In this paper, we propose a novel *projection-free* framework that positively answers this question. Concretely, we make the following contributions:

$\triangleright$ In Sec. 3, we design the novel projection-free method, termed as **Hom-PGD**, that re-parameterizes the optimization over convex compact sets to equivalent ball-constrained optimization. By solving the equivalent problem with gradient descent with closed-form projection and mapping the converged solution back, we obtain the solution to the original constrained problem.

$\triangleright$ In Sec. 4, we establish convergence and complexity analysis for **Hom-PGD**: $\mathcal{O}(1/\epsilon)$ in the convex setting, $\mathcal{O}(\log 1/\epsilon)$ in the strongly convex setting, and $\mathcal{O}(1/\epsilon^2)$ in the non-convex setting, where established convergence rates are *optimal* [2] under unaccelerated settings. Moreover, the per-iteration complexity of Hom-PGD is cheap as $\mathcal{O}(n^2)$ without linear/quadratic optimization oracles. We also extend our framework to optimization over certain non-convex sets in Sec. 5.

[2]Optimal convergence rate means it matches the lower bound on iteration complexity. (i) The optimal convergence is $\mathcal{O}(\epsilon^{-2})$ in the non-convex smooth setting [CDHS20]; (ii) For constant step-size, the optimal convergence rate for GD is $\mathcal{O}(\log 1/\epsilon)$ in the strongly convex smooth setting and $\mathcal{O}(1/\epsilon)$ in the convex smooth setting; see e.g. [AP23].

$\triangleright$ In Sec. 6, through extensive numerical experiments over convex and non-convex problems, including applications to max-cut SDP problems, we demonstrate **Hom-PGD** outperforms existing first-order approaches in computational efficiency, which achieve similar convergence rate but significantly lower per-iteration cost (up to 3-5 orders of magnitude).

To the best of our knowledge, the proposed **Hom-PGD** is the *first* projection-free, first-order framework capable of solving optimization over general convex compact sets while achieving the *optimal* convergence rate under unaccelerated settings without expensive optimization oracles.

2 Problem Statement

We consider the following continuous convex constrained optimization problem:

$$\min_{\mathbf{x}} \ f(\mathbf{x}), \quad \text{s.t. } \mathbf{x} \in \mathcal{K}, \tag{\mathbf{P}}$$

where $\mathbf{x} \in \mathbb{R}^n$ is the decision variable, $f(\cdot)$ is the objective function, and the constraint set $\mathcal{K} \subset \mathbb{R}^n$ is compact and convex. For ease of analysis and without loss of generality, we assume the constraint set $\mathcal{K}$ is defined by inequalities[3] as $\mathcal{K} = \{\mathbf{x} \in \mathbb{R}^n \mid \mathbf{g}(\mathbf{x}) \leq \mathbf{0}\}$ with $\mathbf{g} = (g_1, \cdots, g_m)$, where $g_i : \mathbb{R}^n \to \mathbb{R}$ are functions.

Open Issues: Although convex optimization has been extensively studied, existing methods face significant limitations when dealing with complex constraints. As discussed in Sec. 1, projection-based approaches incur high computational costs beyond simple sets, while projection-free methods such as FW and primal-dual approaches are largely restricted to structured convex sets. For example, when solving semidefinite programming (SDP), the LOO required by FW involves solving an SDP itself, defeating its purpose as a low-cost alternative. These challenges underscore the need for projection-free algorithms that preserve fast convergence and maintain computational efficiency across broader convex programs.

3 Homeomorphic Optimization Approach

Motivated by recent advances in low-complexity schemes and reparameterization techniques for solving constrained optimization problems [LCL23, LMY23, PP23, LCL24, RSW24, LC25], we propose a novel approach that transforms the original constrained problem via a homeomorphic mapping between the convex constraint set $\mathcal{K}$ and the unit ball $\mathcal{B}$. This transformation preserves the essential problem structure while replacing the potentially complex constraint set with the geometrically simple unit ball, thereby enabling efficient gradient-based optimization without expensive projection operations.

Definition 3.1 (Homeomorphic Constrained Optimization)**.** Given a homeomorphism $\boldsymbol{\psi} : \mathcal{B} \to \mathcal{K}$, we define the transformed optimization problem with objective function $h(\mathbf{z}) = (f \circ \boldsymbol{\psi})(\mathbf{z})$ and constraint set $\mathcal{B} = \boldsymbol{\psi}^{-1}(\mathcal{K})$ as:

$$\min_{\mathbf{z}} \ h(\mathbf{z}), \quad \text{s.t. } \mathbf{z} \in \mathcal{B} \tag{\mathbf{H}}$$

Homeomorphism (or homeomorphic mapping) is a bi-continuous bijection from two topological spaces, guaranteeing the topological equivalence. It is a classic result that any compact convex set is homeomorphic to a unit ball [Ges12, Bre13], i.e., there exists a homeomorphism $\boldsymbol{\psi}$ such that $\mathcal{B} = \boldsymbol{\psi}^{-1}(\mathcal{K})$ and $\mathcal{K} = \boldsymbol{\psi}(\mathcal{B})$. Thus, we can transform any optimization problem $\mathbf{P}$ over a compact convex set into a ball-constrained program $\mathbf{H}$. In practice, this transformation relies on an explicit homeomorphic mapping, which we will discuss how to obtain in Sec. 3.2.

Remark: The transformed problem $\mathbf{H}$ has a *non-convex* objective function $h(\cdot)$ due to the non-linear mapping $\boldsymbol{\psi}$ even though the original objective is convex, but it features a simple constraint set as a unit ball, leading to a closed-form projection. Moreover, under the homeomorphic transformation, the original problem and its homeomorphic counterpart are equivalent, i.e., there exists a bijective correspondence between their optimal solution sets $\mathbf{P}^*$ and $\mathbf{H}^*$, where $\mathbf{P}^* = \{\mathbf{x} \mid \mathbf{x} \in \arg\min\{\mathbf{P}\}\}$ and similarly for $\mathbf{H}^*$. Specifically, for any $\mathbf{x} \in \mathbf{P}^*$, there exists a unique $\mathbf{z} \in \mathbf{H}^*$ such that $\mathbf{x} = \boldsymbol{\psi}(\mathbf{z})$, and vice versa. Thus, we can solve the transformed problem $\mathbf{H}$ without expensive projection to obtain the corresponding optimal solution of the original problem $\mathbf{P}$.

[3]Linear equality constraints can be removed without loss of generality, see Appendix B.1 for discussions.

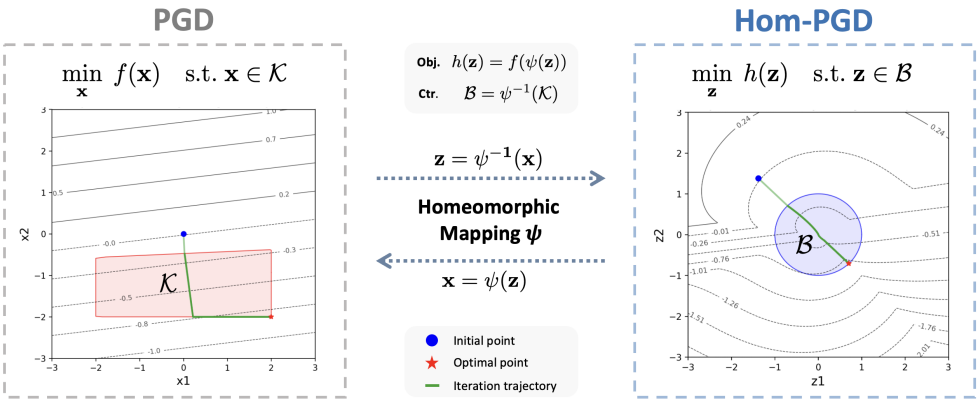


Figure 1: **Hom-PGD Framework**: Hom-PGD conducts the standard PGD algorithm in the transformed space by a homeomorphic mapping $\psi(\cdot)$, where the transformed constraint set $\mathcal{B}$ is a simple unit ball, and the transformed objective function $h(\cdot)$ is non-convex but structured.

3.1 Algorithm Overview

As illustrated in Fig. 1, PGD in the original space suffers from expensive projection operations during iteration over the constraint boundary. To solve problem $\mathbf{P}$ without expensive projection, we transform problem $\mathbf{P}$ into a ball-constrained optimization $\mathbf{H}$ by a homeomorphic mapping ψ. Then we apply regular PGD to efficiently solve the homeomorphic optimization $\mathbf{H}$ with a closed-form projection, thereby termed *projection-free*[4] methods. Finally, we map the obtained solution back to the original space to recover the corresponding solution for the original problem.

Algorithm 1 Hom-PGD

Input: initial point $\mathbf{z}_0$, problem $\mathbf{H}$ with ψ and maximum iteration number K
for $k = 0$ **to** K **do**
  Compute stepsize α_k
  Update: $\mathbf{z}_{k+1} = \Pi_{\mathcal{B}}\left(\mathbf{z}_k - \alpha_k \nabla h(\mathbf{z}_k)\right)$
end for
Output: $\mathbf{x}_K = \psi(\mathbf{z}_K)$

We call the combination of homeomorphic transformation and regular PGD as **Hom-PGD**, shown in Alg. 1. Next, we discuss how to construct an explicit ψ for a general compact convex set $\mathcal{K}$.

3.2 Construction of Homeomorphism

We introduce an explicit-form homeomorphic mapping between convex set $\mathcal{K}$ and unit ball $\mathcal{B}$ as follows, termed Gauge mapping [TZ22, LLC25]:

Definition 3.2 (Gauge Mapping)**.** Let $\gamma_{\mathcal{K}}(\mathbf{x}, \mathbf{x}^\circ) = \inf\{\lambda \geq 0 \mid \mathbf{x} \in \lambda(\mathcal{K} - \mathbf{x}^\circ)\}$ be the Gauge/Minkowski function [BM08] given an interior point $\mathbf{x}^\circ \in \operatorname{int}(\mathcal{K})$. The gauge mapping $\psi : \mathcal{B} \to \mathcal{K}$ is defined between a unit ball and a compact convex set:

$$\psi(\mathbf{z}) = \frac{\|\mathbf{z}\|}{\gamma_{\mathcal{K}}(\mathbf{z}, \mathbf{x}^\circ)}\mathbf{z} + \mathbf{x}^\circ, \ \forall \mathbf{z} \in \mathcal{B}; \qquad \psi^{-1}(\mathbf{x}) = \frac{\gamma_{\mathcal{K}}(\mathbf{x} - \mathbf{x}^\circ, \mathbf{x}^\circ)}{\|\mathbf{x} - \mathbf{x}^\circ\|}(\mathbf{x} - \mathbf{x}^\circ), \ \forall \mathbf{x} \in \mathcal{K}. \tag{1}$$

First, gauge mapping ψ establishes a homeomorphism between any compact convex set and the unit ball, ensuring that $\mathcal{K} = \psi(\mathcal{B})$ and $\mathcal{B} = \psi^{-1}(\mathcal{K})$. Intuitively, this mapping transforms the unit ball by first translating it to align with an interior point of the convex set, then scaling points radially outward from this interior point until the ball's boundary conforms to the convex set's boundary (illustrated in Fig. 2). Moreover, the gauge mapping has *closed-form* expressions for common convex sets (linear, quadratic, second-order cone, and linear matrix inequality constraints) and can be efficiently computed via *bisection* methods for general convex constraints. For a comprehensive property and computation of gauge mapping, we refer readers to Appendix B.

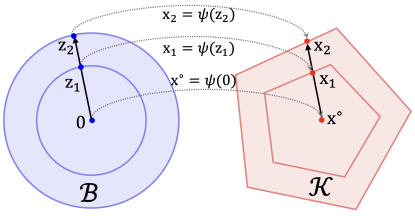


Figure 2: Gauge mapping illustration.

[4]The term "projection-free" specifically refers to avoiding expensive projections onto original complex constraint sets, a usage that aligns with standard conventions in the literature [LMY23, LBGH23].

Leveraging this explicit homeomorphic gauge mapping $\boldsymbol{\psi}$, we transform problem $\mathbf{P}$ to problem $\mathbf{H}$, and apply Hom-PGD (Alg. 1) to solve it without expensive projection. However, the gauge mapping $\boldsymbol{\psi}$ depends on the choice of interior point $\mathbf{x}^\circ$ (shown in Def. 3.2). Different choices of $\mathbf{x}^\circ$ yield distinct gauge mappings that alter the landscape of the transformed problem $\mathbf{H}$ and affect the convergence behavior of Hom-PGD. As we will establish rigorously in Sec. 4, gauge mappings with smaller Lipschitz constants favor faster convergence of Hom-PGD. Thus, we proceed to analyze the Lipschitz properties of the gauge mapping as:

Proposition 3.3 (Bi-Lipschitz Constants of the Gauge Mapping)**.** *Let $\mathcal{K} \subset \mathbb{R}^n$ be a compact convex set and let $\mathbf{x}^\circ \in \operatorname{int}(\mathcal{K})$ be an interior point. Define the inner and outer radii with respect to $\mathbf{x}^\circ$ as*

$$r_{\mathrm{i}} := \sup\{r \geq 0 : \mathcal{B}(\mathbf{x}^\circ, r) \subseteq \mathcal{K}\}, \quad r_{\mathrm{o}} := \inf\{r \geq 0 : \mathcal{K} \subseteq \mathcal{B}(\mathbf{x}^\circ, r)\},$$

such that $\mathcal{B}(\mathbf{x}^\circ, r_{\mathrm{i}}) \subseteq \mathcal{K} \subseteq \mathcal{B}(\mathbf{x}^\circ, r_{\mathrm{o}})$. Then the Lipschitz constant (denoted as $\mathrm{L}(\cdot)$) of gauge mapping $\boldsymbol{\psi}$ associated with $\mathcal{K}$ satisfies the following bounds:

$$\textit{Forward Lipschitz:} \quad \kappa_2 := \mathrm{L}(\boldsymbol{\psi}) \leq 2\, r_{\mathrm{o}} + r_{\mathrm{o}}^2 / r_{\mathrm{i}}, \quad \textit{Inverse Lipschitz:} \quad \frac{1}{\kappa_1} := \mathrm{L}(\boldsymbol{\psi}^{-1}) \leq 2 / r_{\mathrm{i}}.$$

Therefore, to reduce the Lipschitz constant of the gauge mapping and boost the convergence of Hom-PGD, we can select a "central" interior point with large inner radius r_{i} or small outer radius r_{o}. In practice, we may solve a convex problem by minimizing the constraint residual to find a "central" interior point approximately following [THH23] (refer to Appendix B for details).

4 Performance Analysis

In this section, we present a comprehensive performance analysis for Hom-PGD, including the landscape analysis, convergence rate, and run-time complexity.

General Assumptions (with details in Appendix C.2): The objective f and constrained functions g_i in $\mathbf{P}$ are continuously differentiable and smooth. The homeomorphic mapping $\boldsymbol{\psi}$ is invertible with a non-singular Jacobian matrix and is (κ_1, κ_2)-bi-Lipschitz continuous. Additionally, the Jacobian matrix of $\boldsymbol{\psi}$ (denoted as $\mathrm{J}_{\boldsymbol{\psi}}$) exists and is Lipschitz continuous.

We remark that our theoretical results hold for *any* homeomorphism satisfying these assumptions, and we construct a specific homeomorphism, the gauge mapping, in practice. Moreover, gauge mapping meets these assumptions (with details in Appendix B.3).

4.1 Landscape Analysis

First, under general assumptions, the composite function $h = f \circ \boldsymbol{\psi}$ in problem $\mathbf{H}$ inherits favorable properties as follows (more properties of h are included Lemma D.1).

Lemma 4.1. *Denote $L_{h,0}, L_h, \kappa_2, L_{\boldsymbol{\psi}}$ as the Lipschitz constant of $f, \nabla f, \boldsymbol{\psi}, \mathrm{J}_{\boldsymbol{\psi}}$ respectively. Then h is $L_{h,0} := L_{f,0}\kappa_2$ Lipschitz continuous and h is L_h-smooth with $L_h = \kappa_2^2 L_f + L_{\boldsymbol{\psi}} L_{f,0}$.*

Next, recall that a point $\mathbf{x}^*$ is said to be a stationary point of problem $\min_{\mathbf{x}\in\mathcal{K}} f(\mathbf{x})$ with convex set $\mathcal{K}$, if $\nabla f(\mathbf{x}^*)^\top(\mathbf{x} - \mathbf{x}^*) \geq 0$ for any $\mathbf{x} \in \mathcal{K}$. It is well-known that any stationary point is a global optimum for a convex constrained optimization problem. A natural question arises: *does this property also hold for the non-convex optimization problem $\mathbf{H}$ under convex problem $\mathbf{P}$*? For unconstrained cases, this property does hold, as the function h is invex with the property that every stationary point of an invex function is a global optimum [Mar85]. For the constrained case, we provide the following formal statement where the proof is deferred to Appendix D.6.

Proposition 4.2 (Global Optimality of $\mathbf{H}$)**.** *Suppose problem $\mathbf{P}$ is a convex optimization. If $\mathbf{z}^*$ is a stationary point of problem $\mathbf{H}$ and LICQ (Def. D.3) holds at $\mathbf{z}^*$, then $\mathbf{x}^* = \boldsymbol{\psi}(\mathbf{z}^*)$ is a stationary point of problem $\mathbf{P}$ (thus a global optimum). Hence, $\mathbf{z}^*$ is a global optimum.*

We remark that the LICQ assumption is mild in our setting. For problem ($\mathbf{H}$), the only constraint is $\|\mathbf{z}\|^2 \leq 1$, so LICQ holds at any boundary point where the constraint is active. Moreover, since there are no equality constraints, LICQ trivially holds at any interior point. Moreover, we derive that there is a one-to-one correspondence for KKT points and non-degenerate stationary points between $\mathbf{P}$ and $\mathbf{H}$. The relevant definitions, formal statements, and proofs are provided in Appendix D.5.

4.2 Convergence Analysis

In this section, we provide a theoretical convergence analysis of Hom-PGD (Alg. 1) for solving problem $\mathbf{H}$. Our main result demonstrates that: *Hom-PGD achieves the same convergence rate as the standard PGD for the (non-)convex problem* $\mathbf{P}$ *under mild regularity conditions, despite operating on the non-convex formulation* $\mathbf{H}$.

Before moving on, we recall some basic definitions. An ϵ-stationary point $\mathbf{x}^*$ for problem $\mathbf{P}$ is defined by $\|G(\mathbf{x}^*)\| \leq \epsilon$ where the *gradient mapping* $G(\mathbf{x}) := G_{1/\alpha}(\mathbf{x}) = \frac{1}{\alpha}\left[\mathbf{x} - \Pi_{\mathcal{K}}\left(\mathbf{x} - \alpha\nabla f(\mathbf{x})\right)\right]$. Note that in the definition, we omit the dependence of the function G on $\mathcal{K}$ and α.

4.2.1 (Strongly) Convex Objective f

The following theorem provides the convergence analysis of Hom-PGD for convex optimization $\mathbf{P}$.

Theorem 1. *Suppose the strict complementary slackness condition (Def. D.4) holds for both problem* $\mathbf{P}$ *and* $\mathbf{H}$*, and the problem* $\mathbf{P}$ *is convex with a non-degenerate*[5] *minimizer* $\mathbf{x}^*$*. Let* $\{\mathbf{z}_k\}$ *be the sequence generated by Hom-PGD with step-size* $\alpha \in (0, \frac{2}{L_h}]$*. For sufficient small* $\epsilon > 0$*,* $\{\mathbf{z}_k\}_{k=1}^K$ *with* $K = \mathcal{O}(L_h/\epsilon)$ *contains* $\mathbf{z}'$ *such that* $h(\mathbf{z}') - h^\star \leq \epsilon$*.*

Proof Intuition. Under the invertible mapping, $\mathbf{z}^* = \boldsymbol{\psi}(\mathbf{x}^*)$ is also a non-degenerate point from Lemma D.10. Consequently, it satisfies local strong convexity or local PL condition, meaning that $h(\mathbf{z}) - h^* \leq \mathcal{O}(\|G(\mathbf{z})\|^2)$ holds within a sufficiently small ball centered at $\mathbf{z}^*$. As a result, it follows from Theorem 3 that PGD only requires $\mathcal{O}(1/\epsilon)$ complexity to find a $\mathcal{O}(\sqrt{\epsilon})$-stationary point $\mathbf{z}'$ such that $h(\mathbf{z}') - h(\mathbf{z}^*) \leq \mathcal{O}(\epsilon)$. The idea is motivated by [LMY23], which applies specific Hadamard parameterization to transform a simplex-constrained optimization to a sphere-constrained optimization. We extend their results to a general homeomorphic mapping $\boldsymbol{\psi}$, and proof details can be found in Appendix E.1.

If the objective in $\mathbf{P}$ is strongly convex, we will have a faster (i.e., linear) convergence rate as:

Theorem 2. *Suppose problem* $\mathbf{P}$ *is convex with* μ_f*-strongly convex objective* f*. Let* $\{\mathbf{z}_k\}_{k\geq 0}$ *be generated by Hom-PGD with proper constant step-size* $\alpha \in (0, \frac{2}{L_h}]$*. With* $K = \mathcal{O}(\kappa \log 1/\epsilon)$ *where* $\kappa = L_h/(\mu_f\kappa_1)$*, we have* $h(\mathbf{z}_K) - h(\mathbf{z}^*) \leq \epsilon$*, and* $\|\mathbf{z}_K - \mathbf{z}^*\| \leq \epsilon$*.*

Proof Intuition. If f is a strongly convex function over a convex set, it satisfies a generalized PL condition. The homeomorphic mapping preserves this generalized PL condition such that the linear convergence can be established for $\mathbf{H}$. Details can be found in Appendix E.2.

Remark. (i) The convergence rates derived in Theorem 1 and 2 match the lower bounds in the unacclerated convex and strongly convex settings (referring to e.g., [AP23]). (ii) Theorem 1 and Theorem 2 demonstrate that the Hom-PGD algorithm not only maintains projection-free properties over the unit ball, reducing per-iteration computational complexity; but also achieves the same convergence rates as standard PGD when applied to the original convex optimization problem $\mathbf{P}$, which is non-trivial since Hom-PGD operates on the non-convex objective in problem $\mathbf{H}$.

4.2.2 Non-convex Objective f

For a non-convex objective f, a classical result (e.g., Theorem 9.15 [Bec14]) can be leveraged to show that Hom-PGD algorithms can converge to an ϵ-stationary point with $\mathcal{O}(1/\epsilon^2)$ iterations.

Theorem 3. *Consider a problem* $\min_{\mathbf{z}\in\mathcal{Z}} h(\mathbf{z})$ *with a convex set* $\mathcal{Z}$*. Suppose* h *is non-convex and differentiable with* L_h*-Lipschitz continuous gradient. Then the sequence* $\{\mathbf{z}_k\}_{k=0}^K$ *with* $K = \mathcal{O}(L_h/\epsilon^2)$ *generated by Hom-PGD algorithm with a constant step-size* $\alpha \in (0, \frac{2}{L_h}]$ *contains an* ϵ*-stationary point* $\mathbf{z}'$*, i.e,* $\|G(\mathbf{z}')\| \leq \epsilon$ *for some* $\mathbf{z}' \in \{\mathbf{z}_k\}_{k=0}^K$*.*

Remark. (i) The convergence rate in Theorem 3 matches the optimal rate for smooth non-convex optimization problems [CDHS20]. (ii) While Theorem 3 also applies to (strongly) convex objectives, it yields slower convergence than standard PGD in these cases, as it fails to exploit the convex structure of problem $\mathbf{P}$ and the hidden convexity of problem $\mathbf{H}$ [FHH23]. This highlights the significance of

[5] A minimizer of an optimization problem is non-degenerate if the Hessian of the Lagrangian function is positive definite in the critical cone of the minimizer. See Appendix D.4 for details.

our results in Theorems 1 and 2, where Hom-PGD achieves the same convergence rates of standard PGD while avoiding expensive projection in the transformed domain. (iii) For non-convex objectives satisfying regularity conditions such as the *KL property* or *error bound conditions*, linear convergence rates as in Theorem 2 can also be achieved. See Remark E.5 for further discussion.

In these sections, we establish convergence results for Hom-PGD across different problem classes (Theorems 1, 2, and 3). The convergence rates depend on the Lipschitz constants of the constructed gauge mapping, specifically: (i) the forward Lipschitz constant κ_2, which relates to the parameter L_h established in Lemma 4.1 and appears in Theorems 1 and 3; and (ii) the inverse Lipschitz constant $1/\kappa_1$, which appears in Theorem 2. As demonstrated in Prop. 3.3, the choice of interior point directly influences the Lipschitz constants of the gauge mapping $\boldsymbol{\psi}$. Consequently, different interior points modify the Hom-PGD convergence rate by constant factors while preserving the fundamental convergence order.

Additionally, our theoretical analysis assumes access to exact gradients and gauge mappings, which is consistent with standard practice in the optimization literature (e.g., [LBGH23]). However, for general convex sets, the gauge mapping is numerically approximated using the bisection algorithm (Alg. 2) to compute both its value and gradient within a specified error tolerance δ at each iteration. This numerical approximation introduces an additional $\mathcal{O}(\delta)$ term in the optimality gap of convergence results [DGN14]. Since δ can be chosen arbitrarily small, this additional error term remains negligible and does not affect the fundamental convergence guarantees of our algorithm.

4.3 Complexity Analysis

In this subsection, we analyze the total run-time complexity of Hom-PGD, including initialization complexity, per-iteration complexity, and last-step complexity.

Oracles. We list specific oracles in Hom-PGD besides general ones (e.g., zeroth-order oracle for explicit function evaluation).

- *Membership oracle*: Given $\mathbf{x} \in \mathbb{R}^n$, this oracle $\mathcal{M}_{\mathcal{K}}(\mathbf{x}) := \mathbb{I}(\mathbf{x} \in \mathcal{K}) : \mathbb{R}^n \to \{0, 1\}$ returns 1 if and only if $\mathbf{x} \in \mathcal{K}$. This oracle performs only feasibility checking without requiring the solution of optimization subproblems. For common convex sets including polyhedra and second-order cones, the membership oracle can be implemented with computational complexity not exceeding $\mathcal{O}(n^2)$ [Mha22], with significantly lower computational burden than LOO or projection in practice.
- *Interior point oracle*: This oracle returns an interior point of $\mathcal{K}$ by solving a convex feasibility problem. This requirement, common in projection-free frameworks [Mha22, Gri24a, Gri24b], can be addressed using first-order methods with $\tilde{\mathcal{O}}(n^2)$[6] complexity or interior-point methods with $\tilde{\mathcal{O}}(n^{3.5})$ complexity. Notably, this oracle is only called **once** for the entire optimization algorithm.

Basic operations in Hom-PGD (with details in Appendix B.4). Hom-PGD requires computing the gradient of $h = f \circ \boldsymbol{\psi}$ per iteration and transforming the final solution via the gauge mapping.

- *Computing gauge mapping* $\boldsymbol{\psi}$: $\tilde{\mathcal{O}}(n)$. To compute the gauge mapping in the general case, one may evaluate the gauge function $\gamma_{\mathcal{K}}(\cdot, \mathbf{x}^\circ)$ to an accuracy ϵ with $\mathcal{O}(\log 1/\epsilon)$ membership oracle calls, plus $\mathcal{O}(n)$ operations for the scalar-vector product in Def. 3.2.
- *Computing gradient of* h: $\tilde{\mathcal{O}}(n^2)$. Numerical differentiation techniques (finite/automatic differentiation [BF97, BPRS18, LBGH23]) can be applied, e.g., computing each component $\nabla_i h$ ($i = 1, 2, \cdots, n$) requires $\mathcal{O}(1)$ zeroth-order oracle calls of f and $\tilde{\mathcal{O}}(n)$ cost for evaluating $\boldsymbol{\psi}$, yielding a total complexity of $\tilde{\mathcal{O}}(n^2)$.

Total run-time complexity of Hom-PGD.

- *Initialization complexity* (IC): One interior point oracle call to obtain an interior point of $\mathcal{K}$.
- *Per-iteration complexity* (PiC): Each iteration cost $\tilde{\mathcal{O}}(n^2)$, comprising gradient computation $\nabla h(\mathbf{z})$ at $\tilde{\mathcal{O}}(n^2)$ and unit ball projection at $\mathcal{O}(n)$.
- *Last-step complexity* (LsC): It costs $\mathcal{O}(n^2)$ to map the converged solution $\mathbf{z}^*$ back to the original space via $\mathbf{x}^* = \boldsymbol{\psi}(\mathbf{z}^*)$.
- *Number of iterations* (#I): Convergence rate varies from different settings, referring to Sec. 4.2.

[6] Here $\tilde{\mathcal{O}}(\cdot)$ hides polynomial logarithmic factors.

In conclusion, the total complexity of Hom-PGD equals to $\mathrm{IC} + \#\mathrm{I} \cdot \mathrm{Pic} + \mathrm{LsC} = \mathcal{O}(n^2 \cdot \#\mathrm{I})$. This complexity is significantly lower than that of second-order methods, which typically incur $\mathcal{O}(n^3)$ per-iteration cost, thereby highlighting the scalability of Hom-PGD to high-dimensional problems. Moreover, our method achieves an optimal convergence rate under the first-order setting, ensuring both efficiency and theoretical soundness.

5 Discussions

5.1 Beyond Vanilla Gradient Methods

Mirror gradient methods: Mirror descent methods use a mirror map to transform points into a dual space, where optimization steps are performed. Essentially, these methods can be viewed as generalized projection methods [BT03]. For example, with a quadratic mirror map, it becomes standard PGD; with a negative entropy mirror map, it recovers exponential gradient descent for simplex-constrained problems. However, for general convex constraints, mirror descent often lacks explicit mirror maps and suffers from high projection complexity, despite having convergence rates comparable to ours [Bec17].

Advanced gradient methods. One may note that we can replace the unaccelerated gradient methods with any advanced optimizers, such as momentum-based or Nesterov-accelerated methods, or the Adam optimizer popular in deep learning [KB14]. Despite their potential for accelerating optimization of the non-convex landscape in problem $\mathbf{H}$, analyzing the convergence behavior by utilizing the hidden convexity structure remains challenging, which opens directions for future research. We also conduct illustrative experiments comparing vanilla gradient descent and Adam in Section 6.4.

Second-order methods. We may also consider second-order methods with projection to optimize problem $\mathbf{H}$. However, significant challenges arise from both computational and theoretical perspectives. From a computational standpoint, evaluating the Hessian of the composite objective $h = f \circ \boldsymbol{\psi}$ requires computing a third-order tensor of the gauge mapping, which is both computationally expensive and memory intensive. From a convergence perspective, analyzing convergence over the non-convex landscape to a global optimum becomes substantially more difficult.

5.2 Extension to Non-Convex Constraints

Star-shaped set. Our framework can naturally extend to optimization over non-convex sets where an explicit homeomorphism $\boldsymbol{\psi}$ exists such that $\boldsymbol{\psi}(\mathcal{B}) = \mathcal{K}$. For star-shaped sets, all points are visible from a star center $\mathbf{x}^\circ$ [Lee10], allowing the construction of a gauge mapping that bijectively maps a standard ball to the set. Such star-shaped constrained problems arise in machine learning tasks such as ℓ_p-constrained adversarial attacks in neural networks [EBMA21]. Applying Hom-PGD to such problems maintains the $\mathcal{O}(1/\epsilon^2)$ convergence rate for finding an ϵ-stationary point per Theorem 3.

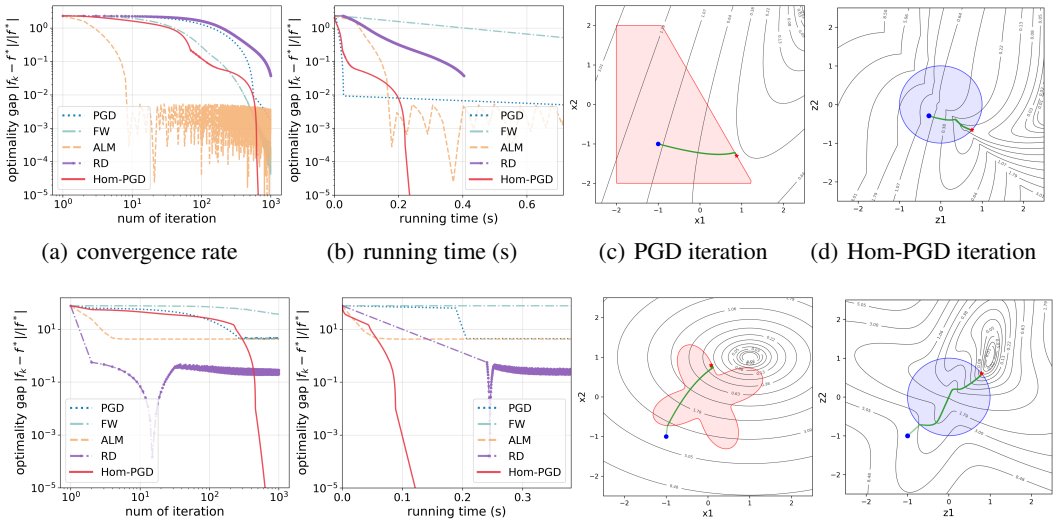
Ball-homeomorphic set. Our framework may also extend to non-convex sets that are homeomorphic to a unit ball. However, determining if a non-convex set is ball-homeomorphic requires examining its topological properties, which is challenging. While a homeomorphism exists for such sets, there is generally no explicit form for the homeomorphism. Learning-based methods for approximating such homeomorphisms [LCL23, LCL24] present promising directions for future research.

General non-convex set. For more general non-convex sets that may not be ball-homeomorphic (e.g., disconnected sets), our framework remains applicable but without optimality guarantees. Given an interior point $\mathbf{x}^\circ$ in $\mathcal{K}$, we can define $\mathcal{X}$ as the largest contained star-shaped set with center $\mathbf{x}^\circ$. Similar to constraint restriction methods [LNDT19], the optimality gap depends on the Hausdorff distance between $\mathcal{K}$ and $\mathcal{X}$.

6 Empirical Study

In this section, we conduct simulations to demonstrate the effectiveness of Hom-PGD on both convex and non-convex constrained optimization problems. The detailed problem formulations, experimental settings, algorithm hyperparameters, and supplementary experiment results are in Appendix F.

Baselines: (i) **PGD**: Regular projected gradient descent applied to problem $\mathbf{P}$, where the projection operation is called at each iteration. (ii) **FW**: Frank-Wolfe methods, which solve an update direction with a linearized objective and update the decision variables. (iii) **ALM**: Augmented Lagrangian



(a) convergence rate (b) running time (s) (c) PGD iteration (d) Hom-PGD iteration

(e) convergence rate (f) running time (s) (g) PGD iteration (h) Hom-PGD iteration

Figure 3: Performance for optimization over **polyhedron** (**a-d**) and **star-shaped** set (**e-h**), respectively. All methods are executed with the same initial points and terminated by a maximum number of iterations (10^3). Complete results are in Appendix G.1 (Fig. 8 - 11).

methods for problem $\mathbf{P}$ that alternately update primal and dual coefficients for the unconstrained formulation to problem $\mathbf{P}$. (iv) **RD**: Radial-Dual framework, which applies radial-dual to formulate the constrained problem into unconstrained min-max optimization. (v) **Hom-PGD**: Projected gradient descent applied to the transformed problem $\mathbf{H}$ shown in Sec. 3.

6.1 Illustrative Examples: Optimization over Polyhedron and Star-shaped Set

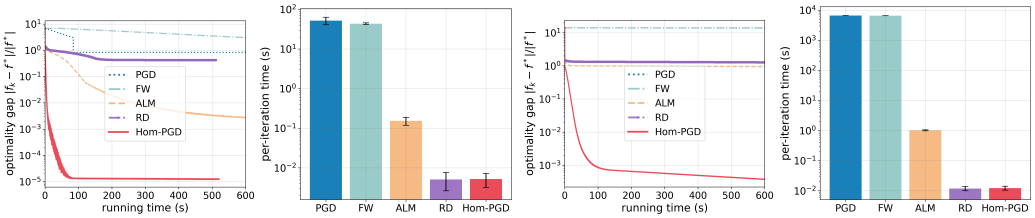
We examine a two-dimensional illustrative optimization problem involving quadratic optimization over both a (*convex*) polyhedron and a (*non-convex*) star-shaped set to demonstrate our method's efficiency. As shown in Fig. 3, Hom-PGD outperforms other first-order algorithms in both settings. The iteration trajectories in the transformed space reveal the mechanism behind this efficiency: Hom-PGD avoids complex projections while effectively performing gradient descent in a structured landscape of problem $\mathbf{H}$ to the optimum, even though it is non-convex.

6.2 Solving Second-order Cone Programming (SOCP)

We next evaluate the performance of algorithms on SOCP, which encompasses fundamental convex programs (LP, QP, convex QCQP) and has widespread applications in portfolio optimization [BBV04] and optimal power flow problems [Low14a]. Problem instances are randomly generated following the CVXPY documents. As shown in Fig. 4, our method not only converges rapidly to the target error tolerance but also demonstrates significantly lower per-iteration costs (up to 3^{-5} orders of magnitude) compared to projection-based or Frank-Wolfe methods, since Hom-PGD does not need complex optimization oracles such as projection or LOO during iterations. Further, we also use a commercial solver, **MOSEK**, which typically applies primal-dual interior point methods to solve convex programs. Notably, the solver costs 5424 seconds to solve the 1000-dim instance, while Hom-PGD takes less than 600 seconds to reach a 10^{-3} objective optimality gap.

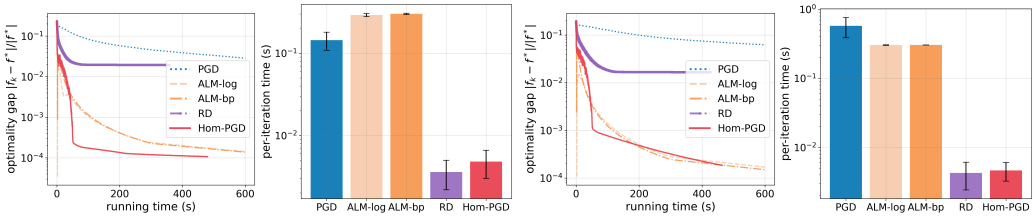
6.3 Solving Max-Cut Semi-Definite Programming (SDP)

We further evaluate our method on the more challenging max-cut SDP problem. While max-cut is an NP-hard combinatorial problem, SDP relaxation with randomized rounding achieves an expected approximation ratio of 0.878 [GW95]. We generate random *Erdős-Rényi* graphs as test instances [HSS08]. Since the optimum of the max-cut SDP is typically low-rank, the gauge mapping encounters non-differentiability at these solutions. To address this practical issue, we apply the smoothing techniques described in Appendix B.3.2. As shown in Figure 5, our approach demonstrates efficient optimization even in high-dimensional decision spaces (50^2 variables) with positive semi-definite cone constraints. Notably, Hom-PGD exhibits a *slower* convergence rate on SDP with linear objectives



(a) $(n, m) = (100, 1000)$ (b) $(n, m) = (1000, 2500)$

Figure 4: **Performance over SOCP**: n is the number of decision variables and m is the number of constraints. All methods are executed with the same initial points and terminated by a maximum iterations (10^5) or running time (600 seconds). Complete results are in Appendix G.2 (Fig. 12-14).



(a) $n = 30^2$ (b) $n = 50^2$

Figure 5: **Performance over SDP**: there are $n = \mathcal{O}(N^2)$ decision variables where N is the graph size. All methods are executed with the same initial points and terminated by a maximum of iterations (10^5) or running time (600 seconds). Complete results are in Appendix G.3 (Fig. 15-19).

compared to SOCP with quadratic objectives. This behavior aligns with our theoretical analysis in Theorems 1 and 2, where strongly convex (quadratic) objectives yield faster convergence rates than convex (linear) objectives. The ALM methods solve the Burer-Monteiro SDP formulation [BM03] with log-rank ($\log N$) and Barvinok-Pataki (bp)-rank ($\sqrt{2N}$) [Bar95, Pat98, BVB20]. Despite the scalability of this low-rank formulation, it incurs violation on additional equality constraint and high iteration cost for each inner minimization. However, the per-iteration complexity of our method still outperforms other approaches, resulting in comparable convergence in terms of total running time.

6.4 Scalability Tests and Ablation Study

We first evaluate the scalability of gauge mapping computation across various constraints and dimensions in Fig. 6, demonstrating efficiency (less than `0.01` seconds) up to `3000`-dimensional constraints. Our ablation study further examines critical framework components: (i) interior point selection (Fig. 20), confirming that central points (smaller Lipschitz) accelerate convergence as predicted by our theory analysis in Sec. 4.2; and (ii) gradient method variants (Fig. 21), revealing that advanced optimization techniques (e.g., Adam [KB14]) further enhance performance for solving non-convex problem **H**, suggesting promising directions for future research.

7 Conclusion and Limitations

In this work, we propose **Hom-PGD**, a projection-free method that transforms constrained optimization over general convex (and certain non-convex) sets into a ball-constrained problem via a homeomorphism. Hom-PGD achieves optimal convergence rates with $\mathcal{O}(n^2)$ per-iteration complexity without expensive projections or oracles. Numerical results show competitive convergence with significantly lower iteration costs. Despite its efficiency, there are several **limitations** to be addressed in future work: (i) Extending Hom-PGD to more general non-convex sets is non-trivial, as discussed in Sec. 5. (ii) From the convergence theory perspective, while Hom-PGD achieves optimal convergence rates under unaccelerated settings, it remains an open question whether acceleration techniques (e.g., Nesterov-style methods) can be incorporated to attain optimal accelerated rates. The challenge stems from the non-convexity of the transformed problem **H**. (iii) The gauge mapping used in this work serves as a simple and explicit homeomorphism but may not be the optimal choice in terms of conditioning or convergence behavior. Exploring alternative homeomorphisms tailored to specific problem structures could further improve performance.

Acknowledments

This work is supported in part by a General Research Fund from Research Grants Council, Hong Kong (Project No. 11214825), a Collaborative Research Fund from Research Grants Council, Hong Kong (Project No. C1049-24G), an InnoHK initiative, The Government of the HKSAR, Laboratory for AI-Powered Financial Technologies, a Shenzhen-Hong Kong-Macau Science & Technology Project (Category C, Project No. SGDX20220530111203026), and a Start-up Research Grant from The Chinese University of Hong Kong, Shenzhen (Project No. UDF01004086). The authors would also like to thank the anonymous reviewers for their helpful comments.

References

[AFP23] Ademir Alves Aguiar, Orizon Pereira Ferreira, and Leandro F Prudente. Inexact gradient projection method with relative error tolerance. *Computational Optimization and Applications*, 84(2):363–395, 2023.

[AP23] Jason Altschuler and Pablo Parrilo. Acceleration by stepsize hedging: Multi-step descent and the silver stepsize schedule. *Journal of the ACM*, 2023.

[B+15] Sébastien Bubeck et al. Convex optimization: Algorithms and complexity. *Foundations and Trends® in Machine Learning*, 8(3-4):231–357, 2015.

[Bar95] Alexander I. Barvinok. Problems of distance geometry and convex properties of quadratic maps. *Discrete & Computational Geometry*, 13:189–202, 1995.

[BBV04] Stephen Boyd, Stephen P Boyd, and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

[Bec14] Amir Beck. *Introduction to nonlinear optimization: Theory, algorithms, and applications with MATLAB*. SIAM, 2014.

[Bec17] Amir Beck. *First-order methods in optimization*. SIAM, 2017.

[Ber76] Dimitri P Bertsekas. On penalty and multiplier methods for constrained minimization. *SIAM Journal on Control and Optimization*, 14(2):216–235, 1976.

[BF97] Richard L Burden and J Douglas Faires. Numerical analysis, brooks, 1997.

[BM03] Samuel Burer and Renato DC Monteiro. A nonlinear programming algorithm for solving semidefinite programs via low-rank factorization. *Mathematical programming*, 95(2):329–357, 2003.

[BM08] Franco Blanchini and Stefano Miani. *Set-theoretic methods in control*, volume 78. Springer, 2008.

[BMR03] Ernesto G Birgin, José Mario Martínez, and Marcos Raydan. Inexact spectral projected gradient methods on convex sets. *IMA Journal of Numerical Analysis*, 23(4):539–559, 2003.

[Bou23] Nicolas Boumal. *An introduction to optimization on smooth manifolds*. Cambridge University Press, 2023.

[BPRS18] Atilim Gunes Baydin, Barak A Pearlmutter, Alexey Andreyevich Radul, and Jeffrey Mark Siskind. Automatic differentiation in machine learning: a survey. *Journal of machine learning research*, 18(153):1–43, 2018.

[BPTW19] Gábor Braun, Sebastian Pokutta, Dan Tu, and Stephen Wright. Blended conditonal gradients. In *International conference on machine learning*, pages 735–743. PMLR, 2019.

[Bre13] Glen E Bredon. *Topology and geometry*, volume 139. Springer Science & Business Media, 2013.

[BRTW22] Bubacarr Bah, Holger Rauhut, Ulrich Terstiege, and Michael Westdickenberg. Learning deep linear neural networks: Riemannian gradient flows and convergence to global minimizers. *Information and Inference: A Journal of the IMA*, 11(1):307–353, 2022.

[BRZ24] Immanuel M Bomze, Francesco Rinaldi, and Damiano Zeffiro. Frank–wolfe and friends: a journey into projection-free first-order optimization methods. *Annals of Operations Research*, 343(2):607–638, 2024.

[BT03] Amir Beck and Marc Teboulle. Mirror descent and nonlinear projected subgradient methods for convex optimization. *Operations Research Letters*, 31(3):167–175, 2003.

[BVB20] Nicolas Boumal, Vladislav Voroninski, and Afonso S Bandeira. Deterministic guarantees for burer-monteiro factorizations of smooth semidefinite programs. *Communications on Pure and Applied Mathematics*, 73(3):581–608, 2020.

[BZK18] Leonard Berrada, Andrew Zisserman, and M Pawan Kumar. Deep frank-wolfe for neural network optimization. *arXiv preprint arXiv:1811.07591*, 2018.

[CDHS20] Yair Carmon, John C Duchi, Oliver Hinder, and Aaron Sidford. Lower bounds for finding stationary points i. *Mathematical Programming*, 184(1):71–120, 2020.

[CHHY22] Xin Chen, Niao He, Yifan Hu, and Zikun Ye. Efficient algorithms for minimizing compositions of convex functions and random functions and its applications in network revenue management. *arXiv preprint arXiv:2205.01774*, 2022.

[Cif21] Diego Cifuentes. On the burer–monteiro method for general semidefinite programs. *Optimization Letters*, 15(6):2299–2309, 2021.

[CJJ+20] Yair Carmon, Arun Jambulapati, Qijia Jiang, Yujia Jin, Yin Tat Lee, Aaron Sidford, and Kevin Tian. Acceleration with a ball optimization oracle. *Advances in Neural Information Processing Systems*, 33:19052–19063, 2020.

[CP11] Antonin Chambolle and Thomas Pock. A first-order primal-dual algorithm for convex problems with applications to imaging. *Journal of mathematical imaging and vision*, 40:120–145, 2011.

[CP20] Cyrille Combettes and Sebastian Pokutta. Boosting frank-wolfe by chasing gradients. In *International conference on machine learning*, pages 2111–2121. PMLR, 2020.

[CP21] Cyrille W Combettes and Sebastian Pokutta. Complexity of linear minimization and projection on some sets. *Operations Research Letters*, 49(4):565–571, 2021.

[CV25] James Chok and Geoffrey M Vasil. Optimization over a probability simplex. *Journal of Machine Learning Research*, 26(73):1–35, 2025.

[DFF56] George Bernard Dantzig, Lester R Ford, and Delbert Ray Fulkerson. A primal-dual algorithm for linear programs. *Linear inequalities and related systems*, 38:171–182, 1956.

[DGN14] Olivier Devolder, François Glineur, and Yurii Nesterov. First-order methods of smooth convex optimization with inexact oracle. *Mathematical Programming*, 146(1):37–75, 2014.

[DKP+24] Oscar Davis, Samuel Kessler, Mircea Petrache, Ismail Ilkan Ceylan, Michael Bronstein, and Avishek Joey Bose. Fisher flow matching for generative modeling over discrete data. *arXiv preprint arXiv:2405.14664*, 2024.

[DP19] Dmitriy Drusvyatskiy and Courtney Paquette. Efficiency of minimizing compositions of convex functions and smooth maps. *Mathematical Programming*, 178:503–558, 2019.

[DR70] Vladimir Fedorovich Demianov and Aleksandr Moiseevich Rubinov. Approximate methods in optimization problems. *(No Title)*, 1970.

[Dun79] Joseph C Dunn. Rates of convergence for conditional gradient algorithms near singular and nonsingular extremals. *SIAM Journal on Control and Optimization*, 17(2):187–211, 1979.

[EBMA21] Ecenaz Erdemir, Jeffrey Bickford, Luca Melis, and Sergul Aydore. Adversarial robustness with non-uniform perturbations. *Advances in Neural Information Processing Systems*, 34:19147–19159, 2021.

[Eva18] LawrenceCraig Evans. *Measure theory and fine properties of functions*. Routledge, 2018.

[FFK98] Francisco Facchinei, Andreas Fischer, and Christian Kanzow. Regularity properties of a semismooth reformulation of variational inequalities. *SIAM Journal on Optimization*, 8(3):850–869, 1998.

[FG16] Robert M Freund and Paul Grigas. New analysis and results for the frank–wolfe method. *Mathematical Programming*, 155(1):199–230, 2016.

[FHH23] Ilyas Fatkhullin, Niao He, and Yifan Hu. Stochastic optimization under hidden convexity. *arXiv preprint arXiv:2401.00108*, 2023.

[FLP22] Orizon Pereira Ferreira, Max Lemes, and Leandro F Prudente. On the inexact scaled gradient projection method. *Computational Optimization and Applications*, pages 1–35, 2022.

[FW+56] Marguerite Frank, Philip Wolfe, et al. An algorithm for quadratic programming. *Naval research logistics quarterly*, 3(1-2):95–110, 1956.

[Ges12] Stefan Geschke. Convex open subsets of rn are homeomorphic to n-dimensional open balls. *Preprint, http://relaunch. hcm. uni-bonn. de/fileadmin/geschke/papers/ConvexOpen. pdf*, 2012.

[GH15] Dan Garber and Elad Hazan. Faster rates for the frank-wolfe method over strongly-convex sets. In *International Conference on Machine Learning*, pages 541–549. PMLR, 2015.

[GH16] Dan Garber and Elad Hazan. A linearly convergent variant of the conditional gradient algorithm under strong convexity, with applications to online and stochastic optimization. *SIAM Journal on Optimization*, 26(3):1493–1528, 2016.

[GLRR25] Kaja Gruntkowska, Hanmin Li, Aadi Rane, and Peter Richtárik. The ball-proximal (=" broximal") point method: a new algorithm, convergence theory, and applications. *arXiv preprint arXiv:2502.02002*, 2025.

[GLS81] Martin Grötschel, László Lovász, and Alexander Schrijver. The ellipsoid method and its consequences in combinatorial optimization. *Combinatorica*, 1(2):169–197, 1981.

[GM86] Jacques Guélat and Patrice Marcotte. Some comments on wolfe's 'away step'. *Mathematical Programming*, 35(1):110–119, 1986.

[Gri24a] Benjamin Grimmer. Radial duality part i: foundations. *Mathematical Programming*, 205(1):33–68, 2024.

[Gri24b] Benjamin Grimmer. Radial duality part ii: applications and algorithms. *Mathematical Programming*, 205(1):69–105, 2024.

[GW95] Michel X Goemans and David P Williamson. Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. *Journal of the ACM (JACM)*, 42(6):1115–1145, 1995.

[HHZ17] Mingyi Hong, Davood Hajinezhad, and Ming-Min Zhao. Prox-pda: The proximal primal-dual algorithm for fast distributed nonconvex optimization and learning over networks. In *International Conference on Machine Learning*, pages 1529–1538. PMLR, 2017.

[HL17] Mingyi Hong and Zhi-Quan Luo. On the linear convergence of the alternating direction method of multipliers. *Mathematical Programming*, 162(1):165–199, 2017.

[HLB20] Wooseok Ha, Haoyang Liu, and Rina Foygel Barber. An equivalence between critical points for rank constraints versus low-rank factorizations. *SIAM Journal on Optimization*, 30(4):2927–2955, 2020.

[HP90] Patrick T Harker and Jong-Shi Pang. Finite-dimensional variational inequality and nonlinear complementarity problems: a survey of theory, algorithms and applications. *Mathematical programming*, 48(1):161–220, 1990.

[HSS08] Aric Hagberg, Pieter J Swart, and Daniel A Schult. Exploring network structure, dynamics, and function using networkx. Technical report, Los Alamos National Laboratory (LANL), Los Alamos, NM (United States), 2008.

[JTFF14] Armand Joulin, Kevin Tang, and Li Fei-Fei. Efficient image and video co-localization with frank-wolfe algorithm. In *European conference on computer vision*, pages 253–268. Springer, 2014.

[KB14] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[KLLJS21] Thomas Kerdreux, Lewis Liu, Simon Lacoste-Julien, and Damien Scieur. Affine invariant analysis of frank-wolfe on strongly convex sets. In *International conference on machine learning*, pages 5398–5408. PMLR, 2021.

[KNS16] Hamed Karimi, Julie Nutini, and Mark Schmidt. Linear convergence of gradient and proximal-gradient methods under the polyak-łojasiewicz condition. In *Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2016, Riva del Garda, Italy, September 19-23, 2016, Proceedings, Part I 16*, pages 795–811. Springer, 2016.

[Koj89] M Kojima. A primal-dual interior point algo-rithm for linear programming. *Progress in mathematical programming: Interior point and related methods/Springer-Verlag*, 1989.

[KS15] Aritra Konar and Nicholas D Sidiropoulos. Hidden convexity in qcqp with toeplitz-hermitian quadratics. *IEEE Signal Processing Letters*, 22(10):1623–1627, 2015.

[Lan13] Guanghui Lan. The complexity of large-scale convex programming under a linear optimization oracle. *arXiv preprint arXiv:1309.5550*, 2013.

[LBGH23] Zhou Lu, Nataly Brukhim, Paula Gradu, and Elad Hazan. Projection-free adaptive regret with membership oracles. In *International Conference on Algorithmic Learning Theory*, pages 1055–1073. PMLR, 2023.

[LC25] Enming Liang and Minghua. Chen. Efficient bisection projection to ensure neural-network solution feasibility for optimization over general set. In *International Conference on Machine Learning*. PMLR, 2025.

[LCL23] Enming Liang, Minghua Chen, and Steven H. Low. Low complexity homeomorphic projection to ensure neural-network solution feasibility for optimization over (non-)convex set. In *International Conference on Machine Learning*. PMLR, 2023.

[LCL24] Enming Liang, Minghua Chen, and Steven H. Low. Homeomorphic projection to ensure neural-network solution feasibility for constrained optimization. *Journal of Machine Learning Rsearch*, 2024.

[Lee10] John Lee. *Introduction to topological manifolds*, volume 202. Springer Science & Business Media, 2010.

[LG23] Ning Liu and Benjamin Grimmer. Gauges and accelerated optimization over smooth and/or strongly convex sets. *arXiv preprint arXiv:2303.05037*, 2023.

[LKB24] Eitan Levin, Joe Kileel, and Nicolas Boumal. The effect of smooth parametrizations on nonconvex optimization landscapes. *Mathematical Programming*, pages 1–49, 2024.

[LL12] John M Lee and John M Lee. *Smooth manifolds*. Springer, 2012.

[LLC25] Xinpeng Li, Enming Liang, and Minghua Chen. Gauge flow matching for efficient constrained generative modeling over general convex set. In *ICLR 2025 Workshop on Deep Generative Model in Machine Learning: Theory, Principle and Efficacy*, 2025.

[LMX22] Qihang Lin, Runchao Ma, and Yangyang Xu. Complexity of an inexact proximal-point penalty method for constrained smooth non-convex optimization. *Computational optimization and applications*, 82(1):175–224, 2022.

[LMY23] Qiuwei Li, Daniel McKenzie, and Wotao Yin. From the simplex to the sphere: faster constrained optimization using the hadamard parametrization. *Information and Inference: A Journal of the IMA*, 12(3):1898–1937, 2023.

[LNDT19] Dongchan Lee, Hung D Nguyen, Krishnamurthy Dvijotham, and Konstantin Turitsyn. Convex restriction of power flow feasibility sets. *IEEE Transactions on Control of Network Systems*, 6(3):1235–1245, 2019.

[Low14a] Steven H Low. Convex relaxation of optimal power flow—part i: Formulations and equivalence. *IEEE Transactions on Control of Network Systems*, 1(1):15–27, 2014.

[Low14b] Steven H Low. Convex relaxation of optimal power flow—part ii: Exactness. *IEEE Transactions on Control of Network Systems*, 1(2):177–189, 2014.

[LP66] Evgeny S Levitin and Boris T Polyak. Constrained minimization methods. *USSR Computational mathematics and mathematical physics*, 6(5):1–50, 1966.

[LT92] Zhi-Quan Luo and Paul Tseng. On the convergence of the coordinate descent method for convex differentiable minimization. *Journal of Optimization Theory and Applications*, 72(1):7–35, 1992.

[LT93a] Zhi-Quan Luo and Paul Tseng. Error bounds and convergence analysis of feasible descent methods: a general approach. *Annals of Operations Research*, 46(1):157–178, 1993.

[LT93b] Zhi-Quan Luo and Paul Tseng. On the convergence rate of dual ascent methods for linearly constrained convex minimization. *Mathematics of Operations Research*, 18(4):846–867, 1993.

[Mar85] D H. Martin. The essence of invexity. *Journal of optimization Theory and Applications*, 47:65–76, 1985.

[Meh92] Sanjay Mehrotra. On the implementation of a primal-dual interior point method. *SIAM Journal on optimization*, 2(4):575–601, 1992.

[Mha22] Zakaria Mhammedi. Efficient projection-free online convex optimization with membership oracle. In *Conference on Learning Theory*, pages 5314–5390. PMLR, 2022.

[MHK20] Aryan Mokhtari, Hamed Hassani, and Amin Karbasi. Stochastic conditional gradient methods: From convex minimization to submodular maximization. *Journal of machine learning research*, 21(105):1–49, 2020.

[MHSY25] Hoomaan Maskan, Yikun Hou, Suvrit Sra, and Alp Yurtsever. Revisiting frank-wolfe for structured nonconvex optimization. *arXiv preprint arXiv:2503.08921*, 2025.

[MMBS14] Bamdev Mishra, Gilles Meyer, Silvere Bonnabel, and Rodolphe Sepulchre. Fixed-rank matrix factorizations and riemannian low-rank optimization. *Computational Statistics*, 29:591–621, 2014.

[MZWG16] Cun Mu, Yuqian Zhang, John Wright, and Donald Goldfarb. Scalable robust matrix recovery: Frank–wolfe meets proximal methods. *SIAM Journal on Scientific Computing*, 38(5):A3291–A3317, 2016.

[N+18] Yurii Nesterov et al. *Lectures on convex optimization*, volume 137. Springer, 2018.

[Nes05] Yu Nesterov. Smooth minimization of non-smooth functions. *Mathematical programming*, 103(1):127–152, 2005.

[ÑFSA14] Ricardo Ñanculef, Emanuele Frandi, Claudio Sartori, and Héctor Allende. A novel frank–wolfe algorithm. analysis and applications to large-scale svm training. *Information Sciences*, 285:66–99, 2014.

[NW99] Jorge Nocedal and Stephen J Wright. *Numerical optimization*. Springer, 1999.

[Pan97] Jong-Shi Pang. Error bounds in mathematical programming. *Mathematical Programming*, 79(1):299–332, 1997.

[Pat98] Gábor Pataki. On the rank of extreme matrices in semidefinite programs and the multiplicity of optimal eigenvalues. *Mathematics of operations research*, 23(2):339–358, 1998.

[Pen23] Javier F Pena. Affine invariant convergence rates of the conditional gradient method. *SIAM Journal on Optimization*, 33(4):2654–2674, 2023.

[PI21] Andrei Patrascu and Paul Irofti. Computational complexity of inexact proximal point algorithm for convex optimization under holderian growth. *arXiv preprint arXiv:2108.04482*, 2021.

[PN18] Andrei Patrascu and Ion Necoara. On the convergence of inexact projection primal first-order methods for convex minimization. *IEEE Transactions on Automatic Control*, 63(10):3317–3329, 2018.

[PP23] Clarice Poon and Gabriel Peyré. Smooth over-parameterized solvers for non-smooth structured optimization. *Mathematical programming*, 201(1):897–952, 2023.

[PW00] Florian A Potra and Stephen J Wright. Interior-point methods. *Journal of computational and applied mathematics*, 124(1-2):281–302, 2000.

[RSW24] Akshay Ramachandran, Kevin Shu, and Alex L Wang. Hidden convexity, optimization, and algorithms on rotation matrices. *Mathematics of Operations Research*, 2024.

[RW09] R Tyrrell Rockafellar and Roger J-B Wets. *Variational analysis*, volume 317. Springer Science & Business Media, 2009.

[SCB+97] Alice E Smith, David W Coit, Thomas Baeck, David Fogel, and Zbigniew Michalewicz. Penalty functions. *Handbook of evolutionary computation*, 97(1):C5, 1997.

[SG24] Thabo Samakhoana and Benjamin Grimmer. Scalable projection-free optimization methods via multiradial duality theory. *arXiv preprint arXiv:2403.13688*, 2024.

[SRB11] Mark Schmidt, Nicolas Roux, and Francis Bach. Convergence rates of inexact proximal-gradient methods for convex optimization. *Advances in neural information processing systems*, 24, 2011.

[THH23] Jesus Tordesillas, Jonathan P How, and Marco Hutter. Rayen: Imposition of hard convex constraints on neural networks. *arXiv preprint arXiv:2307.08336*, 2023.

[THZK21] Kenya Tajima, Yoshihiro Hirohashi, Esmeraldo Ronnie Rey Zara, and Tsuyoshi Kato. Frank-wolfe algorithm for learning svm-type multi-category classifiers. In *Proceedings of the 2021 SIAM International Conference on Data Mining (SDM)*, pages 432–440. SIAM, 2021.

[Tob86] Roger L Tobin. Sensitivity analysis for variational inequalities. *Journal of optimization theory and applications*, 48(1):191–204, 1986.

[TT24] Tianyun Tang and Kim-Chuan Toh. Optimization over convex polyhedra via hadamard parametrizations. *Mathematical Programming*, pages 1–41, 2024.

[TZ22] Daniel Tabas and Baosen Zhang. Computationally efficient safe reinforcement learning for power systems. In *2022 American Control Conference (ACC)*, pages 3303–3310. IEEE, 2022.

[WKP23] Elias Wirth, Thomas Kerdreux, and Sebastian Pokutta. Acceleration of frank-wolfe algorithms with open-loop step-sizes. In *International Conference on Artificial Intelligence and Statistics*, pages 77–100. PMLR, 2023.

[WL06] Changyu Wang and Qian Liu. Convergence properties of inexact projected gradient methods. *Optimization*, 55(3):301–310, 2006.

[Wol70] Philip Wolfe. Convergence theory in nonlinear programming. *Integer and nonlinear programming*, pages 1–36, 1970.

[WPP24] Elias Wirth, Javier Pena, and Sebastian Pokutta. Fast convergence of frank-wolfe algorithms on polytopes. *arXiv preprint arXiv:2406.18789*, 2024.

[WPP25] Elias Wirth, Javier Pena, and Sebastian Pokutta. Accelerated affine-invariant convergence rates of the frank–wolfe algorithm with open-loop step-sizes. *Mathematical Programming*, pages 1–45, 2025.

[Wri97] Stephen J Wright. *Primal-dual interior-point methods*. SIAM, 1997.

[Ye94] Yinyu Ye. Combining binary search and newton s method to compute real roots for a class of real functions. *Journal of Complexity*, 10(3):271–280, 1994.

[Ye01] Yinyu Ye. Quadratic programming over an ellipsoid. In *Encyclopedia of Optimization*, pages 2112–2116. Springer, 2001.

[ZL20] Jiawei Zhang and Zhi-Quan Luo. A proximal alternating direction method of multiplier for linearly constrained nonconvex minimization. *SIAM Journal on Optimization*, 30(3):2272–2302, 2020.

[ZL22] Jiawei Zhang and Zhi-Quan Luo. A global dual error bound and its application to the analysis of linearly constrained nonconvex optimization. *SIAM Journal on Optimization*, 32(3):2319–2346, 2022.

[ZPL22] Jiawei Zhang, Wenqiang Pu, and Zhi-Quan Luo. On the iteration complexity of smoothed proximal alm for nonconvex optimization problem with convex constraints. *arXiv preprint arXiv:2207.06304*, 2022.

[ZWWY20] Fan Zhang, Hao Wang, Jiashan Wang, and Kai Yang. Inexact primal–dual gradient projection methods for nonlinear optimization on convex set. *Optimization*, 69(10):2339–2365, 2020.

Contents

A Related Work

We discuss related work on methods for reducing computational costs and achieving speedups in constrained optimization, organized into three parts: (i) classical projection and projection-free methods, (ii) recent advances, and (iii) work related to ball-constrained optimization.

A.1 Classical Methods

In convex optimization, three classical approaches have been widely studied: Frank-Wolfe methods [FW$^+$56], which avoids projection through linear minimization oracles over constrained sets; primal-dual methods [DFF56], which address primarily linear constraints through simultaneous updates to primal and dual variables; and penalty methods [Ber76], which incorporate constraints into the objective function using penalty functions [SCB$^+$97]. Each approach has its limitations.

Frank-Wolfe methods. Frank-Wolfe (FW) methods are first-order optimization algorithms that offer several attractive properties: they are easy to implement, projection-free, affine-invariant [Lan13, KLLJS21, Pen23],and their iterates naturally form sparse convex combinations of extreme points in the feasible region, making them particularly valuable for various machine learning applications [ÑFSA14, JTFF14, BZK18, MHK20, THZK21]. Convergence analysis of classic FW methods have been widely studied [LP66, DR70, Dun79, GM86]. However, FW methods face two key limitations: they require an efficient linear minimization oracle [CP21], and they often exhibit slower convergence rates [BRZ24, FG16]. Specifically, under the *Wolfe's lower bound* setting [Wol70], the FW algorithm cannot achieve convergence rates better than $\mathcal{O}(\epsilon^{-1+\delta})$ for any $\delta > 0$. Overcoming this fundamental barrier requires either algorithmic modifications or additional strong assumptions [GH15, GH16, BPTW19, CP20, WKP23, WPP25]. For advanced convergence analysis, one could refer to [Pen23, WPP24, MHSY25].

Penalty methods. Penalty methods struggle with ill-conditioning as penalty parameters increase and perform badly, especially for complex constraints. Primal-dual methods are primarily used to handle linear constraints [Koj89, Meh92, CP11], and their second-order variants face scalability issues due to high computational complexity.

First-order primal-dual methods. First-order primal-dual methods iteratively update primal and dual variables using inexact gradient steps (see, e.g., [HHZ17, CP11]). However, analyzing the convergence of such methods remains a challenging problem, particularly for nonconvex objectives [ZL22]. Most existing convergence results focus on problems with linear constraints [LT93b, CP11]. A widely used technique to address this challenge is error-bound analysis [LT93a, Pan97], which has been effective in establishing convergence rates for first-order methods in the convex setting [LT92, HL17]. However, these results typically provide only local convergence guarantees—ensuring convergence only when iterates are sufficiently close to the solution set—and depend on an error-bound constant that is often unknown or difficult to estimate. Recent work [ZL20, ZL22, ZPL22] introduces the smooth augmented Lagrangian primal-dual algorithm for constrained optimization. While this method achieves an optimal convergence rate matching the lower bound of optimization complexity for nonconvex objectives, it incurs additional *projection complexity* in each iteration.

A.2 Recent Advances

To reduce the cost and accelerate the convergence for solving (non-)convex optimization over convex sets, recent novel projection-free methods and other advanced techniques involve inexact projection, radial dual formulation, re-parameterizing optimization problems, and uncovering hidden convexity.

Inexact projection. In many cases, the projection operator lacks an analytic solution or is computationally expensive to compute exactly, motivating the analysis of inexact projected methods. For convex optimization, such methods achieve the same convergence rate as PGD if the cumulative projection error is bounded [SRB11, PN18], with new results derived under specific settings [PI21]. For nonconvex objectives with convex constraints, their convergence has been analyzed in [BMR03, WL06, ZWWY20]. Recent advances further generalize inexact projection operators to broader settings [FLP22, AFP23].

Radial duality. Beyond classical projection-free methods, recent advancements have introduced novel approaches based on gauge and radial duality theory. Radial duality theory for nonnegative

optimization problems [Gri24a, Gri24b] demonstrates that constrained optimization problems can be reformulated as unconstrained problems using the gauge of their constraints. This framework has led to the development of new families of projection-free methods with optimal convergence guarantees [LG23], as well as relaxed conditions [SG24] that enable more efficient line search operators for the reformulated unconstrained problems.

Reparameterization. Reparameterization optimization problems aim to mitigate challenging properties, such as non-smoothness or non-convexity, via invertible transformations while preserving equivalent optima. Parameterization is widely used in optimization and learning tasks, including semi-definite programming [Cif21], low-rank optimization [MMBS14, HLB20], and risk minimization [BRTW22]. Recent advancements include parameterizing simplex [LMY23] and polyhedron [TT24] optimization via Hadamard transformation to reduce projection complexity, smooth over-parameterization to accelerate non-smooth optimization algorithms [PP23], parameterizing discrete data as continuous for generative learning [DKP$^+$24], and analyzing the optimization landscape under parameterization transformations in non-convex settings [LKB24].

Hidden convexity. Hidden convexity refers to transformations that reveal the convex structure of non-convex sets or functions, which has been exploited in problems such as rotation matrix optimization [RSW24], non-linear least squares [DP19], revenue management and inventory control [CHHY22], and quadratically constrained quadratic programming (QCQP) with Toeplitz-Hermitian quadratics [KS15]. For non-convex stochastic optimization with hidden structure, projected gradient-based algorithms can achieve the same convergence rate as in convex optimization for both strongly convex [FHH23] and convex objectives [CHHY22] under certain assumptions. Furthermore, QCQP, which is generally NP-hard, can be solved in polynomial time when hidden convexity is present [KS15].

A.3 Ball-Constrained Optimization

To improve algorithmic performance and reduce the computational cost of constrained optimization, recent work has explored the use of ball-constrained optimization. The idea dates back to the ellipsoid method [GLS81], which iteratively encloses the feasible region in shrinking ellipsoids that contain the optimal solution. Despite its theoretical appeal and linear convergence, the ellipsoid method suffers from high computational complexity as $\mathcal{O}\left(n^4\right)$, making it impractical for large-scale problems. More recently, [CJJ$^+$20, GLRR25] studied acceleration techniques using ball-optimization oracles for specific problem settings. Inherently, ball-constrained optimization exhibits favorable properties; for example, solving quadratic problems over a ball using a combination of bisection and Newton's method can achieve a convergence rate of $\mathcal{O}(\log\log(1/\epsilon))$ [Ye94, Ye01], which matches the fast rate of Newton's method for unconstrained problems [BBV04]. These developments highlight the potential of ball-constrained techniques in designing efficient and scalable optimization algorithms.

B Homeomorphism of Convex set via Gauge Mapping

In this section, we provide the omitted details in Sec. 2 and Sec. 3.

B.1 Handling Constraint Set with Equality

We first explain how to apply our framework to handle constraints with linear equality constraints as mentioned in Sec. 2. Consider the constrained set $\mathcal{K}$ as follows

$$\mathcal{K} = \{\mathbf{x} \mid \mathbf{q}(\mathbf{x}) = \mathbf{0}, g_1(\mathbf{x}) \leq 0, \cdots, g_m(\mathbf{x}) \leq 0\}$$

where $\mathbf{q}(\cdot) = (q_1, q_2, \cdots, q_{m_{\text{eq}}})$ with $q_i : \mathbb{R}^n \to \mathbb{R}$ are linear/affine functions.

Note that the rank of $\mathrm{J}_{\mathbf{q}}$ is constant for all $\mathbf{x}$, i.e.,

$$\operatorname{rank}(\mathrm{J}_{\mathbf{q}}(\mathbf{x})) = r, \quad \forall \mathbf{x} \in \mathcal{K}.$$

Then $\{\mathbf{q}(\mathbf{x}) = \mathbf{0}\}$ is of dimension $n - r$ by the Constant-Rank Level Set Theorem [LL12]. In other words, we can use a subset of decision variables $\mathbf{x}_1 \in \mathbb{R}^{n-r}$ and reconstruct full decision variable $[\mathbf{x}_1, \mathbf{x}_2] \in \mathbb{R}^n$ via the equality constraint, where $\mathbf{x}_2 = \boldsymbol{\phi}(\mathbf{x}_1)$ and $\mathbf{q}([\mathbf{x}_1, \boldsymbol{\phi}(\mathbf{x}_1)]) = \mathbf{0}$. Such a reconstruction process ensures the feasibility of the equality constraint. Then the constraint $\mathcal{K}$ can be reformulated as

$$\mathcal{K}^s = \{\mathbf{x}_1 \in \mathbb{R}^{n-r} \mid g_1(\mathbf{x}_1, \boldsymbol{\phi}(\mathbf{x}_1)) \leq 0, \cdots, g_m(\mathbf{x}_1, \boldsymbol{\phi}(\mathbf{x}_1)) \leq 0\}.$$

It follows from the reconstruction that

$$(\mathbf{x}_1, \mathbf{x}_2 = \boldsymbol{\phi}(\mathbf{x}_2)) \in \mathcal{K} \Leftrightarrow \mathbf{x}_1 \in \mathcal{K}^s.$$

Thus, we can assume the constrained set has no inequalities without loss of generality.

B.2 Gauge Mapping for Convex Set

We first recall the definitions of gauge function/mapping.

Definition B.1 (Gauge/Minkowski Function [BM08])**.** Let $\mathcal{C} \subset \mathbb{R}^n$ be a compact convex set with a non-empty interior. The gauge/Minkowski function $\gamma_{\mathcal{C}} : \mathbb{R}^n \times \operatorname{int}(\mathcal{C}) \to \mathbb{R}_{\geq 0}$ is defined as

$$\gamma_{\mathcal{C}}(\mathbf{x}, \mathbf{x}^\circ) = \inf\{\lambda \geq 0 \mid \mathbf{x} \in \lambda(\mathcal{C} - \mathbf{x}^\circ)\},$$

where $\mathbf{x}^\circ \in \operatorname{int}(\mathcal{C})$.

Building upon this foundation, we define the gauge mapping between two compact convex sets:

Definition B.2 (Gauge Mapping [TZ22])**.** Let $\mathcal{Z}, \mathcal{C} \subset \mathbb{R}^n$ be compact convex sets with interior points $\mathbf{z}^\circ \in \operatorname{int}(\mathcal{Z})$ and $\mathbf{x}^\circ \in \operatorname{int}(\mathcal{C})$, respectively. Then

1) the gauge mapping $\boldsymbol{\psi} : \mathcal{Z} \to \mathcal{C}$ is defined as:

$$\boldsymbol{\psi}(\mathbf{z}) = \frac{\gamma_{\mathcal{Z}}(\mathbf{z} - \mathbf{z}^\circ, \mathbf{z}^\circ)}{\gamma_{\mathcal{C}}(\mathbf{z} - \mathbf{z}^\circ, \mathbf{x}^\circ)}(\mathbf{z} - \mathbf{z}^\circ) + \mathbf{x}^\circ, \ \mathbf{z} \in \mathcal{Z},$$

2) and the inverse gauge mapping $\boldsymbol{\psi}^{-1} : \mathcal{C} \to \mathcal{Z}$ is given by:

$$\boldsymbol{\psi}^{-1}(\mathbf{x}) = \frac{\gamma_{\mathcal{C}}(\mathbf{x} - \mathbf{x}^\circ, \mathbf{x}^\circ)}{\gamma_{\mathcal{Z}}(\mathbf{x} - \mathbf{x}^\circ, \mathbf{z}^\circ)}(\mathbf{x} - \mathbf{x}^\circ) + \mathbf{z}^\circ, \ \mathbf{x} \in \mathcal{C}.$$

We have the following remarks based on the definition.

- In essence, the gauge mapping scales the boundary of a convex set from an interior point to another convex set and with translation to its interior point.
- When $\mathcal{Z}$ is a unit ball, the gauge mapping in Def. B.2 is simplified as Def. 3.2:

$$\boldsymbol{\psi}(\mathbf{z}) = \frac{\|\mathbf{z}\|}{\gamma_{\mathcal{C}}(\mathbf{z}, \mathbf{x}^\circ)}\mathbf{z} + \mathbf{x}^\circ, \ \forall \mathbf{z} \in \mathcal{B}, \qquad \boldsymbol{\psi}^{-1}(\mathbf{x}) = \frac{\gamma_{\mathcal{C}}(\mathbf{x} - \mathbf{x}^\circ, \mathbf{x}^\circ)}{\|\mathbf{x} - \mathbf{x}^\circ\|}(\mathbf{x} - \mathbf{x}^\circ), \ \forall \mathbf{x} \in \mathcal{C}.$$

B.3 Properties of Gauge Mapping

The gauge function satisfies the following properties.

Proposition B.3 (Basic Properties of Gauge Function)**.** *Let $\mathcal{C}$ be a compact and convex set. For all $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$ and $\alpha \geq 0$, gauge function $\gamma_{\mathcal{C}}(\cdot, \mathbf{x}^\circ)$ given an interior point $\mathbf{x}^\circ \in \operatorname{int}(\mathcal{C})$ satisfies:*

- *Non-negativity:* $\gamma_{\mathcal{C}}(\mathbf{x}, \mathbf{x}^\circ) \geq 0$.
- *Positive homogeneity:* $\gamma_{\mathcal{C}}(\alpha\mathbf{x}, \mathbf{x}^\circ) = \alpha\gamma_{\mathcal{C}}(\mathbf{x}, \mathbf{x}^\circ)$.
- *Subadditivity:* $\gamma_{\mathcal{C}}(\mathbf{x} + \mathbf{y}, \mathbf{x}^\circ) \leq \gamma_{\mathcal{C}}(\mathbf{x}, \mathbf{x}^\circ) + \gamma_{\mathcal{C}}(\mathbf{y}, \mathbf{x}^\circ)$.
- *Convexity.*
- *Differentiability: Gauge function is twice differentiable almost everywhere.*
- *Upper/lower bounds:* $\gamma_{\mathcal{C}}(\mathbf{x}, \mathbf{x}^\circ) \in [\|\mathbf{x}\|/r_{\mathrm{o}}, \|\mathbf{x}\|/r_{\mathrm{i}}]$.
- *Lipschitz continuous:* $\|\gamma_{\mathcal{C}}(\mathbf{x}, \mathbf{x}^\circ) - \gamma_{\mathcal{C}}(\mathbf{y}, \mathbf{x}^\circ)\| \leq \frac{1}{r_{\mathrm{i}}}\|\mathbf{x} - \mathbf{y}\|$.

Proof. We show them one by one in the following.

1)2) The definition can directly derive non-negativity and positive homogeneity.

3)To show subadditivity, let

$$\lambda_x = \gamma_{\mathcal{C}}\left(\mathbf{x}, \mathbf{x}^\circ\right) \quad \text{and} \quad \lambda_y = \gamma_{\mathcal{C}}\left(\mathbf{y}, \mathbf{x}^\circ\right).$$

By the definition of the gauge function, there exist points $\mathbf{u}, \mathbf{v} \in \mathcal{C} - \mathbf{x}^\circ$ such that $\mathbf{x} = \lambda_x\mathbf{u}$ and $\mathbf{y} = \lambda_y\mathbf{v}$. Now, consider the sum:$\mathbf{x} + \mathbf{y} = \lambda_x\mathbf{u} + \lambda_y\mathbf{v}$. Write this sum as

$$\mathbf{x} + \mathbf{y} = (\lambda_x + \lambda_y)\left(\frac{\lambda_x}{\lambda_x + \lambda_y}\mathbf{u} + \frac{\lambda_y}{\lambda_x + \lambda_y}\mathbf{v}\right).$$

Since $\mathcal{C} - \mathbf{x}^\circ$ is convex (as a translation of the convex set $\mathcal{C}$), the convex combination

$$\frac{\lambda_x}{\lambda_x + \lambda_y}\mathbf{u} + \frac{\lambda_y}{\lambda_x + \lambda_y}\mathbf{v} \in \mathcal{C} - \mathbf{x}^\circ.$$

Thus, $\mathbf{x} + \mathbf{y} \in (\lambda_x + \lambda_y)(\mathcal{C} - \mathbf{x}^\circ)$ which implies by the definition of $\gamma_{\mathcal{C}}$ that $\gamma_{\mathcal{C}}\left(\mathbf{x} + \mathbf{y}, \mathbf{x}^\circ\right) \leq \lambda_x + \lambda_y$. Since λ_x and λ_y can be arbitrarily approximated by sequences converging to $\gamma_{\mathcal{C}}\left(\mathbf{x}, \mathbf{x}^\circ\right)$ and $\gamma_{\mathcal{C}}\left(\mathbf{y}, \mathbf{x}^\circ\right)$ (if the infimum is not attained exactly), we conclude $\gamma_{\mathcal{C}}\left(\mathbf{x} + \mathbf{y}, \mathbf{x}^\circ\right) \leq \gamma_{\mathcal{C}}\left(\mathbf{x}, \mathbf{x}^\circ\right) + \gamma_{\mathcal{C}}\left(\mathbf{y}, \mathbf{x}^\circ\right)$.

4) Convexity is induced by Positive homogeneity and Subadditivity.

5) Differentiability is from the fact that Convex functions over a compact set are twice differentiable almost everywhere [Eva18].

6) By the definition of gauge function (Def. B.5), we have:

$$\gamma_{\mathcal{C}}(\mathbf{x}, \mathbf{x}^\circ) = \frac{\|\mathbf{x}\|}{d_{\mathcal{C}}(\mathbf{x}^\circ, \mathbf{x}/\|\mathbf{x}\|)} \in [\|\mathbf{x}\|/r_{\mathrm{o}}, \|\mathbf{x}\|/r_{\mathrm{i}}]$$

7) By the subadditivity, we have:

$\gamma_{\mathcal{C}}(\mathbf{x}, \mathbf{x}^\circ) - \gamma_{\mathcal{C}}(\mathbf{y}, \mathbf{x}^\circ) \leq \gamma_{\mathcal{C}}(\mathbf{x} - \mathbf{y}, \mathbf{x}^\circ) + \gamma_{\mathcal{C}}(\mathbf{y}, \mathbf{x}^\circ) - \gamma_{\mathcal{C}}(\mathbf{y}, \mathbf{x}^\circ) = \gamma_{\mathcal{C}}(\mathbf{x} - \mathbf{y}, \mathbf{x}^\circ) \leq \|\mathbf{x} - \mathbf{y}\|/r_{\mathrm{i}}$

Similarly, we have:

$$\gamma_{\mathcal{C}}(\mathbf{y}, \mathbf{x}^\circ) - \gamma_{\mathcal{C}}(\mathbf{x}, \mathbf{x}^\circ) \leq \gamma_{\mathcal{C}}(\mathbf{y} - \mathbf{x}, \mathbf{x}^\circ) \leq \|\mathbf{x} - \mathbf{y}\|/r_{\mathrm{i}}$$

Thus, we have

$$\|\gamma_{\mathcal{C}}(\mathbf{x}, \mathbf{x}^\circ) - \gamma_{\mathcal{C}}(\mathbf{y}, \mathbf{x}^\circ)\| \leq \frac{1}{r_{\mathrm{i}}}\|\mathbf{x} - \mathbf{y}\|.$$

□

Based on the non-negativity, positive homogeneity, and subadditivity, the gauge function generalizes the concept of a norm. For a set $\mathcal{C}$ that is symmetric about the origin, the gauge function $\gamma_{\mathcal{C}}(\mathbf{x}, \mathbf{0})$ defines a norm. In particular, when $\mathcal{C} = \mathcal{B}_p = \{\mathbf{x} \in \mathbb{R}^n \mid \|\mathbf{x}\|_p \leq 1\}$ is the unit ball of the p-norm, we have $\gamma_{\mathcal{B}_p}(\mathbf{x}, \mathbf{0}) = \|\mathbf{x}\|_p$.

B.3.1 Proof of Proposition 3.3: Bi-Lipschitz Constants of Gauge Mapping

Next, we prove Proposition 3.3, stating that the gauge mapping is bi-Lipschitz continuous with constant depending on r_{i} and r_{o}.

Proof of Proposition 3.3. We begin with the forward gauge mapping from the 2-norm ball $\mathcal{B}$ to $\mathcal{C}$ as

$$\boldsymbol{\psi}(\mathbf{z}) = \frac{\|\mathbf{z}\|}{\gamma_{\mathcal{C}}(\mathbf{z}, \mathbf{x}^\circ)}\, \mathbf{z} + \mathbf{x}^\circ,\ \forall \mathbf{z} \in \mathcal{B}$$

Differentiating $\boldsymbol{\psi}(\mathbf{z})$ with respect to $\mathbf{z}$ (using the product and quotient rules) yields

$$\mathrm{J}_{\boldsymbol{\psi}}(\mathbf{z}) = \frac{\|\mathbf{z}\|}{\gamma_{\mathcal{C}}(\mathbf{z}, \mathbf{x}^\circ)}\mathbf{I} + \frac{\mathbf{z}\mathbf{z}^\top}{\gamma_{\mathcal{C}}(\mathbf{z}, \mathbf{x}^\circ)} - \frac{\|\mathbf{z}\|}{\gamma_{\mathcal{C}}(\mathbf{z}, \mathbf{x}^\circ)^2}\, \mathbf{z}\Big(\nabla_{\mathbf{z}} \gamma_{\mathcal{C}}(\mathbf{z}, \mathbf{x}^\circ)\Big)^\top.$$

Taking the operator norm and applying the triangle inequality gives

$$\begin{aligned}\Big\|\mathrm{J}_{\boldsymbol{\psi}}(\mathbf{z})\Big\| &\leq \frac{\|\mathbf{z}\|}{\gamma_{\mathcal{C}}(\mathbf{z}, \mathbf{x}^\circ)} + \frac{\|\mathbf{z}\|}{\gamma_{\mathcal{C}}(\mathbf{z}, \mathbf{x}^\circ)} + \frac{\|\mathbf{z}\|^2}{\gamma_{\mathcal{C}}(\mathbf{z}, \mathbf{x}^\circ)^2} \Big\|\nabla_{\mathbf{z}} \gamma_{\mathcal{C}}(\mathbf{z}, \mathbf{x}^\circ)\Big\| \\ &\leq r_{\mathrm{o}} + r_{\mathrm{o}} + \frac{r_{\mathrm{o}}^2}{r_{\mathrm{i}}},\end{aligned}$$

where in the last inequality we have used the facts that (i) for $\mathbf{z} \in \mathcal{B}$ one has $\|\mathbf{z}\| \leq 1$, (ii) the gauge function satisfies $\gamma_{\mathcal{C}}(\mathbf{z}, \mathbf{x}^\circ) \in [\|\mathbf{z}\|/r_{\mathrm{o}}, \|\mathbf{z}\|/r_{\mathrm{i}}]$, and (iii) $\|\nabla_{\mathbf{z}} \gamma_{\mathcal{C}}(\mathbf{z}, \mathbf{x}^\circ)\|$ is bounded by $1/r_{\mathrm{i}}$. In summary, we obtain

$$\Big\|\mathrm{J}_{\boldsymbol{\psi}}(\mathbf{z})\Big\| \leq 2r_{\mathrm{o}} + \frac{r_{\mathrm{o}}^2}{r_{\mathrm{i}}},$$

which proves that the forward Lipschitz constant of $\boldsymbol{\psi}$ satisfies

$$\mathrm{Lip}(\boldsymbol{\psi}) \leq 2r_{\mathrm{o}} + \frac{r_{\mathrm{o}}^2}{r_{\mathrm{i}}}.$$

Next, consider the inverse gauge mapping from $\mathcal{C}$ to the 2-norm ball as

$$\boldsymbol{\psi}^{-1}(\mathbf{x}) = \frac{\gamma_{\mathcal{C}}(\mathbf{x} - \mathbf{x}^\circ, \mathbf{x}^\circ)}{\|\mathbf{x} - \mathbf{x}^\circ\|}\, (\mathbf{x} - \mathbf{x}^\circ),\ \forall \mathbf{x} \in \mathcal{C}.$$

Differentiating with respect to $\mathbf{x}$ gives

$$\mathrm{J}_{\boldsymbol{\psi}^{-1}}(\mathbf{x}) = \nabla \gamma_{\mathcal{C}}(\mathbf{x} - \mathbf{x}^\circ, \mathbf{x}^\circ) \left(\frac{\mathbf{x} - \mathbf{x}^\circ}{\|\mathbf{x} - \mathbf{x}^\circ\|}\right)^\top + \gamma_{\mathcal{C}}(\mathbf{x} - \mathbf{x}^\circ, \mathbf{x}^\circ) \cdot \frac{I - \frac{(\mathbf{x}-\mathbf{x}^\circ)(\mathbf{x}-\mathbf{x}^\circ)^\top}{\|\mathbf{x}-\mathbf{x}^\circ\|^2}}{\|\mathbf{x} - \mathbf{x}^\circ\|}.$$

Taking norms and again using the triangle inequality leads to

$$\Big\|\mathrm{J}_{\boldsymbol{\psi}^{-1}}(\mathbf{x})\Big\| \leq \left\|\nabla \gamma_{\mathcal{C}}(\mathbf{x} - \mathbf{x}^\circ, \mathbf{x}^\circ) \left(\frac{\mathbf{x} - \mathbf{x}^\circ}{\|\mathbf{x} - \mathbf{x}^\circ\|_2}\right)^\top\right\| + \gamma_{\mathcal{C}}(\mathbf{x} - \mathbf{x}^\circ, \mathbf{x}^\circ) \left\|\frac{I - \frac{(\mathbf{x}-\mathbf{x}^\circ)(\mathbf{x}-\mathbf{x}^\circ)^\top}{\|\mathbf{x}-\mathbf{x}^\circ\|^2}}{\|\mathbf{x} - \mathbf{x}^\circ\|}\right\|.$$

Using the bound $\gamma_{\mathcal{C}}(\mathbf{x} - \mathbf{x}^\circ, \mathbf{x}^\circ) \in [\|\mathbf{x} - \mathbf{x}^\circ\|/r_{\mathrm{o}}, \|\mathbf{x} - \mathbf{x}^\circ\|/r_{\mathrm{i}}]$ and the projection matrix related term has norm at most $1/\|\mathbf{x} - \mathbf{x}^\circ\|$), we have

$$\Big\|\mathrm{J}_{\boldsymbol{\psi}^{-1}}(\mathbf{x})\Big\| \leq \frac{1}{r_{\mathrm{i}}} + \frac{1}{r_{\mathrm{i}}} = \frac{2}{r_{\mathrm{i}}}.$$

Thus, the inverse Lipschitz constant is bounded by

$$\mathrm{Lip}(\boldsymbol{\psi}^{-1}) \leq \frac{2}{r_{\mathrm{i}}}.$$

This completes the proof. □

B.3.2 Smoothing Technique for Gauge Mapping

Moreover, we have the following property for gauge mapping as a corollary from Proposition B.3.

Corollary B.4. *The gauge mapping $\boldsymbol{\psi}$ defined in Def. B.2 is twice differentiable almost everywhere.*

This corollary follows directly from Proposition B.3 and aligns with our general assumptions (see Appendix C.2). *However, the existence of a zero-measure set of non-differentiable points creates a gap between our theoretical requirements and the current formulation*. For instance, in max-cut SDP problems, low-rank optimal solutions lead to non-differentiable points on the PSD cone constraint boundary due to eigenvalue multiplicity at the maximum eigenvalue, causing convergence behavior to oscillate around the optimum. While these points may have no practical impact on most engineering problems, mathematical rigor demands that we address this discrepancy.

To bridge this gap, we introduce a smoothing technique. Recall that the gauge mapping $\boldsymbol{\psi}$ from the unit ball $\mathcal{B}$ to a convex set $\mathcal{C}$ is computed (see Appendix B.4) as:

$$\boldsymbol{\psi}(\mathbf{z}) = \frac{1}{\kappa_{\mathcal{C}}\left(\mathbf{x}^\circ, \mathbf{z}/\|\mathbf{z}\|\right)} \cdot \mathbf{z} + \mathbf{x}^\circ$$

where $\kappa_{\mathcal{C}}$ is the inverse distance function (Def. B.5).

In practice, most constraints we encounter are listed in Table 2. The non-smoothness of $\kappa_{\mathcal{C}}$ (and thus $\boldsymbol{\psi}$) arises from the use of the max operator (including the ReLU operator $[\cdot]^+$) when handling multiple constraints. We define $\kappa_i(\mathbf{x}^\circ, \mathbf{v})$ such that $\kappa_{\mathcal{C}_i}(\mathbf{x}^\circ, \mathbf{v}) = [\kappa_i(\mathbf{x}^\circ, \mathbf{v})]^+$ in Table 2. Hence, each κ_i is smooth. We then express $\kappa_{\mathcal{C}}(\mathbf{x}^\circ, \mathbf{v}) = \max\limits_{1\le i\le m} \{\kappa_i(\mathbf{x}^\circ, \mathbf{v}), 0\}$ and, letting $\kappa_{m+1} = 0$, we simplify this as

$$\kappa_{\mathcal{C}}(\mathbf{x}^\circ, \mathbf{v}) = \max_{1\le i\le m} \{\kappa_i(\mathbf{x}^\circ, \mathbf{v})\},$$

with loss of generality. In the following, for convenience, we fix $\mathbf{x}^\circ$ and write $\kappa_i(\mathbf{v})$ instead of $\kappa_i(\mathbf{x}^\circ, \mathbf{v})$.

We smooth $\kappa_{\mathcal{C}}$ using the log-sum-exp approximation (also called Nesterov smoothing) with parameter $\eta > 0$ following [Gri24b, Nes05]:

$$\kappa_\eta(\mathbf{v}) = \eta \log\left(\sum_{i=0}^{n} \exp\left(\frac{\kappa_i(\mathbf{v})}{\eta}\right)\right),$$

with the corresponding gradient

$$\nabla\kappa_\eta(\mathbf{v}) = \sum_{i=1}^{m} \lambda_i \nabla\kappa_i(\mathbf{v}) \quad \text{where} \quad \lambda_i = \frac{\exp\left(\kappa_i(\mathbf{v})/\eta\right)}{\sum_{j=1}^{m} \exp\left(\kappa_j(\mathbf{v})/\eta\right)}.$$

By replacing the objective $h = f \circ \boldsymbol{\psi}$ with $h_\eta = f \circ \boldsymbol{\psi}_\eta$ where $\boldsymbol{\psi}_\eta(\mathbf{z}) = \frac{1}{\kappa_\eta(\mathbf{x}^\circ, \mathbf{z}/\|\mathbf{z}\|)} \cdot \mathbf{z} + \mathbf{x}^\circ$, we obtain a twice-differentiable homeomorphism. The smoothing introduces an optimality gap of $\mathcal{O}(\eta)$ [Nes05, Gri24b] (here the hidden constant is dependent on the Lipschitz constant of f), which can be made arbitrarily small by choosing η sufficiently small.

B.4 Computation of Gauge Mapping

For computational purposes, we introduce a point-to-boundary distance function and its inverse.

Definition B.5 (Point-to-Boundary Distance [THH23])**.** Let $\mathcal{C} \subset \mathbb{R}^n$ be a compact convex set and $\mathbf{x}^\circ \in \operatorname{int}(\mathcal{C})$. For any unit vector $\mathbf{v} \in \mathbb{S}^{n-1} = \{\mathbf{u} \in \mathbb{R}^n \mid \|\mathbf{u}\| = 1\}$, we define the interior-point-to-boundary distance function $d_{\mathcal{C}} : \operatorname{int}(\mathcal{C}) \times \mathbb{S}^{n-1} \to \mathbb{R}_{\ge 0}$ along direction $\mathbf{v}$ as

$$d_{\mathcal{C}}(\mathbf{x}^\circ, \mathbf{v}) = \sup\{\lambda \ge 0 \mid \mathbf{x}^\circ + \lambda\mathbf{v} \in \mathcal{C}\}.$$

The inverse distance function $\kappa_{\mathcal{C}} : \operatorname{int}(\mathcal{C}) \times \mathbb{S}^{n-1} \to \mathbb{R}_{\ge 0}$ is defined as $\kappa_{\mathcal{C}}(\mathbf{x}^\circ, \mathbf{v}) := 1/d_{\mathcal{C}}(\mathbf{x}^\circ, \mathbf{v})$.

We then have the following rules for computing the gauge mapping with the help of the point-to-boundary distance function.

- This distance function relates to the gauge function as:

$$\gamma_{\mathcal{C}}(\mathbf{x}, \mathbf{x}^\circ) = \kappa_{\mathcal{C}}(\mathbf{x}^\circ, \mathbf{x}/\|\mathbf{x}\|) \cdot \|\mathbf{x}\| = \frac{\|\mathbf{x}\|}{d_{\mathcal{C}}(\mathbf{x}^\circ, \mathbf{x}/\|\mathbf{x}\|)}.$$

- Further, considering the positive homogeneity, we have:

$$\gamma_{\mathcal{C}}(\mathbf{x}/\|\mathbf{x}\|, \mathbf{x}^\circ) = \kappa_{\mathcal{C}}(\mathbf{x}^\circ, \mathbf{x}/\|\mathbf{x}\|) = \frac{1}{d_{\mathcal{C}}(\mathbf{x}^\circ, \mathbf{x}/\|\mathbf{x}\|)}.$$

- The gauge mapping between $\mathcal{B}$ and $\mathcal{C}$ can be simplified as:

$$\boldsymbol{\psi}(\mathbf{z}) = d_{\mathcal{C}}(\mathbf{x}^\circ, \mathbf{z}/\|\mathbf{z}\|) \cdot \mathbf{z} + \mathbf{x}^\circ, \ \forall \mathbf{z} \in \mathcal{B}$$
$$\boldsymbol{\psi}^{-1}(\mathbf{x}) = \frac{\mathbf{x} - \mathbf{x}^\circ}{d_{\mathcal{C}}(\mathbf{x}^\circ, (\mathbf{x} - \mathbf{x}^\circ)/\|\mathbf{x} - \mathbf{x}^\circ\|)}, \ \forall \mathbf{x} \in \mathcal{C}.$$

Therefore, we can compute the inverse distance function to obtain the gauge mapping. In practice, we can compute the inverse distance function $\kappa_{\mathcal{C}}(\mathbf{x}^\circ, \cdot)$ as follows.

- *Closed-form for common constrained set.* For various common constraint types, Table 2 provides closed-form expressions of the inverse distance function and Fig. 6 show the actual computational cost for it. Most matrix calculations can be computed and stored offline before being applied online given $\mathbf{v}$.
- *Bisection-based algorithm for general constrained set* . When the inverse distance function lacks a closed-form expression, we employ an efficient bisection algorithm detailed in Alg. 2. This algorithm supports batch processing, enabling efficient parallel computation for multiple inputs simultaneously.

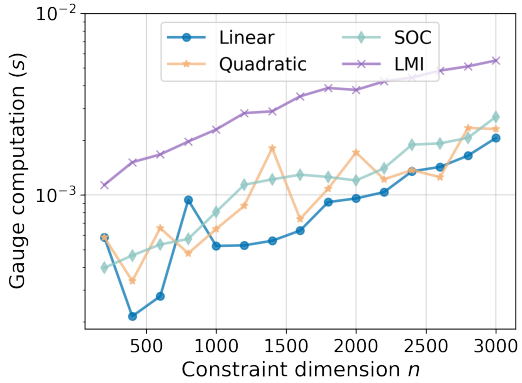


Figure 6: Illustration of gauge function calculation time as constraint dimension varies.

B.5 Gauge Mapping in Hom-PGD

Optimizing the interior point in Hom-PGD. As shown in Sec. 4, one can select a central interior point with large inner radius r_{i} or smaller outer radius r_{o} to reduce the Lipschitz constant of the gauge mapping $\boldsymbol{\psi}$ thereby boosting the convergence rate of Hom-PGD. Here, we introduce two types of interior points which maximize r_{i} and minimize r_{o}, respectively, below.

- Chebyshev center: maximizes the minimum distance from the point to the boundary of the set for a large inner radius r_{i}:

$$\mathbf{x}^\circ = \arg\max_{\mathbf{x} \in \mathcal{K}} \min_{\mathbf{y} \in \partial\mathcal{K}} \|\mathbf{x} - \mathbf{y}\|.$$

- Min-max center: minimizes the maximum distance from the point to the boundary of the set for a smaller outer radius r_{o}:

$$\mathbf{x}^\circ = \arg\min_{\mathbf{x} \in \mathcal{K}} \max_{\mathbf{y} \in \partial\mathcal{K}} \|\mathbf{x} - \mathbf{y}\|.$$

Table 2: Closed-form Expressions for Inverse Distance Functions [THH23]

Constrained Set $\mathcal{C}$	Formulation	Inverse Distance Function $\kappa_{\mathcal{C}}(\mathbf{x}^\circ, \mathbf{v})$
Linear	$\{\mathbf{x} : \mathbf{a}^\top \mathbf{x} \leq b\}$	$(\frac{\mathbf{a}^\top \mathbf{v}}{b - \mathbf{a}^\top \mathbf{x}^\circ})^+$
Quadratic	$\{\mathbf{x} : \mathbf{x}^\top \mathbf{Q}\mathbf{x} + \mathbf{a}^\top \mathbf{x} \leq b\}$	$(1/\mathrm{root}\,(A_{\mathbf{Q}}, B_{\mathbf{Q}}, C_{\mathbf{Q}}))^+$
Second Order Cone	$\{\mathbf{x} : \|\mathbf{A}^\top \mathbf{x} + \mathbf{p}\|_2 \leq \mathbf{a}^\top \mathbf{x} + b\}$	$(1/\mathrm{root}(A_{\mathrm{S}}, B_{\mathrm{S}}, C_{\mathrm{S}}))^+$
Linear Matrix Inequality	$\{\mathbf{x} : \sum_{i=1}^n x_i \cdot \mathbf{F}_i + \mathbf{F}_0 \succeq 0\}$	$\left(\mathrm{eig}\left(\mathbf{L}^\top (-\mathbf{S})\, \mathbf{L}\right)\right)^+$
Intersections	$\mathcal{C} = \bigcap_{i=1}^m \mathcal{C}_i$	$\max_{1 \leq i \leq m} \{\kappa_{\mathcal{C}_i}(\mathbf{x}^\circ, \mathbf{v})\}$

[1] Notation: $\mathbf{x}, \mathbf{a} \in \mathbb{R}^n$, $b \in \mathbb{R}$, $\mathbf{Q} \in \mathbb{S}_+^n$, $\mathbf{A} \in \mathbb{R}^{n \times m}$, $\mathbf{p} \in \mathbb{R}^m$, $\mathbf{F}_0, \cdots, \mathbf{F}_n \in \mathbb{R}^{m \times m}$, $\mathbf{X} \in \mathbb{R}^{n \times n}$, $(\cdot)^+ = \max(\cdot, 0)$.

[2] $\mathrm{root}(x_1, x_2, x_3) = \frac{-x_2 \pm \sqrt{x_2^2 - 4x_1x_3}}{2x_1}$ denotes the roots for quadratic equation.

[3] $A_{\mathbf{Q}} = \mathbf{v}^\top \mathbf{Q}\mathbf{v}$, $B_{\mathbf{Q}} = 2\mathbf{x}^{\circ\top}\mathbf{Q}\mathbf{v} + \mathbf{a}^\top \mathbf{v}$, $C_{\mathbf{Q}} = \mathbf{x}^{\circ\top}\mathbf{Q}\mathbf{x}^\circ + \mathbf{a}^\top \mathbf{x}^\circ - b$.

[4] $A_{\mathrm{S}} = (\mathbf{A}^\top \mathbf{v})^\top (\mathbf{A}^\top \mathbf{v}) - (\mathbf{a}^\top \mathbf{v})^2$, $B_{\mathrm{S}} = 2(\mathbf{A}^\top \mathbf{x}^\circ + \mathbf{p})^\top (\mathbf{A}^\top \mathbf{v}) - 2(\mathbf{a}^\top \mathbf{x}^\circ + b)(\mathbf{a}^\top \mathbf{v})$, $C_{\mathrm{S}} = (\mathbf{A}^\top \mathbf{x}^\circ + \mathbf{p})^\top (\mathbf{A}^\top \mathbf{x}^\circ + \mathbf{p}) - (\mathbf{a}^\top \mathbf{x}^\circ + b)^2$.

[5] $\mathbf{H} = \mathbf{F}_0 + \sum_{i=1}^n \mathbf{x}_i^\circ \mathbf{F}_i$, $\mathbf{H}^{-1} = \mathbf{L}^\top \mathbf{L}$, $\mathbf{S} = \sum_{i=1}^n v_i \mathbf{F}_i$. $\mathrm{eig}(\mathbf{X}) = \lambda_1, \cdots, \lambda_n$ denotes the eigenvalues satisfying $\det(\mathbf{X} - \lambda \mathbf{I}) = 0$. Note that only the maximum eigenvalue is needed. Thus, power iteration methods can be applied to compute it efficiently.

[6] Note that all $\mathbf{v}$-independent terms can be computed only once and stored for use.

Algorithm 2 Bisection Algorithm for Inverse Point-to-Boundary Distance

Input: A compact convex set $\mathcal{C}$, an interior point $\mathbf{x}^\circ \in \mathcal{C}$, and a unit vector $\mathbf{v}$.

$\alpha_l = 0$, and $\alpha_u = 1$
while $|\alpha_l - \alpha_u| \geq 10^{-3}$ **do**
  if $\mathbf{x}^\circ + \alpha_u \cdot \mathbf{v} \in \mathcal{C}$ **then**
    increase lower bound: $\alpha_l \leftarrow \alpha_u$
    double upper bound: $\alpha_u \leftarrow 2 \cdot \alpha_m$
  else
    bisection: $\alpha_m = (\alpha_l + \alpha_u)/2$
    if $\mathbf{x}^\circ + \alpha_m \cdot \mathbf{v} \in \mathcal{C}$ **then**
      increase lower bound: $\alpha_l \leftarrow \alpha_m$
    else
      decrease upper bound: $\alpha_u \leftarrow \alpha_m$
    end if
  end if
end while

Output: inverse distance: $\kappa_{\mathcal{C}}(\mathbf{x}^\circ, \mathbf{v}) \approx 1/\alpha_m$

According to the definitions, the Chebyshev center maximizes r_{i} and the min-max center minimizes r_{o}. For certain convex sets, one can efficiently compute these centers. For instance, for a polyhedron, one can find the Chebyshev center by solving an LP (see e.g., [BBV04]). However, for general convex sets, computing the Chebyshev center entails solving a convex optimization problem that may be very hard due to the infinitely many constraints where each corresponding to a boundary point of the set.

In practice, we may minimize the constraint residual to find an approximate central interior point following [THH23]. Concretely, we solve the following convex program to obtain an interior point $\mathbf{x}^\circ \in \mathcal{K} := \{\mathbf{g}(\mathbf{x}) \leq \mathbf{0}\}$.

$$\begin{aligned} (\mathbf{x}^\circ, \epsilon) = \arg\max_{\mathbf{x}, \epsilon} \quad & \epsilon \\ \text{s.t.} \quad & \mathbf{g}(\mathbf{x}) \leq -\epsilon \mathbf{1}, \\ & \epsilon > 0. \end{aligned}$$

Note that in the main experiments, we set $\epsilon = 10^{-3}$ as a constant to solve this convex feasibility problem to obtain an interior point (which may be close to the boundary). In the ablation study, we solve this residual maximization problem to obtain a "central" interior point for comparison.

Computation of basic operations in Hom-PGD. We now summarize the practical methods used to compute the key operations in Hom-PGD, including the gauge function/mapping and the gradient of $h = f \circ \boldsymbol{\psi}$:

- *Computing the gauge function* $\gamma_{\mathcal{K}}(\cdot, \mathbf{x}^\circ)$: Compute the inverse distance function $\kappa_{\mathcal{K}}(\mathbf{x}^\circ, \cdot)$ via Algorithm 2 or using the closed form when available.
- *Computing the gauge mapping* $\boldsymbol{\psi}$: Derived from Definition B.2, given $\gamma_{\mathcal{K}}(\cdot, \mathbf{x}^\circ)$.
- *Computing the gradient of* $h = f \circ \boldsymbol{\psi}$: Approximated using numerical differentiation methods. If h has a closed-form, the gradient can be computed using automatic differentiation methods (see e.g., [BPRS18]). For general cases, we adopt the finite difference method for computing the gradient of the gauge function in [LBGH23], i.e., for $i = 1, 2, \ldots, n$:

$$\nabla_i h(\mathbf{x}) = \frac{h(\mathbf{x} + \lambda \mathbf{e}_i) - h(\mathbf{x})}{\lambda}$$

given a proper small number $\lambda > 0$ where $\mathbf{e}_i$ denotes the standard basis over $\mathbb{R}^n$.

Next, we summarize the complexity for computing the above basic operations.

Proposition B.6. *With the zeroth-order oracle of f and the membership oracle, the computational complexity of the basic operations in Hom-PGD is as follows.*

(i) Computing gauge function $\gamma_{\mathcal{K}}(\cdot, \mathbf{x}^\circ)$ to an error ϵ costs $\mathcal{O}(\log 1/\epsilon)$.

(ii) Computing gauge mapping $\boldsymbol{\psi}(\mathbf{x})$ costs $\tilde{\mathcal{O}}(n)$.

(iii) Computing the gradient of $h = f \circ \boldsymbol{\psi}$ costs $\tilde{\mathcal{O}}(n^2)$.

Proof. We show them one by one in the following.

(i) This is a classical result of the bisection-based algorithm. Using $\mathcal{O}(\log 1/\epsilon)$ number of calls to the membership oracle, we can get an ϵ_{bis}-approximate solution to the gauge function $\gamma_{\mathcal{K}}(\cdot, \mathbf{x}^\circ)$. One could refer to, e.g., [LCL23, LBGH23, Mha22] for a detailed proof.

(ii) From Def. B.2, computing $\boldsymbol{\psi}$ requires $\tilde{\mathcal{O}}(1)$ complexity to evaluate $\gamma_{\mathcal{C}}(\cdot, \mathbf{x}^\circ)$ and $\mathcal{O}(n)$ complexity to compute the scalar-vector product $d_{\mathcal{C}}(\mathbf{x}^\circ, \cdot) \cdot \mathbf{z}$.

(iii) From the finite difference method described above, computing each partial derivative $\nabla_i h$ requires evaluating h, which involves a zeroth-order oracle call to f and a computation of the gauge mapping $\boldsymbol{\psi}$. Therefore, computing each $\nabla_i h(\mathbf{x})$ $(i = 1, 2, \ldots, n)$ costs $\tilde{\mathcal{O}}(n)$, leading to a total of $\tilde{\mathcal{O}}(n^2)$ to compute the gradient mapping ∇h. □

C Preliminaries for Technical Proof

In this section, we summarize the related basic concepts, notations, assumptions, and fundamental propositions and lemmas.

C.1 Basic Concepts

We list the basic concepts used in this paper below.

- Indicator function. For a set $\mathcal{X} \subset \mathbb{R}^n$, the indicator function $\delta_{\mathcal{X}} : \mathbb{R}^n \to \mathbb{R}$ is defined as

$$\delta_{\mathcal{X}}(\mathbf{x}) := \begin{cases} 1 & \text{if } \mathbf{x} \in \mathcal{X}, \\ \infty & \text{if } \mathbf{x} \notin \mathcal{X}. \end{cases}$$

- Distance between a point and a set. For a closed set $\mathcal{X} \in \mathbb{R}^n$ and any $\mathbf{x} \in \mathbb{R}^n$, the distance between $\mathbf{x}$ and $\mathcal{X}$ is defined as $\operatorname{dist}(\mathbf{x}, \mathcal{X}) = \inf_{\mathbf{y} \in \mathcal{X}} \|\mathbf{x} - \mathbf{y}\|$.
- Orthogonal projection. For a closed set $\mathcal{X}$, the orthogonal projection of a point $\mathbf{x} \in \mathbb{R}^n$ onto $\mathcal{X}$ is defined as $\Pi_{\mathcal{X}}(\mathbf{x}) = \arg\min_{\mathbf{y} \in \mathcal{X}} \|\mathbf{x} - \mathbf{y}\|$.
- Function convexity. For a differentiable function $f : \mathcal{X} \subseteq \mathbb{R}^n \to \mathbb{R}$, it is said to be convex if one of the following holds:

1) Jensen's inequality. For θ with $0 \leq \theta \leq 1$, we have $f(\theta \mathbf{x}+(1-\theta)\mathbf{y}) \leq \theta f(\mathbf{x})+(1-\theta) f(\mathbf{y})$ for all $\mathbf{x}, \mathbf{y} \in \mathcal{X}$.
2) first-order condition. $f(\mathbf{y}) \geq f(\mathbf{x})+\langle \nabla f(\mathbf{x}), \mathbf{y}-\mathbf{x}\rangle, \forall \mathbf{x}, \mathbf{y} \in \mathcal{X}$.
3) monotone gradient. $(\nabla f(\mathbf{x})-\nabla f(\mathbf{y}))^T(\mathbf{x}-\mathbf{y}) \geq 0$ for all $\mathbf{x}, \mathbf{y} \in \mathcal{X}$.

- L-Smoothness. A differentiable function $f: \mathcal{X} \subseteq \mathbb{R}^n \to \mathbb{R}$ is said be L-smooth if one of the following holds:
  1) zeroth-order condition. $f(\lambda \mathbf{x}+(1-\lambda)\mathbf{y}) \geq \lambda f(\mathbf{x})+(1-\lambda) f(\mathbf{y})-\frac{L}{2}\lambda(1-\lambda)\|\mathbf{y}-\mathbf{x}\|^2$, for all $\mathbf{x}, \mathbf{y} \in \mathcal{X}, \lambda \in [0,1]$.
  2) first-order condition. $f(\mathbf{y}) \leq f(\mathbf{x})+\langle \nabla f(\mathbf{x}), \mathbf{y}-\mathbf{x}\rangle+\frac{L}{2}\|\mathbf{y}-\mathbf{x}\|^2$, for all $\mathbf{x}, \mathbf{y} \in \mathcal{X}$.
  3) Lipschitz gradient. $\|\nabla f(\mathbf{y})-\nabla f(\mathbf{x})\| \leq L\|\mathbf{y}-\mathbf{x}\|$, for all $\mathbf{x}, \mathbf{y}$.
- μ-Strong convexity. A differentiable function $f: \mathcal{X} \subseteq \mathbb{R}^n \to \mathbb{R}$ is said be μ-strongly convex holds if one of the following holds:
  1) zeroth-order condition. $f(\lambda \mathbf{x}+(1-\lambda)\mathbf{y}) \leq \lambda f(\mathbf{x})+(1-\lambda) f(\mathbf{y})-\frac{\mu}{2}\lambda(1-\lambda)\|\mathbf{y}-\mathbf{x}\|^2$, for all $\mathbf{x}, \mathbf{y} \in \mathcal{X}, \lambda \in [0,1]$.
  2) first-order condition. $f(\mathbf{y}) \geq f(\mathbf{x})+\langle \nabla f(\mathbf{x}), \mathbf{y}-\mathbf{x}\rangle+\frac{\mu}{2}\|\mathbf{y}-\mathbf{x}\|^2$, for all $\mathbf{x}, \mathbf{y} \in \mathcal{X}$.
  3) strictly monotone gradient. $\langle \nabla f(\mathbf{y})-\nabla f(\mathbf{x}), \mathbf{y}-\mathbf{x}\rangle \geq \mu\|\mathbf{y}-\mathbf{x}\|^2$, for all $\mathbf{x}, \mathbf{y} \in \mathcal{X}$.
- Stationary point. Consider a program $\{\min_{\mathbf{x}} f(\mathbf{x}),\ \text{s.t.},\ \mathbf{x} \in \mathcal{X}\}$ where f is differentiable and $\mathcal{X}$ is a convex set. Let $\alpha > 0$. Then a point $\mathbf{x}^*$ is called a stationary point for the program if
$$\Pi_{\mathcal{X}}(\mathbf{x}-\alpha \nabla f(\mathbf{x}))=\mathbf{x},$$
or equivalently (see e.g. [Bec14])
$$\langle \nabla f(\mathbf{x}^*), \mathbf{x}-\mathbf{x}^*\rangle \geq 0,\ \forall \mathbf{x} \in \mathcal{X}.$$
- Jacobian matrix. Suppose $\mathbf{f}: \mathbb{R}^n \to \mathbb{R}^m$ is a function such that each of its first-order partial derivatives exists on $\mathbb{R}^n$. Then the Jacobian matrix of $\mathbf{f}$, denoted $\mathrm{J}_{\mathbf{f}} \in \mathbb{R}^{m \times n}$, is defined as $\mathrm{J}_{\mathbf{f}}=(\frac{\partial f_i}{\partial x_j})_{ij}$.
- A Hessian of a function $f: \mathbb{R}^n \to \mathbb{R}$ is defined as $\nabla^2 f=(\frac{\partial^2 f}{\partial x_i \partial x_j})_{ij} \in \mathbb{R}^{n \times n}$, if its second-order partial derivatives exist. Moreover, for a mapping $\mathbf{f}: \mathbb{R}^n \to \mathbb{R}^m$ with existed second-order partial derivatives of each component f_i $(i=1,2,\cdots,m)$. The Hessian of $\mathbf{f}$ is defined as
$$\mathrm{H}(\mathbf{f})=(\nabla^2 f_1, \cdots, \nabla^2 f_m).$$

C.2 Basic Assumptions and Notations

In the following, we make assumptions throughout the paper.

- Assumptions on f and constraints g_i $(i=1,2,\cdots,m)$ in problem $\mathbf{P}$:
  1) f is $L_{f,0}$ Lipschitz continuous, i.e., $\|f(\mathbf{x})-f(\mathbf{y})\| \leq L_{f,0}\|\mathbf{x}-\mathbf{y}\|$ for any $\mathbf{x}, \mathbf{y}$.
  2) f in problem $\mathbf{P}$ is differentiable and L_f smooth.
  3) $f^* > -\infty$ where $f^* := \min_{\mathbf{x} \in \mathcal{K}} f(\mathbf{x})$.
  4) Each g_i is $L_{g_i,0}$-Lipschitz continuous, differentiable, and L_{g_i}-smooth.
- Assumptions on the homeomorphic mapping $\boldsymbol{\psi}: \mathbb{R}^n \to \mathbb{R}^n$:
  1) $\boldsymbol{\psi}$ is differentiable with non-singular Jacobian $\mathrm{J}_{\boldsymbol{\psi}}(\cdot)$,
  2) $\boldsymbol{\psi}$ is (κ_1, κ_2)-bi-Lipschitz continuous for $\kappa_2 \geq \kappa_1 > 0$, i.e.,
  $$\kappa_1\|\mathbf{u}-\mathbf{v}\| \leq \|\boldsymbol{\psi}(\mathbf{u})-\boldsymbol{\psi}(\mathbf{v})\| \leq \kappa_2\|\mathbf{u}-\mathbf{v}\|.$$
  Then the Jacobian matrix, $\mathrm{J}_{\boldsymbol{\psi}}(\cdot)$ and $\mathrm{J}_{\boldsymbol{\psi}^{-1}}(\cdot)$ will satisfy
  $$\|\mathrm{J}_{\boldsymbol{\psi}}(\mathbf{z})\| \leq \kappa_2,\ \forall \mathbf{z}, \quad \|\mathrm{J}_{\boldsymbol{\psi}^{-1}}(\mathbf{x})\| \leq \frac{1}{\kappa_1},\ \forall \mathbf{x}.$$
  3) $\boldsymbol{\psi}$ has $L_{\boldsymbol{\psi}}$-Lipschitz continuous Jacobian matrix, i.e.,
  $$\|\mathrm{J}_{\boldsymbol{\psi}}(\mathbf{u})-\mathrm{J}_{\boldsymbol{\psi}}(\mathbf{v})\| \leq L_{\boldsymbol{\psi}}\|\mathbf{u}-\mathbf{v}\|,\ \forall \mathbf{u}, \mathbf{v}.$$
  4) $\boldsymbol{\psi}$ has continuous Hessian, i.e.,
  $$\mathrm{H}_{\boldsymbol{\psi}}(\mathbf{z})=(\nabla^2 \boldsymbol{\psi}_1, \cdots, \nabla^2 \boldsymbol{\psi}_n)$$
  exists and is continuous.

In addition, we summarize the commonly used notations in this paper in Table 3.

Table 3: Summary of Notations. The notations shown in the table is for problem $\mathbf{P}$ and we use the same type notations for problem $\mathbf{H}$.

Notation	Definition
$\|\cdot\|$	$l2$-norm $\|\cdot\|_2$
$\mathcal{B}$	unit ball centered at 0
$L_{f,0}$	Lipschitz constant of f
L_f	L_f-smooth property of f
μ_f	μ_f-strong convexity of f
κ_1, κ_2	bi-Lipschitz constant of $\boldsymbol{\psi}$
D	distortion of $\boldsymbol{\psi}$, i.e., κ_2/κ_1
$L_{\boldsymbol{\psi}}$	Lipschitz constant of $\mathrm{J}_{\boldsymbol{\psi}}$
$\mathrm{int}(\mathcal{K})$,$\partial\mathcal{K}$	the interior,boundary of $\mathcal{K}$

C.3 Basic Facts

In this section, we list the fundamental facts we will use in this paper.

Proposition C.1 (Global Optimality Condition, see e.g., [BBV04, Bec17])**.** *For convex constrained optimization* $\{\min_{\mathbf{x}} f(\mathbf{x}),\ s.t.\ \mathbf{x} \in \mathcal{X}\}$*,* $\mathbf{x}^*$ *is a global optimum if and only if* $\mathbf{x}^*$ *is a stationary point of the problem, i.e.,*

$$\langle \nabla f(\mathbf{x}^*), \mathbf{x} - \mathbf{x}^* \rangle \geq 0.$$

Proposition C.2 (Properties of Orthogonal Projection, see e.g., [Bec14])**.** *The projection operator* $\Pi_{\mathcal{C}}$ *over a closed and convex set* $\mathcal{C}$ *satisfies the following properties.*

1) Optimality condition: $\forall \mathbf{y} \in \mathcal{C}$, $\langle \mathbf{x} - \Pi_{\mathcal{C}}(\mathbf{x}), \mathbf{y} - \Pi_{\mathcal{C}}(\mathbf{x}) \rangle \leq 0$.

2) Non-Expansiveness: $\|\Pi_{\mathcal{C}}(\mathbf{x}) - \Pi_{\mathcal{C}}(\mathbf{y})\| \leq \|\mathbf{x} - \mathbf{y}\|$.

3) Monotonicity: $\langle \Pi_{\mathcal{C}}(\mathbf{x}) - \Pi_{\mathcal{C}}(\mathbf{y}), \mathbf{x} - \mathbf{y} \rangle \geq 0$.

We have the following lemma related to $\boldsymbol{\psi}$ to help with the computation.

Lemma C.3. *Suppose* $\mathrm{J}_{\boldsymbol{\psi}}$ *is* $L_{\boldsymbol{\psi}}$ *Lipschitz, i.e.,* $\|\mathrm{J}_{\boldsymbol{\psi}}(\mathbf{u}) - \mathrm{J}_{\boldsymbol{\psi}}(\mathbf{z})\| \leq L_{\boldsymbol{\psi}}\|\mathbf{u} - \mathbf{z}\|$ *for any* $\mathbf{u}$ *and* $\mathbf{z}$*. Then, we have*

$$\|\boldsymbol{\psi}(\mathbf{u}) - \boldsymbol{\psi}(\mathbf{z}) - \mathrm{J}_{\boldsymbol{\psi}}(\mathbf{z})(\mathbf{u} - \mathbf{z})\| \leq \frac{L_{\boldsymbol{\psi}}\|\mathbf{u} - \mathbf{z}\|^2}{2},\ \forall \mathbf{u}, \mathbf{z}.$$

One can refer to Lemma 1.2.3 [N$^+$18] for the proof.

Next, we list the following rules for basic computation:

- Jacobian equivalence: $\mathrm{J}_{\boldsymbol{\psi}^{-1}}(\mathbf{x}) = \mathrm{J}_{\boldsymbol{\psi}}^{-1}(\mathbf{z})$ for $\mathbf{z} = \boldsymbol{\psi}(\mathbf{x})$.
- Chain rule for computing gradient of $h = f \circ \boldsymbol{\psi}$:

$$\nabla h(\mathbf{z}) = \mathrm{J}_{\boldsymbol{\psi}}(\mathbf{z})^\top \nabla f(\boldsymbol{\psi}(\mathbf{z})) = \mathrm{J}_{\boldsymbol{\psi}}(\mathbf{z})^\top \nabla f(\mathbf{x}).$$

- Chain rule for computing gradient of f:

$$\nabla f(\mathbf{x}) = \mathrm{J}_{\boldsymbol{\psi}^{-1}}(\mathbf{x})^\top \nabla h(\mathbf{z}) = \mathrm{J}_{\boldsymbol{\psi}}^{-1}(\mathbf{z})^\top \nabla h(\mathbf{z}).$$

- Chain rule for computing Hessian of $h = f \circ \boldsymbol{\psi}$:

$$\nabla^2 h(\mathbf{z}) = \mathrm{J}_{\boldsymbol{\psi}}(\mathbf{z})^\top \nabla^2 f(\boldsymbol{\psi}(\mathbf{z})) \mathrm{J}_{\boldsymbol{\psi}}(\mathbf{z}) + \sum_{i=1}^{n} \frac{\partial f}{\partial \mathbf{x}_i}(\boldsymbol{\psi}(\mathbf{z})) \nabla^2 \boldsymbol{\psi}_i(\mathbf{z}).$$

D Landscape Analysis

In this section, we provide landscape analysis to understand important relationships between problem $\mathbf{P}$ and $\mathbf{H}$.

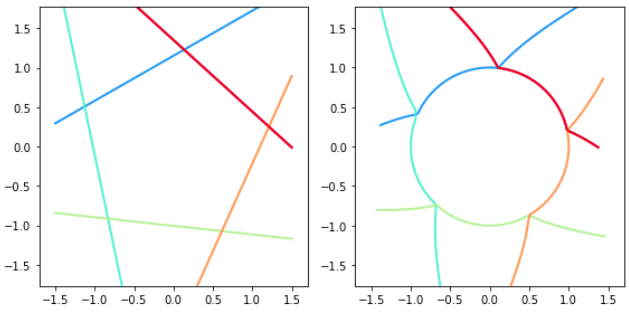


Figure 7: Illustration of the action of homeomorphism on a polyhedron. The left figure is the polyhedral constraints of problem $\mathbf{P}$. Each color of line represents a constraint inequality $\{\mathbf{a}_i^\top \mathbf{x} \leq b_i\}$ for some i. Under the homeomorphic mapping $\boldsymbol{\psi}$, the constrained set is transformed to a ball (right figure). Each constraint inequality $\{G_i(\mathbf{z}) \leq 0\}$ (colored differently) is non-convex in general.

D.1 Action of Homeomorphism on a Constrained Set

Recall that the constrained set is $\mathcal{K} = \{\mathbf{x} \in \mathbb{R}^n \mid \mathbf{g}(\mathbf{x}) \leq 0\}$ with $\mathbf{g} = (g_1, g_2, \cdots, g_m)$ where g_i $(i = 1, 2, \cdots, m)$ is a convex function. For problem $\mathbf{H}$,

$$\mathcal{B} = \boldsymbol{\psi}^{-1}(\mathcal{K}) = \{\mathbf{z} \in \mathbb{R}^n \mid \boldsymbol{\psi}(\mathbf{z}) \in \mathcal{K}\} = \{\mathbf{z} \in \mathbb{R}^n \mid \mathbf{G}(\mathbf{z}) := \mathbf{g}(\boldsymbol{\psi}(\mathbf{z})) \leq \mathbf{0}\}$$

where G_i might be non-convex even for convex g_i. However, $\mathcal{B}$ is assumed to be convex (actually a ball set) in this paper. One can refer to Fig. 7 for an illustration.

Moreover, we assume there are no redundant inequalities in $\mathcal{K}$, i.e., there is no g_i such that it can be represented as the positive linear combination of other inequalities. In this case, any feasible point $\mathbf{x}$ satisfying $g_i(\mathbf{x}) = 0$ for some i is on the boundary of the set $\mathcal{K}$. Moreover, we have

$$\{\mathbf{x} \in \mathcal{K} \mid g_j(\mathbf{x}) = 0, g_k(\mathbf{x}) \neq 0\} \bigcap \{\mathbf{x} \in \mathcal{K} \mid g_k(\mathbf{x}) = 0, g_j(\mathbf{x}) \neq 0\} = \emptyset$$

for any $k \neq j$. Note $\mathcal{B} = \{\mathbf{z} \mid G_i(\mathbf{z}) \leq 0, i = 1, 2, \cdots, m\} = \{\mathbf{z} \mid \|\mathbf{z}\|^2 \leq 1\}$. Moreover, $\{G_i(\mathbf{z}) \leq 0, i = 1, 2, \cdots, m\}$ also has no redundant constraints by the non-singularity of the Jacobian of $\boldsymbol{\psi}$ and similarly,

$$\{\mathbf{z} \in \mathcal{B} \mid G_j(\mathbf{z}) = 0, G_k(\mathbf{z}) \neq 0\} \bigcap \{\mathbf{z} \in \mathcal{B} \mid G_k(\mathbf{z}) = 0\} = \emptyset$$

for any $j \neq k$. Hence if $\mathbf{z} \in \mathcal{B}$ satisfies $G_i(\mathbf{z}) = 0$ for some i, it lies on the boundary of $\mathcal{B}$. Clearly, we have

$$G_i(\mathbf{z}) = \|\mathbf{z}\|^2 - 1 \quad \text{at} \quad \mathbf{z}' \in \mathcal{B}, G_i(\mathbf{z}') = 0, \tag{2}$$

and

$$\nabla G_i(\mathbf{z}) = 2\mathbf{z}, \nabla^2 G_i(\mathbf{z}) = 2\mathbf{I}_n \quad \text{at} \quad \mathbf{z}' \in \mathcal{B}, G_i(\mathbf{z}') = 0. \tag{3}$$

where $\mathbf{I}_n$ is the identity matrix of n by n.

D.2 Properties of Function $h = f \circ \boldsymbol{\psi}$

Lemma D.1 (Properties of $h = f \circ \boldsymbol{\psi}$)**.** *Under the general assumptions C.2, $h = f \circ \boldsymbol{\psi}$ has the following properties.*

1) h is $L_{h,0} := L_{f,0}\kappa_2$ Lipschitz continuous.

2) h is L_h-smooth with $L_h = \kappa_2^2 L_f + L_{\boldsymbol{\psi}} L_{f,0}$.

3) If f is convex, then h is ℓ_h-weakly convex with $\ell_h = L_{f,0} L_{\boldsymbol{\psi}}$.

4) If f is μ_f-strongly convex. Then h satisfies the quadratic growth condition with $m_h = \frac{\mu_f \kappa_1}{2}$ on the ball set $\mathcal{B}$, i.e.,

$$h(\mathbf{z}) - h^* \geq m_h \|\mathbf{z} - \mathbf{z}^*\|^2, \ \forall \mathbf{z} \in \mathcal{B}. \tag{4}$$

Proof. We prove them one by one in the following.

1) We can directly derive from basic definitions:

$$\begin{aligned}\|h(\mathbf{u}) - h(\mathbf{v})\| &\leq \|f(\boldsymbol{\psi}(\mathbf{u})) - f(\boldsymbol{\psi}(\mathbf{v}))\| \\ &\leq L_{f,0}\|\boldsymbol{\psi}(\mathbf{u}) - \boldsymbol{\psi}(\mathbf{v})\| \\ &\leq L_{f,0}L_{\boldsymbol{\psi}}\|\mathbf{u} - \mathbf{v}\|.\end{aligned}$$

2) From L_f-smoothness of f, we have

$$\|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\| \leq L_f\|\mathbf{x} - \mathbf{y}\|. \tag{5}$$

Then we derive with $\mathbf{x} = \boldsymbol{\psi}(\mathbf{z}), \mathbf{v} = \boldsymbol{\psi}(\mathbf{y})$,

$$\begin{aligned}\|\nabla h(\mathbf{z}) - \nabla h(\mathbf{v})\| &= \left\|\mathrm{J}_{\boldsymbol{\psi}}(\mathbf{z})^\top \nabla f(\mathbf{x}) - \mathrm{J}_{\boldsymbol{\psi}}(\mathbf{v})^\top \nabla f(\mathbf{y})\right\| \\ &= \left\|\mathrm{J}_{\boldsymbol{\psi}}(\mathbf{z})^\top (\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})) + (\mathrm{J}_{\boldsymbol{\psi}}(\mathbf{z}) - \mathrm{J}_{\boldsymbol{\psi}}(\mathbf{v}))^\top \nabla f(\mathbf{y})\right\| \\ &\leq \left\|\mathrm{J}_{\boldsymbol{\psi}}(\mathbf{z})^\top (\nabla f(\mathbf{x}) - \nabla f(\mathbf{y}))\right\| + \left\|(\mathrm{J}_{\boldsymbol{\psi}}(\mathbf{z}) - \mathrm{J}_{\boldsymbol{\psi}}(\mathbf{v}))^\top \nabla f(\mathbf{y})\right\| \\ &\leq \kappa_2 L_f\|\boldsymbol{\psi}(\mathbf{z}) - \boldsymbol{\psi}(\mathbf{v})\| + L_{\boldsymbol{\psi}}L_{f,0}\|\mathbf{z} - \mathbf{v}\| \\ &\leq \left(\kappa_2^2 L_f + L_{\boldsymbol{\psi}}L_{f,0}\right)\|\mathbf{z} - \mathbf{v}\|.\end{aligned}$$

Let $L_h = \kappa_2^2 L_f + L_{\boldsymbol{\psi}}L_{f,0}$. We have the conclusion.

3) One hope to show $h(\cdot) + \frac{\ell_h}{2}\|\cdot + \mathbf{v}\|^2$ is a convex function, i.e.,

$$h(\mathbf{v}) + \frac{\ell_h}{2}\|\mathbf{v}\|^2 \geq h(\mathbf{z}) + \frac{\ell_h}{2}\|\mathbf{z}\|^2 + \langle \nabla h(\mathbf{z}) + \ell_h\mathbf{z}, \mathbf{v} - \mathbf{z}\rangle,\ \forall \mathbf{z}, \mathbf{v}.$$

This is equivalent to show

$$h(\mathbf{v}) + \frac{\ell_h}{2}\|\mathbf{v} - \mathbf{z}\|^2 \geq h(\mathbf{z}) + \langle \nabla h(\mathbf{z}), \mathbf{v} - \mathbf{z}\rangle,\ \forall \mathbf{z}, \mathbf{v}.$$

We drive with $\mathbf{x} = \boldsymbol{\psi}(\mathbf{z}), \mathbf{y} = \boldsymbol{\psi}(\mathbf{v})$ as follows,

$$\begin{aligned}\langle \nabla h(\mathbf{z}), \mathbf{v} - \mathbf{z}\rangle &= \left\langle \nabla \mathrm{J}_{\boldsymbol{\psi}}(\mathbf{z})^\top f(\mathbf{x}), \mathbf{v} - \mathbf{z}\right\rangle \\ &= \langle \nabla f(\mathbf{x}), \mathrm{J}_{\boldsymbol{\psi}}(\mathbf{z})(\mathbf{v} - \mathbf{z})\rangle \\ &= \langle \nabla f(\mathbf{x}), -\boldsymbol{\psi}(\mathbf{v}) + \boldsymbol{\psi}(\mathbf{z}) + \mathrm{J}_{\boldsymbol{\psi}}(\mathbf{z})(\mathbf{v} - \mathbf{z})\rangle + \langle \nabla f(\mathbf{x}), \boldsymbol{\psi}(\mathbf{v}) - \boldsymbol{\psi}(\mathbf{z})\rangle \\ &\leq \|\nabla f(\mathbf{x})\| \cdot \|\boldsymbol{\psi}(\mathbf{v}) - \boldsymbol{\psi}(\mathbf{z}) - \mathrm{J}_{\boldsymbol{\psi}}(\mathbf{z})(\mathbf{v} - \mathbf{z})\| + \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x}\rangle \\ &\leq L_{f,0}L_{\boldsymbol{\psi}}\|\mathbf{z} - \mathbf{v}\|^2 + f(\mathbf{y}) - f(\mathbf{x}) \\ &= L_{f,0}L_{\boldsymbol{\psi}}\|\mathbf{z} - \mathbf{v}\|^2 + h(\mathbf{v}) - h(\mathbf{z})\end{aligned}$$

where the first inequality is from triangular inequality and the second inequality is from Lemma C.3 and the convexity of f.

4) Note that optimality implies stationarity. By the strong convexity of f, for any $\mathbf{x} \in \mathcal{K}$ and $\mathbf{x}^* \in \mathcal{K}^*$, we have

$$f(\mathbf{x}) \geq f(\mathbf{x}^*) + \langle \nabla f(\mathbf{x}^*), \mathbf{x} - \mathbf{x}^*\rangle + \frac{\mu_f}{2}\|\mathbf{x} - \mathbf{x}^*\|^2 \geq f(\mathbf{x}^*) + \frac{\mu_f}{2}\|\mathbf{x} - \mathbf{x}^*\|^2.$$

Here the last inequality is from the global optimality condition C.1.

With $\mathbf{x} = \boldsymbol{\psi}(\mathbf{z})$, we have

$$h(\mathbf{z}) \geq h(\mathbf{z}^*) + \frac{\mu_f}{2}\|\boldsymbol{\psi}(\mathbf{z}) - \boldsymbol{\psi}(\mathbf{z}^*)\|^2 \geq h(\mathbf{z}^*) + \frac{\mu_f}{2}\kappa_1\|\mathbf{z} - \mathbf{z}^*\|^2.$$

□

D.3 Stationary Points and KKT Points

Recall that the stationary point $\mathbf{x}^\star$ of a set $\mathcal{K}$ is defined by the variational inequality:

$$\langle \nabla f(\mathbf{x}^*), \mathbf{x} - \mathbf{x}^\star \rangle \geq 0, \forall \mathbf{x} \in \mathcal{K}. \tag{6}$$

Besides, a tuple $(\mathbf{x}, \boldsymbol{\lambda})$ is said to satisfy the Karush–Kuhn–Tucker (KKT) condition of problem $\mathbf{P}$ if the following holds

$$\begin{aligned} \nabla f(\mathbf{x}) + \sum_{i=1}^{m} \lambda_i \nabla g_i(\mathbf{x}) &= \mathbf{0}, \\ g_i(\mathbf{x}) &\leq 0, \quad i = 1, 2, \cdots, m \\ \boldsymbol{\lambda} \geq \mathbf{0},\ \lambda_i g_i(\mathbf{x}) &= 0, \quad i = 1, 2, \cdots, m \end{aligned} \tag{7}$$

where $\boldsymbol{\lambda}$ is the dual variable corresponding to inequality constraints.

Definition D.2 (KKT Stationary Point)**.** A point $\mathbf{x}^*$ is said to be a KKT stationary point of $\mathbf{P}$ if there exists $\boldsymbol{\lambda}^*$ such that $(\mathbf{x}^*, \boldsymbol{\lambda}^*)$ satisfies KKT condition (7).

Actually, these two definitions are equivalent under mild conditions, which will be discussed later. Before moving on, we introduce related definitions.

Definition D.3 (Constraints Qualification)**.** Let $\bar{\mathbf{x}}$ be feasible for a problem with constraint set

$$\{\mathbf{x} \mid g_i(\mathbf{x}) \leq 0, q_j(\mathbf{x}) = 0, \text{ for } i = 1, 2, \cdots, m, \text{ and } j = 1, 2, \cdots, p\}$$

and put $\mathcal{A}(\bar{\mathbf{x}}) := \{i \mid g_i(\bar{\mathbf{x}}) = 0\}$. We say that

1) the linear independence constraint qualification (LICQ) holds at $\bar{\mathbf{x}}$ (and write $\mathrm{LICQ}(\bar{\mathbf{x}})$) if the gradients

$$\nabla g_i(\bar{\mathbf{x}})(i \in \mathcal{A}(\bar{\mathbf{x}})), \quad \nabla q_j(\bar{\mathbf{x}})(j = 1, \ldots, p)$$

are linearly independent.

2) the Mangasarian-Fromovitz constraint qualification (MFCQ) holds at $\bar{\mathbf{x}}$ (and write $\mathrm{MFCQ}(\bar{\mathbf{x}})$) if the gradients

$$\nabla q_j(\bar{\mathbf{x}})(j = 1, \ldots, p)$$

are linearly independent and there exists a vector $d \in \mathbb{R}^n$ such that

$$\nabla g_i(\bar{\mathbf{x}})^\top d < 0 (i \in \mathcal{A}(\bar{\mathbf{x}})), \quad \nabla q_j(\bar{\mathbf{x}})^\top d = 0 (j = 1, \ldots, p).$$

Definition D.4 (Strict Slackness)**.** It is said that the strict complementary slackness condition holds for problem $\mathbf{P}$, if

$$\lambda_i^* > 0 \quad \text{for} \quad g_i(\mathbf{x}^*) = 0,$$

and for problem $\mathbf{H}$, if

$$\nu_* > 0 \quad \text{for} \quad \|\mathbf{z}^*\| = 1.$$

Definition D.5 (Slater's Condition)**.** It is said that the Slater's condition holds for a problem with constrained set $\mathcal{K} := \{g_i(\mathbf{x}) \leq 0, i = 1, 2, \cdots, m\}$, if there exists $\mathbf{x}_0$ such that for all $i = 1, 2, \cdots, m$,

$$g_i(\mathbf{x}_0) < 0.$$

It is established that the variational inequality (6) and the KKT system (7) coincide with a strong relationship if certain constraints qualification holds. One could refer to [HP90, FFK98, Tob86] for detailed discussion. We summarize the conclusion in the following proposition.

Proposition D.6. *We have the following relationship between (6) and (7).*

1) If $\mathbf{x}^*$ is a solution of the variational inequality (6) and a certain constraint qualification (e.g., LICQ, MFCQ, see Def. D.3) holds at $\mathbf{x}^*$, then there exists λ^* such that the tuple $(\mathbf{x}^*, \boldsymbol{\lambda}^*)$ satisfy the KKT system (7).

2) Suppose f and g_i $(i = 1, 2, \cdots, m)$ are convex and Slater's condition D.5 is satisfied. If $\mathbf{x}^*$ is a solution of the variational inequality (6), then there exists λ^* such that the tuple $(\mathbf{x}^*, \boldsymbol{\lambda}^*)$ satisfy the KKT system (7).

3) If $(\mathbf{x}^*, \boldsymbol{\lambda}^*)$ satisfies the KKT system (7) and g is convex, then $\mathbf{x}^*$ satisfy the variational inequalities (6).

Proof. One can refer to e.g. [Tob86, HP90] for item (1) and (3) and refer to e.g. Theorem 3.78 [Bec17] for item (2). □

D.4 KKT Conditions of Problem P and H

We first introduce the following definition.

- The normal cone $\mathrm{N}_S(\mathbf{x})$ of a closed and convex set $\mathcal{K}$ at $\mathbf{x} \in \mathcal{K}$ is defined as
$$\mathrm{N}_{\mathcal{K}}(\mathbf{x}) = \{\mathbf{y} : \langle \mathbf{y}, \mathbf{z} - \mathbf{x} \rangle \leq 0 \text{ for any } \mathbf{z} \in \mathcal{K}\}.$$
- The tangent cone $\mathrm{T}_{\mathcal{K}}(\mathbf{x})$ of a closed and convex set $\mathcal{K}$ at $\mathbf{x} \in \mathcal{K}$ is defined as
$$\mathrm{T}_{\mathcal{K}}(\mathbf{x}) = \mathrm{cl}\{\mathbf{y} : \exists \lambda > 0 \text{ s.t } \mathbf{x} + \lambda \mathbf{y} \in \mathcal{K}\}$$
where $\mathrm{cl}(\cdot)$ denotes the closure of a set.
- The critical cone $\mathrm{C}_{\mathcal{K}}(\mathbf{x})$ of a closed and convex set $\mathcal{K}$ at $\mathbf{x} \in \mathcal{K}$ with related to a objective function f is defined as
$$\mathrm{C}_{\mathcal{K}}(\mathbf{x}) = \left\{\mathbf{d} \in \mathrm{T}_{\mathcal{K}}(\mathbf{x}) : \nabla f(\mathbf{x})^\top \mathbf{d} = 0\right\}.$$

To define the second-order KKT condition for the optimization problems, we first recall that the critical cone of problem $\mathbf{P}$ can be written as [NW99]

$$\mathbf{d} \in \mathrm{C}_{\mathcal{K}}(\mathbf{x}^*) \Leftrightarrow \begin{cases} \nabla g_i(\mathbf{x}^*)^T \mathbf{d} = 0, & \text{for all } i \in \mathcal{A}(\mathbf{x}^*) \text{ with } \lambda_i^* > 0, \\ \nabla g_i(\mathbf{x}^*)^T \mathbf{d} \geq 0, & \text{for all } i \in \mathcal{A}(\mathbf{x}^*) \text{ with } \lambda_i^* = 0. \end{cases}$$

Here $\boldsymbol{\lambda}^*$ is the Lagrangian multiplier of inequality constraints g_i and $\mathcal{A}(\mathbf{x}^*)$ is the index of active constraints. Moreover, if *strict complementary slackness* holds, the critical cone is simplified as

$$\mathrm{C}_{\mathcal{K}}(\mathbf{x}^*) = \{\mathbf{d} \in \mathbb{R}^n \mid \nabla g_i(\mathbf{x}^*)^T \mathbf{d} = 0, \text{ for all } i \in \mathcal{A}(\mathbf{x}^*)\}.$$

Suppose *strict complementary slackness* holds for problem $\mathbf{P}$ and $\mathbf{H}$. Then, we can write KKT conditions for problem $\mathbf{P}$ and $\mathbf{H}$ in the following.

First-order KKT conditions on $\mathbf{x}^*$. The Lagrangian of $\mathbf{P}$ is

$$\mathcal{L}_{\mathrm{P}}(\mathbf{x}, \boldsymbol{\lambda}) = f(\mathbf{x}) + \sum_{i=1}^m \lambda_i g_i(\mathbf{x}).$$

The first-order KKT conditions of $\mathbf{P}$ are: there exists $\boldsymbol{\lambda}^*$ such that

$$\nabla f(\mathbf{x}^*) + \sum_{i=1}^m \lambda_i^* \nabla g_i(\mathbf{x}^*) = \mathbf{0}, \tag{8a}$$

$$g_i(\mathbf{x}^*) \leq 0, \quad i = 1, 2, \cdots, m \tag{8b}$$

$$\boldsymbol{\lambda}^* \geq \mathbf{0}, \ \lambda_i^* g_i(\mathbf{x}^*) = 0, \quad i = 1, 2, \cdots, m. \tag{8c}$$

Second-order KKT conditions on $\mathbf{x}^*$. It adds the following condition

$$\mathbf{w}^\top \nabla_{\mathbf{x}}^2 \mathcal{L}_{\mathrm{P}}(\mathbf{x}^*, \boldsymbol{\lambda}^*) \mathbf{w} \geq 0 \tag{9}$$

for any $\mathbf{w}$ satisfying $\mathbf{w}^\top \nabla g_i(\mathbf{x}^*) = 0$ with $i \in \mathcal{A}(\mathbf{x}^*)$.

First-order KKT conditions on $\mathbf{z}^*$. The Lagrangian of $\mathbf{H}$ is

$$\mathcal{L}_{\mathrm{H}}(\mathbf{z}, \nu) = h(\mathbf{z}) + \nu(\|\mathbf{z}\|^2 - 1).$$

The first-order KKT conditions of $\mathbf{H}$ are: there exists ν^* such that

$$\nabla h(\mathbf{z}^*) + 2\nu^* \mathbf{z}^* = \mathbf{0}, \tag{10a}$$

$$\|\mathbf{z}^*\|^2 \leq 1, \tag{10b}$$

$$\nu^* \geq 0, \ \nu^*(\|\mathbf{z}^*\|^2 - 1) = 0. \tag{10c}$$

Second-order KKT condition on $\mathbf{z}^*$. It will add the following condition.

$$\mathbf{d}^\top \nabla_{\mathbf{z}}^2 \mathcal{L}_{\mathrm{H}}(\mathbf{z}^*, \nu^*) \mathbf{d} \geq 0 \tag{11}$$

for any $\mathbf{d} \in \mathrm{C}_{\mathcal{B}}(\mathbf{z}^*)$. Here recall that

$$\mathrm{C}_{\mathcal{B}}(\mathbf{z}^*) = \begin{cases} \mathbb{R}^n, & \text{if } \mathbf{z}^* \in \mathrm{int}(\mathcal{B}), \\ \{\mathbf{d} : \mathbf{d}^\top \mathbf{z}^* = 0\}, & \text{if } \mathbf{z}^* \in \partial\mathcal{B}. \end{cases}$$

D.5 Relationships of KKT Points between Problem P and H

Lemma D.7. *We have that* $\mathbf{x}^*$ *is a KKT stationary point of* $\mathbf{P}$ *if and only if* $\mathbf{z}^*$ *is also a KKT stationary point of* $\mathbf{H}$.

Proof. 1) First, we assume that $\mathbf{z}^*$ is a KKT stationary point of $\mathbf{P}$. By assumption, there exists $\boldsymbol{\lambda}^*$ such that the KKT condition holds (8) holds. Then we have

$$\begin{aligned} \mathrm{J}_{\boldsymbol{\psi}}(\mathbf{z}^*)^\top \nabla f(\mathbf{x}^*) + \sum_{i=1}^m \lambda_i^* \mathrm{J}_{\boldsymbol{\psi}}(\mathbf{z}^*)^\top \nabla g_i(\mathbf{x}^*) = \mathbf{0}, \\ g_i(\boldsymbol{\psi}(\mathbf{z}^*)) \le 0, \quad i = 1, 2, \cdots, m \\ \boldsymbol{\lambda}^* \ge \mathbf{0},\ \lambda_i^* g_i(\boldsymbol{\psi}(\mathbf{z}^*)) = 0, \quad i = 1, 2, \cdots, m. \end{aligned}$$

This is equivalent to

$$\nabla h(\mathbf{z}^*) + \sum_{i=1}^m \lambda_i^* \nabla G_i(\mathbf{z}^*) = \mathbf{0}, \tag{12a}$$

$$G_i(\mathbf{z}^*) \le 0, \quad i = 1, 2, \cdots, m \tag{12b}$$

$$\boldsymbol{\lambda}^* \ge \mathbf{0},\ \lambda_i^* G_i(\mathbf{z}^*) = 0, \quad i = 1, 2, \cdots, m. \tag{12c}$$

Let $\nu^* = \sum_{i=1} \lambda_i^*$. According to the eq. (2,3), eq. (12a) is actually

$$\nabla h(\mathbf{z}^*) + 2\nu^* \mathbf{z}^* = \mathbf{0}.$$

By assumption, eq. (12b) is equivalent to

$$\|\mathbf{z}^*\|^2 \le 1.$$

Note that if $G_i(\mathbf{z}^*) < 0$ for all i, then $\boldsymbol{\lambda}^* = 0$ and thus $\nu^* = 0$. In this case, $\nu^*(\|\mathbf{z}^*\|^2 - 1) = 0$. If $\mathbf{z}^*$ makes at least one $G_i(\mathbf{z}^*) = 0$, then we have $\|\mathbf{z}^*\|^2 = 1$. In this case, we also have $\nu^*(\|\mathbf{z}^*\|^2 - 1) = 0$. Hence, eq. (12c) imply

$$\nu^* \ge 0, \nu^*(\|\mathbf{z}^*\|^2 - 1) = 0.$$

In conclusion, there exists $\mathbf{z}^*, \nu*$ such the KKT condition holds.

2) Now, we assume $\mathbf{z}^*, \nu^*$ satisfy KKT condition for problem $\mathbf{H}$, i.e.,

$$\begin{aligned} \nabla h(\mathbf{z}^*) + 2\nu^* \mathbf{z}^* = \mathbf{0}, \\ \|\mathbf{z}^*\|^2 \le 1, \\ \nu^* \ge 0,\ \nu^*(\|\mathbf{z}^*\|^2 - 1) = 0. \end{aligned}$$

If $\mathbf{z}^* \in \mathrm{int}(\mathcal{B})$, then $G_i(\mathbf{z}^*) < 0$ for all i and $\nu^* = 0$. In this case, there exists $\boldsymbol{\lambda}^* = 0$ such that the KKT condition of problem $\mathbf{P}$ holds at $\mathbf{x}^* = \boldsymbol{\psi}(\mathbf{z}^*), \boldsymbol{\lambda}^*$ as eq. (10).

If $\mathbf{z}^* \in \partial\mathcal{B}$, then there exists at least one $i \in \{1, 2, \cdots, m\}$ such that $G_i(\mathbf{z}^*) = 0$ and $\nu^* > 0$ from strict complementary slackness. Denote $\mathcal{A} = \{i : G_i(\mathbf{z}^*) = 0\}$. Note we define $\lambda_i^* = 0$ if $i \notin \mathcal{A}$ and $\lambda_i^* = \nu^*/|\mathcal{A}|$. Then we have $\mathbf{z}^*, \boldsymbol{\lambda}^*$ such that eq. 12 holds which implies $\mathbf{x}^* = \boldsymbol{\psi}(\mathbf{z}^*), \boldsymbol{\lambda}^*$ make the KKT condition of problem $\mathbf{P}$ holds.

□

Lemma D.8. *Suppose strict complementary slackness condition holds for both problem* $\mathbf{P}$ *and* $\mathbf{H}$. *Then* $\mathbf{x}^*$ *is a second-order KKT stationary point of* $\mathbf{P}$ *if and only if* $\mathbf{z}^* = \boldsymbol{\psi}^{-1}(\mathbf{x}^*)$ *is also a second-order KKT stationary point of* $\mathbf{H}$.

Proof. From Lemma D.7, there exists $\boldsymbol{\lambda}^*$ and ν^* such that $(\mathbf{x}^*, \boldsymbol{\lambda}^*)$ holds for first-order KKT condition of $\mathbf{P}$ if and only if $(\mathbf{z}^*, \nu^*)$ holds for first-order KKT condition of $\mathbf{H}$. Hence, it suffices to show the equivalence of condition 11 and 9.

1) Let's first suppose $\mathbf{x}^*$ is a second-order KKT stationary point, i.e., eq. (9) holds.

Note

$$\nabla^2_{\mathbf{z}} \mathcal{L}_{\mathrm{H}}(\mathbf{z}^*, \nu^*) = \nabla^2 h(\mathbf{z}^*) + 2\nu^* \mathbf{I}_n,$$

where $\mathbf{I}_n$ is identity matrix of size $n \times n$. We just need to show $\mathbf{d}^\top \nabla \mathcal{L}_{\mathrm{H}}(\mathbf{z}^*, \nu^*)\mathbf{d} \geq 0$ for any $\mathbf{d} \in \mathrm{C}_{\mathcal{B}}(\mathbf{z}^*)$. Recall that

$$\nabla^2 h(\mathbf{z}^*) = \mathrm{J}_{\boldsymbol{\psi}}(\mathbf{z}^*)^\top \nabla^2 f(\boldsymbol{\psi}(\mathbf{z}^*))\mathrm{J}_{\boldsymbol{\psi}}(\mathbf{z}^*) + \sum_{i=1}^{n} \frac{\partial f}{\partial \mathbf{x}_i}(\boldsymbol{\psi}(\mathbf{z}^*))\nabla^2 \boldsymbol{\psi}_i(\mathbf{z}^*),$$

and

$$\nabla^2 G_i(\mathbf{z}^*) = \mathrm{J}_{\boldsymbol{\psi}}(\mathbf{z}^*)^\top \nabla^2 g_i(\boldsymbol{\psi}(\mathbf{z}^*))\mathrm{J}_{\boldsymbol{\psi}}(\mathbf{z}^*) + \sum_{k=1}^{n} \frac{\partial g_i}{\partial \mathbf{x}_k}(\boldsymbol{\psi}(\mathbf{z}^*))\nabla^2 \boldsymbol{\psi}_k(\mathbf{z}^*), \;\; k = 1, 2, \cdots, m.$$

From eq. (2), note that

$$\nabla^2 G_k(\mathbf{z}^*) = 2\mathbf{I}_n, \forall k \in \mathcal{A}(\mathbf{x}^*) \cap \{k : G_k(\mathbf{z}^*) = 0\}.$$

From Lemma D.7, $\nu^* = \sum_i \lambda_i^*$. Then we have

$$\nabla^2 \mathcal{L}_{\mathrm{H}}(\mathbf{z}^*, \nu^*) = \nabla^2 h(\mathbf{z}^*) + \sum_{i=1}^{m} \lambda_i^* \nabla^2 G_i(\mathbf{z}^*) \tag{13a}$$

$$= \mathrm{J}_{\boldsymbol{\psi}}(\mathbf{z}^*)^\top \nabla^2 f(\boldsymbol{\psi}(\mathbf{z}^*))\mathrm{J}_{\boldsymbol{\psi}}(\mathbf{z}^*) + \sum_{i=1}^{m} \mathrm{J}_{\boldsymbol{\psi}}(\mathbf{z}^*)^\top \lambda_i^* \nabla^2 g_i(\boldsymbol{\psi}(\mathbf{z}^*))\mathrm{J}_{\boldsymbol{\psi}}(\mathbf{z}^*) \tag{13b}$$

$$+ \sum_{k=1}^{n} \frac{\partial f}{\partial \mathbf{x}_k}(\boldsymbol{\psi}(\mathbf{z}^*))\nabla^2 \psi_k(\mathbf{z}^*) + \sum_{k=1}^{n}\sum_{i=1}^{m} \lambda_i^* \frac{\partial g_i}{\partial \mathbf{x}_k}(\boldsymbol{\psi}(\mathbf{z}^*))\nabla^2 \psi_k(\mathbf{z}^*). \tag{13c}$$

From first-order KKT stationarity of $\mathbf{P}$, i.e.,

$$\nabla f(\mathbf{x}^*) + \sum_{i=1}^{m} \lambda_i^* \nabla g_i(\mathbf{x}^*) = \mathbf{0},$$

We have

$$\frac{\partial f}{\partial \mathbf{x}_k}(\mathbf{x}^*) + \sum_{i=1}^{m} \lambda_i^* \frac{\partial g_i}{\partial \mathbf{x}_k}(\mathbf{x}^*) = 0.$$

Hence for any $\mathbf{d} \in \mathrm{C}_{\mathcal{B}}(\mathbf{z}^*)$, we have the second term (13c) is equal to 0.

Now we note it's trivial that $\mathrm{C}_{\mathcal{K}}(\mathbf{x})^* = \mathrm{C}_{\mathcal{B}}(\mathbf{z}^*) = \mathbb{R}^n$ if $\mathbf{z}^* \in \mathrm{int}(\mathcal{K})$ where $\mathbf{x} = \boldsymbol{\psi}(\mathbf{z}^*)$. Hence in this case if $\mathbf{d} \in \mathrm{C}_{\mathcal{B}}(\mathbf{z}^*)$, we will have $\mathrm{J}_{\boldsymbol{\psi}}(\mathbf{z}^*)\mathbf{d} \in \mathrm{C}_{\mathcal{K}}(\mathbf{x}^*)$

If $\mathbf{x}^* \in \partial\mathcal{K}$. Then $\mathcal{A}(\mathbf{x}^*) \neq \emptyset$. For $\mathbf{d} \in \mathrm{C}_{\mathcal{B}}(\mathbf{z}^*)$, i.e., $\mathbf{d}^\top \mathbf{z}^* = 0$, we have

$$(\mathrm{J}_{\boldsymbol{\psi}}(\mathbf{z}^*)\mathbf{d})^\top \nabla g_i(\mathbf{x}^*) = \mathbf{d}^\top \mathrm{J}_{\boldsymbol{\psi}}(\mathbf{z}^*)^\top \nabla g_i(\mathbf{x}^*) = \mathbf{d}^\top G_i(\mathbf{z}^*) = 2\mathbf{d}^\top \mathbf{z}^* = 0, \quad \text{for } i \in \mathcal{A}(\mathbf{x}^*),$$

or $\mathrm{J}_{\boldsymbol{\psi}}(\mathbf{z}^*)\mathbf{d} \in \mathrm{C}_{\mathcal{K}}(\mathbf{x}^*)$.

So for $\mathbf{d}^\top \in \mathrm{C}_{\mathcal{B}}(\mathbf{z}^*)$, we have the following holds about the first term of $\nabla^2 \mathcal{L}_{\mathrm{H}}(\mathbf{z}^*, \nu^*)$.

$$\left(\mathrm{J}_{\boldsymbol{\psi}}\left(\mathbf{z}^*\right)\mathbf{d}\right)^\top \nabla^2 f\left(\boldsymbol{\psi}\left(\mathbf{z}^*\right)\right) \mathrm{J}_{\boldsymbol{\psi}}\left(\mathbf{z}^*\right)\mathbf{d} + \left(\mathrm{J}_{\boldsymbol{\psi}}\left(\mathbf{z}^*\right)\mathbf{d}\right)^\top \left(\sum_{i=1}^{m} \lambda_i^* \nabla^2 g_i\left(\boldsymbol{\psi}\left(\mathbf{z}^*\right)\right)\right) \mathrm{J}_{\boldsymbol{\psi}}\left(\mathbf{z}^*\right)\mathbf{d} \geq 0$$

where the last '$\geq$' is from the assumption that $\mathbf{x}^*$ is the second-order KKT stationary point of $\mathbf{P}$. Hence, we have $\mathbf{d}^\top \nabla^2 \mathcal{L}_{\mathrm{H}}(\mathbf{z}^*, \nu^*)\mathbf{d} \geq 0$ for any $\mathbf{d}^T \in \mathrm{C}_{\mathcal{B}}(\mathbf{z}^*)$, i.e., $\mathbf{z}^* = \boldsymbol{\psi}^{-1}(\mathbf{x}^*)$ is also a second-order KKT stationary point.

2) Let's suppose $\mathbf{z}^*$ is a second-order KKT stationary point and show that $\mathbf{x}^*$ is a second-order KKT stationary point.

If $\mathbf{z}^* \in \mathrm{int}(\mathcal{B})$, the proof is trivial because $\nu^* = 0$ according to the similar analysis. So we assume $\mathbf{z}^* \in \partial\mathcal{B}$. Define $\mathcal{A}(\mathbf{z}^*) = \{i : G_i(\mathbf{z}^*) = 0\}$, and $\lambda_i^* = 0$ for $i \notin \mathcal{A}(\mathbf{z}^*)$, $\lambda_i^* = \nu^*/|\mathcal{A}(\mathbf{z}^*)|$ for $i \in \mathcal{A}(\mathbf{z}^*)$.

Note for any $\mathbf{w} \in \mathrm{C}_{\mathcal{K}}(\mathbf{x}^*)$, we have

$$\mathbf{0} = \mathbf{w}^\top \nabla g_i(\mathbf{x}^*) = \mathbf{w}^\top \mathrm{J}_{\boldsymbol{\psi}}^{-1}(\mathbf{z}^*)\nabla G_i(\mathbf{z}^*) = (\mathrm{J}_{\boldsymbol{\psi}}^{-1}(\mathbf{z}^*)\mathbf{w})^\top \mathbf{z}^*, \;\; \text{for } \; i \in \mathcal{A}(\mathbf{x}^*) = \mathcal{A}(\mathbf{z}^*).$$

Hence $\mathrm{J}_{\boldsymbol{\psi}}^{-1}(\mathbf{z}^*)\mathbf{w} \in \mathrm{C}_{\mathcal{B}}(\mathbf{z}^*)$. Then for any $\mathbf{w} \in \mathrm{C}_{\mathcal{K}}(\mathbf{x}^*)$,

$$
\begin{aligned}
&\mathbf{w}^\top \nabla^2_{\mathbf{x}} \mathcal{L}_{\mathrm{P}}(\mathbf{x}^*, \boldsymbol{\lambda}^*)\mathbf{w} \\
&= \mathbf{w}^\top \nabla^2 f(\mathbf{x}^*)\mathbf{w} + \mathbf{w}^\top \sum_{i=1}^m \lambda_i^* \nabla^2 g_i(\mathbf{x}^*)\mathbf{w} \\
&= (\mathrm{J}_{\boldsymbol{\psi}}^{-1}(\mathbf{z}^*)\mathbf{w})^\top \mathrm{J}_{\boldsymbol{\psi}}(\mathbf{z}^*) \nabla^2 f(\mathbf{x}^*) \mathrm{J}_{\boldsymbol{\psi}}(\mathbf{z}^*) \mathrm{J}_{\boldsymbol{\psi}}^{-1}(\mathbf{z}^*)\mathbf{w} \\
&+ (\mathrm{J}_{\boldsymbol{\psi}}^{-1}(\mathbf{z}^*)\mathbf{w})^\top \mathrm{J}_{\boldsymbol{\psi}}(\mathbf{z}^*) (\sum_{i=1}^m \lambda_i^* \nabla^2 g_i(\mathbf{x}^*)) \mathrm{J}_{\boldsymbol{\psi}}(\mathbf{z}^*) \mathrm{J}_{\boldsymbol{\psi}}^{-1}(\mathbf{z}^*)\mathbf{w} \\
&+ (\mathrm{J}_{\boldsymbol{\psi}}^{-1}(\mathbf{z}^*)\mathbf{w})^\top [\sum_{k=1}^n \frac{\partial f}{\partial \mathbf{x}_k}(\boldsymbol{\psi}(\mathbf{z}^*)) \nabla^2 \psi_k(\mathbf{z}^*) + \sum_{k=1}^n \sum_{i=1}^m \lambda_i^* \frac{\partial g_i}{\partial \mathbf{x}_k}(\boldsymbol{\psi}(\mathbf{z}^*)) \nabla^2 \psi_k(\mathbf{z}^*)] \mathrm{J}_{\boldsymbol{\psi}}^{-1}(\mathbf{z}^*)\mathbf{w} \\
&= (\mathrm{J}_{\boldsymbol{\psi}}^{-1}(\mathbf{z}^*)\mathbf{w})^\top \mathcal{L}_{\mathrm{H}}(\mathbf{z}^*, \nu^*) \mathrm{J}_{\boldsymbol{\psi}}^{-1}(\mathbf{z}^*)\mathbf{w} \geq 0
\end{aligned}
$$

where the sum of last term of the second '$=$' is exactly 0 and the last '$\geq$' is from the assumption that $\mathbf{z}^*$ is a second-order KKT stationary point.

□

Definition D.9 (Non-degenerate KKT Stationary Point)**.** A second-order KKT point $\mathbf{x}^*$ of $\mathbf{P}$ is said to be non-degenerate if there exists $\boldsymbol{\lambda}^*$ such that

$$\mathbf{d}^\top \nabla^2 \mathcal{L}(\mathbf{x}^*, \boldsymbol{\lambda}^*)\mathbf{d} > 0$$

for all $0 \neq \mathbf{d} \in \mathrm{C}_{\mathcal{K}}(\mathbf{x}^*)$. Here the Lagrangian function is

$$\mathcal{L}(\mathbf{x}, \boldsymbol{\lambda}) = f(\mathbf{x}) + \sum_{i=1}^m \lambda_i g_i(\mathbf{x}).$$

Lemma D.10. *Suppose strict complementary slackness holds for problem* $\mathbf{P}$ *and* $\mathbf{H}$*. Then* $\mathbf{x}^\star$ *is a non-degenerate KKT point of optimization* $\mathbf{P}$ *if and only if* $\mathbf{z}^\star$ *satisfying* $\mathbf{x}^\star = \boldsymbol{\psi}(\mathbf{z}^*)$ *is also a non-degenerate KKT point of problem* $\mathbf{H}$*.*

Proof. 1) Suppose $\mathbf{x}^*$ is a non-degenerate KKT stationary point. Note that for $\mathbf{d} \in \mathrm{C}_{\mathcal{B}}(\mathbf{z}^*)$, we have $\mathrm{J}_{\boldsymbol{\psi}}(\mathbf{z}^*)\mathbf{d} \in \mathrm{C}_{\mathcal{K}}(\mathbf{x}^*)$ from the proof of Lemma D.8. Moreover, from $\mathrm{J}_{\boldsymbol{\psi}}(\mathbf{z}^*) \neq 0$ we have $\mathrm{J}_{\boldsymbol{\psi}}(\mathbf{z}^*)\mathbf{d} \neq 0$ if and only if $\mathbf{d} \neq 0$. Then the conclusion is trivial from eq. (13) in the proof of Lemma D.8.

2) Now, we suppose $\mathbf{z}^*$ is a non-degenerate KKT stationary point. It follows from the proof of Lemma D.8 that for any $\mathbf{w} \in \mathrm{C}_{\mathcal{K}}(\mathbf{x}^*)$, we have $\mathrm{J}_{\boldsymbol{\psi}}^{-1}(\mathbf{z}^*)\mathbf{w} \in \mathrm{C}_{\mathcal{B}}(\mathbf{z}^*)$. Hence, the conclusion is also trivial from the proof of item (2) of Lemma D.8.

□

D.6 Global Optimality Property of Optimization Problem H

Proposition 4.2. *Suppose problem* $\mathbf{P}$ *is a convex optimization. Then the following holds.*

1) If $\mathbf{x}^*$ *is a stationary point (thus a global optimum) in problem* $\mathbf{P}$*, then* $\mathbf{z}^* = \boldsymbol{\psi}^{-1}(\mathbf{x}^*)$ *is a global optimum in problem* $\mathbf{H}$ *and* $\langle \nabla h(\mathbf{z}^*), \mathbf{z} - \mathbf{z}^* \rangle \geq 0$*.*

2) If $\mathbf{z}^*$ *is a stationary point of problem* $\mathbf{H}$ *and LICQ holds at* $\mathbf{z}^*$*, then* $\mathbf{x}^*$ *is a stationary point of problem* $\mathbf{P}$ *(thus a global optimum). Hence,* $\mathbf{z}^*$ *is a global optimum.*

Proof. 1) Due to the global optimality of $\mathbf{x}^*$, we have

$$f(\mathbf{x}^*) \leq f(\mathbf{x}), \forall \mathbf{x} \in \mathcal{K}.$$

Hence with the homeomorphism $\boldsymbol{\psi}$ and $\mathbf{x} = \psi(\mathbf{z})$, we have

$$f(\boldsymbol{\psi}(\mathbf{z}^*)) \leq f(\boldsymbol{\psi}(\mathbf{z})), \forall \mathbf{z} \in \mathcal{B},$$

that is

$$h(\mathbf{z}^*) \leq h(\mathbf{z}), \forall \mathbf{z} \in \mathcal{B}.$$

Therefore, $\langle \nabla h(\mathbf{z}^*), \mathbf{z} - \mathbf{z}^* \rangle \geq 0$ referring to Prop. C.1.

2) By assumption, LICQ holds at $\mathbf{z}^*$. Hence by Proposition D.6, there exists a tuple $(\mathbf{z}^*, \nu^*)$ such that the KKT condition of problem $\mathbf{H}$ holds. It follows from Lemma D.7, there exists $\boldsymbol{\lambda}^*$ such that $(\mathbf{x}^*, \boldsymbol{\lambda}^*)$ also satisfy the KKT condition of problem $\mathbf{P}$. Then it follows from the convexity of $\mathbf{g} = (g_1, \cdots, g_m)$ that $\mathbf{x}^*$ is a global optimum of $\mathbf{P}$. From the above conclusion (item (1)), $\mathbf{z}^*$ is also a global optimum. □

E Convergence Analysis

In this section, we will provide omitted details in Section 4.2.

E.1 Proof of Theorem 1: Convex Case

Define for any $\mathbf{z}$, $\mathrm{T}(\mathbf{z}) := \mathrm{T}_{\mathcal{S}^{n-1}}(\mathbf{z}) = \{\mathbf{d} : \mathbf{d}^\top \mathbf{z} = 0\}$ and denote $\mathrm{Proj}_{\mathbf{z}}(\mathbf{d}) = \mathrm{Proj}_{\mathrm{T}(\mathbf{z})}(\mathbf{d}) = \mathbf{d} - (\mathbf{d}^\top \mathbf{z})\mathbf{z}$. We define

$$\mathrm{grad}\, h(\mathbf{z}) := \mathrm{Proj}_{\mathbf{z}}(\nabla h(\mathbf{z}))$$

and

$$\mathrm{hess}\, h(\mathbf{z}) = \mathrm{Proj}_{\mathbf{z}} \circ (\nabla^2 h(\mathbf{z}) - \nabla h(\mathbf{z})^\top \mathbf{z} \cdot I_n) \circ \mathrm{Proj}_{\mathbf{z}}.$$

Moreover, we define $\nabla_{\mathcal{B}} h(\mathbf{z}) = \nabla h(\mathbf{z})$ if $\mathbf{z} \in \mathrm{int}(\mathcal{B})$ and $\nabla_{\mathcal{B}} h(\mathbf{z}) = \mathrm{grad}\, h(\mathbf{z})$ if $\mathbf{z} \in \partial\mathcal{B}$. Similarly, define $\nabla^2_{\mathcal{B}} h(\mathbf{z}) = \nabla^2 h(\mathbf{z})$ if $\mathbf{z} \in \mathrm{int}(\mathcal{B})$ and $\nabla^2_{\mathcal{B}} h(\mathbf{z}) = \mathrm{hess}\, h(\mathbf{z})$ if $\mathbf{z} \in \partial\mathcal{B}$.

With these notations, the definition of the second-order KKT stationary point $\mathbf{z}^*$ for $\mathbf{H}$ is equivalent to the following (see e.g. [LMY23])

$$\nabla_{\mathcal{B}}\, h(\mathbf{z}^*) = 0, \min \mathrm{eig}(\nabla^2_{\mathcal{B}} h(\mathbf{z}^*)) \geq 0,$$

where $\min \mathrm{eig}(\cdot)$ represents the minimum eigenvalue. Moreover, $\mathbf{z}^*$ is non-degenerate if in addition $\min \mathrm{eig}(\nabla^2_{\mathcal{B}} h(\mathbf{z}^*)) > 0$.

We first give some help lemmas in the following.

Lemma E.1 (Local PL Condition)**.** *Suppose h is twice continuously differentiable.*

1) If $\mathbf{z}^* \in \partial\mathcal{B}$ is a non-degenerate minimizer for $\mathbf{H}$, there exists $\delta := \delta(\mathbf{z}^*)$ and $\tau := \tau(\mathbf{z}^*)$ such that PL inequality holds locally, i.e.,

$$h(\mathbf{z}) - h(\mathbf{z}^*) \leq \frac{1}{2\tau} \|\mathrm{grad}\, h(\mathbf{z})\|^2$$

for any $\mathbf{z} \in \mathcal{B}(\mathbf{z}^*, \delta) \cap \partial\mathcal{B}$.

2) If $\mathbf{z}^* \in \mathrm{int}(\mathcal{B})$ is a non-degenerate minimizer for $\mathbf{H}$, there exists $\delta := \delta(\mathbf{z}^*)$ and $\tau := \tau(\mathbf{z}^*)$ such that PL inequality holds at the ball $\mathcal{B}(\mathbf{z}^*, \delta)$, i.e.,

$$h(\mathbf{z}) - h(\mathbf{z}^*) \leq \frac{1}{2\tau} \|\nabla h(\mathbf{z})\|^2$$

for any $\mathbf{z} \in \mathcal{B}(\mathbf{z}^*, \delta)$.

Proof. For item (1), $\mathbf{z}^*$ is on the boundary of $\mathcal{B}$. We can consider $h : \partial\mathcal{B} = \mathcal{S}_{n-1} \to \mathbb{R}$. Because $\mathbf{z}^*$ is a non-degenerate minimizer, by continuity there exists a ball $\mathcal{B}(\mathbf{z}^*, \delta)$ and upon $\mathbf{z} \in \mathcal{B}(\mathbf{z}^*, \delta) \cap \partial\mathcal{B}$ we have $\mathrm{hess}\, h(\mathbf{z})$ is positive definite, i.e., h is τ-strongly convex over the ball. Hence PL inequality holds over $\mathbf{z} \in \mathcal{B}(\mathbf{z}^*, \delta) \cap \partial\mathcal{B}$ (refer to e.g. Lemma 11.28 [Bou23]), i.e.,

$$h(\mathbf{z}) - h(\mathbf{z}^*) \leq \frac{1}{2\tau} \|\mathrm{grad}\, h(\mathbf{z})\|^2$$

for any $\mathbf{z} \in \mathcal{B}(\mathbf{z}^*, \delta) \cap \partial\mathcal{B}$.

It's easy to show item (2). Actually, by continuity there exists a ball $\mathcal{B}(\mathbf{z}^*, \delta)$ upon which $\nabla^2 h(\mathbf{z})$ is positive definite, i.e., h is τ-strongly convex over the ball. Hence PL inequality holds over the ball (refer to e.g. [KNS16]). □

Definition E.2 (Approximate Stationary Point)**.** A point $\mathbf{x}^*$ is called ϵ-stationary point for problem $\min_{\mathbf{x}\in\mathcal{K}} f(\mathbf{x})$ with convex set $\mathcal{K}$, if the gradient norm mapping

$$G(\mathbf{x}) := G_{1/\alpha} = \frac{1}{\alpha}[\mathbf{x} - \Pi_{\mathcal{K}}(\mathbf{x} - \alpha\nabla f(\mathbf{x}))]$$

satisfies $\|G(\mathbf{x})\| \leq \epsilon$ for proper $\alpha > 0$.

Lemma E.3. *Suppose* $\mathbf{z}$ *and* $\mathbf{z}^+ = \Pi_{\mathcal{B}}(\mathbf{z} - \alpha\nabla h(\mathbf{z}))$ *for some* $\alpha > 0$ *are both on the boundary of* $\mathcal{B}$*. If the gradient norm mapping*

$$G(\mathbf{z}) := G_{1/\alpha}(\mathbf{z}) = \frac{1}{\alpha}[\mathbf{z} - \mathbf{z}^+]$$

satisfies $\|G(\mathbf{z})\| \leq \epsilon$*, then* $\|\text{grad } h(\mathbf{z})\| \leq \mathcal{O}(\epsilon)$*.*

Proof. From properties of the orthogonal projection C.2, we have

$$\mathbf{z} - \alpha\nabla h(\mathbf{z}) - \mathbf{z}^+ = \mu\mathbf{z}^+$$

for some $\mu \geq 0$, or

$$\nabla h(\mathbf{z}) = G(\mathbf{z}) - \alpha\mu G(\mathbf{z}) + \frac{\mu}{\alpha}\mathbf{z} = c_1 G(\mathbf{z}) + c_2\mathbf{z}$$

where $c_1 = 1 - \alpha\mu > 0$ (for small enough α) and $c_2 = \mu/\alpha$. We have then

$$\begin{aligned}\text{grad } h(\mathbf{z}) &= \nabla h(\mathbf{z}) - \langle\nabla h(\mathbf{z}), \mathbf{z}\rangle\mathbf{z} \\ &= c_1 G(\mathbf{z}) - \langle c_1 G(\mathbf{z}), \mathbf{z}\rangle\mathbf{z}.\end{aligned}$$

Thus,

$$\|\text{grad } h(\mathbf{z})\| \leq 2c_1\|G(\mathbf{z})\| \leq \mathcal{O}(\epsilon).$$

□

Lemma E.4. *Suppose* h *is twice continuously differentiable and* $\mathbf{z}^*$ *is a unique minimizer (stationary point) for* $\mathbf{H}$*. Then for any* $\delta > 0$*, there exists an* $\epsilon_\delta > 0$ *such that* $\mathbf{z}$ *is an* ϵ_δ *stationary point can imply* $\|\mathbf{z} - \mathbf{z}^*\| < \delta$*.*

Proof. We suppose there does not exist such $\epsilon_\delta > 0$. Then for any $\epsilon := 1/k$, we can find an ϵ stationary point $\mathbf{z}_k$ such that $\|\mathbf{z}_k - \mathbf{z}^*\| \geq \delta$. As $\mathcal{B}$ is compact, we may assume $\mathbf{z}_k \to \bar{\mathbf{z}} \in \mathcal{B}$. Then $\bar{\mathbf{z}}$ is a stationary point by continuity of the gradient norm mapping:

$$\|G(\bar{\mathbf{z}})\| = \|G(\lim_{k\to\infty}\mathbf{z}_k)\| = \lim_{k\to\infty}\|G(\mathbf{z}_k)\| \leq \lim_{k\to\infty}\epsilon_k = 0.$$

This contradicts $\|\mathbf{z}_k - \mathbf{z}^*\| \geq \delta$. □

Theorem 1. *Suppose the strict complementary slackness condition holds for both problem* $\mathbf{P}$ *and* $\mathbf{H}$ *and we suppose the problem* $\mathbf{P}$ *is convex and has a non-degenerate minimizer* $\mathbf{x}^*$*. Let* $\{\mathbf{z}_k\}$ *be the sequence generated by Hom-PGD with step-size* $\alpha \in (0, \frac{2}{L_h}]$*. For sufficient small* $\epsilon > 0$*,* $\{\mathbf{z}_k\}_{k=1}^K$ *with* $K = \mathcal{O}(L_h/\epsilon)$ *contains* $\mathbf{z}'$ *such that*

$$h(\mathbf{z}') - h^\star \leq \epsilon.$$

Proof. From Lemma D.10, $\mathbf{z}^\star$ is also a non-degenerate stationary point.

1) We first assume $\mathbf{z}^* \in \partial\mathcal{B}$.

By Lemma E.1, there exists $\tau^* > 0$ and δ^* such that for any $\mathbf{z} \in \mathcal{B}(\mathbf{z}^\star, \delta^\star) \cap \partial\mathcal{B}$,

$$h(\mathbf{z}) - h(\mathbf{z}^*) \leq \frac{1}{2\tau^*}\|\text{grad } h(\mathbf{z})\|^2.$$

Moreover, we assume this δ^* is small enough to satisfy the condition that for any $\mathbf{z} \in \mathcal{B}(\mathbf{z}^\star, \delta^\star)$, $\mathbf{z} - \alpha\nabla h(\mathbf{z})$ is outside the ball, which implies $\mathbf{z}^+ = \Pi_{\mathcal{B}}(\mathbf{z} - \alpha h(\mathbf{z})) \in \partial\mathcal{B}$. This δ^* exists by the continuity of ∇h and note $\nabla h(\mathbf{z}^*) = \mu\mathbf{z}^*$ for some $\mu^* > 0$ (where '$>$' is from strict complementary slackness condition). Hence if PGD can find an approximate stationary point $\mathbf{z}' \in \mathcal{B}(\mathbf{z}^\star, \delta^\star)$, we can assume $\mathbf{z}' \in \partial\mathcal{B}$.

From Lemma E.4, there exists $\epsilon_{\delta^*} > 0$ such that if $\mathbf{z}$ is an ϵ_{δ^*} stationary point then $\mathbf{z} \in \mathcal{B}(\mathbf{z}^\star, \delta^*)$.

So, suppose $\epsilon > 0$ is small enough that $c\sqrt{\epsilon} < \epsilon_{\delta^*}$ where $c > 0$ is a fixed constant determined later. By Theorem 3, PGD finds a $c\sqrt{\epsilon}$-stationary point $\mathbf{z}' \in \partial\mathcal{B}$, within $K = \mathcal{O}\left(L_h/(\sqrt{\epsilon})^2\right) = \mathcal{O}(1/\epsilon)$ iterations, i.e., $\|G(\mathbf{z}')\| \leq c\sqrt{\epsilon}$. From Lemma E.3, we can choose proper c such that $\|\mathrm{grad}\, h(\mathbf{z}')\| \leq \sqrt{2\tau^*\epsilon}$.

As $c\sqrt{\epsilon} < \epsilon_{\delta^*}$ we know $\mathbf{z}'$ is also an ϵ_{δ^*}-stationary point within $B(\mathbf{z}^\star, \delta^*)$.

$$h(\mathbf{z}') - h(\mathbf{z}^\star) \leq \frac{1}{2\tau^*}\|\mathrm{grad}\, h(\mathbf{z}')\|^2 \leq \frac{1}{2\tau^*}(\sqrt{2\tau^*\epsilon})^2 \leq \epsilon.$$

2) The case when $\mathbf{z}^* \in \mathrm{int}(\mathcal{B})$ is similar.

By Lemma E.1, there exists $\tau^* > 0$ and $\delta^* > 0$ such that for any $\mathbf{z} \in \mathcal{B}(\mathbf{z}^\star, \delta^\star)$,

$$h(\mathbf{z}) - h(\mathbf{z}^*) \leq \frac{1}{2\tau^*}\|\nabla h(\mathbf{z})\|^2.$$

From Lemma E.4, there exists $\epsilon_{\delta^*} > 0$ such that if $\mathbf{z}$ is an ϵ_{δ^*} stationary point then $\mathbf{z} \in \mathcal{B}(\mathbf{z}^\star, \delta)$.

Similarly, suppose $\epsilon > 0$ is small enough that $c\sqrt{\epsilon} < \epsilon_{\delta^*}$ where $c > 0$ is a fixed constant determined later. By Theorem 3, PGD finds a $c\sqrt{\epsilon}$-stationary point $\mathbf{z}'$, within $K = \mathcal{O}\left(1/(\sqrt{\epsilon})^2\right) = \mathcal{O}(1/\epsilon)$ iterations, i.e., $\|G(\mathbf{z}')\| \leq c\sqrt{\epsilon}$. Because $\mathbf{z}^*$ is in the interior of $\mathcal{B}$, hence when $\mathbf{z}'$ is close enough to $\mathbf{z}^*$ (δ is sufficient small), the gradient norm mapping $G(\mathbf{z}')$ is exactly $\nabla h(\mathbf{z}')$. So we can choose proper c such that $\|\nabla h(\mathbf{z}')\| \leq \sqrt{2\tau^*\epsilon}$.

As $c\sqrt{\epsilon} < \epsilon_{\delta^*}$ we know $\mathbf{z}'$ is also an ϵ_{δ^*} stationary point within $B(\mathbf{z}^\star, \delta^*)$. Then we have

$$h(\mathbf{z}') - h(\mathbf{z}^\star) \leq \frac{1}{2\tau^*}\|\mathrm{grad}\, h(\mathbf{z}')\|^2 \leq \frac{1}{2\tau^*}(\sqrt{2\tau^*\epsilon})^2 \leq \epsilon.$$

□

E.2 Proof of Theorem 2: Strongly Convex Case

In this section, we show Theorem 2. Actually, Theorem 2 is a corollary of the following theorem.

Theorem 4. *Suppose f is μ_f-strongly convex and $\mathcal{K}$ is a convex set. Then the updating sequence $\{\mathbf{z}_k\}$ by Hom-PGD algorithm with a constant step size $\alpha \in (0, \frac{2}{L_h}]$, converges to a global optimum point $\mathbf{z}^\star$ linearly, i.e.,*

$$h(\mathbf{z}_K) - h(\mathbf{z}^*) \leq \sigma^K\left(h(\mathbf{z}_0) - h(\mathbf{z}^*)\right), \mathrm{dist}(\mathbf{z}_K, \mathcal{B}^*) \leq \frac{2}{\mu_f\kappa_1}\sigma^K\left(h(\mathbf{z}_0) - h(\mathbf{z}^*)\right)$$

where $\sigma = 1 - \frac{\mu_f\kappa_1}{L_h} \in (0, 1)$.

Proof. By strong convexity of f, and convexity of $\mathcal{K}$, one can show that f satisfy proximal PL condition over $\mathcal{K}$ (see e.g. Appendix G [KNS16]), i.e.,

$$\mathcal{D}_{\delta_\mathcal{K}}(\mathbf{x}, \mu_f; f) \geq 2\mu_f(f(\mathbf{x}) - f(\mathbf{x}^*))$$

where

$$\mathcal{D}_{\delta_\mathcal{K}}(\mathbf{x}, \lambda; f) = -2\lambda\min_{y\in\mathcal{K}}\left[\langle\nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x}\rangle + \frac{\lambda}{2}\|\mathbf{y} - \mathbf{x}\|^2\right].$$

Next, we show that $h = f \circ \boldsymbol{\psi}$ also satisfies the proximal-PL condition over $\mathcal{B}$. We derive with $\mathbf{x} = \boldsymbol{\psi}(\mathbf{z}), \mathbf{y} = \boldsymbol{\psi}(\mathbf{u})$ as follows

$$
\begin{aligned}
&\langle \nabla h(\mathbf{z}), \mathbf{u} - \mathbf{z} \rangle + \frac{L_{f,0} L_{\boldsymbol{\psi}}}{2} \|\mathbf{u} - \mathbf{z}\|^2 \\
=& \left\langle \mathrm{J}_{\boldsymbol{\psi}}(\mathbf{z})^\top \nabla f(\boldsymbol{\psi}(\mathbf{z})), \mathbf{u} - \mathbf{z} \right\rangle + \frac{L_{f,0} L_{\boldsymbol{\psi}}}{2} \|\mathbf{u} - \mathbf{z}\|^2 \\
=& \langle \nabla f(\boldsymbol{\psi}(\mathbf{z})), \mathrm{J}_{\boldsymbol{\psi}}(\mathbf{z})(\mathbf{u} - \mathbf{z}) \rangle + \frac{L_{f,0} L_{\boldsymbol{\psi}}}{2} \|\mathbf{u} - \mathbf{z}\|^2 \\
=& \langle \nabla f(\boldsymbol{\psi}(\mathbf{z})), -\boldsymbol{\psi}(\mathbf{u}) + \boldsymbol{\psi}(\mathbf{z}) + \mathrm{J}_{\boldsymbol{\psi}}(\mathbf{z})(\mathbf{u} - \mathbf{z}) \rangle + \langle \nabla f(\boldsymbol{\psi}(\mathbf{z})), \boldsymbol{\psi}(\mathbf{u}) - \boldsymbol{\psi}(\mathbf{z}) \rangle + \frac{L_{f,0} L_{\boldsymbol{\psi}}}{2} \|\mathbf{u} - \mathbf{z}\|^2 \\
\leq& \frac{L_{f,0} L_{\boldsymbol{\psi}}}{2} \|\mathbf{u} - \mathbf{z}\|^2 + \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle + \frac{L_{f,0} L_{\boldsymbol{\psi}}}{2} \|\mathbf{u} - \mathbf{z}\|^2 \\
=& \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle + \frac{2 L_{f,0} L_{\boldsymbol{\psi}}}{2} \left\| \boldsymbol{\psi}^{-1}(\mathbf{y}) - \boldsymbol{\psi}^{-1}(\mathbf{x}) \right\|^2 \\
\leq& \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle + \frac{2 L_{f,0} L_{\boldsymbol{\psi}}}{2 \kappa_1} \|\mathbf{y} - \mathbf{x}\|^2,
\end{aligned}
$$

where
the 2-nd line is from the chain rule of the gradient,
the 3-rd line is from property of inner product, i.e., for vector $\mathbf{a}, \mathbf{b}$ and matrix A, $\langle A^\top \mathbf{a}, \mathbf{b} \rangle = \langle \mathbf{a}, A\mathbf{b} \rangle$,
the 4-th line is based on simple caculation,
the 5-th line is from Lemma C.3, and the transformation $\mathbf{z} = \boldsymbol{\psi}(\mathbf{x}), \mathbf{y} = \boldsymbol{\psi}(\mathbf{u})$,
the 6-th line is from the inverse transformation $\boldsymbol{\psi}^{-1}$,
and the last line is from the bi-Lipschitz property of $\boldsymbol{\psi}$.

Briefly, we get

$$
\langle \nabla h(\mathbf{z}), \mathbf{u} - \mathbf{z} \rangle + \frac{L_{f,0} L_{\boldsymbol{\psi}}}{2} \|\mathbf{u} - \mathbf{z}\|^2 \leq \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle + \frac{2 L_{f,0} L_{\boldsymbol{\psi}}}{2 \kappa_1} \|\mathbf{y} - \mathbf{x}\|^2 \tag{14}
$$

Next, we assume $c_1 := \frac{2 L_{f,0} L_{\boldsymbol{\psi}}}{\kappa_1} \geq \mu_f$ without loss of generality. This is because if $c_1 < \mu_f$, we have

$$
\langle \nabla h(\mathbf{z}), \mathbf{u} - \mathbf{z} \rangle + \frac{L_{f,0} L_{\boldsymbol{\psi}}}{2} \|\mathbf{u} - \mathbf{z}\|^2 \leq \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle + \frac{\mu_f}{2} \|\mathbf{y} - \mathbf{x}\|^2
$$

or

$$
\begin{aligned}
-2 \min_{\mathbf{u} \in \mathcal{B}} \left\{ \langle \nabla h(\mathbf{z}), \mathbf{u} - \mathbf{z} \rangle + \frac{L_{f,0} L_{\boldsymbol{\psi}}}{2} \|\mathbf{u} - \mathbf{z}\|^2 \right\} &\geq -2 \min_{\mathbf{y} \in \mathcal{K}} \left\{ \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle + \frac{\mu_f}{2} \|\mathbf{y} - \mathbf{x}\|^2 \right\} \\
&\geq 2 \left(f(\mathbf{x}) - f^* \right) = 2 \left(h(\mathbf{z}) - h^* \right).
\end{aligned}
$$

That is, if $c_1 < \mu_f$, we directly get the conclusion that h satisfy proximal PL condition over $\mathcal{B}$:

$$
\mathcal{D}_{\delta_{\mathcal{B}}}(\mathbf{z}, L_{f,0} L_{\boldsymbol{\psi}}; h) \geq 2 L_{f,0} L_{\boldsymbol{\psi}} \left(h\left(\mathbf{z}\right) - h\left(\mathbf{z}^*\right) \right).
$$

In the following, we assume $c_1 \geq \mu_f$. By Lemma 1 in [KNS16], for any convex set $\mathcal{K}$ and differentiable function f, $\mathcal{D}_{\delta_{\mathcal{K}}}(\mathbf{x}, \mu; f)$ is monotone increasing in μ, i.e., $\mathcal{D}_{\delta_{\mathcal{K}}}(\mathbf{x}, \mu_1; f) \geq \mathcal{D}_{\delta_{\mathcal{K}}}(\mathbf{x}, \mu_2; f)$ given $\mu_2 \geq \mu_1 > 0$. Hence, we have

$$
\mathcal{D}_{\delta_{\mathcal{K}}}(\mathbf{x}, c_1; f) \geq \mathcal{D}_{\delta_{\mathcal{K}}}(\mathbf{x}, \mu_f; f) \geq 2 \mu_f (f(\mathbf{x}) - f^*).
$$

Then it following from Eq. (14) that

$$
\frac{1}{L_{f,0} L_{\boldsymbol{\psi}}} \mathcal{D}_{\delta_{\mathcal{B}}}(\mathbf{z}, L_{f,0} L_{\boldsymbol{\psi}}; h) \geq \frac{1}{c_1} \mathcal{D}_{\delta_{\mathcal{K}}}(\mathbf{x}, c_1; f) \geq \frac{1}{c_1} \mathcal{D}_{\delta_{\mathcal{K}}}(\mathbf{x}, \mu_f; f) \geq 2 \frac{\mu_f}{c_1} (f(\mathbf{x}) - f^*).
$$

Hence we have h satisfy proximal PL condition over $\mathcal{B}$:

$$
\mathcal{D}_{\delta_{\mathcal{B}}}(\mathbf{z}, L_{f,0} L_{\boldsymbol{\psi}}; h) \geq 2 L_{f,0} L_{\boldsymbol{\psi}} \frac{\mu_f}{c_1} (h(\mathbf{z}) - h(\mathbf{z}^*)) = 2 \frac{\mu_f \kappa_1}{2} (h(\mathbf{z}) - h(\mathbf{z}^*)). \tag{15}
$$

Finally, we show the linear convergence rate of the projected gradient descent algorithm. We derive

$$\begin{aligned}\mathbf{z}_{k+1} &= \Pi_{\mathcal{B}}\left(\mathbf{z}_k - \alpha\nabla h\left(\mathbf{z}_k\right)\right)\\ &= \arg\min_{\mathbf{u}\in\mathcal{B}} \left\|\mathbf{u} - \left(\mathbf{z}_k - \alpha\nabla h\left(\mathbf{z}_k\right)\right)\right\|^2\\ &= \arg\min_{\mathbf{u}\in\mathcal{B}} \left\{\alpha\left\langle\nabla h\left(\mathbf{z}_k\right), \mathbf{u}-\mathbf{z}_k\right\rangle + \left\|\mathbf{u}-\mathbf{z}_k\right\|^2\right\}\\ &= \arg\min_{\mathbf{u}\in\mathcal{B}} \left\{\left\langle\nabla h\left(\mathbf{z}_k\right), \mathbf{u}-\mathbf{z}_k\right\rangle + \frac{1}{\alpha}\left\|\mathbf{u}-\mathbf{z}_k\right\|^2\right\},\end{aligned}$$

where the 2-nd line is from the definition of orthogonal projection Π, the 3-rd and last line is from simple calculation.

This implies

$$-\frac{\alpha}{2}\mathcal{D}_{\delta_{\mathcal{B}}}(\mathbf{z}, \frac{2}{\alpha}; h) = \langle\nabla h(\mathbf{z}_k), \mathbf{z}_{k+1}-\mathbf{z}_k\rangle + \frac{1}{\alpha}\|\mathbf{z}_{k+1}-\mathbf{z}_k\|^2 \geq \langle h(\mathbf{z}_k), \mathbf{z}_{k+1}-\mathbf{z}_k\rangle + \frac{L_h}{2}\|\mathbf{z}_{k+1}-\mathbf{z}_k\|^2 \tag{16}$$

where the last inequality is from the selection of stepsize $\alpha \in (0, \frac{2}{L_h}]$ and recall $L_h = \kappa_2^2 L_{f,1} + L_{\psi} L_{f,0}$ from Prop. D.1.

Then the iterative of PGD satisfies,

$$\begin{aligned}h(\mathbf{z}_{k+1}) &\leq h(\mathbf{z}_k) + \langle\nabla h(\mathbf{z}_k), \mathbf{z}_{k+1}-\mathbf{z}_k\rangle + \frac{L_h}{2}\|\mathbf{z}_{k+1}-\mathbf{z}_k\|^2\\ &\leq h(\mathbf{z}_k) - \frac{\alpha}{2}\mathcal{D}_{\delta_{\mathcal{B}}}(\mathbf{z}_k, \frac{2}{\alpha}; h)\\ &\leq h(\mathbf{z}_k) - \frac{\alpha}{2}\mathcal{D}_{\delta_{\mathcal{B}}}(\mathbf{z}_k, L_{f,0}L_{\psi}; h)\\ &\leq h(\mathbf{z}_k) - \frac{\alpha\mu_f\kappa_1}{2}(h(\mathbf{z}_k) - h(\mathbf{z}^*)),\end{aligned}$$

where
the 1-st line is from the L_h-smoothness of h,
the 2-nd line is from Eq. (16),
the 3-rd line is from monotone increasing property of $\mathcal{D}_{\delta_{\mathcal{B}}}(\mathbf{z}, \cdot; h)$ and the choice of $\alpha \in (0, \frac{2}{L_h}]$,
and the last line is from the proximal PL condition Eq. (15) of h.

Take $\alpha = \frac{2}{L_h}$,

$$h(\mathbf{z}_{k+1}) - h(\mathbf{z}^*) \leq (1 - \frac{\mu_f\kappa_1}{L_h})(h(\mathbf{z}_k) - h(\mathbf{z}^*)).$$

Then we rewrite

$$h(\mathbf{z}_K) - h(\mathbf{z}^*) \leq \sigma^K(h(\mathbf{z}_0) - h(\mathbf{z}^*)), \quad \text{where} \quad \sigma = 1 - \frac{\mu_f\kappa_1}{L_h}.$$

From quadratic growth condition D.1, we have

$$\operatorname{dist}(\mathbf{z}_K, \mathcal{B}^*) \leq \frac{2}{\mu_f\kappa_1}(h(\mathbf{z}_K) - h(\mathbf{z}^*)) \leq \frac{2}{\mu_f\kappa_1}\sigma^K(h(\mathbf{z}_0) - h(\mathbf{z}^*)).$$

Let $\sigma^K(h(\mathbf{z}_0) - h(\mathbf{z}^*)) \leq \epsilon$. We have

$$K = \mathcal{O}(\frac{\log 1/\epsilon}{\log 1/\sigma}) = \mathcal{O}\left((1-\sigma)^{-1}\log 1/\epsilon\right)$$

where the last '=' holds when σ is closed to 1. □

Remark E.5. It follows directly from the proof that the assumption of strong convexity can be relaxed. It suffices for the objective to satisfy the proximal-PL condition over the *convex* constraint set $\mathcal{K}$ in order to achieve the convergence rate of $\mathcal{O}(\log 1/\epsilon)$. Moreover, there exist many alternative and equivalent assumptions for the proximal-PL condition. We list some commonly used equivalent assumptions below, where the proof of the equivalence can be referred to, e.g, [KNS16].

- Proximal error bounds (Proximal-EB): There exists $c > 0$ such that

$$\|\mathbf{x} - \mathbf{x}_{\mathcal{K}^*}\| \leq c \left\| \mathbf{x} - \operatorname{prox}_{\frac{1}{L_f}\delta_{\mathcal{K}}} \left(\mathbf{x} - \frac{1}{L_f} \nabla f(\mathbf{x}) \right) \right\|$$

where $\mathbf{x}_{\mathcal{K}^*}$ is the orthogonal projection of $\mathbf{x}$ onto the optimal solution set $\mathcal{K}^*$ and $\operatorname{prox}_g(\mathbf{x}) = \arg\min_{\mathbf{u}} \left\{ g(\mathbf{u}) + \frac{1}{2} \|\mathbf{u} - \mathbf{x}\|^2 \right\}$. Moreover, note that for $\lambda > 0$, we have $\operatorname{prox}_{\lambda\delta_{\mathcal{K}}} = \Pi_{\mathcal{K}}$.

- Kurdyka-Łojasiewicz (KL) condition: There exists $\mu_f > 0$ such that

$$\min_{s \in \partial F(\mathbf{x})} \|s\|^2 \geq 2\mu_f \left(F(\mathbf{x}) - F^* \right)$$

where $F(\mathbf{x}) = f(\mathbf{x}) + \delta_{\mathcal{K}}(\mathbf{x})$ and $\partial F(\mathbf{x})$ is the Frechet subdifferential [RW09]. In this case,

$$\partial F(\mathbf{x}) = \{ \nabla f(\mathbf{x}) + \boldsymbol{\xi} \mid \boldsymbol{\xi} \in \partial \delta_{\mathcal{K}}(\mathbf{x}) \}$$

where $\partial \delta_{\mathcal{K}}(\mathbf{x})$ can be simlified as $\mathrm{N}_{\mathcal{K}}(\mathbf{x})$, with $\mathrm{N}_{\mathcal{K}}(\mathbf{x}) = \{ \mathbf{y} : \langle \mathbf{y}, \mathbf{z} - \mathbf{x} \rangle \leq 0, \forall \mathbf{z} \in \mathcal{K} \}$ for $\mathbf{x} \in \mathcal{K}$ and $\mathrm{N}_{\mathcal{K}}(\mathbf{x}) = \emptyset$ for $\mathbf{x} \notin \mathcal{K}$. Note that KL condition can imply PL condition ($\|\nabla f(\mathbf{x})\|^2 \geq 2\mu_f(f(\mathbf{x}) - f^*)$) but the converse does not hold in general.

F Experiments Setting

F.1 Problem Formulations and Instance Generation

Optimization over polyhedron: We first consider a two-dimensional optimization over a convex polyhedron to illustrate the effectiveness of our methods. The problem is defined as:

$$\min_{\mathbf{L} \leq \mathbf{x} \leq \mathbf{U}} \quad \sum_{i=1}^{2} w_i (x_i - 1)^2 \qquad \text{s.t.} \quad \mathbf{a}_i^\top \mathbf{x} \leq b_i, \quad i = 1, \ldots, n_{\text{lin}} \tag{17}$$

where $\mathbf{x} \in \mathbb{R}^2$ is the decision variable, w_i is the positive coefficients, and $\mathbf{L}, \mathbf{U} \in \mathbb{R}^2$ represent the lower and upper bounds on the variables. $\mathbf{a}_i \in \mathbb{R}^2$ and $b_i \in \mathbb{R}$ represents the coefficients in n_{lin} linear constraints. The homeomorphic counterparts are derived by a closed-form gauge mapping as discussed in B.4.

Optimization over star-shaped set: We then consider a two-dimensional optimization over a non-convex star-shaped set to illustrate the effectiveness of our methods. The problem and its homeomorphic counterpart are defined as:

$$\begin{array}{lcl} \min\limits_{x} & \sum_{i=1}^{2} w_i (x_i - 1)^2 & \overset{\boldsymbol{\psi}^{-1}}{\Longrightarrow} \\ \text{s.t.} & \|\mathbf{x}\| \leq \Gamma_{\alpha,n}(\mathbf{x}) & \overset{\boldsymbol{\psi}}{\Longleftarrow} \end{array} \quad \begin{array}{lcl} \min\limits_{z} & \sum_{i=1}^{2} w_i (z_i \cdot \Gamma_{\alpha,n}(\mathbf{z}) - 1)^2 \\ \text{s.t.} & \|\mathbf{z}\| \leq 1 \end{array} \tag{18}$$

$$\boldsymbol{\psi}(\mathbf{z}) = [z_1 \cdot \Gamma_{\alpha,n}(\mathbf{z}), z_2 \cdot \Gamma_{\alpha,n}(\mathbf{z})], \quad \boldsymbol{\psi}^{-1}(\mathbf{x}) = [x_1 / \Gamma_{\alpha,n}(\mathbf{x}), x_2 / \Gamma_{\alpha,n}(\mathbf{x})], \tag{19}$$

where $\mathbf{z}, \mathbf{x} \in \mathbb{R}^2$ and $\Gamma_{\alpha,n}$ is a non-linear function with parameters $\alpha > 0$ and $n \in \mathbb{Z}^+$, defined as $\Gamma_{\alpha,n}([x_1, x_2]) := 1 + \alpha \sin(n \arctan(x_2/x_1))$. Under the homeomorphic mapping $\boldsymbol{\psi}$, the non-convex-constrained optimization problem can be transformed into a ball-constrained non-convex optimization. We then compared different iterative algorithms over this problem.

Second-order cone programming: We then consider convex second-order cone programming (SOCP), which encompasses linear programming (LP), quadratic programming (QP), and convex quadratically constrained quadratic programming (QCQP) problems. This formulation has wide applications in portfolio optimization and optimal power flow problems [Low14a, Low14b].

$$\min_{\mathbf{L} \leq \mathbf{x} \leq \mathbf{U}} \quad \frac{1}{2} \mathbf{x}^\top \mathbf{Q} \mathbf{x} + \mathbf{p}^\top \mathbf{x} \qquad \text{s.t.} \quad \|\mathbf{G}_i \mathbf{x} + \mathbf{h}_i\| \leq \mathbf{c}_i^\top \mathbf{x} + d_i, \quad i = 1, \ldots, n_{\text{soc}} \tag{20}$$

where $\mathbf{x} \in \mathbb{R}^n$ is the decision variable, $\mathbf{Q} \in \mathbb{R}^{n \times n}$ is a symmetric positive semidefinite matrix, $\mathbf{p} \in \mathbb{R}^n$ is a vector of linear cost coefficients, and $\mathbf{L}, \mathbf{U} \in \mathbb{R}^n$ represent the lower and upper bounds on the variables. For the second-order cone constraints, $\mathbf{G}_i \in \mathbb{R}^{m_i \times n}$ and $\mathbf{h}_i \in \mathbb{R}^{m_i}$ define the affine function inside the norm, while $\mathbf{c}_i \in \mathbb{R}^n$ and $d_i \in \mathbb{R}$ define the affine function on the right-hand side.

The parameter n_{soc} represents the number of second-order cone constraints in the problem. The total number of constraints also includes the upper/lower bound on the decision variables.

Max-cut semidefinite programming: We consider an important class of SDP in the max-cut problem. Given a graph $G = \{\mathcal{N}, \mathcal{E}\}$ with node set $i \in \mathcal{N}$ and edge set $(i, j) \in \mathcal{E}$, the max-cut SDP problem is formulated as:

$$\max_{-1 \leq \mathbf{X} \leq 1} \quad \sum_{(i,j)\in\mathcal{E}} (1 - x_{ij})/2 \tag{21}$$

$$\text{s.t.} \quad x_{ii} = 1, \quad i = 1, \cdots, n \tag{22}$$

$$\mathbf{X} \succeq \mathbf{0}, \tag{23}$$

where $\mathbf{X} \succeq \mathbf{0}$ indicates that $\mathbf{X}$ is positive semidefinite. Define the upper triangle off-diagonal elements in $\mathbf{X}$ as $\mathbf{y} \in \mathbb{R}^{(N^2-N)/2}$, then the SDP can be equivalently reformulated in Linear Matrix Inequality (LMI)-based form as:

$$\max_{-1 \leq \mathbf{y} \leq 1} \quad \sum_{k=(i,j)\in\mathcal{E}} (1 - y_k)/2 \tag{24}$$

$$\text{s.t.} \quad \mathbf{I} + \sum_k y_k \cdot \mathbf{A}_k \succeq \mathbf{0}, \tag{25}$$

where $\mathbf{A}_k$ is a symmetric matrix with zeros on the diagonal and with a 1 in the (i, j) and (j, i) positions corresponding to the k-th off-diagonal entry, and zeros elsewhere. Given such an LMI-based formulation, we can construct the homeomorphic counterpart based on a closed gauge mapping as discussed in B.4. Note that a "central" interior point for such a PSD cone is naturally the zero vector $\mathbf{y}^\circ = \mathbf{0}$.

We also consider solving the well-known Burer-Monteiro (BM) factorization-based semidefinite program via augmented Lagrangian methods [BM03]. Let $\mathbf{X} = \mathbf{V}\mathbf{V}^\mathbf{T}$, where $\mathbf{V} \in \mathbb{R}^{N\times r}$ and r is the selected rank, then we have the following low-rank SDP:

$$\max_{\mathbf{X}=\mathbf{V}\mathbf{V}^\mathbf{T}} \quad \sum_{(i,j)\in\mathcal{E}} (1 - x_{ij})/2 \tag{26}$$

$$\text{s.t.} \quad x_{ii} = 1, \quad i = 1, \cdots, n \tag{27}$$

$$-\mathbf{1} \leq \mathbf{X} \leq \mathbf{1} \tag{28}$$

In our experiments, we consider both log-rank ($r = \log(N)$) and Barvinok-Pataki (bp)-rank ($r = \sqrt{2N}$) factorization-based SDP methods [Bar95, Pat98, BVB20].

F.2 Baseline Algorithms and Hyper-Parameters

We implement the baselines as follows:

- **PGD**:

$$\mathbf{x}_{k+1} = \Pi_{\mathcal{K}}(\mathbf{x}_k - \alpha_k \nabla f(\mathbf{x}_k)) \tag{29}$$

  where $\Pi_{\mathcal{K}}$ denotes the orthogonal projection onto the feasible set $\mathcal{K}$, $\alpha_k > 0$ is the step size at iteration k, and $\nabla f(\mathbf{x}_k)$ is the gradient of the objective function at point $\mathbf{x}_k$. The quadratic projection problem is solved via MOSEK for convex problems and by ALM for non-convex problems.

- **FW**:

$$\mathbf{s}_k = \arg\min_{\mathbf{s}\in\mathcal{K}} \langle \nabla f(\mathbf{x}_k), \mathbf{s} \rangle, \tag{30}$$

$$\mathbf{x}_{k+1} = (1 - \alpha)\mathbf{x}_k + \alpha_k \mathbf{s}_k, \tag{31}$$

  where $\alpha_k \in [0, 1]$ is the step size at iteration k. The linear minimization problem is solved via MOSEK for convex problems and by ALM for non-convex problems.

- **ALM**:

$$\mathbf{x}_{k+1} = \arg\min_{\mathbf{x}} \{ f(\mathbf{x}) + \boldsymbol{\lambda}_k^T \mathbf{g}(\mathbf{x}) + \rho_k \cdot \mathbf{1}^T [\mathbf{g}(\mathbf{x})]_+^2 \}, \tag{32}$$

$$\boldsymbol{\lambda}_{k+1} = [\boldsymbol{\lambda}_k + \rho_k \cdot \mathbf{g}(\mathbf{x}_{k+1})]_+, \tag{33}$$

where $\boldsymbol{\lambda}_k$ is the Lagrange multipliers, $g(\mathbf{x})$ represents the constraint functions, and $\rho_k > 0$ is the dual step size. The inner unconstrained optimization problem is solved by gradient descent.

- **RD**:

$$\mathbf{y}_{k+1} = \mathbf{y}_k - \alpha_k \nabla \max\{f^\Gamma(\mathbf{y}_k), \gamma_\mathcal{K}(\mathbf{y}_k)\} \tag{34}$$

where f^Γ is the radial dual of the objective function and $\gamma_\mathcal{K}$ is the gauge function [Gri24b]. The solution is mapped to the original space after convergence as $\mathbf{x}^* = \mathbf{y}^*/f^\Gamma(\mathbf{y}^*)$ via radial dual.

- **Hom-PGD**:

$$\mathbf{z}_{k+1} = \Pi_\mathcal{B}(\mathbf{z}_k - \alpha_k \nabla f(\boldsymbol{\psi}(\mathbf{z}_k))) \tag{35}$$

where $\Pi_\mathcal{B}$ denotes the projection onto unit ball $\mathcal{B}$, and $\boldsymbol{\psi}$ is the homeomorphism. The solution is mapped to the original space after convergence as $\mathbf{x}^* = \boldsymbol{\psi}(\mathbf{z}^*)$.

- **MOSEK**: A commercial interior-point optimizer that solves conic optimization problems efficiently using highly optimized primal-dual interior-point methods with predictor-corrector techniques and sparse linear algebra. Note that we use an Academic license for MOSEK.

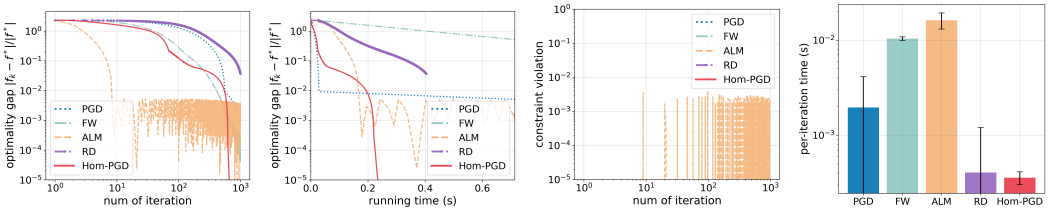
Gradient calculation: For simple quadratic objective functions, gradients are calculated via closed-form formulations. Other non-trivial gradient calculations across the various algorithms are implemented using auto-differentiation in PyTorch. We note that replacing auto-differentiation with closed-form gradient implementations could further improve the computational efficiency of the algorithms.

Step-size: Theoretically, different algorithms employ their own step size selection strategies, such as explicit dependence on smoothness and convexity parameters, or implicit step sizes that depend on the optimal objective value [Gri24b]. For practical implementation, we initialize a fixed step size (e.g., 10^{-3}) and decay it by a factor of 0.999 if the objective value does not decrease, which helps identify a sufficient step size for convergence. Although more sophisticated backtracking line search methods or adaptive step size schemes could accelerate convergence, we do not implement these for the sake of fair comparison.

Computational Environment: We conduct our experiments across two computational platforms to accommodate different problem scales. For small-scale illustrative examples, we execute algorithm comparisons on a MacBook Pro 2023. For larger-scale SOCP and SDP experiments, we implement all algorithms in PyTorch and execute them on an Ubuntu server equipped with an NVIDIA A800 GPU and an AMD EPYC 7763 64-Core Processor.

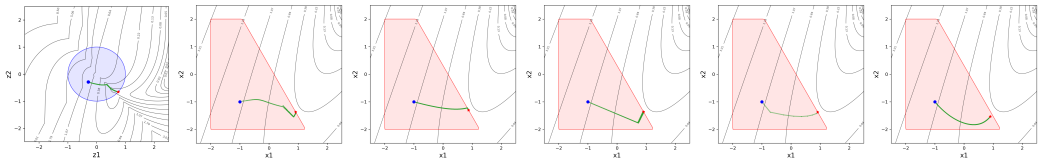
G Supplementary Experiments Results

G.1 Illustrative Examples



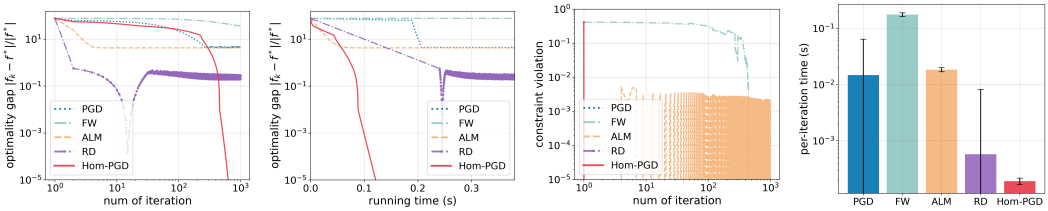
(a) convergence rate (b) running time (s) (c) constraint violation (d) per-iter. time (s)

Figure 8: Convergence performance for optimization over polyhedron.



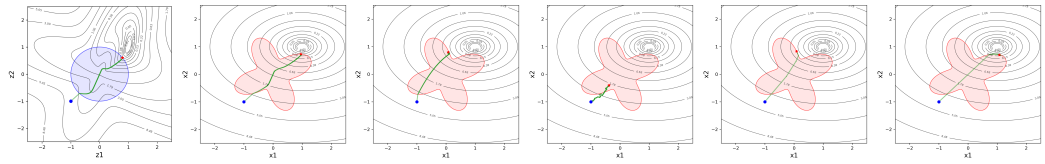
(a) Hom-PGD (b) Hom-PGD (c) PGD (d) FW (e) ALM (f) RD

Figure 9: Iteration trajectory for optimization over polyhedron. Hom-PGD (a) and RD (f) are also mapped to the original space to visualize their trajectories.



(a) convergence rate (b) running time (s) (c) constraint violation (d) per-iter. time (s)

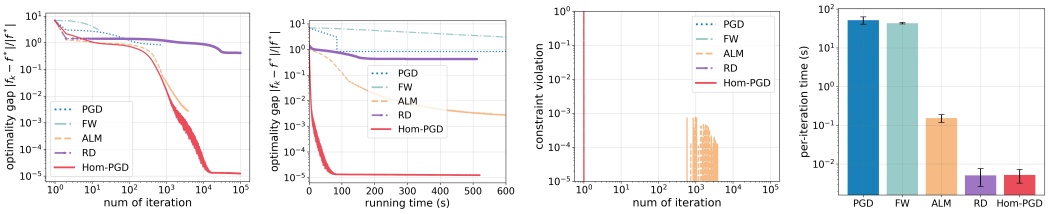
Figure 10: Convergence performance for optimization over a star-shaped set.



(a) Hom-PGD (b) Hom-PGD (c) PGD (d) FW (e) ALM (f) RD

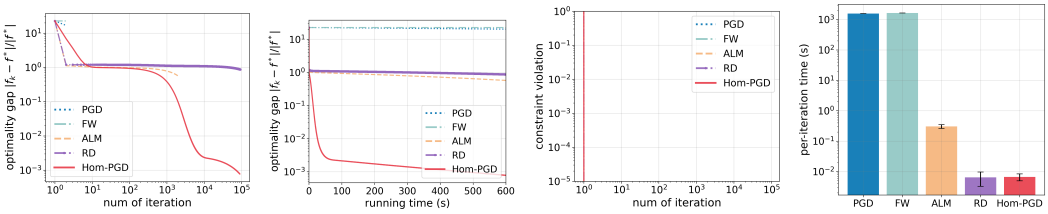
Figure 11: Iteration trajectory for optimization over a star-shaped set. Hom-PGD (a) and RD (f) are also mapped to the original space to visualize their trajectories.

G.2 Second-order Cone Programming



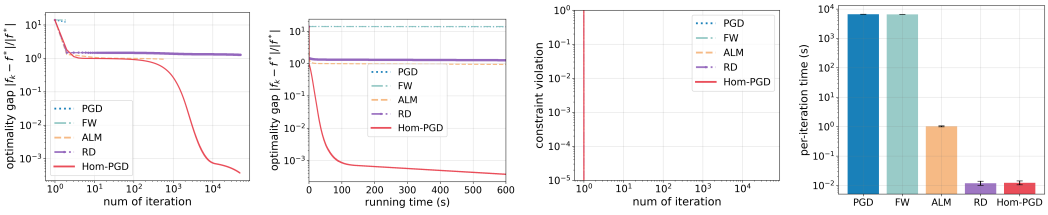
(a) convergence rate (b) running time (s) (c) constraint violation (d) per-iter. time (s)

Figure 12: Convergence performance over SOCP with $(n, m) = (100, 1000)$.



(a) convergence rate (b) running time (s) (c) constraint violation (d) per-iter. time (s)

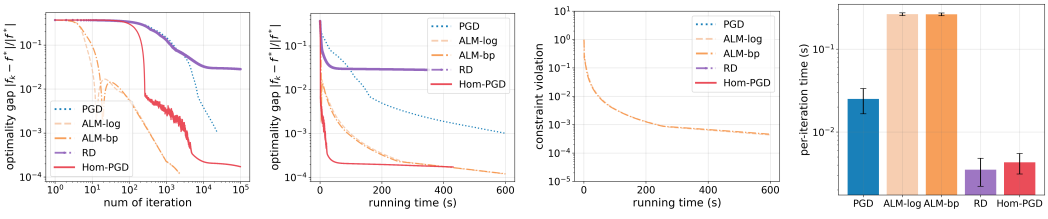
Figure 13: Convergence performance over SOCP with $(n, m) = (500, 1500)$.



(a) convergence rate (b) running time (s) (c) constraint violation (d) per-iter. time (s)

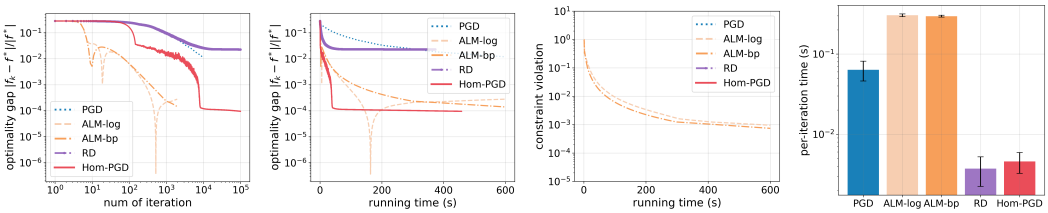
Figure 14: Convergence performance over SOCP with $(n, m) = (1000, 2500)$.

G.3 Max-Cut Semidefinite Programming



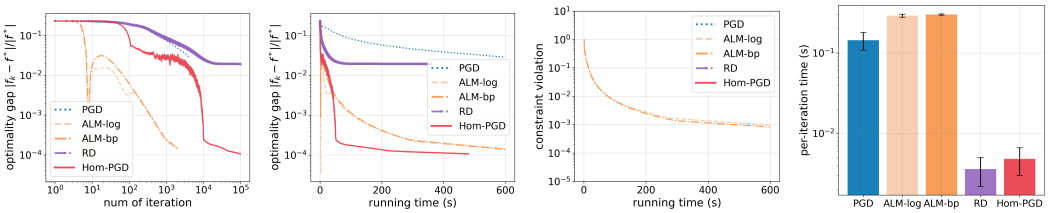
(a) convergence rate (b) running time (s) (c) constraint violation (d) per-iter. time (s)

Figure 15: Convergence performance over SDP with $n = 10^2$.



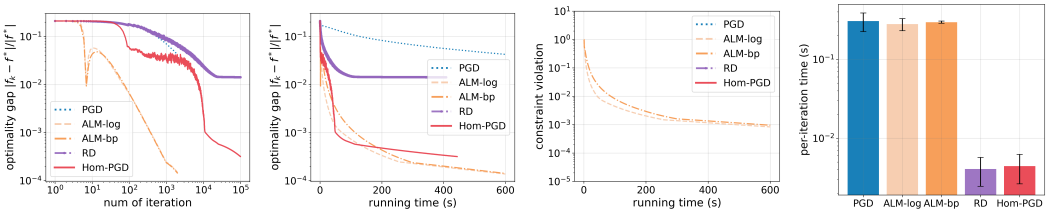
(a) convergence rate (b) running time (s) (c) constraint violation (d) per-iter. time (s)

Figure 16: Convergence performance over SDP with $n = 20^2$.



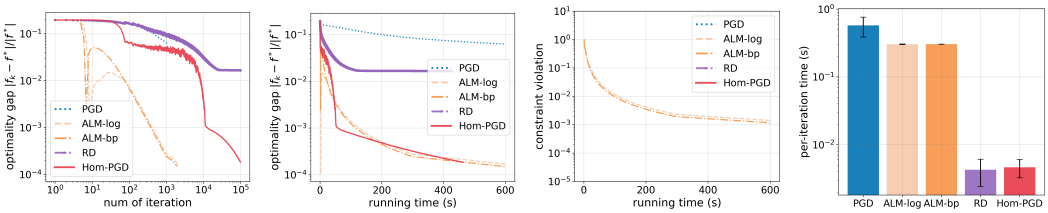
(a) convergence rate (b) running time (s) (c) constraint violation (d) per-iter. time (s)

Figure 17: Convergence performance over SDP with $n = 30^2$.



(a) convergence rate (b) running time (s) (c) constraint violation (d) per-iter. time (s)

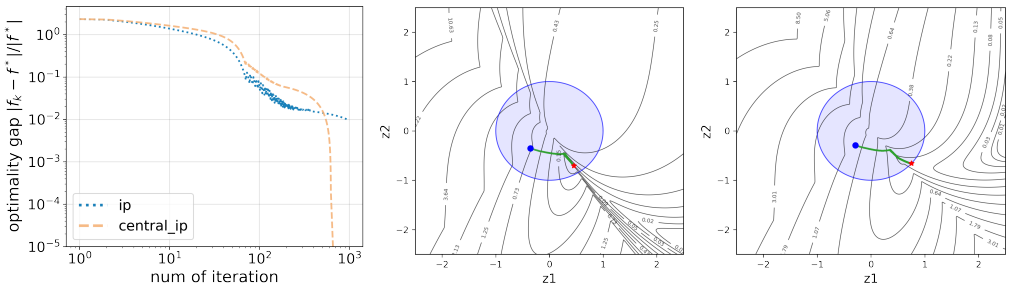
Figure 18: Convergence performance over SDP with $n = 40^2$.



(a) convergence rate (b) running time (s) (c) constraint violation (d) per-iter. time (s)

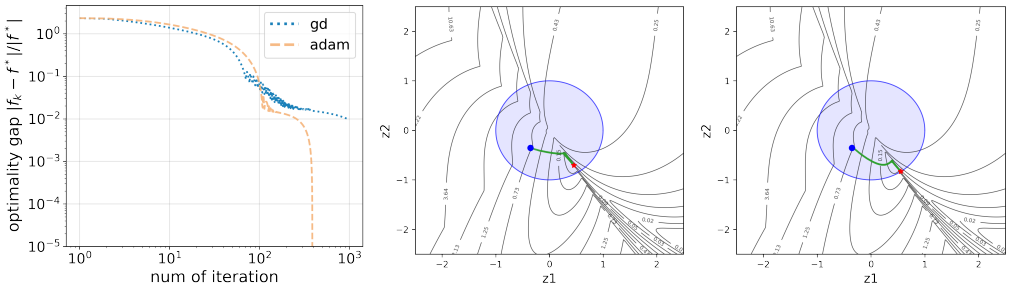
Figure 19: Convergence performance over SDP with $n = 50^2$.

G.4 Ablation Study



(a) Convergence comparison. (b) Near-Boundary IP (c) "Central" IP

Figure 20: Effect of **interior point** (IP) selection on convergence behavior: Gauge mapping with a near-boundary IP results in larger Bi-Lipschitz constants, which distorts the landscape of the transformed problem $\mathbf{H}$, consequently reducing convergence speed in practice.



(a) Convergence comparison. (b) Gradient descent (c) Adam [KB14]

Figure 21: Comparison of **gradient methods**: We evaluate Hom-PGD under Gauge mapping with a near-boundary IP. In this non-convex landscape, standard gradient descent exhibits slower convergence, while the Adam optimizer demonstrates superior performance due to its momentum acceleration and adaptive step size adjustment.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: See the abstract part and the introduction part (Sec. 1).

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: See Sec. 7.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

Justification: See Sec. 4 and Appendix C.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: See Sec. 6 and Appendix F.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [No]

Justification: it does not include code.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: See Sec. 6 and Appendix F.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: See Sec. 6 and Appendix G.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.

- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: See Sec. 6 and Appendix F.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: It conforms with the NeurIPS Code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [NA]

    Justification: There is no societal impact of the work performed.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to

generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.

- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: The paper poses no such risks.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [NA]

    Justification: The paper does not use existing assets.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [NA]

Justification: The paper does not release new assets.

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [NA]

Justification: The paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [NA]

Justification: The paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer: [NA]

Justification: The core method development in this research does not involve LLMs as any important, original, or non-standard components.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.