Convex Optimization over Intersection of Simple Sets: improved Convergence Rate Guarantees via an Exact Penalty Approach

Achintya Kundu
Department of CSA
Indian Institute of Science

Francis Bach
INRIA - École Normale Supérieure
PSL Research University

Chiranjib Bhattacharyya
Department of CSA
Indian Institute of Science

Abstract

We consider the problem of minimizing a convex function over the intersection of finitely many simple sets which are easy to project onto. This is an important problem arising in various domains such as machine learning. The main difficulty lies in finding the projection of a point in the intersection of many sets. Existing approaches yield an infeasible point with an iteration-complexity of $O(1/\varepsilon^2)$ for nonsmooth problems with no guarantees on the in-feasibility. By reformulating the problem through exact penalty functions, we derive first-order algorithms which not only guarantees that the distance to the intersection is small but also improve the complexity to $O(1/\varepsilon)$ and $O(1/\sqrt{\varepsilon})$ for smooth functions. For composite and smooth problems, this is achieved through a saddle-point reformulation where the proximal operators required by the primal-dual algorithms can be computed in closed form. We illustrate the benefits of our approach on a graph transduction problem and on graph matching.

1 Introduction

We call a closed convex set *simple* if there is an oracle available for computing Euclidean projection onto the set. In this paper we consider the problem of minimizing a convex function f over a convex set $\mathcal{C}$ where $\mathcal{C}$ is given as the intersection of finitely many simple closed convex sets $\mathcal{C}_1, \ldots, \mathcal{C}_m$ ($m \geq 2$). Specifically, we focus on optimization problems of the following form:

$$f_* \;=\; \min_{\mathbf{x} \,\in\, \mathcal{X}} \Big[\; f(\mathbf{x}) \;+\; \sum_{i=1}^{m} \mathbf{1}_{\mathcal{C}_i}(\mathbf{x}) \;\Big], \qquad (1)$$

where $\mathbf{1}_{\mathcal{C}_i}$ is the indicator function for set $\mathcal{C}_i$ and $\mathcal{X}$ ($\mathcal{C} \subset \mathcal{X}$) represents the domain of f.

Optimization problems of the form (1) arise in many machine learning tasks such as learning over doubly stochastic matrices, matrix completion [1], graph transduction [2]; sparse principal component analysis can be posed as optimization over the intersection of the set of positive semidefinite (PSD) matrices with unit trace and an ℓ_1-norm ball [3]; in learning correlation matrices, the feasible set is the intersection of the PSD cone and the set of symmetric matrices with diagonal elements equal to one [4]. Another area of computer science where problems of type (1) occur is in convex relaxations of various combinatorial optimization problems such as correlation clustering [5], graph-matching [6], etc.

Over the last few decades a large number of first-order algorithms have been proposed to solve (1) efficiently assuming $\mathcal{C}$ to be simple [7, 8, 9]. But, in many practical problems such as those mentioned above, projection onto the feasible set $\mathcal{C} = \cap_{i=1}^{m} \mathcal{C}_i$ is difficult to compute whereas oracles for projecting onto each of $\mathcal{C}_1, \ldots, \mathcal{C}_m$ are readily available. Note that many sets $\mathcal{C}$ where Frank-Wolfe algorithms can sometimes be used [10], i.e., when maximizing linear functions on $\mathcal{C}$ is supposed to be efficient, can often be decomposed as the intersection of sets with projection oracles (a classical example being the set of doubly stochastic matrices, as done in our experiments).

This calls for developing efficient first-order algorithms which access $\mathcal{C}$ only through the projection oracles of the individual sets $\mathcal{C}_1, \ldots, \mathcal{C}_m$. We mention here that such algorithms have been well-studied in the context of two specific problems: (a) the convex feasibility problem (corresponding to $f = 0$), which aims at finding a point in $\mathcal{C} = \cap_{i=1}^{m} \mathcal{C}_i$ [11, 12] and (b) the problem of

Proceedings of the 21^{st} International Conference on Artificial Intelligence and Statistics (AISTATS) 2018, Lanzarote, Spain. PMLR: Volume 84.

computing Euclidean projections onto $\cap_{i=1}^m \mathcal{C}_i$ [13, 14]. Existing algorithms for (a) and (b) ensure a feasible solution only in the asymptotic sense and in general produce only an infeasible approximate solution when terminated after a finite number of iterations. Therefore, aiming for feasible approximate solution without access to projection oracle for $\mathcal{C}$ seems too big a goal to achieve for problems of the form (1). Hence, we relax the feasibility requirement and introduce the following notion of approximate solution:

Definition 1 *For a given $\varepsilon > 0$, we call $\mathbf{x}_\varepsilon \in \mathcal{X}$ to be an ε-optimal ε-feasible solution of (1) if $f(\mathbf{x}_\varepsilon) - f_* \leq \varepsilon$ and $d_{\mathcal{C}}(\mathbf{x}_\varepsilon) \leq \varepsilon/L_f$, where $d_{\mathcal{C}}(\mathbf{x}_\varepsilon) \triangleq \inf_{\mathbf{x}\in\mathcal{C}} \|\mathbf{x} - \mathbf{x}_\varepsilon\|$ and L_f is a Lipschitz constant of f.*

Note that $f(\mathbf{x}_\varepsilon) \geq f_*$ holds if $\mathbf{x}_\varepsilon$ is feasible. Since $\mathbf{x}_\varepsilon$ is allowed to be infeasible as per the above definition, $f(\mathbf{x}_\varepsilon)$ might be well below f_*. The bound on the distance to feasible set $d_{\mathcal{C}}(\mathbf{x}_\varepsilon) \leq \varepsilon/L_f$ not only characterizes that feasibility violation of $\mathbf{x}_\varepsilon$ is small but also ensures $f(\mathbf{x}_\varepsilon) - f_* \geq -\varepsilon$. With the notion of approximate solution in place, the key question now is the following: given access to projection oracles of the $\mathcal{C}_i$'s how many oracle calls does a first-order method need in order to produce an ε-optimal ε-feasible solution of (1)? In this paper we aim to address this question. We summarize our contributions below.

Contributions. To the best of our knowledge, we are the first one to derive general complexity results for problems of the form (1) where f is given by a first-order oracle and the feasible set $\mathcal{C} = \cap_{i=1}^m \mathcal{C}_i$ can be accessed only through projections onto $\mathcal{C}_i$s. Note that our complexity estimates not only guarantee closeness of the approximate solution to the optimal objective value but also provide guarantees on the distance of such infeasible solutions from the feasible set. More precisely (see summary in Table 1):

- Utilizing a standard constraint qualification assumption on problem (1), we present in Proposition 2 an exact penalty based reformulation whose ε-optimal feasible solutions are in fact the desired ε-accurate ε-feasible solution of (1).
- We show in Proposition 3 that an adaptation of the standard subgradient method achieves the $O(1/\varepsilon^2)$ iteration complexity for obtaining an ε-optimal ε-feasible solution of (1) where f belongs to the class of general nonsmooth convex functions given by a first-order oracle. Specifically, an iteration of the proposed algorithm asks for one call to the first-order oracle of f and one call each to the projection oracles of $\mathcal{C}_1, \ldots, \mathcal{C}_m$. Additionally, assuming f to be strongly convex we show in Proposition 4 that the same subgradient based algorithm achieves the $O(1/\varepsilon)$ iteration complexity for obtaining an ε-optimal ε-feasible solution of (1). We mention that existing approaches [15] with $O(1/\varepsilon^2)$ complexity produce only an infeasible solution without any guarantee on the distance of the infeasible solutions from the feasible set. For the strongly convex case $O(1/\varepsilon)$ complexity was reported [14] but applicable only to a limited class of functions f where gradients of Fenchel conjugate of f can be computed easily. In contrast, our method relies on the availability of only subgradient of f.
- Through a novel saddle-point reformulation and employing existing primal-dual methods we show that the resulting approach achieves $O(1/\varepsilon)$ iteration complexity (with per iteration cost similar to the subgradient approach) for obtaining an ε-optimal ε-feasible solution of (1) when f belongs to the class of smooth convex functions given by a first-order oracle. Further, assuming f to be strongly convex the same primal-dual approach achieves $O(1/\sqrt{\varepsilon})$ iteration complexity for obtaining an ε-optimal ε-feasible solution of (1). Moreover, for nonsmooth convex functions with specific structure, for example, when the minimization problem has a smooth convex-concave saddle-point representation, we show that an adaptation of the mirror-prox technique achieves an iteration complexity of $O(1/\varepsilon)$ to produce an ε-optimal ε-feasible solution of (1). For the same class of functions, existing approaches [16] using mirror-prox reported $O(\frac{1}{\varepsilon}\log\frac{1}{\varepsilon})$ complexity.

Table 1: Complexity of the proposed first-order algorithms for obtaining an ε-optimal ε-feasible solution of (1) under 4 different classes of functions f.

Class of functions f	**Nonsmooth**	**Smooth**
Convex	$O(1/\varepsilon^2)$	$O(1/\varepsilon)$
Strongly convex	$O(1/\varepsilon)$	$O(1/\sqrt{\varepsilon})$

Notation. Through out this paper $\|\cdot\|$ denotes the standard Euclidean norm. Let $\mathcal{A} \subset \mathbb{R}^n$ be a nonempty closed convex set and $\mathbf{x} \in \mathbb{R}^n$. The Euclidean projection of $\mathbf{x}$ onto $\mathcal{A}$ is $\mathsf{P}_{\mathcal{A}}(\mathbf{x}) \triangleq \operatorname{argmin}_{\mathbf{a}\in\mathcal{A}} \|\mathbf{x} - \mathbf{a}\|$. The distance of $\mathbf{x}$ from $\mathcal{A}$ is given by $d_{\mathcal{A}}(\mathbf{x}) \triangleq \min_{\mathbf{a}\in\mathcal{A}} \|\mathbf{x} - \mathbf{a}\|$. The support function of $\mathcal{A}$ is defined as $\sigma_{\mathcal{A}}(\mathbf{x}) \triangleq \sup_{\mathbf{a}\in\mathcal{A}} \langle \mathbf{x}, \mathbf{a}\rangle$. Proximal operator of a convex function $\psi : \mathcal{A} \to \mathbb{R}$ is defined as $\mathsf{Prox}_{\gamma\psi}(\mathbf{x}) \triangleq \operatorname{argmin}_{\mathbf{a}\in\mathcal{A}} \left[\psi(\mathbf{a}) + \frac{1}{2\gamma}\|\mathbf{x} - \mathbf{a}\|^2\right]$, $\gamma > 0$. Whenever $\frac{0}{0}$ appears we will treat it to be 0. Proofs of all Propositions & Lemmas and details of the proposed algorithms are given in the supplementary material.

2 Problem Set-up & Related Work

In this paper we focus on developing efficient first-order algorithms for solving problems of the form (1). For the rest of this paper, we make the following assumptions on (1):

A1. $\mathcal{X}, \mathcal{C}_1, \ldots, \mathcal{C}_m$ are simple closed convex sets in $\mathbb{R}^n$ such that $\mathcal{X} \supset \mathcal{C} \triangleq \bigcap_{i=1}^m \mathcal{C}_i$ and $\mathcal{X}$ is bounded,

A2. $f : \mathcal{X} \to \mathbb{R}$ is convex and Lipschitz continuous with Lipschitz constant $L_f > 0$,

A3. the family of sets $\{\mathcal{C}_1, \ldots, \mathcal{C}_m\}$ satisfies the standard constraint qualification condition [12]:

$$\exists\, \bar{\mathbf{x}} \;\in\; \bigcap_{i=1}^m \mathrm{ri}(\mathcal{C}_i), \tag{2}$$

where $\mathrm{ri}(\mathcal{C}_i)$ denotes the relative interior of $\mathcal{C}_i$. If $\mathcal{C}_i$ is polyhedral then $\mathrm{ri}(\mathcal{C}_i)$ in the above condition can be replaced by $\mathcal{C}_i$.

Note that we have access to oracles for computing Euclidean projections onto each of the following sets: $\mathcal{X}, \mathcal{C}_1, \ldots, \mathcal{C}_m$ as these sets have been assumed to be simple. Typically, the domain $\mathcal{X}$ is equal to one of the $\mathcal{C}_i$'s. The standard constraint qualification condition (2) enables us to avoid pathological cases. It is automatically satisfied in the following cases: (a) the feasible set $\mathcal{C}$ has a nonempty interior; (b) all $\mathcal{C}_i$'s are polyhedral and $\mathcal{C} \neq \emptyset$. By virtue of assumptions [A1-A3], the set of optimal solutions of (1) is nonempty as f is continuous over the nonempty compact set $\mathcal{C}$. Our goal in this paper is to develop efficient algorithms which can produce for any given $\varepsilon > 0$ an ε-optimal ε-feasible solution of (1) with access to only projection oracles of $\mathcal{X}, \mathcal{C}_1, \ldots, \mathcal{C}_m$ and a first-order oracle which returns a subgradient of f. Below we provide a brief survey of the existing literature.

2.1 Related work

Many algorithms with $O(1/\varepsilon^2)$ complexity have been suggested in the stochastic setting [17, 18, 19]. But these randomized approaches do not provide any insight on how to obtain an approximate solution with a deterministic guarantee on the distance to the feasible set. In [15] an incremental subgradient approach was proposed for solving general convex optimization problems of the form (1) through an exact penalty reformulation. Though their approach produces an ε-optimal solution of the penalized problem in $O(1/\varepsilon^2)$ iterations, such solutions need not be ε-optimal ε-feasible solutions of the original problem (1) as they come with no guarantee on their distance to the feasible set.

Another line of research considers problem (1) with $\mathcal{C}_i$'s given by functional constraints: $\mathcal{C}_i = \{\mathbf{x} \in \mathbb{R}^n \mid g_i(\mathbf{x}) \leq 0\}$ for some convex function g_i. In such setting, the convergence analysis of existing algorithms [20, 21, 22, 23] crucially depends on the assumption $\exists \bar{\mathbf{x}} \in \mathbb{R}^n$ such that $g_i(\bar{\mathbf{x}}) < 0$. Hence, these methods can not be applied for abstract set constraints by taking the distance function $d_{\mathcal{C}_i}$ as g_i. Also their dependence on the existence of a strictly feasible point makes them inapplicable in the presence of affine equality constraints. Another shortcoming of their approach is its sub-optimal performance when the objective function has smoothness structure. Hence, in this paper we explore alternative approaches without assuming any functional representation for the constraint sets. Using the standard constraint qualification (2) for problem (1) we derive in Section 6 a primal-dual formulation and an improved convergence guarantee of $O(1/\varepsilon)$ under additional smoothness / structural assumptions on f. In [24] a smooth penalty based approach was proposed for minimizing convex function over intersections of convex sets; however, their approach does not provide any guarantee on the feasibility violation of the approximate solutions. In addition, their method requires the penalty constant to approach infinity, which our method does not require.

A special case of (1) where $\mathcal{C}$ is given by the inverse image of a convex cone under affine transformation was studied in [25]. Although any convex set $\mathcal{C}$ can be expressed as inverse image of a convex cone under an affine transformation, this representation may not result in tractable algorithm unless projecting onto the cone is easy. This makes the approach of [25] unsuitable for the the general case. [2] proposed an inexact proximal method to solve a graph transduction problem which is cast as an instance of (1) with $m = 2$. They substituted the projection step in the standard subgradient method with an approximate projection which is computed through an iterative algorithm. Due to the use of repeated projections onto $\mathcal{C}_i$'s to compute one approximate projection onto $\mathcal{C}$ their method can be shown to require $O(1/\varepsilon^3)$ projections onto each of the $\mathcal{C}_i$s for producing an ε-optimal ε-feasible solution of (1).

When the objective f is strongly convex the fast dual proximal gradient (FDPG) method of [14] can be applied to (1); they showed that the primal iterates (and corresponding primal objective function values) generated by the FDPG method converge to the optimal solution (optimal primal objective value) at $O(1/T)$-rate, where T is the number of iterations. But every iteration of the FDPG method requires solving a subproblem for computing the gradient of the Fenchel-conjugate of f. This makes the FDPG method unsuitable for a general strongly convex objective f where f is accessed only through a first-order oracle.

We note that our problem (1) can be posed in the

following form suitable for applying splitting methods such as the alternating direction method of multipliers (ADMM) [26] or proximal method of multipliers [27]:

$$\min_{\mathbf{x}\in\mathcal{X},\,\mathbf{Z}} \quad f(\mathbf{x}) + \sum_{i=1}^{m} 1_{\mathcal{C}_i}(\mathbf{z}_i) \;\text{ s.t. }\; \mathbf{A}\mathbf{x} = \mathbf{Z}\,, \tag{3}$$

where $\mathbf{Z} \triangleq (\mathbf{z}_1, \ldots, \mathbf{z}_m)$, $\mathbf{A}$ denotes the mapping $\mathbf{x} \mapsto (\mathbf{x}, \ldots, \mathbf{x}) \in \otimes_{i=1}^{m}\mathbb{R}^n$. Note that splitting based approach requires solving a subproblem of the following form at every iteration: $\operatorname{argmin}_{\mathbf{x}\in\mathcal{X}} f(\mathbf{x}) + \frac{\rho}{2}\|\mathbf{A}\mathbf{x} - \mathbf{Z}\|^2$ for a fixed $\rho > 0$ and $\mathbf{Z}$. Hence, these methods are suitable only when solving the above mentioned subproblem is easy. Therefore, in this paper we aim at developing algorithms which deals with the general case and make no assumption on availability of efficient oracles for solving such subproblems. We mention here that (3) is a special case of semi-separable problem considered in [16]. For that they proposed a first-order algorithm with $O(\frac{1}{\varepsilon}\log\frac{1}{\varepsilon})$ complexity when f possesses a special saddle-point structure. Specifically, their algorithm proceeds in stages with each stage solving a saddle-point formulation through composite mirror-prox [16] technique. Our exact penalty based approach enables us to achieve improved complexity of $O(\frac{1}{\varepsilon})$ through a similar mirror-prox based algorithm; notably our approach does not need several stages unlike that of [16].

Finally, we mention the connection to the literature on error bounds [28]. For convex feasibility problem (the case when $f = 0$) there is a rich history of using the distance to the individual sets $\mathcal{C}_1, \ldots, \mathcal{C}_m$ as a proxy for minimizing the distance to the intersection [12]. However, we explore the use of the same in the context of optimization problems ($f \neq 0$). Notably, utilizing distance to the individual sets we construct an exact penalty based formulation whose approximate solutions have guarantees on their distance to the intersection of the sets. Note that error bound properties (characterizations of the distance to the set of optimal solutions) in constrained convex optimization have been shown to hold only for a limited set of problems [29]. Assuming certain error bound conditions there have been attempt to establish better convergence rate guarantees [23]. However, in this paper we deal with the general case with out assuming any error bound property for problem (1).

3 Exact Penalty-based Reformulation

In this section we show that standard constraint qualification (2) allows us to find a suitable penalty function-based reformulation of (1). Towards that we first recall the concept of *linear regularity* of a collection of convex sets:

Definition 2 *The collection of closed convex sets* $\{\mathcal{C}_1, \ldots, \mathcal{C}_m\}$ *is linearly regular if* $\exists \Upsilon > 0$ *such that*

$$\forall \mathbf{x} \in \mathbb{R}^n : \; d_{\mathcal{C}}(\mathbf{x}) \;\leq\; \Upsilon \max_{1\leq i\leq m} d_{\mathcal{C}_i}(\mathbf{x}). \tag{4}$$

A sufficient condition for $\{\mathcal{C}_1, \ldots, \mathcal{C}_m\}$ to be linearly regular is that $\mathcal{C} = \bigcap_{i=1}^{m}\mathcal{C}_i$ is bounded and the standard constraint qualification (2) holds [30]. Thus, for problem (1) we have linear regularity of $\{\mathcal{C}_1, \ldots, \mathcal{C}_m\}$ as a consequence of the assumptions [A1-A3]. In this context we mention that [17, 18, 19] assumed linear regularity property of the sets for designing stochastic algorithms for problem (1). For the rest of the paper Υ will denote the linear regularity constant of $\{\mathcal{C}_1, \ldots, \mathcal{C}_m\}$. Let $R, r > 0$ be such that $\mathcal{C} = \cap_{i=1}^{m}\mathcal{C}_i$ contains a ball of radius r and $\mathcal{C}$ is contained in a ball of radius R; then the ratio R/r can be taken as the regularity constant Υ [31]. Please refer to [11] for details about linear regularity and how to estimate the corresponding constant. In the supplementary material we discuss an algorithmic strategy based on a "doubling trick" to deal with the case when the regularity constant Υ is not available.

We now discuss the availability of suitable penalty functions for $\mathcal{C} = \bigcap_{i=1}^{m}\mathcal{C}_i$ such that we can solve the penalty-based reformulation efficiently using existing first-order methods without requiring projection onto $\mathcal{C}$; however, the method can make use of the oracles for projecting onto each of the $\mathcal{C}_i$s. Below we characterize a class of such penalty functions through the notion of absolute norm [32].

Definition 3 *A norm P on $\mathbb{R}^m$ is called an absolute norm if $\forall \mathbf{u} \in \mathbb{R}^m$ we have $P(\mathbf{u}) = P(|\mathbf{u}|)$, where $|\mathbf{u}|$ denotes the vector obtained by taking element-wise modulus of $\mathbf{u}$.*

Proposition 1 *Let P be an absolute norm on $\mathbb{R}^m$ and $h_P : \mathbb{R}^n \to \mathbb{R}_+$ be defined as*

$$h_P(\mathbf{x}) \;=\; P(d_{\mathcal{C}_1}(\mathbf{x}), \ldots, d_{\mathcal{C}_m}(\mathbf{x})),\; \mathbf{x} \in \mathbb{R}^n. \tag{5}$$

Then (a) h_P is a convex function, (b) $h_P(\mathbf{x}) = 0$ if and only if $\mathbf{x} \in \mathcal{C}$, (c) $\exists$ a regularity constant $\Upsilon_P > 0$ such that

$$\forall\, \mathbf{x} \in \mathbb{R}^n : \quad d_{\mathcal{C}}(\mathbf{x}) \;\leq\; \Upsilon_P\, h_P(\mathbf{x}). \tag{6}$$

With h_P as defined in (5), we consider the following penalized version of problem (1):

$$\boxed{f_*^\lambda = \inf_{\mathbf{x}\in\mathcal{X}} \left[f^\lambda(\mathbf{x}) \equiv f(\mathbf{x}) + \lambda\, h_P(\mathbf{x})\right],\; \lambda > 0.} \tag{7}$$

An exact penalty function of the form $\sum_{i=1}^{m}\gamma_i d_{\mathcal{C}_i}(\cdot)$, where the constants γ_i are chosen solely based on the

Lipschitz constant L_f (without taking linear regularity of the sets into account) was proposed in [15]. In Appendix we provide a counter example to show that choosing γ_i's as per their prescription does not always work (in fact our example shows that Proposition 11 in [15] does not hold in absence of the standard constraint qualification). Secondly, the penalty-based reformulation suggested in [15] does not provide any guarantee on the feasibility violation (distance from the feasible set) of the approximate solutions. In this paper, making use of the standard constraint qualification condition (2) we propose the penalty-based reformulation (7) which we will show to be an exact reformulation of the original problem (1) with the added property that the approximate solutions of (7) are also the approximate solutions of (1) with the desired in-feasibility guarantee. We now show that the use of linear regularity property (6) allows us to relate the solution set of (7) to that of (1) and the constant λ can be set independent of the desired accuracy of the solution (but big enough) unlike other penalty methods [24] where $\lambda \to \infty$ is needed.

Proposition 2 *Consider problem (1) and the corresponding penalty-based formulation (7) with penalty function h_P as in (5). Then we have:*

a. *If $\lambda \geq \Upsilon_P L_f$ then $f_*^\lambda = f_*$ and every optimal solution of (1) is an optimal solution of (7).*

b. *If $\lambda > \Upsilon_P L_f$ then every optimal solution of (7) is an optimal solution of (1).*

c. *Let $\lambda \geq 2\Upsilon_p L_f$ and $\mathbf{x}_\varepsilon$ be an ε-optimal solution of (7) for a given $\varepsilon > 0$. Then $\mathbf{x}_\varepsilon$ is an ε-optimal ε-feasible solution of (1), that is, $f(\mathbf{x}_\varepsilon) - f_* \leq \varepsilon$ and $d_{\mathcal{C}}(\mathbf{x}_\varepsilon) \leq \frac{\varepsilon}{L_f}$. Moreover, $\mathsf{P}_{\mathcal{C}}(\mathbf{x}_\varepsilon)$ is an ε-optimal feasible solution of (1).*

4 Nonsmooth Objective Functions

In this section we propose an adaptation of the standard subgradient method for efficiently solving nonsmooth convex optimization problems over intersections of simple convex sets. Specifically, we consider problem (1) with f represented by a black-box oracle of first-order, that is, the oracle returns a subgradient $f'(\mathbf{x})$ of f at $\mathbf{x} \in \mathcal{X}$. Without loss of generality we assume that the subgradients returned by the oracle are bounded by the Lipschitz constant L_f. Note that direct application of the subgradient method to solve (1) requires projection onto the feasible set $\mathcal{C}$ which may be hard to compute even when $\mathcal{C}$ is given by the intersection of finitely many simple sets. Our adaptation of the subgradient method, which we call the "split-projection subgradient" (SPS) algorithm, overcomes this difficulty by requiring projections only onto each $\mathcal{C}_i$. We achieve this by applying the standard subgradient algorithm to problem (7) instead of (1). In order to apply subgradient method to (7) we present the following Lemma:

Lemma 1 *Let h_P be as defined in (5). Then h_P is Lipschitz-continuous on $\mathbb{R}^n$ with Lipschitz constant $P(\mathbf{1})$ where $\mathbf{1} \in \mathbb{R}^m$ is the vector of all ones. Moreover, a subgradient of h_P at $\mathbf{x} \in \mathbb{R}^n$ is given by*

$$h_P'(\mathbf{x}) = \sum_{i=1}^m \frac{u_i^*}{d_i}[\mathbf{x} - \mathsf{P}_{\mathcal{C}_i}(\mathbf{x})],$$

where $\mathbf{u}^* := \operatorname{argmax}_{\mathbf{u} \in \mathbb{R}^m} \{\sum_{i=1}^m u_i d_i \,|\, P_*(\mathbf{u}) \leq 1\}$, $d_i = \|\mathbf{x} - \mathsf{P}_{\mathcal{C}_i}(\mathbf{x})\|$ and P_* denotes the dual norm of P.

The above lemma shows that we can compute a subgradient of the penalty term h_P utilizing the projection oracles of $\mathcal{C}_1, \ldots, \mathcal{C}_m$ and a linear optimization oracle for the unit ball of P_*. Moreover, the Lipschtiz continuity of h_P ensures that such subgradients are bounded by the constant $P(\mathbf{1})$. Also, recall from the previous section that solving (7) is equivalent to (1) under $\lambda > \Upsilon_P L_f$. Therefore, we can apply subgradient method to (7) with $\lambda \geq 2\Upsilon_P L_f$. This results in the SPS algorithm for solving (1). The key recursion in SPS algorithm is the following:

$$\mathbf{x}^{(t+1)} := \mathsf{P}_{\mathcal{X}}\left(\mathbf{x}^{(t)} - \gamma_t[f'(\mathbf{x}^{(t)}) + \lambda\, h_P'(\mathbf{x}^{(t)})]\right),\ t \geq 1.$$

Algorithmic details are given in the supplementary material. Now, the following proposition states the convergence behavior of the proposed SPS algorithm:

Proposition 3 *Consider the SPS algorithm applied to problem (1) with $\lambda \geq 2\Upsilon_P L_f$. Then, for a given $\varepsilon > 0$, SPS algorithm produces an ε-optimal ε-feasible solution of (1) in no more than $O(1/\varepsilon^2)$ iterations where each iteration involves computation of a subgradient of f and projections onto each of $\mathcal{X}, \mathcal{C}_1, \ldots, \mathcal{C}_m$.*

Now, we present an improved complexity estimate for the class of strongly convex functions.

Proposition 4 *Consider the SPS algorithm applied to problem (1) where f is strongly convex with strong convexity parameter $\mu_f > 0$. Let $\lambda \geq 2\Upsilon_P L_f$. Then, for a given $\varepsilon > 0$, SPS algorithm produces an ε-optimal ε-feasible solution of (1) in no more than $O(1/\varepsilon)$ iterations where each iteration involves computation of a subgradient of f and projections onto each of $\mathcal{X}, \mathcal{C}_1, \ldots, \mathcal{C}_m$.*

5 Smooth Objective Functions

In this section we consider solving problem (1) under the additional assumption that f is smooth. Specif-

ically, we assume through out this section that the gradient of f, denoted as ∇f, is Lipschitz-continuous on $\mathcal{X}$ with Lipschitz constant M_f. Recall that we have access to only a first-order oracle which returns the gradient $\nabla f(\mathbf{x})$ of f at $\mathbf{x} \in \mathcal{X}$ and projection oracles for computing projections onto the simple sets $\mathcal{X}, \mathcal{C}_1, \ldots, \mathcal{C}_m$. If we had access to projection oracle for $\mathcal{C}$ then applying accelerated gradient methods we can obtain an ε-optimal solution of (1) in $O(1/\sqrt{\varepsilon})$ iterations. But, in the absence of projection oracle for $\mathcal{C}$, problem (1) is essentially an instance of nonsmooth optimization as the nonsmooth part $\sum_{i=1}^m 1_{\mathcal{C}_i}$ does not possess a tractable proximal operator. Therefore, existing first-order methods for smooth/composite convex minimization can not be applied directly to (1).

One of the main contributions of the paper is to show that we can use first-order methods through an adaptation of the primal-dual framework of [33]. To apply the primal-dual framework we first propose a saddle-point reformulation of (7) by exploiting the structure of the nonsmooth penalty function h_P. Before going into the details, we introduce the following notation: $\mathbf{Y} \triangleq (\mathbf{y}_1, \ldots, \mathbf{y}_m) \in \otimes_{i=1}^m \mathbb{R}^n$. We have:

Lemma 2 *Let h_P be as defined in (5). Then the following holds for all $\mathbf{x} \in \mathbb{R}^n$:*

$$h_P(\mathbf{x}) = \max_{\mathbf{Y} \in \mathcal{Y}_P} \sum_{i=1}^m \left[\mathbf{x}^\top \mathbf{y}_i - \sigma_{\mathcal{C}_i}(\mathbf{y}_i) \right], \tag{8}$$

where $\mathcal{Y}_P \triangleq \{\mathbf{Y} \in \otimes_{i=1}^m \mathbb{R}^n \mid P_*(\|\mathbf{y}_1\|, \ldots, \|\mathbf{y}_m\|) \leq 1\}$, P_* denotes the dual norm of P and $\sigma_{\mathcal{C}_i}$ denotes the support function of set $\mathcal{C}_i$.

Exploiting the above structure of h_P we have the following saddle-point reformulation of (7) for any $\lambda > 0$:

$$\boxed{\min_{\mathbf{x} \in \mathcal{X}} \max_{\mathbf{Y} \in \mathcal{Y}_P^\lambda} \Big[\mathcal{L}(\mathbf{x}, \mathbf{Y}) \equiv f(\mathbf{x}) + \sum_{i=1}^m \mathbf{x}^\top \mathbf{y}_i - g(\mathbf{Y}) \Big],} \tag{9}$$

where $\mathcal{Y}_P^\lambda \triangleq \{\mathbf{Y} \in \otimes_{i=1}^m \mathbb{R}^n \mid P_*(\|\mathbf{y}_1\|, \ldots, \|\mathbf{y}_m\|) \leq \lambda\}$,

$$g(\mathbf{Y}) \triangleq \sum_{i=1}^m \sigma_{\mathcal{C}_i}(\mathbf{y}_i) + 1_{\mathcal{Y}_P^\lambda}(\mathbf{Y}), \quad \mathbf{Y} \in \otimes_{i=1}^m \mathbb{R}^n. \tag{10}$$

We now connect the saddle-point formulation (9) with the original problem (1).

Lemma 3 *Consider the saddle-point formulation (9) with $\lambda \geq 2\Upsilon_P L_f$. Fix $\varepsilon > 0$. Let $(\mathbf{x}_\varepsilon, \mathbf{Y}_\varepsilon) \in \mathcal{X} \times \mathcal{Y}_P^\lambda$ be an ε-optimal solution of (9) in the following sense:*

$$\sup_{\mathbf{x} \in \mathcal{X},\, \mathbf{Y} \in \mathcal{Y}_P^\lambda} \; [\; \mathcal{L}(\mathbf{x}_\varepsilon, \mathbf{Y}) - \mathcal{L}(\mathbf{x}, \mathbf{Y}_\varepsilon) \;] \;\leq\; \varepsilon. \tag{11}$$

Then $\mathbf{x}_\varepsilon$ is an ε-optimal ε-feasible solution of (1).

In order to solve (9) efficiently, the following lemma shows that the proximal operator of the nonsmooth convex function g can be evaluated in closed form through projections onto each of the sets $\mathcal{C}_1, \ldots, \mathcal{C}_m$ and a projection onto a dual norm ball of P.

Lemma 4 *Let g be defined in (10). Then for any $\gamma > 0$ and $\mathbf{Y} \in \otimes_{i=1}^m \mathbb{R}^n$ the proximal operator of g is given by $\mathsf{Prox}_{\gamma g}(\mathbf{Y}) = (r_1 \hat{\mathbf{y}}_1 / \|\hat{\mathbf{y}}_1\|, \ldots, r_m \hat{\mathbf{y}}_m / \|\hat{\mathbf{y}}_m\|)$, where $\hat{\mathbf{y}}_i \triangleq \mathbf{y}_i - \gamma \mathsf{P}_{\mathcal{C}_i}(\gamma^{-1}\mathbf{y}_i)$ and $(r_1, \ldots, r_m)$ is the projection of $(\|\hat{\mathbf{y}}_1\|, \ldots, \|\hat{\mathbf{y}}_m\|)$ onto $\{\mathbf{u} \in \mathbb{R}^m \mid P_*(\mathbf{u}) \leq \lambda\}$.*

With the proximal operator of g being computable and f being smooth, we define a primal-dual iteration of the following form:

$$\begin{array}{l} \text{Iteration: } (\mathbf{x}^+, \mathbf{Y}^+) = \mathcal{PD}_{\tau,\gamma}(\mathbf{x}, \mathbf{Y}, \tilde{\mathbf{x}}, \tilde{\mathbf{Y}}) \\ \left\{ \begin{array}{l} \mathbf{x}^+ := \mathsf{P}_{\mathcal{X}}(\mathbf{x} - \tau \left[\nabla f(\mathbf{x}) + \sum_{i=1}^m \tilde{\mathbf{y}}_i \right]), \\ \mathbf{Y}^+ := \mathsf{Prox}_{\gamma g}(\mathbf{Y} + \gamma \mathbf{A} \tilde{\mathbf{x}}), \end{array} \right. \end{array} \tag{12}$$

where $\mathbf{A}$ denotes the map $\mathbf{x} \mapsto (\mathbf{x}, \ldots, \mathbf{x}) \in \otimes_{i=1}^m \mathbb{R}^n$. With this we can now apply primal-dual algorithms of [33] to problem (9) with primal-dual iteration defined by (12). Since $\mathcal{X}$ and $\mathcal{Y}_P^\lambda$ are compact sets, we can obtain an ε-optimal solution of (9) in the sense of (11) by applying $O(1/\varepsilon)$ iterations of the non-linear primal-dual algorithm of [33]. Moreover, when f is smooth as well as strongly convex we can apply the accelerated primal-dual algorithm of [33] which needs only $O(1/\sqrt{\varepsilon})$ iterations of the form (12). Note that Lemma 3 guarantees that such solutions are enough to output an ε-optimal ε-feasible solution of (1). Therefore, complexity of obtaining an ε-optimal ε-feasible solution of (1) is $O(1/\sqrt{\varepsilon})$ when f is smooth and $O(1/\sqrt{\varepsilon})$ for smooth and strongly convex f. Thus, utilizing existing primal-dual machinery to a saddle-point reformulation of the exact penalty based equivalent problem (7), we achieve better complexity for problems of the form (1) under a smoothness assumption on f. We call this approach exact penalty primal-dual (EPPD) method.

6 Nonsmooth Objective Functions with Structure

In many machine learning problems such as kernel learning [34], learning optimal embedding for graph transduction [2], etc., the objective function f is defined as the optimal value of a maximization problem. In most of these cases, in spite of f being non-smooth, the problem of minimizing f can be cast as a smooth saddle-point problem. It is well-known that by exploiting such structure in the problem, first-order algorithms with improved convergence rate of $O(1/\varepsilon)$ can be obtained even for non-smooth problems [8]. Hence, in the context of problem (1) we would like to address the following question: by exploiting structure in f is it possible to

design first-order algorithms with $O(1/\varepsilon)$ complexity for (1)? For this we make the following additional assumptions on problem (1). Through out this section we will assume that the objective function f possesses the following structure:

$$\boxed{f(\mathbf{x}) \;=\; \max_{\mathbf{z}\,\in\,\mathcal{Z}} \; \Phi(\mathbf{x},\mathbf{z}),\; \mathbf{x}\in\mathcal{X},} \tag{13}$$

where $\mathcal{Z}$ is a simple compact convex set and $\Phi : \mathcal{X}\times\mathcal{Z}\to\mathbb{R}$ is a convex-concave function with gradient $\nabla\Phi$ being Lipschitz continuous on $\mathcal{X}\times\mathcal{Z}$. We also assume availability of a first-order oracle for computing $\nabla\Phi$ at any $(\mathbf{x},\mathbf{z})\in\mathcal{X}\times\mathcal{Z}$.

Recall that only projections onto $\mathcal{X},\mathcal{C}_1,\ldots,\mathcal{C}_m$ are available and algorithms can not ask for projections onto $\mathcal{C}$. This makes the existing mirror-prox algorithm [8] unsuitable for problem (1) even when f has the above structure. We overcome this difficulty by considering the penalty based formulation (7) where $\lambda \geq 2\Upsilon_P L_f$, f as in (13) and h_P given by (8). This results in the following saddle-point formulation like (9):

$$\boxed{\min_{\mathbf{x}\in\mathcal{X}} \max_{\mathbf{z}\in\mathcal{Z},\,\mathbf{Y}\in\mathcal{Y}_P^{\lambda}} \; [\,\Phi(\mathbf{x},\mathbf{z}) + \sum_{i=1}^{m} \mathbf{x}^\top\mathbf{y}_i - g(\mathbf{Y})\,].} \tag{14}$$

We see that the objective above has a nonsmooth term g and the remaining part has Lipschitz continuous gradient. Also, as given in Lemma 4, the proximal operator of g can be computed through projections onto $\mathcal{C}_i$'s. Hence, the mirror-prox-"a" (MPa) algorithm [8] can be applied to problem (14). The resulting approach we call split-mirror prox (SMP), with the following complexity estimate:

Proposition 5 *Given $\varepsilon > 0$, SMP algorithm requires no more than $O(1/\varepsilon)$ calls to the first order oracle of Φ and $O(1/\varepsilon)$ projections onto each of $\mathcal{X},\mathcal{Z},\mathcal{C}_1,\ldots,\mathcal{C}_m$ to produce an ε-optimal ε-feasible solution of (1) where f is of the form (13).*

7 Experimental Results

In this section we illustrate the benefits of the proposed algorithms on two problems: a graph transduction problem where the objective function is non-smooth but can be cast in the form (13) and on a graph matching problem where the objective is a smooth function with Lipschitz continuous gradient. We performed all the experiments on a CPU with Intel Core i7 processor and 8GB memory. In the implementation of the proposed methods we choose the norm P to be the standard ℓ_1-norm.

7.1 Learning orthonormal embedding for graph-transduction

Consider a simple graph $G = (V,E)$, with vertex set $V = \{1,\ldots,N\}$ and edge set $E \subset V\times V$. If $S \subseteq V$ is labelled with binary values denoted by $\mathbf{y}_S \in \{-1,1\}^{|S|}$ the problem of graph transduction can be posed as learning the labels of the remaining vertices. Recently the following formulation was posed in [2] for learning the optimal orthonormal embedding of the graph for a graph transduction problem:

$$\min_{\mathbf{K}\in\mathcal{K}(G)} \; [\,\omega_C(\mathbf{K},\mathbf{y}_S) \,+\, \beta\,\lambda_{max}(\mathbf{K})\,], \tag{15}$$
$$\omega_C(\mathbf{K},\mathbf{y}_S) = \max_{\boldsymbol{\alpha}\in\mathcal{A}} \sum_{i\in S}\alpha_i - \tfrac{1}{2}\sum_{i,j\in S}\alpha_i\alpha_j y_i y_j K_{ij},$$
$$\mathcal{A} = \{\boldsymbol{\alpha}\in\mathbb{R}^N \,|\, 0\leq\alpha_i\leq C\;\forall i\in S,\; \alpha_j = 0\;\forall j\notin S\},$$

where $C > 0$ and the set $\mathcal{K}(G)$ consists of positive semidefinite (PSD) kernel matrices arising due to an orthonormal embedding characterization. Specifically, $\mathcal{K}(\mathbf{G}) := \{\mathbf{K}\in\mathbb{R}^{N\times N} \,|\, \mathbf{K}\; is\; PSD,\; \mathbf{K}_{ii} = 1\,\forall i, \mathbf{K}_{ij} = 0\,\forall (i,j)\notin E\}$. The set $\mathcal{K}(G)$ is an elliptope lying in the intersection of PSD cone with affine constraints. The objective function consists of two nonsmooth functions, $\omega_C(\mathbf{K},\mathbf{y}_S)$ and $\lambda_{max}(\mathbf{K})$, the largest eigenvalue of $\mathbf{K}$ where $\beta > 0$ is user defined. In [2] an inexact infeasible proximal method (IIPM) was proposed which does not provide any feasibility guarantee on the approximate solutions. To illustrate the effect of infeasibility we compare the proposed SMP method with IIPM on solving (15).

We experimented on a subset of the MNIST dataset [35] where corresponding to each pair of digit classes we constructed a graph with $n = 1000$ nodes as follows: (a) first randomly select 500 samples from each digit; (b) for each pair of samples put an edge in the graph if the cosine distance between samples is less than a threshold value (we set 0.4 as the threshold). For IIPM the number of inner-iteration (S) to compute approximate projection was set to 5. The regularization parameters β and C were selected through 5-fold cross-validation. Table 2 summarizes the results, averaged over 5 random training/test partitioning. Labels for 10 percent of the nodes were used for training. Entries in the table represent classification accuracy (mean ± standard deviation) which we calculate as the percentage of un-labelled nodes classified correctly. To compare the effect of in-feasibility of the iterates generated by the two methods we report in Table 3 the objective function value reached for one particular training/test partitioning of the data. Under the infeasible column we present the objective value at the infeasible solutions returned by SMP/ IIPM whereas in the feasible case we project the output from both the algorithms to the feasible set before computing the objective value. We observe that IIPM misleadingly reports smaller objec-

Table 2: Comparison of classification accuracy (mean ± standard deviation) on MNIST digit recognition dataset.

Dataset	SMP	IIPM
1 vs 2	**96.9** ± 0.4	96.2 ± 1.2
1 vs 7	**97.6** ± 0.5	95.0 ± 3.0
3 vs 8	**89.7** ± 1.3	86.8 ± 2.7
4 vs 9	**83.0** ± 1.9	77.5 ± 2.1
6 vs 8	**97.6** ± 0.3	93.8 ± 2.0

Table 3: Comparison of objective function value on MNIST digit recognition dataset.

Dataset	Infeasible		Feasible	
	SMP	IIPM	SMP	IIPM
1 vs 2	5.77	3.97	**5.77**	5.80
1 vs 7	5.31	3.30	**5.32**	5.33
3 vs 8	6.46	4.33	**6.46**	6.54
4 vs 9	6.13	4.19	**6.14**	6.18
6 vs 8	6.35	4.38	**6.35**	6.38

tive values; this is result of the iterates being far from the feasible set. As the new SMP algorithm ensures that iterates are not far from the feasible set; also, objective function values do not change much even after projecting the infeasible solution onto the feasible set. Also, we see better predictive performance for SMP in Table 2.

7.2 Graph matching

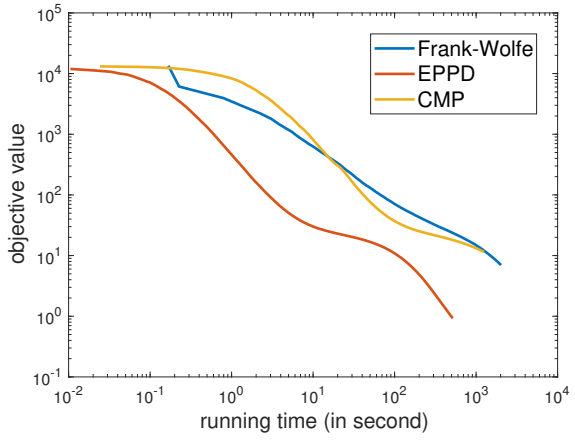

Figure 1: Convergence plot of Frank-Wolfe, Exact Penalty Primal-Dual (EPPD) and Composite Mirror-Prox (CMP) method on the Graph Matching problem.

We consider a graph matching problem, where two adjacency matrices A and B in $\mathbb{R}^{n\times n}$ are given, and we aim to minimize $\|A\Pi - \Pi B\|_F^2$ with respect to Π over the set of doubly stochastic matrices. This can be seen as a natural convex relaxation of optimizing on the set of permutation matrices [6]. The set of doubly stochastic matrices is defined as the intersection of two products of n simplices in dimension n (which are indeed simple sets with efficient projection oracles). We compare the proposed Exact Penalty based Primal-Dual (EPPD) approach to the Frank-Wolfe algorithm, for which the linear maximization oracle is an assignment problem, which can be solved in $O(n^3)$. Note that both algorithms have the same convergence rate in terms of number t of iterations, as $O(1/t)$. We also include in comparison the Composite Mirror-Prox (CMP) based approach for solving semi-separable problems [16]. We present experimental results on randomly generated undirected graphs with number nodes = 200. In Figure 1 we compare the convergence behavior of the methods. Although all the algorithms have very similar convergence rate, our EPPD method takes considerably lesser time. This is due to the fact the EPPD method just need projection onto simplices where as Frank-Wolfe needs to solve a linear maximization problem over the set of doubly stochastic matrices which is computationally more demanding. As the composite Mirror-Prox method requires two gradient computations and 2 proximal evaluations per iteration, every iteration of CMP is at least twice as costly as our primal-dual iteration; moreover, their formulation introduces m more variables; this makes CMP approach slower than EPPD.

8 Conclusions

In this paper, we presented algorithms to minimize convex functions over intersections of simple convex sets, with explicit convergence guarantees for feasibilty and optimality of function values. Our work not only bounds the level of in-feasibility, currently missing in existing literature but also improves the convergence rate. This is mostly based on a new saddle-point formulation with an explicit proximity operator, and led to improved experimental behavior in two situations. Our work opens up several avenues for future work: (a) we can imagine letting the number m of sets grow large or even to infinity and using a stochastic oracle [17] with efficient stochastic gradient techniques [36], (b) we could consider other geometries than the Euclidean one by considering mirror descent extensions.

Acknowledgement

This work was supported by a generous grant under the CEFIPRA project and the INRIA associated team BIGFOKS2. FB acknowledges support from the European Research Council (grant SEQUOIA 724063).

References

[1] A. Aragon, J. Francisco, J. M. Borwein, and T. K. Mathew. Douglas-Rachford feasibility methods for matrix completion problmes. *The ANZIAM Journal*, 55(4):299–326, 2014.

[2] R. Shivanna, B. Chatterjee, R. Sankaran, C. Bhattacharyya, and F. Bach. Spectral Norm Regularization of Orthonormal Representations for Graph Transduction. In *Proceedings of the 28th International Conference on Neural Information Processing Systems*, 2015.

[3] Alexandre d'Aspremont, Laurent El Ghaoui, Michael I. Jordan, and Gert R. G. Lanckriet. A Direct Formulation for Sparse PCA Using Semidefinite Programming. *SIAM Review*, 49(3):434–448, 2007.

[4] Nicholas J. Higham and Nataša Strabić. Anderson acceleration of the alternating projections method for computing the nearest correlation matrix. *Numerical Algorithms*, 72(4):1021–1042, 2016.

[5] Claire Mathieu and Warren Schudy. Correlation Clustering with Noisy Input. In *Proceedings of the Twenty-first Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '10, pages 712–728, Philadelphia, PA, USA, 2010. Society for Industrial and Applied Mathematics.

[6] M. Zaslavskiy, F. Bach, and J. P. Vert. A Path Following Algorithm for the Graph Matching Problem. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(12):2227–2242, Dec 2009.

[7] Anatoli Juditsky, Arkadi Nemirovski, et al. First order methods for nonsmooth convex large-scale optimization, I: general purpose methods. *Optimization for Machine Learning*, pages 121–148, 2011.

[8] Anatoli Juditsky, Arkadi Nemirovski, et al. First order methods for nonsmooth convex large-scale optimization, II: utilizing problems structure. *Optimization for Machine Learning*, pages 149–183, 2011.

[9] Yurii Nesterov. Smooth minimization of non-smooth functions. *Math. Program.*, 103(1):127–152, 2005.

[10] Martin Jaggi. Revisiting Frank-Wolfe: Projection-Free Sparse Convex Optimization. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 427–435, 2013.

[11] Heinz H. Bauschke and Jonathan M. Borwein. On Projection Algorithms for Solving Convex Feasibility Problems. *SIAM Rev.*, 38(3):367–426, September 1996.

[12] Amir Beck and Marc Teboulle. Convergence rate analysis and error bounds for projection algorithms in convex feasibility problems. *Optimization Methods and Software*, 18(4):377–394, 2003.

[13] James P. Boyle and Richard L. Dykstra. *A Method for Finding Projections onto the Intersection of Convex Sets in Hilbert Spaces*, pages 28–47. Springer New York, New York, NY, 1986.

[14] Amir Beck and Marc Teboulle. A fast dual proximal gradient algorithm for convex minimization and applications. *Operations Research Letters*, 42(1):1–6, 2014.

[15] Dimitri P. Bertsekas. Incremental proximal methods for large scale convex optimization. *Mathematical Programming*, 129(2):163, 2011.

[16] Niao He, Anatoli Juditsky, and Arkadi Nemirovski. Mirror Prox Algorithm for Multi-Term Composite Minimization and Alternating Directions. *Computational Optimization and Applications*, 61(2):275–319, 2015.

[17] Angelia Nedić. Random algorithms for convex minimization problems. *Mathematical Programming*, 129(2):225–253, 2011.

[18] M. Wang, Y. Chen, J. Liu, and Y. Gu. Random Multi-Constraint Projection: Stochastic Gradient Methods for Convex Optimization with Many Constraints. *ArXiv e-prints*, November 2015.

[19] Mengdi Wang and Dimitri P. Bertsekas. Incremental constraint projection methods for variational inequalities. *Mathematical Programming*, 150(2):321–363, 2015.

[20] A. Nedić and A. Ozdaglar. Subgradient Methods for Saddle-Point Problems. *Journal of Optimization Theory and Applications*, 142(1):205–228, 2009.

[21] Mehrdad Mahdavi, Tianbao Yang, Rong Jin, Shenghuo Zhu, and Jinfeng Yi. Stochastic Gradient Descent with Only One Projection. In *Advances in Neural Information Processing Systems 25*, pages 494–502. Curran Associates, Inc., 2012.

[22] Andrew Cotter, Maya Gupta, and Jan Pfeifer. A Light Touch for heavily constrained SGD. In *Conference on Learning Theory*, pages 729–771, 2016.

[23] Tianbao Yang, Qihang Lin, and Lijun Zhang. A Richer Theory of Convex Constrained Optimization with Reduced Projections and Improved Rates. In *Proceedings of the 34th International Conference on Machine Learning (ICML)*, pages 3901–3910, 2017.

[24] K. L. Keys, H. Zhou, and K. Lange. Proximal Distance Algorithms: Theory and Practice. *ArXiv e-prints*, April 2016.

[25] Guanghui Lan and Renato D. C. Monteiro. Iteration-complexity of first-order penalty methods for convex programming. *Mathematical Programming*, 138(1):115–139, 2013.

[26] P. Giselsson and S. Boyd. Linear Convergence and Metric Selection for Douglas-Rachford Splitting and ADMM. *IEEE Transactions on Automatic Control*, 62(2):532–544, Feb 2017.

[27] Ron Shefi and Marc Teboulle. Rate of Convergence Analysis of Decomposition Methods Based on the Proximal Method of Multipliers for Convex Minimization. *SIAM Journal on Optimization*, 24(1):269–297, 2014.

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

[29] Zirui Zhou and Anthony Man-Cho So. A unified approach to error bounds for structured convex optimization problems. *Mathematical Programming*, pages 1–40, 2017.

[30] Heinz H. Bauschke, Jonathan M. Borwein, and Wu Li. Strong conical hull intersection property, bounded linear regularity, Jameson's property (G), and error bounds in convex optimization. *Mathematical Programming*, 86(1):135–160, 1999.

[31] H. Hu. Geometric Condition Measures and Smoothness Condition Measures for Closed Convex Sets and Linear Regularity of Infinitely Many Closed Convex Sets. *Journal of Optimization Theory and Applications*, 126(2):287–308, 2005.

[32] F. L. Bauer, J. Stoer, and C. Witzgall. Absolute and Monotonic Norms. *Numer. Math.*, 3(1):257–264, December 1961.

[33] Antonin Chambolle and Thomas Pock. On the ergodic convergence rates of a first-order primal–dual algorithm. *Mathematical Programming*, 159(1):253–287, 2016.

[34] Francis R. Bach, Gert R. G. Lanckriet, and Michael I. Jordan. Multiple Kernel Learning, Conic Duality, and the SMO Algorithm. In *Proceedings of the Twenty-first International Conference on Machine Learning*, 2004.

[35] Y. LeCun and C. Cortes. The MNIST database of handwritten digits. 1998.

[36] Mark Schmidt, Nicolas Le Roux, and Francis Bach. Minimizing finite sums with the stochastic average gradient. *Mathematical Programming*, 162(1):83–112, 2017.

[37] Maurice Sion. On general minimax theorems. *Pacific J. Math.*, 8(1):171–176, 1958.

[38] Angelia Nedić and Soomin Lee. On Stochastic Subgradient Mirror-Descent Algorithm with Weighted Averaging. *SIAM Journal on Optimization*, 24(1):84–107, 2014.

9 Appendix

Before going into the proofs of the propositions and lemmas stated in the paper, we state the following proposition:

Proposition 6 *Let $\mathcal{A}$ be a nonempty closed convex set in $\mathbb{R}^n$. The distance function $d_{\mathcal{A}}$ given by*

$$d_{\mathcal{A}}(\mathbf{x}) \triangleq \inf_{\mathbf{a}\in\mathcal{A}} \|\mathbf{x}-\mathbf{a}\|,\ \mathbf{x}\in\mathbb{R}^n, \tag{16}$$

has the following properties:

1. *$d_{\mathcal{A}}$ is convex and Lipschitz continuous on $\mathbb{R}^n$ with Lipschitz constant 1.*
2. *$d_{\mathcal{A}}$ has the following representation:*

$$d_{\mathcal{A}}(\mathbf{x}) = \sup_{\mathbf{y}\in\mathbb{R}^n:\|\mathbf{y}\|\le 1} [\mathbf{x}^\top\mathbf{y} - \sigma_{\mathcal{A}}(\mathbf{y})],\ \mathbf{x}\in\mathbb{R}^n, \tag{17}$$

   where $\sigma_{\mathcal{A}}$ denotes the support function of $\mathcal{A}$.
3. *$\frac{\mathbf{x}-\mathsf{P}_{\mathcal{A}}(\mathbf{x})}{\|\mathbf{x}-\mathsf{P}_{\mathcal{A}}(\mathbf{x})\|}$ is a subgradient of $d_{\mathcal{A}}$ at $\mathbf{x}\in\mathbb{R}^n$.*

Proof: The convexity of $d_{\mathcal{A}}$ is clear from (16) as $(\mathbf{x},\mathbf{a})\mapsto\|\mathbf{x}-\mathbf{a}\|$ is jointly convex in $\mathbf{x}$ and $\mathbf{a}$.

Let $\mathbf{x},\mathbf{x}'\in\mathbb{R}$ and $\mathbf{a}\in\mathcal{A}$. By triangle inequality of the Euclidean norm, we have:

$$\|\mathbf{x}-\mathbf{a}\| \ \le\ \|\mathbf{x}-\mathbf{x}'\| + \|\mathbf{x}'-\mathbf{a}\|.$$

By taking minimum on both sides over $\mathbf{a}\in\mathcal{A}$ we get

$$d_{\mathcal{A}}(\mathbf{x}) \le \|\mathbf{x}-\mathbf{x}'\| + d_{\mathcal{A}}(\mathbf{x}')$$

Now, interchange the role of $\mathbf{x}$ and $\mathbf{x}'$ to arrive at the following:

$$\forall\mathbf{x},\mathbf{x}'\in\mathbb{R}^n:\ \ |d_{\mathcal{A}}(\mathbf{x}) - d_{\mathcal{A}}(\mathbf{x}')| \le \|\mathbf{x}-\mathbf{x}'\|. \tag{18}$$

This shows that $d_{\mathcal{A}}$ is Lipschitz continuous with Lipschitz constraint 1.

To prove the 2nd part we have for any $\mathbf{x}\in\mathbb{R}^n$:

$$\begin{aligned}
d_{\mathcal{A}}(\mathbf{x}) &= \min_{\mathbf{a}\in\mathcal{A}} \|\mathbf{x}-\mathbf{a}\| \\
&= \min_{\mathbf{a}\in\mathcal{A}} \max_{\mathbf{y}\in\mathbb{R}^n:\|\mathbf{y}\|\le 1} \mathbf{y}^\top(\mathbf{x}-\mathbf{a}) \qquad (19)\\
&= \max_{\mathbf{y}\in\mathbb{R}^n:\|\mathbf{y}\|\le 1} \min_{\mathbf{a}\in\mathcal{A}} \mathbf{y}^\top(\mathbf{x}-\mathbf{a}) \\
&= \max_{\mathbf{y}\in\mathbb{R}^n:\|\mathbf{y}\|\le 1} [\mathbf{y}^\top\mathbf{x} - \max_{\mathbf{a}\in\mathcal{A}} \mathbf{y}^\top\mathbf{a}] \\
&= \max_{\mathbf{y}\in\mathbb{R}^n:\|\mathbf{y}\|\le 1} [\mathbf{x}^\top\mathbf{y} - \sigma_{\mathcal{A}}(\mathbf{y})], \qquad (20)
\end{aligned}$$

where the 3rd equality follows from Min-Max Theorem [37] as $\{\mathbf{y}\in\mathbb{R}^n : \|\mathbf{y}\|\le 1\}$ is compact.

Let $\mathsf{P}_{\mathcal{A}}(\mathbf{x})$ denote the Euclidean projection of $\mathbf{x}$ onto $\mathcal{A}$. Then we have $d_{\mathcal{A}}(\mathbf{x}) = \|\mathbf{x}-\mathsf{P}_{\mathcal{A}}(\mathbf{x})\| = (\mathbf{x}-\mathbf{a}^*)^\top\mathbf{y}^*$, where $\mathbf{a}^* = \mathsf{P}_{\mathcal{A}}(\mathbf{x})$ and $\mathbf{y}^* = \frac{\mathbf{x}-\mathsf{P}_{\mathcal{A}}(\mathbf{x})}{\|\mathbf{x}-\mathsf{P}_{\mathcal{A}}(\mathbf{x})\|}$. Therefore, $(\mathbf{a}^*,\mathbf{y}^*)$ is a saddle-point of (19). So, $\mathbf{y}^*$ is an optimal solution of (20). Now, note that any maximizer $\mathbf{y}$ of (20) is a subgradient of $d_{\mathcal{A}}$ at $\mathbf{x}$. This completes the proof of the 3rd part of the proposition. ■

9.1 Proof of Proposition 1

Since, P is an absolute norm, it is non-decreasing in the absolute values of its components [32]. Therefore, (a) follows from the convexity of the distance functions $d_{\mathcal{C}_i}$ (part 1 of Proposition 6) and monotonicity of the norm P.

To prove part (b) we note that $\mathbf{x}\in\mathcal{C}$ iff $d_{\mathcal{C}_i}(\mathbf{x}) = 0\,\forall i$. Also, $h_P(\mathbf{x})$ is zero iff $d_{\mathcal{C}_i}(\mathbf{x}) = 0\,\forall i$ as P is a norm.

Recall that $(\mathcal{C}_1,\ldots,\mathcal{C}_m)$ is linearly regular with constant Υ. Therefore, as a consequence of (4), we have (6) with $\Upsilon_P = \Upsilon\ \max\{\|\mathbf{u}\|_\infty : P(\mathbf{u}) = 1\}$, where $\|\cdot\|_\infty$ denotes the standard ℓ_∞-norm on $\mathbb{R}^m$.

9.2 Counter Examples for Proposition 11 of [15]

Here we present counter examples to falsify the claims made by [15] in their Proposition 11. We first show that their proof of Proposition 11 does not hold always. Then we present another counter example to prove that their Proposition 11 is not true in general, specifically, when standard constraint qualification condition (2) is violated.

Let $X_1,\ldots,X_m$ be closed convex subsets of $Y\subset\mathbb{R}^n$ with nonempty intersection and $f: Y\to\mathbb{R}$ be Lipschitz continuous with Lipschitz constant L_f . We now state the incorrect claims of [15] supported by our counter examples.

Claim 1: The construction given in proof of Proposition 11 in [15] claims that the set of minima of f over $\cap_{i=1}^m X_i$ coincides with the set of minima of

$$F(\mathbf{x}) \ \triangleq\ f(\mathbf{x}) + \gamma\sum_{i=1}^m d_{X_i}(\mathbf{x}) \tag{21}$$

over Y if

$$\gamma_0 = 0,\ \forall k\ge 1:\ \gamma_k > L_f + \sum_{i=1}^{k-1}\gamma_i,\ \text{and}\ \gamma\ge\gamma_m. \tag{22}$$

Counter Example 1: We present the following counter example to show that the above claim

of [15] is not always true. Consider the following set-up:

$$\begin{aligned} n = 2,\ Y = \mathbb{R}^2,\ m = 2,\\ X_1 \ = \ \{(x,y) \in \mathbb{R}^2 \,|\, 0.1x + y \leq 1\},\\ X_2 \ = \ \{(x,y) \in \mathbb{R}^2 \,|\, 0.1x - y \leq 1\},\\ f(x,y) = -x - y,\ \ \forall (x,y) \in \mathbb{R}^2. \end{aligned}$$

Clearly, f is Lipschitz continuous with Lipschitz constant $L_f = \sqrt{2}$. Now, satisfying the conditions given in (22), we choose $\gamma_1 = 1.5$, $\gamma_2 = 3$ and $\gamma = 4$. With this F defined in (21) becomes:

$$F(x,y) = -x - y + \frac{4([0.1x + y - 1]_+ + [0.1x - y - 1]_+)}{\sqrt{1.01}},$$

where $[a]_+ = \max\{a, 0\}$ for any $a \in \mathbb{R}$.
Note that $(10, 0)$ is the only minima of f over $X_1 \cap X_2$. But, $(10, 0)$ can not be a minima of F as we have $F(20, 0) < F(10, 0)$. In fact, F does not have any minima on $\mathbb{R}^2$ as $F(x, 0) \to -\infty$ when $x \to \infty$. Hence, the set of minima of F over Y need not be the same as the set of minima of f over $\cap_{i=1}^m X_i$ even if γ satisfies the condition given in (22). Thus, the proof of Proposition 11 in [15] stands void.

We mention that X_1, X_2 in the above example possesses the following linear regularity property:

$$\forall \mathbf{x} \in \mathbb{R}^2 :\ d_X(\mathbf{x}) \ \leq \ \Upsilon \max_{1 \leq i \leq 2} d_{X_i}(\mathbf{x}), \qquad (23)$$

where $X = X_1 \cap X_2$ and $\Upsilon = 1/\sin(\tan^{-1}(0.1))$. So, for the above example one can be verify that setting $\gamma > \Upsilon L_f$ in (21) suffices for the set of minima of F over Y to coincide with that of f over $X_1 \cap X_2$. This is expected as per our proposition 2.

Claim 2: Proposition 11 in [15] claims that $\exists \bar{\gamma} > 0$ such that the set of minima of f over $\cap_{i=1}^m X_i$ coincides with the set of minima of F (as defined in (21)) over Y for all $\gamma \geq \bar{\gamma}$.

Counter Example 2: We construct the following counter example to show that the above claim of [15] is false. Consider the following set-up:

$$\begin{aligned} n = 2,\ Y = \mathbb{R}^2,\ m = 2,\\ X_1 \ = \ \{(x,y) \in \mathbb{R}^2 \,|\, y \geq x^2\},\\ X_2 \ = \ \{(x,y) \in \mathbb{R}^2 \,|\, y = 0\},\\ f(x,y) = -2x,\ \ \forall (x,y) \in \mathbb{R}^2. \end{aligned}$$

Clearly, f is Lipschitz continuous with Lipschitz constant $L_f = 2$. We note that $X_1 \cap X_2 = \{(0, 0)\}$. Therefore, $(0, 0)$ is the only minima of f over $X_1 \cap X_2$. Fix any $\gamma > 0$ in (21). Now, for all $x \in \mathbb{R}$ we have

$$F(x, x^2) = -2x + \gamma x^2.$$

Setting $x = \gamma^{-1}$ we have

$$F(\gamma^{-1}, \gamma^{-2}) = -\gamma^{-1} \ < \ F(0, 0).$$

Therefore, $(0, 0)$ is not a minima of F over $\mathbb{R}^2$ for any $\gamma > 0$. Hence, $\nexists \bar{\gamma} > 0$ such that $(0, 0)$ is a minima of F for any $\gamma \geq \bar{\gamma}$. This establishes that claim 2 is not true in general.

One can verify that $\nexists \Upsilon > 0$ such that (23) holds where X_1, X_2 are as in example 2. We mention that standard constraint qualification (SCQ) condition (2) is not satisfied in this example. Recall that SCQ, although a very mild requirement, is sufficient to ensure linear regularity property when the intersection of the closed convex sets is bounded.

9.3 Proof of Proposition 2

Recall that $f_* = \min_{\mathbf{x} \in \mathcal{C}} f(\mathbf{x})$. As f is Lipschitz continuous on $\mathcal{X}$ with Lipschitz constant L_f, we have $\forall \rho \geq L_f$:

$$\begin{aligned} & \forall \quad \mathbf{x}, \mathbf{y} \in \mathcal{X} :\ f(\mathbf{y}) - f(\mathbf{x}) \ \leq \ \rho \, \|\mathbf{y} - \mathbf{x}\|,\\ \Rightarrow \quad & \forall \quad \mathbf{x} \in \mathcal{X} :\ \inf_{\mathbf{y} \in \mathcal{C}} f(\mathbf{y}) - f(\mathbf{x}) \ \leq \ \rho \inf_{\mathbf{y} \in \mathcal{C}} \|\mathbf{y} - \mathbf{x}\|,\\ \Rightarrow \quad & \forall \quad \mathbf{x} \in \mathcal{X} :\ f_* \ \leq \ f(\mathbf{x}) + \rho \, d_{\mathcal{C}}(\mathbf{x}). \end{aligned} \qquad (24)$$

Using regularity of h_P from (6), we have $\forall \lambda \geq 0$:

$$f_*^\lambda = \inf_{\mathbf{x} \in \mathcal{X}} f(\mathbf{x}) + \lambda \, h_P(\mathbf{x}) \geq \inf_{\mathbf{x} \in \mathcal{X}} f(\mathbf{x}) + \frac{\lambda}{\Upsilon_P} \, d_{\mathcal{C}}(\mathbf{x}). \qquad (25)$$

If $\lambda \geq \Upsilon_P L_f$ then applying (24) in (25), we obtain $f_*^\lambda \geq f_*$. On the other hand, we have $f_*^\lambda \leq f_*$ for all $\lambda \geq 0$ as $\mathcal{C} \subset \mathcal{X}$ and h_P is zero on $\mathcal{C}$. Thus, $f_*^\lambda = f_*$ and a minima of f over $\mathcal{C}$ is also a minima of f^λ over $\mathcal{X}$.

To prove part [b] of the proposition it is enough to show the following: if $\mathbf{x}_\lambda^*$ is an optimal solution of (7) then $\mathbf{x}_\lambda^* \in \mathcal{C}$ when $\lambda > \Upsilon_P L_f$. Assume $\mathbf{x}_\lambda^* \notin \mathcal{C}$. So, $d_{\mathcal{C}}(\mathbf{x}_\lambda^*) > 0$. Now, using (6) and $\lambda > \Upsilon_P L_f$ we have

$$\begin{aligned} f_*^\lambda = f(\mathbf{x}_\lambda^*) + \lambda \, h_P(\mathbf{x}_\lambda^*) \quad &\geq \quad f(\mathbf{x}_\lambda^*) + \frac{\lambda}{\Upsilon_p} \, d_{\mathcal{C}}(\mathbf{x}_\lambda^*)\\ &> \quad f(\mathbf{x}_\lambda^*) + L_f \, d_{\mathcal{C}}(\mathbf{x}_\lambda^*). \end{aligned}$$

Now, applying 24 we obtain $f_*^\lambda > f_*$ which is a contradiction to the first part of the theorem. Thus, every minimizer $\mathbf{x}_\lambda^*$ of f^λ over $\mathcal{X}$ must belong to $\mathcal{C}$ when $\lambda > \Upsilon_P L_f$. This together with the fact that $f_*^\lambda = f_*$ shows that $\mathbf{x}_\lambda^*$ is also a minimizer of f over $\mathcal{C}$.

Now, we focus on part [c] of the proposition. By definition of ε-optimal solution of (7), we have

$$f(\mathbf{x}_\varepsilon) + \lambda \, h_P(\mathbf{x}_\varepsilon) \ \leq \ f_*^\lambda + \varepsilon. \qquad (26)$$

Now, applying (6) and using $f_*^\lambda = f_*$ from part [a], we get

$$f(\mathbf{x}_\varepsilon) + \frac{\lambda}{\Upsilon_P} \, d_{\mathcal{C}}(\mathbf{x}_\varepsilon) \ \leq \ f_* + \varepsilon. \qquad (27)$$

Putting $\rho = L_f$ in (24) we have

$$f(\mathbf{x}_\varepsilon) + L_f\, d_{\mathcal{C}}(\mathbf{x}_\varepsilon) \;\geq\; f_*. \tag{28}$$

Now, combining (27), (28) and using $\lambda \geq 2\Upsilon_P L_f$ we achieve

$$d_{\mathcal{C}}(\mathbf{x}_\varepsilon) \leq \frac{\varepsilon \Upsilon_P}{\lambda - \Upsilon_P L_f} \leq \frac{\varepsilon}{L_f}.$$

Note that we also have $f(\mathbf{x}_\varepsilon) - f_* \leq \varepsilon$ from (26) as h_P is always non-negative. This proves that $\mathbf{x}_\varepsilon$ is also an ε-optimal ε-feasible solution of (1). Now it remains to show that $f(\mathsf{P}_{\mathcal{C}}(\mathbf{x}_\varepsilon)) \leq f_* + \varepsilon$. This follows from Lipschitz continuity of f and (6) as shown below:

$$\begin{aligned} f(\mathsf{P}_{\mathcal{C}}(\mathbf{x}_\varepsilon)) &\leq [f(\mathsf{P}_{\mathcal{C}}(\mathbf{x}_\varepsilon)) - f(\mathbf{x}_\varepsilon)] + f(\mathbf{x}_\varepsilon) \\ &\leq L_f \,\|\mathsf{P}_{\mathcal{C}}(\mathbf{x}_\varepsilon) - \mathbf{x}_\varepsilon\| + f(\mathbf{x}_\varepsilon) \\ &= L_f\, d_{\mathcal{C}}(\mathbf{x}_\varepsilon) + f(\mathbf{x}_\varepsilon) \\ &\leq L_f \Upsilon_P\, h_P(\mathbf{x}_\varepsilon) + f(\mathbf{x}_\varepsilon) \\ &\leq \lambda\, h_P(\mathbf{x}_\varepsilon) + f(\mathbf{x}_\varepsilon) \;\leq\; f_* + \varepsilon. \end{aligned}$$

9.4 Proof of Lemma 1

Let $\mathbf{x}, \mathbf{x} \in \mathbb{R}^n$. To establish Lipschitz continuity of h_P, we have

$$\begin{aligned} & |h_P(\mathbf{x}) - h_P(\mathbf{x}')| \\ = \;& |P(d_{\mathcal{C}_1}(\mathbf{x}), \ldots, d_{\mathcal{C}_m}(\mathbf{x})) - P(d_{\mathcal{C}_1}(\mathbf{x}'), \ldots, d_{\mathcal{C}_m}(\mathbf{x}'))| \\ \leq \;& P(d_{\mathcal{C}_1}(\mathbf{x}) - d_{\mathcal{C}_1}(\mathbf{x}'), \ldots, d_{\mathcal{C}_m}(\mathbf{x}) - d_{\mathcal{C}_m}(\mathbf{x}')) \\ = \;& P(|d_{\mathcal{C}_1}(\mathbf{x}) - d_{\mathcal{C}_1}(\mathbf{x}')|, \ldots, |d_{\mathcal{C}_m}(\mathbf{x}) - d_{\mathcal{C}_m}(\mathbf{x}')|) \\ \leq \;& P(\|\mathbf{x} - \mathbf{x}'\|, \ldots, \|\mathbf{x} - \mathbf{x}'\|) \\ = \;& \|\mathbf{x} - \mathbf{x}'\|\, P(\mathbf{1}), \end{aligned}$$

where the first inequality is a result of triangle inequality of norm P and the 2nd equality is due to P being an absolute norm. Recall that an absolute norm is monotonic, that is, the norm is monotonically non-decreasing in the absolute values of its components. Using this monotonicity property of P together with (18) results in the 2nd inequality above. Thus, h_P is Lipschitz continuous on $\mathbb{R}^n$ and $P(\mathbf{1})$ is a Lipschitz constant.

Using the property that dual norm of an absolute norm is also an absolute norm, we have P_* as an absolute norm. Now, to find a subgradient we first present the following characterization of h_P using dual norm P_*:

$$\begin{aligned} h_P(\mathbf{x}) &= P(d_{\mathcal{C}_1}(\mathbf{x}), \ldots, d_{\mathcal{C}_m}(\mathbf{x})) \\ &= \max_{\mathbf{u} \in \mathbb{R}^m :\, P_*(\mathbf{u}) \leq 1} \; \sum_{i=1}^m u_i d_{\mathcal{C}_i}(\mathbf{x}) \qquad (29) \\ &= \max_{\mathbf{u} \in \mathbb{R}^m_+ :\, P_*(\mathbf{u}) \leq 1} \; \sum_{i=1}^m u_i d_{\mathcal{C}_i}(\mathbf{x}), \qquad (30) \end{aligned}$$

where the last equality follows because $d_{\mathcal{C}_i}(\mathbf{x}) \geq 0 \,\forall i$ and P_* is an absolute norm. Since distance functions are convex, (30) represents h_P as maximum of a family of convex functions. Therefore, if $\mathbf{u}^*$ is a maximizer of (29) or (30) and ξ_i denotes a subgradient of $d_{\mathcal{C}_i}$ at $\mathbf{x}$, then $\sum_{i=1}^m u_i^* \xi_i$ is a subgradient of h_P at $\mathbf{x}$. Now, part 3 of Proposition 6 says we can use $\xi_i = \frac{\mathbf{x} - \mathsf{P}_{\mathcal{C}_i}(\mathbf{x})}{\|\mathbf{x} - \mathsf{P}_{\mathcal{C}_i}(\mathbf{x})\|}$. This completes the proof.

9.5 Split-Projection subgradient (SPS) Algorithm

Algorithm 1 Split-Projection Subgradient (SPS) Algorithm to solve (1) when f is nonsmooth

Input: $\lambda > 0$, number of iterations T.
Initialization: $\mathbf{x}^{(1)} \in \mathcal{X}$.
for $t = 1$ **to** T **do**
  Get subgradient $f'(\mathbf{x}^{(t)})$ of f at $\mathbf{x}^{(t)}$.
  Get projections: $\mathsf{P}_{\mathcal{C}_i}(\mathbf{x}^{(t)}), \ldots, \mathsf{P}_{\mathcal{C}_m}(\mathbf{x}^{(t)})$.
  Compute $h'_P(\mathbf{x}^{(t)})$ using Lemma 1.
  Set $\xi^{(t)} := f'(\mathbf{x}^{(t)}) + \lambda\, h'_P(\mathbf{x}^{(t)})$.
  Choose step-size $\gamma_t > 0$.
  Update $\mathbf{x}^{(t+1)} := \mathsf{P}_{\mathcal{X}}\big(\mathbf{x}^{(t)} - \gamma_t \xi^t\big)$.
end for
Output: $\widehat{\mathbf{x}}^{(T)} \triangleq [\sum_{t=1}^T \gamma_t^{-1} \mathbf{x}^{(t)}] \,/\, [\sum_{i=1}^T \gamma_i^{-1}]$.

9.6 Proof of Proposition 3

We consider the SPS algorithm applied to problem (1) with $\lambda \geq 2\Upsilon_P L_f$. Let $D \triangleq \max_{\mathbf{x}, \mathbf{x}' \in \mathcal{X}} \|\mathbf{x} - \mathbf{x}'\|$ be the diameter of the compact set $\mathcal{X}$. Let the stepsizes be chosen as $\gamma_t = \frac{\eta}{\sqrt{t}}, 1 \leq t \leq T$, for some $\eta > 0$. Note that SPS algorithm for problem (1) is nothing but application of standard subgradient algorithm to the exact penalty based reformulation (7). Therefore, we apply the convergence rate guarantee of standard subgradient method from Corollary 2 of [38] which provides the following bound on the output $\widehat{\mathbf{x}}^{(T)}$:

$$f^\lambda(\widehat{\mathbf{x}}^{(T)}) - f_*^\lambda \;\leq\; \frac{3}{4\sqrt{T}} \left(\frac{D^2}{\eta} + \eta [L_f + \lambda P(\mathbf{1})]^2 \right), \tag{31}$$

where $[L_f + \lambda P(\mathbf{1})]$ is a Lipschitz constant of $f^\lambda \equiv f + \lambda h_P$. By minimizing the right hand side of (31) we obtain optimal value of η as $\frac{D}{L_f + \lambda P(\mathbf{1})}$. Now, substituting the optimal value of η in (31) we get:

$$f^\lambda(\widehat{\mathbf{x}}^{(T)}) - f_*^\lambda \leq \frac{3D[L_f + \lambda P(\mathbf{1})]}{2\sqrt{T}}.$$

Recall that $\lambda \geq 2\Upsilon_P L_f$. Thus, to produce an ε-optimal solution of (7) we need $O\big(\frac{L_f^2 D^2}{\varepsilon^2}\big)$ iterations of the SPS-algorithm. As per Proposition 2 such ε-optimal solutions of (7) are in fact ε-optimal ε-feasible solutions of (1). This completes the proof.

9.7 Proof of Proposition 4

For strongly convex case, we set stepsizes in the SPS algorithm as $\gamma_t = \frac{2}{\mu_f(t+1)}, 1 \le t \le T$. Recall that SPS algorithm is nothing but an instance of standard subgradient algorithm applied to (7). Now, we quote the following convergence rate guarantee of standard subgradient method from Corollary 1 of [38]:

$$f^\lambda(\widehat{\mathbf{x}}^{(T)}) - f^\lambda_* \;\le\; \frac{2}{\mu_f T}[L_f + \lambda P(\mathbf{1})]^2.$$

Also, we choose $\lambda \ge 2\Upsilon_P L_f$. Therefore, SPS algorithm produces an ε-optimal solution of (7) in $O(\frac{L_f^2}{\mu_f \varepsilon})$ iterations. Now, applying Proposition 2 we achieve the desired $O(1/\varepsilon)$ complexity for an ε-optimal ε-feasible solution of (1).

9.8 Proof of Lemma 2

Applying Proposition 6 to $\mathcal{C}_i$ we have $\forall \mathbf{x} \in \mathbb{R}^n$:

$$d_{\mathcal{C}_i}(\mathbf{x}) = \max_{\mathbf{y}_i \in \mathbb{R}^n : \|\mathbf{y}_i\| \le 1} [\mathbf{x}^\top \mathbf{y}_i - \sigma_{\mathcal{A}}(\mathbf{y}_i)], \tag{32}$$

where the maxima is achieved at a point with $\|y_i\| = 1$. Using the above characterization of $d_{\mathcal{C}_i}$ in (30) we get

$$h_P(\mathbf{x}) = \max_{\mathbf{u} \in \mathbb{R}^m_+ :\, P_*(\mathbf{u}) \le 1,\; \|\mathbf{y}_i\| = 1 \,\forall i} \sum_{i=1}^m u_i [\mathbf{x}^\top \mathbf{y}_i - \sigma_{\mathcal{C}_i}(\mathbf{y}_i)].$$

Note that $\forall\, u_i \ge 0 :\;\; u_i \sigma_{\mathcal{C}_i}(\mathbf{y}_i) = \sigma_{\mathcal{C}_i}(u_i \mathbf{y}_i)$. Therefore, making the variable transformation $u_i \mathbf{y}_i \mapsto \tilde{\mathbf{y}}_i$ we achieve the desired form:

$$h_P(\mathbf{x}) = \max_{\tilde{\mathbf{y}}_i \in \mathbb{R}^n \forall i :\, P_*(\|\tilde{\mathbf{y}}_1\|, \ldots, \|\tilde{\mathbf{y}}_m\|) \le 1} \sum_{i=1}^m [\mathbf{x}^\top \tilde{\mathbf{y}}_i - \sigma_{\mathcal{C}_i}(\tilde{\mathbf{y}}_i)].$$

9.9 Proof of Lemma 3

From (7), (8) and (9) we have

$$f^\lambda(\mathbf{x}) = \max_{\mathbf{Y} \in \mathcal{Y}^\lambda_P} \mathcal{L}(\mathbf{x}, \mathbf{Y})$$

and

$$\begin{aligned} f^\lambda_* = \min_{\mathbf{x}} f^\lambda(\mathbf{x}) \;&=\; \min_{\mathbf{x} \in \mathcal{X}} \max_{\mathbf{Y} \in \mathcal{Y}^\lambda_P} \mathcal{L}(\mathbf{x}, \mathbf{Y}) \\ &=\; \max_{\mathbf{Y} \in \mathcal{Y}^\lambda_P} \min_{\mathbf{x} \in \mathcal{X}} \mathcal{L}(\mathbf{x}, \mathbf{Y}) \\ &=\; \max_{\mathbf{Y} \in \mathcal{Y}^\lambda_P} q(\mathbf{Y}), \end{aligned} \tag{33}$$

where we define $q(\mathbf{Y}) \triangleq \min_{\mathbf{x} \in \mathcal{X}} \mathcal{L}(\mathbf{x}, \mathbf{Y})$ and swapping of min & max holds due to Min-Max Theorem [37].

We are given $(\mathbf{x}_\varepsilon, \mathbf{Y}_\varepsilon) \in \mathcal{X} \times \mathcal{Y}^\lambda_P$ such that

$$\begin{aligned} &\sup_{\mathbf{x} \in \mathcal{X},\, \mathbf{Y} \in \mathcal{Y}^\lambda_P} [\, \mathcal{L}(\mathbf{x}_\varepsilon, \mathbf{Y}) - \mathcal{L}(\mathbf{x}, \mathbf{Y}_\varepsilon)\,] \;\le\; \varepsilon \\ \Rightarrow\;\; &[\, \sup_{\mathbf{Y} \in \mathcal{Y}^\lambda_P} \mathcal{L}(\mathbf{x}_\varepsilon, \mathbf{Y}) - \inf_{\mathbf{x} \in \mathcal{X}} \mathcal{L}(\mathbf{x}, \mathbf{Y}_\varepsilon)] \;\le\; \varepsilon \\ \Rightarrow\;\; &[f^\lambda(\mathbf{x}_\varepsilon) - q(\mathbf{Y}_\varepsilon)] \;\le\; \varepsilon \\ \Rightarrow\;\; &[f^\lambda(\mathbf{x}_\varepsilon) - f^\lambda_*] + [f^\lambda_* - q(\mathbf{Y}_\varepsilon)] \;\le\; \varepsilon. \end{aligned}$$

Now, from (33) we have $q(\mathbf{Y}_\varepsilon) \le \max_{\mathbf{Y} \in \mathcal{Y}^\lambda_P} q(\mathbf{Y}) = f^\lambda_*$. Thus, we have $f^\lambda(\mathbf{x}_\varepsilon) - f^\lambda_* \le \varepsilon$. Therefore, $\mathbf{x}_\varepsilon$ is an ε-optimal solution of (7). Now, by virtue of part-c of Proposition 2 $\mathbf{x}_\varepsilon$ is an ε-optimal ε-feasible solution of (1).

9.10 Proof of Lemma 4

By definition of proximal operator, we have

$$\mathsf{Prox}_{\gamma g}(\mathbf{Y}) = \operatorname*{argmin}_{\mathbf{Y}' \in \mathcal{Y}^\lambda_P} \sum_{i=1}^m \left[\frac{1}{2\gamma} \|\mathbf{y}_i - \mathbf{y}'_i\|^2 + \sigma_{\mathcal{C}_i}(\mathbf{y}'_i) \right],$$

where $\mathcal{Y}^\lambda_P \triangleq \{\mathbf{Y} \in \otimes_{i=1}^m \mathbb{R}^n \,|\, P_*(\|\mathbf{y}_1\|, \ldots, \|\mathbf{y}_m\|) \le \lambda\}$. Utilizing monotonicity of norm P_*, we break the above minimization problem as follows:

$$\min_{\mathbf{u} \in \mathbb{R}^m_+ :\, P_*(\mathbf{u}) \le 1} \sum_{i=1}^m \left[\frac{1}{2\gamma} \|\mathbf{y}_i - \mathbf{y}^*_i\|^2 + \sigma_{\mathcal{C}_i}(\mathbf{y}^*_i) \right], \tag{34}$$

where $\mathbf{y}^*_i$ depends on $\mathbf{u} = (u_1, \ldots, u_m) \in \mathbb{R}^m_+$ as given below

$$\begin{aligned} \mathbf{y}^*_i &= \operatorname*{argmin}_{\mathbf{y}'_i \in \mathbb{R}^n : \|\mathbf{y}'_i\| \le u_i} \frac{1}{2\gamma} \|\mathbf{y}_i - \mathbf{y}'_i\|^2 + \sigma_{\mathcal{C}_i}(\mathbf{y}'_i) \\ &= \operatorname*{argmin}_{\mathbf{y}'_i \in \mathbb{R}^n : \|\mathbf{y}'_i\| \le u_i} \max_{\mathbf{x}_i \in \mathcal{C}_i} \frac{1}{2\gamma} \|\mathbf{y}_i - \mathbf{y}'_i\|^2 + \mathbf{x}_i^\top \mathbf{y}'_i \\ &= \max_{\mathbf{x}_i \in \mathcal{C}_i} \operatorname*{argmin}_{\mathbf{y}'_i \in \mathbb{R}^n : \|\mathbf{y}'_i\| \le u_i} \frac{1}{2\gamma} \|\mathbf{y}_i - \mathbf{y}'_i\|^2 + \mathbf{x}_i^\top \mathbf{y}'_i \end{aligned} \tag{35}$$

Now, for a fixed $\mathbf{x}_i$, the inner minimization w.r.t. $\mathbf{y}'_i$ is achieved at

$$\mathbf{y}'^*_i = \min\left\{1, \frac{u_i}{\|\mathbf{y}_i - \gamma \mathbf{x}_i\|}\right\} [\mathbf{y}_i - \gamma \mathbf{x}_i].$$

Substituting $\mathbf{y}'_i$ with $\mathbf{y}'^*_i$ in (35) we now find the value of $\mathbf{x}_i$ as

$$\begin{aligned} &\operatorname*{argmax}_{\mathbf{x}_i \in \mathcal{C}_i} \frac{1}{2\gamma} \|\mathbf{y}_i - \mathbf{y}'^*_i\|^2 + \mathbf{x}_i^\top \mathbf{y}^*_i \\ =\;\; &\operatorname*{argmax}_{\mathbf{x}_i \in \mathcal{C}_i} \|(\mathbf{y}_i - \gamma \mathbf{x}_i) - \mathbf{y}'^*_i\|^2 - \|(\mathbf{y}_i - \gamma \mathbf{x}_i)\|^2 \\ =\;\; &\operatorname*{argmin}_{\mathbf{x}_i \in \mathcal{C}_i} \|\mathbf{y}_i - \gamma \mathbf{x}_i\|^2 (1 - [1 - \min\{1, \frac{u_i}{\|\mathbf{y}_i - \gamma \mathbf{x}_i\|}\}]^2). \end{aligned}$$

The last minimization actually boils down to $\operatorname{argmin}_{\mathbf{x}_i \in \mathcal{C}_i} \|\mathbf{y}_i - \gamma \mathbf{x}_i\|$ which is achieved at $\mathbf{x}_i =$

$\mathsf{P}_{\mathcal{C}_i}(\gamma^{-1}\mathbf{y}_i)$. Now substituting this value of $\mathbf{x}_i$ in $\mathbf{y}_i'^*$ and defining $\hat{\mathbf{y}}_i \triangleq \mathbf{y}_i - \gamma\mathsf{P}_{\mathcal{C}_i}(\gamma^{-1}\mathbf{y}_i)$ we have

$$\mathbf{y}_i^* = \min\{u_i, \|\hat{\mathbf{y}}_i\|\}\ \frac{\hat{\mathbf{y}}_i}{\|\hat{\mathbf{y}}_i\|}.$$

This bring us to the problem of finding optimal u_i by solving (34) with y_i^* as above. Since, P_* is a monotonic norm we can equivalently solve the following:

$$\min_{\boldsymbol{\eta}\in\mathbb{R}^m_+:\, P_*(\boldsymbol{\eta})\le 1} \sum_{i=1}^m \left[\frac{1}{2\gamma}\|\mathbf{y}_i - \mathbf{y}_i^*\|^2 + \sigma_{\mathcal{C}_i}(\mathbf{y}_i^*)\right], \quad (36)$$

where $\eta_i = \min\{u_i, \|\mathbf{y}_i\|\}$, $\mathbf{y}_i^* = \frac{\eta_i}{\|\hat{\mathbf{y}}_i\|}\hat{\mathbf{y}}_i$ and $\hat{\mathbf{y}}_i = \mathbf{y}_i - \gamma\mathsf{P}_{\mathcal{C}_i}(\gamma^{-1}\mathbf{y}_i)$. Using the definition of $\mathsf{P}_{\mathcal{C}_i}(\gamma^{-1}\mathbf{y}_i)$ we find $\sigma_{\mathcal{C}_i}(\mathbf{y}_i^*) = \mathsf{P}_{\mathcal{C}_i}(\gamma^{-1}\mathbf{y}_i)^\top\mathbf{y}_i^*$ which we substitute in (36). Therefore (36) becomes

$$\begin{aligned}
&\operatorname*{argmin}_{\boldsymbol{\eta}\in\mathbb{R}^m_+:\, P_*(\boldsymbol{\eta})\le 1} \sum_{i=1}^m \left[\frac{1}{2\gamma}\|\mathbf{y}_i - \mathbf{y}_i^*\|^2 + \mathsf{P}_{\mathcal{C}_i}(\gamma^{-1}\mathbf{y}_i)^\top\mathbf{y}_i^*\right] \\
= &\operatorname*{argmin}_{\boldsymbol{\eta}\in\mathbb{R}^m_+:\, P_*(\boldsymbol{\eta})\le 1} \sum_{i=1}^m \|\hat{\mathbf{y}}_i - \mathbf{y}_i^*\|^2 \\
= &\operatorname*{argmin}_{\boldsymbol{\eta}\in\mathbb{R}^m_+:\, P_*(\boldsymbol{\eta})\le 1} \sum_{i=1}^m \left\|\hat{\mathbf{y}}_i - \frac{\eta_i}{\|\hat{\mathbf{y}}_i\|}\hat{\mathbf{y}}_i\right\|^2 \\
= &\operatorname*{argmin}_{\boldsymbol{\eta}\in\mathbb{R}^m_+:\, P_*(\boldsymbol{\eta})\le 1} \sum_{i=1}^m [\,\eta_i - \|\hat{\mathbf{y}}_i\|\,]^2. \\
= &\operatorname*{argmin}_{\boldsymbol{\eta}\in\mathbb{R}^m:\, P_*(\boldsymbol{\eta})\le 1} \sum_{i=1}^m [\,\eta_i - \|\hat{\mathbf{y}}_i\|\,]^2.
\end{aligned}$$

where the last equality holds as P_* is an absolute norm. Thus, the optimal $\boldsymbol{\eta}$ is the projection of $(\|\hat{\mathbf{y}}_1\|, \ldots, \|\hat{\mathbf{y}}_m\|)$ onto $\{\boldsymbol{\eta} \in \mathbb{R}^m : P_*(\boldsymbol{\eta}) \le 1\}$. Let $(r_1, \ldots, r_m)$ be the projection. Now, substituting the optimal value of η_i in $\mathbf{y}_i^* = \frac{\eta_i}{\|\hat{\mathbf{y}}_i\|}\hat{\mathbf{y}}_i$ we achieve the desired result.

9.11 Proof of Proposition 5

From [8] we have that iteration complexity of the Mirror Prox-a (MPa) Algorithm is $O(1/\varepsilon)$. Note that our SMP algorithm is nothing but standard MPa algorithm applied to problem (14). As (14) is the saddle-point version of the primal problem (7), SMP ãlgorithm will produce an ε-optimal solution of (7) in $O(1/\varepsilon)$ iterations. Now apply Proposition 2 which ensures that ε-optimal solution of (7) is an ε-optimal ε-feasible solution of the original problem (1).

9.12 Exact Penalty based Primal Dual (EPPD) Algorithm

Before stating the algorithm we recall the following notation: $\mathbf{Y} \triangleq (\mathbf{y}_1, \ldots, \mathbf{y}_m) \in \otimes_{i=1}^m\mathbb{R}^n$ and the operator $\mathbf{A}$ maps $\mathbf{x}$ to $(\mathbf{x}, \ldots, \mathbf{x}) \in \otimes_{i=1}^m\mathbb{R}^n$. Let $M_f > 0$ be the Lipschitz constant of the gradient of f and D be the diameter of the set $\mathcal{X}$. We set the stepsize parameters τ, γ in the algorithm as follows:

$$\gamma = \frac{\lambda}{D},\ \tau = \frac{1}{M_f + m\gamma}.$$

Algorithm 2 Exact Penalty based Primal Dual (EPPD) Algorithm to solve (1) when f is smooth

Input: $\lambda > 0$, number of iterations T.
Initialization: $\mathbf{x}^{(0)} \in \mathcal{X}$, $\mathbf{Y}^{(0)} = 0$.
Choose stepsize parameters $\tau, \gamma > 0$.
for $t = 0$ **to** $T-1$ **do**
$\mathbf{x}^{(t+1)} := \mathsf{P}_{\mathcal{X}}\left(\mathbf{x}^{(t)} - \tau[\nabla f(\mathbf{x}^{(t)}) + \sum_{i=1}^m \mathbf{y}_i^{(t)}]\right)$.
$\mathbf{Y}^{(t+1)} := \mathsf{Prox}_{\gamma g}\left(\mathbf{Y}^{(t)} + \gamma\mathbf{A}[2\mathbf{x}^{(t+1)} - \mathbf{x}^{(t)}]\right)$.
end for
Output: $\widehat{\mathbf{x}}^{(T)} \triangleq \frac{1}{T}\sum_{t=1}^T \mathbf{x}^{(t)}$.

9.13 Exact Penalty based Accelerated Primal Dual (EPAPD) Algorithm

Algorithm 3 Exact Penalty based Accelerated Primal Dual (EPAPD) Algorithm to solve (1) when f is smooth and strongly convex

Input: $\lambda > 0$, number of iterations T.
Initialization: $\mathbf{x}^{(0)} \in \mathcal{X}$, $\mathbf{Y}^{(0)} = 0, \mathbf{x}^{(-1)} = \mathbf{x}^{(0)}$.
for $t = 0$ **to** $T-1$ **do**
Choose parameters τ_t, γ_t, θ_t.
$\mathbf{Y}^{(t+1)} := \mathsf{Prox}_{\gamma g}\left(\mathbf{Y}^{(t)} + \gamma_t\mathbf{A}[\mathbf{x}^{(t)} + \theta_t(\mathbf{x}^{(t)} - \mathbf{x}^{(t-1)})]\right)$.
$\mathbf{x}^{(t+1)} := \mathsf{P}_{\mathcal{X}}\left(\mathbf{x}^{(t)} - \frac{\tau_t}{(1+\mu_f\tau_t)}[\nabla f(\mathbf{x}^{(t)}) + \sum_{i=1}^m \mathbf{y}_i^{(t+1)}]\right)$.
end for
Output: $\widehat{\mathbf{x}}^{(T)} \triangleq \frac{1}{T}\sum_{t=1}^T \mathbf{x}^{(t)}$.

Let $\mu_f > 0$ be the modulus of strong convexity of f and $M_f > 0$ be the Lipschitz constant of the gradient of f. As in [33] we choose the algorithm parameters τ_t, γ_t, θ_t based on the following recursions:

$$\begin{aligned}
\theta_0 &= 1,\ \tau_0 = \frac{1}{2M_f},\ \gamma_0 = \frac{M_f}{m}, \\
\theta_{t+1} &:= \frac{1}{\sqrt{1+\mu_f\tau_t}}, \\
\tau_{t+1} &:= \theta_{t+1}\tau_t, \\
\gamma_{t+1} &:= \gamma_t/\theta_{t+1}.
\end{aligned}$$

9.14 Split Mirror Prox (SMP) Algorithm

We recall from Lemma 4 that proximal operator of g is computed through projections onto the sets $\mathcal{C}_1, \ldots, \mathcal{C}_m$. The same projections were used to

Algorithm 4 Split Mirror Prox (SMP) Algorithm to solve (1) when f is given by (13)

Input: $\lambda > 0$, number of iterations T.
Initialization: $\mathbf{x}^{(1)} \in \mathcal{X}$, $\mathbf{z}^{(1)} \in \mathcal{Z}$, $\mathbf{Y}^{(1)} = 0$.
Choose stepsizes γ_x, γ_y, $\gamma_z > 0$.
for $t = 1$ **to** T **do**

- $\tilde{\mathbf{x}}^{(t)} := \mathsf{P}_{\mathcal{X}}\Big(\mathbf{x}^{(t)} - \gamma_x[\nabla_{\mathbf{x}}\Phi(\mathbf{x}^{(t)}, \mathbf{z}^{(t)}) + \sum_{i=1}^m \mathbf{y}_i^{(t)}]\Big)$.
- $\tilde{\mathbf{z}}^{(t)} := \mathsf{P}_{\mathcal{Z}}\big(\mathbf{z}^{(t)} + \gamma_z \nabla_{\mathbf{z}}\Phi(\mathbf{x}^{(t)}, \mathbf{z}^{(t)})\big)$.
- $\tilde{\mathbf{Y}}^{(t)} := \mathsf{Prox}_{\gamma_y g}\big(\mathbf{Y}^{(t)} + \gamma_y \mathbf{A}\mathbf{x}^{(t)}\big)$.
- $g'(\tilde{\mathbf{Y}}^{(t)}) := \Big(\mathsf{P}_{\mathcal{C}_1}(\frac{\mathbf{y}_1^{(t)}}{\gamma_y}), \ldots, \mathsf{P}_{\mathcal{C}_m}(\frac{\mathbf{y}_m^{(t)}}{\gamma_y})\Big)$.
- $\mathbf{x}^{(t+1)} := \mathsf{P}_{\mathcal{X}}\Big(\mathbf{x}^{(t)} - \gamma_x[\nabla_{\mathbf{x}}\Phi(\tilde{\mathbf{x}}^{(t)}, \tilde{\mathbf{z}}^{(t)}) + \sum_{i=1}^m \tilde{\mathbf{y}}_i^{(t)}]\Big)$.
- $\mathbf{z}^{(t+1)} := \mathsf{P}_{\mathcal{Z}}\big(\mathbf{z}^{(t)} + \gamma_z \nabla_{\mathbf{z}}\Phi(\tilde{\mathbf{x}}^{(t)}, \tilde{\mathbf{z}}^{(t)})\big)$.
- $\mathbf{Y}^{(t+1)} := \mathsf{P}_{\mathcal{Y}_P^\lambda}\big(\mathbf{Y}^{(t)} + \gamma_y[\mathbf{A}\tilde{\mathbf{x}}^{(t)} - g'(\tilde{\mathbf{Y}}^{(t)})]\big)$.

end for
Output: $\widehat{\mathbf{x}}^{(T)} \triangleq \frac{1}{T}\sum_{t=1}^T \tilde{\mathbf{x}}^{(t)}$.

construct $g'(\tilde{\mathbf{Y}}^{(t)})$ in the above algorithm. Hence, every iteration of SMP algorithm requires projecting onto each $\mathcal{C}_i$'s only once. Also, the last projection onto the set $\mathcal{Y}_P^\lambda$ can be computed as follows: $\mathsf{P}_{\mathcal{Y}_P^\lambda}(\mathbf{Y}) = (r_1\mathbf{y}_1/\|\mathbf{y}_1\|, \ldots, r_m\mathbf{y}_m/\|\mathbf{y}_m\|)$, where $(r_1, \ldots, r_m)$ is the projection of $(\|\mathbf{y}_1\|, \ldots, \|\mathbf{y}_m\|)$ onto $\{\mathbf{u} \in \mathbb{R}^m \,|\, P_*(\mathbf{u}) \leq \lambda\}$. As done in standard mirror-prox [8] we set the stepsize parameters γ_x, γ_y, γ_z based on the diameters of the sets $\mathcal{X}, \mathcal{Z}, \mathcal{Y}_P^\lambda$ and the Lipschitz constant of F.

9.15 Dealing with unknown Regularity Constant

We note from Proposition 2 that setting $\lambda \geq 2\Upsilon_P L_f$ ensures that an ε-optimal solution of (7) is also an ε-optimal ε-feasible solution of the original problem (1). Here we state an algorithmic strategy to deal with the case when the regularity constant Υ_P is not known. Proposed methods find an ε-optimal ε-feasible solution of (1) by solving (7). We fix an $\varepsilon > 0$ and consider a first-order method M for obtaining an ε-optimal solution of (7). Let the number of iterations required be $T_\lambda = C\lambda^a/\varepsilon^b$, where C, a, b are positive constants. When the regularity constant Υ_P is not known we start with $\lambda = \lambda_0$ for some $\lambda_0 > 0$ and run T_{λ_0} iterations of M. If the output after T_{λ_0} iterations satisfy the ε-feasibility then we are done; otherwise we double the value of λ and run T_λ iterations of M with the new value of λ. If we proceed in this way at one point we will have $\lambda \geq 2\Upsilon_P L_f$ and the algorithm will stop with an ε-optimal ε-feasible solution of (1). It is easy to see that the total number of iterations required to finally stop is only a constant factor times the number of iterations required if the regularity constant was known. Hence, we achieve the same complexity of $O(1/\varepsilon^b)$.