A Max-Min-Max Algorithm for Large-Scale Robust Optimization

Kai Tu[1], Zhi Chen[2], and Man-Chung Yue[3]

[1] *Shenzhen University,* kaitu_02@163.com

[2] *The Chinese University of Hong Kong,* zhi.chen@cuhk.edu.hk

[3] *The University of Hong Kong,* mcyue@hku.hk

April 8, 2024

Abstract

Robust optimization (RO) is a powerful paradigm for decision making under uncertainty. Existing algorithms for solving RO, including the reformulation approach and the cutting-plane method, do not scale well, hindering the application of RO to large-scale decision problems. In this paper, we devise a first-order algorithm for solving RO based on a novel max-min-max perspective. Our algorithm operates directly on the model functions and sets through the subgradient and projection oracles, which enables the exploitation of problem structures and is especially suitable for large-scale RO. Theoretically, we prove that the oracle complexity of our algorithm for attaining an ε-approximate optimal solution is $\mathcal{O}(\varepsilon^{-3})$ or $\mathcal{O}(\varepsilon^{-2})$, depending on the smoothness of the model functions. The algorithm and its theoretical results are then extended to RO with projection-unfriendly uncertainty sets. We also show via extensive numerical experiments that the proposed algorithm outperforms the reformulation approach, the cutting-plane method and two other recent first-order algorithms.

Keywords. (Distributionally) Robust Optimization; Decision Making under Uncertainty; First-Order Methods; Oracle Complexity; Max-Min-Max Problems.

1 Introduction

Optimization models often require the input of some instance-specific parameters, which are unfortunately uncertain in most applications. The uncertainty could come from errors in estimating or measuring the parameters. Another reason for the uncertainty could be that the parameters are intrinsically random. For example, the beamforming problem in wireless communication (Wu et al. 2017) aims at finding the optimal transmission angle and power concerning some objective (*e.g.*, least interference or largest throughput) subject to certain physical constraints. The parameters required to specify the beamforming optimization model include the transmitter and receiver antennas' geographical locations, the obstacles between them, and the spectrum of other network users. Misspecification of these parameters may lead to poor signal quality or even a breakdown of the communication network. As one of the most powerful and popular paradigms for optimization under uncertainty, robust optimization (RO) has attracted intense research in recent years and found applications across a wide range of areas such as machine learning (Singla et al. 2020), operations management (Bertsimas et al. 2023), health care (Meng et al. 2015) and finance (Gregory et al. 2011), to name a few.

To set the scene, consider the following nominal optimization problem:

$$\begin{array}{rl} \min & f_0(\boldsymbol{x}) \\ \text{s.t.} & g_m(\boldsymbol{x}, \boldsymbol{z}_m) \leq 0 \quad \forall m \in [M] \\ & \boldsymbol{x} \in \mathcal{X}, \end{array}$$

where f_0 is the objective function, $g_1, \ldots, g_M$ are the constraint functions, $\boldsymbol{x} \in \mathbb{R}^N$ is the decision vector, the set $\mathcal{X} \subseteq \mathbb{R}^N$ models further constraints on $\boldsymbol{x}$, and $\boldsymbol{z}_1 \ldots, \boldsymbol{z}_M \in \mathbb{R}^{J_m}$ are the parameters specifying the optimization model. In the face of uncertainty, RO postulates that each parameter $\boldsymbol{z}_m$ resides in a subset $\mathcal{Z}_m \subseteq \mathbb{R}^{J_m}$—called the uncertainty set—that represents the modeler's belief about the possible range of the uncertain parameter $\boldsymbol{z}_m$. RO then takes a pessimistic point of view: whatever decision is chosen, the worst parameters over the uncertainty sets will be realized accordingly. More precisely, RO prescribes choosing the decision as an optimal solution to the problem

$$\begin{aligned} \min \quad & f_0(\boldsymbol{x}) \\ \text{s.t.} \quad & \max_{\boldsymbol{z}_m \in \mathcal{Z}_m} g_m(\boldsymbol{x}, \boldsymbol{z}_m) \le 0 \quad \forall m \in [M] \\ & \boldsymbol{x} \in \mathcal{X}. \end{aligned} \tag{Robust}$$

Despite the wide applicability, computational approaches for solving RO are to some extent limited and do not meet the needs of modern RO users who often require to solve high-dimensional non-linear and/or non-smooth RO problems. The difficulty stems primarily from the embedded optimization problems $\max_{\boldsymbol{z}_m \in \mathcal{Z}_m} g_m(\boldsymbol{x}, \boldsymbol{z}_m)$, rendering standard optimization algorithms non-viable.

Currently, there are two dominant approaches: the reformulation approach (see, *e.g.*, Ben-Tal and Nemirovski 1998) and the cutting-plane method (see, *e.g.*, Mutapcic and Boyd 2009). Roughly speaking, the reformulation approach solves the Robust problem by converting it into a deterministic reformulation—an equivalent optimization problem without embedded optimization problems. Here, "deterministic" refers to the uncertainty-free nature of the reformulation. This is often achieved by either solving the embedded optimization problem analytically or replacing it with its dual problem. Since the deterministic reformulation is a standard optimization problem, it can be solved by many sophisticated off-the-shelf optimization solvers such as CPLEX, Gurobi and MOSEK. The reformulation approach works for a large and useful class of Robust problems wherein the constraint functions $g_1, \ldots, g_M$ and the uncertainty sets $\mathcal{Z}_1, \ldots, \mathcal{Z}_M$ possess certain special structures; see Ben-Tal and Nemirovski (1998), Ben-Tal et al. (2009).

The cutting-plane method is essentially Kelley's cutting-plane method (Kelley 1960) specialized to RO. It is an iterative algorithm alternating between two steps: the *optimization step* and the *pessimization step*. Given finite subsets $\widehat{\mathcal{Z}}_m \subseteq \mathcal{Z}_m$, $m \in [M]$, the optimization step solves the following approximation of Robust to find a new $\boldsymbol{x}$:

$$\begin{aligned} \min \quad & f_0(\boldsymbol{x}) \\ \text{s.t.} \quad & \max_{\boldsymbol{z}_m \in \widehat{\mathcal{Z}}_m} g_m(\boldsymbol{x}, \boldsymbol{z}_m) \le 0 \quad \forall m \in [M] \\ & \boldsymbol{x} \in \mathcal{X}. \end{aligned}$$

Such an approximation has only a finite number of constraints and thus can be solved readily. The pessimization step computes a maximizer $\hat{\boldsymbol{z}}_m$ for each embedded problem $\max_{\boldsymbol{z}_m \in \mathcal{Z}_m} g_m(\boldsymbol{x}, \boldsymbol{z}_m)$ and if the maximum value is positive, expands the approximate uncertainty set $\widehat{\mathcal{Z}}_m$ by setting $\widehat{\mathcal{Z}}_m \leftarrow \widehat{\mathcal{Z}}_m \cup \{\hat{\boldsymbol{z}}_m\}$. The algorithm alternates between these two steps until reaching a certain stopping criterion.

There are several weaknesses of existing computational approaches to RO, which significantly hinder its development and applications. First, off-the-shelf optimization solvers typically applied to solve the deterministic reformulations of RO are mainly based on interior-point methods (Nesterov and Nemirovskii 1994). As evidenced by many numerical studies (Toh and Yun 2010, Liu et al. 2023), interior-point methods have relatively worse scalability when compared to some other classes of optimization algorithms, such as first-order and second-order methods, and thus are unsuitable for large-scale problems. Moreover, solvers may ignore useful structures of the problem (*e.g.*, sparsity or low-rank-ness of matrices), which, if suitably exploited, can substantially improve the computational speed.

Second, as mentioned earlier, one way to eliminate the embedded maximization in Robust is to dualize it into a minimization problem. The dualization process possibly introduces a large number of extra variables and/or constraints, resulting in a high-dimensional and/or highly constrained reformulation that is hard to solve even for well-developed solvers.

Third, for many cases of g_m and $\mathcal{Z}_m$, the reformulation technique often lifts Robust to a more difficult and general class of optimization problems. As an example, suppose that the constraint function is $g_m(\boldsymbol{x}, \boldsymbol{z}_m) = \boldsymbol{x}^\top \boldsymbol{A}^\top(\boldsymbol{z}_m)\boldsymbol{A}(\boldsymbol{z}_m)\boldsymbol{x} + \boldsymbol{b}^\top(\boldsymbol{z}_m)\boldsymbol{x} + c(\boldsymbol{z}_m)$, with $\boldsymbol{A}(\boldsymbol{z}_m) \in \mathbb{R}^{L\times N}$, $\boldsymbol{b}(\boldsymbol{z}_m) \in \mathbb{R}^N$ and $c(\boldsymbol{z}_m) \in \mathbb{R}$ being affine functions in $\boldsymbol{z}_m$, and the uncertainty set $\mathcal{Z}_m$ is a unit ball in $\mathbb{R}^{J_m}$. In this case, the function g_m is quadratic in both $\boldsymbol{x}$ and $\boldsymbol{z}_m$, and the uncertainty set $\mathcal{Z}_m$ is defined by a quadratic inequality. Its reformulation, however, is a semidefinite programming problem (Ben-Tal and Nemirovski 1998). The quadratic nature of the robust constraint is disregarded.

Fourth, it is well-known that the cutting-plane method may perform poorly in both theory and practice (Mitchell 2009) due to instability. In particular, its iteration complexity (*i.e.*, the number of iterations required to achieve an ε-optimal solution) is $(1 + \mathcal{O}(\varepsilon^{-1}))^N$, exponential in the dimension (Mutapcic and Boyd 2009, Section 5.2). Therefore, its performance on RO problems could potentially be equally bad. Moreover, it has been shown in a comprehensive computational study (Bertsimas et al. 2016) that the cutting-plane method performs on par with or even worse than the reformulation approach.

Motivated by the above discussions, we aim to develop specialized iterative algorithms for efficiently solving large-scale Robust problems. The idea of specialized iterative algorithms for RO is not entirely new and has been recently pursued. Using tools from online convex optimization (Hazan 2022), Ben-Tal et al. (2015b) developed an iterative algorithm with an iteration complexity $\mathcal{O}(\varepsilon^{-2})$ for solving RO, each iteration of which requires an optimization step similar to that in the cutting-plane method. Extending the online convex optimization idea, significant improvements have been obtained in the papers Ho-Nguyen and Kılınç-Karzan 2018, 2019, where the authors developed a first-order method (each iteration requires only first-order updates but neither the optimization nor pessimization step) with an iteration complexity of $\mathcal{O}(\varepsilon^{-1}\log\frac{1}{\varepsilon})$. One drawback of this algorithm is that it requires a binary search of the optimal value, which incurs extra computational overhead. In the very recent work Postek and Shtern (2021), another first-order method, named SGSP, has been developed based on perspective transformations (see Boyd and Vandenberghe 2004 for reference). Its iteration complexity is $\mathcal{O}(\varepsilon^{-2})$, with respect to more complicated oracles than the ones assumed in Ho-Nguyen and Kılınç-Karzan 2018, 2019 and this paper tough.

The point of departure of our work is the following max-min-max problem:

$$\max_{\boldsymbol{\lambda}\geq\mathbf{0}}\ \min_{\boldsymbol{x}\in\mathcal{X}}\ \max_{\boldsymbol{z}\in\mathcal{Z}}\ \mathcal{K}(\boldsymbol{\lambda},\boldsymbol{x},\boldsymbol{z}), \tag{Max-Min-Max}$$

where $\mathcal{K}(\boldsymbol{\lambda},\boldsymbol{x},\boldsymbol{z}) = f_0(\boldsymbol{x}) + \sum_{m\in[M]}\lambda_m g_m(\boldsymbol{x},\boldsymbol{z}_m)$ and $\mathcal{Z} = \mathcal{Z}_1\times\cdots\times\mathcal{Z}_M$. This is essentially the Lagrangian dual of the Robust problem and therefore equivalent to it under mild assumptions (see Proposition 2 below). The advantage of our framework is that neither complicated reformulation (as in the reformulation approach) nor perspective transformation (as in Postek and Shtern 2021) is needed. Furthermore, the algorithm will operate directly on the functions $f_0, g_1, \ldots, g_M$ as well as the sets $\mathcal{X}, \mathcal{Z}_1, \ldots, \mathcal{Z}_M$ through their gradient and projection oracles, respectively. The useful structures and theoretical properties of the functions and sets are all preserved and can be easily exploited in algorithmic design.

However, designing and theoretically analyzing max-min-max algorithms are substantially more difficult than those for max-min problems. Our contributions are as follows.

- We devise a first-order method, called ProM^3, for solving the Max-Min-Max problem (hence, the Robust problem) that utilizes the structure of $\mathcal{K}(\boldsymbol{\lambda}, \boldsymbol{x}, \boldsymbol{z})$ and operates directly on the constituent functions and sets. Our design views Max-Min-Max as a max-min problem
$$\max_{\boldsymbol{\lambda} \geq \mathbf{0}} \min_{\boldsymbol{x} \in \mathcal{X}} \mathcal{L}(\boldsymbol{\lambda}, \boldsymbol{x}), \tag{1}$$
where $\mathcal{L}(\boldsymbol{\lambda}, \boldsymbol{x}) = \max_{\boldsymbol{z} \in \mathcal{Z}} \mathcal{K}(\boldsymbol{\lambda}, \boldsymbol{x}, \boldsymbol{z})$ is the Lagrangian function, and adopts the framework of alternating proximal algorithm for max-min problems. We call problem (1) the *outer saddle-point problem*. Nevertheless, since the objective function $\mathcal{L}(\boldsymbol{\lambda}, \boldsymbol{x})$ in the outer saddle-point problem is itself a maximum value, efficiently updating $\boldsymbol{\lambda}$ and $\boldsymbol{x}$ as prescribed by the standard alternating proximal algorithm is non-trivial. Our algorithm therefore requires extra ideas. In particular, the updating step for $\boldsymbol{x}$ turns out to be a strongly-convex-concave min-max problem with a non-linear coupling term, which we call the *inner saddle-point problem*. A customized algorithm for the inner saddle-point problem is also developed. Combining the algorithms proposed for the outer and inner saddle-point problems leads to our algorithm ProM^3 for solving RO.

- We prove that under similar conditions as in Ho-Nguyen and Kılınç-Karzan (2018, 2019), Postek and Shtern (2021), the proposed algorithms for both the outer and inner saddle-point problems enjoy a sublinear convergence rate. Since the per-iteration cost of different algorithms may vary, the oracle complexity—the number of calls to the projection and subgradient oracles—for achieving an ε-optimal solution would be a more faithful measure of computational efficiency. Based on the convergence analysis of our algorithms for the outer and inner saddle-point problems, we prove that our algorithm ProM^3 enjoys the oracle complexity of $\mathcal{O}(\varepsilon^{-3})$, and the oracle complexity can be strengthened to $\mathcal{O}(\varepsilon^{-2})$ if the functions f and $g_1, \ldots, g_M$ are all smooth.

- Third, we extend our algorithm ProM^3 and convergence analysis to RO problems where the uncertainty sets $\mathcal{Z}_1, \ldots, \mathcal{Z}_M$ take certain intersection form and do not admit easy projection. Such a setting is particularly useful in distributionally robust optimization, where the uncertainty $\boldsymbol{z}_m$ is a probability vector lying in the intersection of the probability simplex and a ball defined by some norm, such as those based on the popular (type-∞) Wasserstein distance (Mohajerin Esfahani and Kuhn 2018, Xie 2020, Bertsimas et al. 2022, Gao and Kleywegt 2023) or the support space is a finite set (Ben-Tal et al. 2013, Wiesemann et al. 2014).

We conclude the introduction with a few remarks. First, although the oracle complexity of SGSP in Postek and Shtern (2021) is $\mathcal{O}(\varepsilon^{-2})$, as mentioned above, it relies on oracles that are generally more complicated than those assumed in this paper. Thus, the complexity $\mathcal{O}(\varepsilon^{-2})$ or $\mathcal{O}(\varepsilon^{-3})$ of ProM^3 should not be directly compared to the complexity $\mathcal{O}(\varepsilon^{-2})$ of SGSP. Second, contrary to saddle-point problems that have received intense research recently, the literature on max-min-max problems, as pointed out by Polak and Royset 2003, is much scarcer. Therefore, our work could be of independent interest to researchers working on max-min-max problems. Third, to the best of our knowledge, strongly-convex-concave min-max problems with a non-linear coupling term have not been thoroughly investigated yet. Our proposed algorithm for the inner saddle-point problem and its convergence analysis partially fill this gap in the fast-growing literature on saddle-point problems.

2 Proximal Max-Min-Max Algorithm (ProM^3)

We develop a first-order algorithm for solving the Max-Min-Max problem (thus, the Robust problem), called the proximal max-min-max algorithm (ProM^3). Our line of attack to Max-Min-Max is to view it as two layers of saddle-point problems: the outer saddle-point problem is the max-min problem (1), whereas the inner saddle-point problem is a step towards solving the outer problem in our algorithmic framework. For simplicity, we call the algorithms for solving the outer and inner saddle-point problems the outer and inner algorithms, respectively. The proposed ProM^3 for solving Max-Min-Max is obtained by combining the outer and inner algorithms.

2.1 Outer Algorithm

To describe the outer algorithm, we re-state the outer saddle-point problem (1) as

$$\max_{\boldsymbol{\lambda} \geq \mathbf{0}} \min_{\boldsymbol{x} \in \mathcal{X}} \left\{ f_0(\boldsymbol{x}) + \boldsymbol{\lambda}^\top \boldsymbol{f}(\boldsymbol{x}) \right\}, \tag{\textsc{Outer}}$$

where $\boldsymbol{f}(\boldsymbol{x}) = (f_1(\boldsymbol{x}), \ldots, f_M(\boldsymbol{x}))$ and $f_m(\boldsymbol{x}) = \max_{\boldsymbol{z}_m \in \mathcal{Z}_m} g_m(\boldsymbol{x}, \boldsymbol{z}_m)$ for each $m \in [M]$. The general algorithmic framework we adopt for the outer saddle-point problem, Outer, is an alternating proximal update scheme:

$$\begin{cases} \boldsymbol{\lambda}^{k+1} = \underset{\boldsymbol{\lambda} \geq \mathbf{0}}{\operatorname{argmax}} \ \boldsymbol{\lambda}^\top \boldsymbol{f}(\boldsymbol{x}^k) - \dfrac{1}{2\beta} \|\boldsymbol{\lambda} - \boldsymbol{\lambda}^k\|_2^2 = [\boldsymbol{\lambda}^k + \beta \boldsymbol{f}(\boldsymbol{x}^k)]_+, \\ \boldsymbol{x}^{k+1} = \underset{\boldsymbol{x} \in \mathcal{X}}{\operatorname{argmin}} \ f_0(\boldsymbol{x}) + (\boldsymbol{\lambda}^{k+1})^\top \boldsymbol{f}(\boldsymbol{x}) + \dfrac{1}{2\alpha} \|\boldsymbol{x} - \boldsymbol{x}^k\|_2^2, \end{cases} \tag{2}$$

where $\alpha, \beta > 0$ are step sizes and $[a]_+ = \max\{a, 0\}$ for any $a \in \mathbb{R}$. It is well-known that scheme (2) is generally divergent. A correction for the scheme is proposed in Chambolle and Pock (2011), which has a modified $\boldsymbol{\lambda}$-update but keeps the $\boldsymbol{x}$-update unchanged. When specialized to our Outer problem, the modified $\boldsymbol{\lambda}$-update reads

$$\boldsymbol{\lambda}^{k+1} = \underset{\boldsymbol{\lambda} \geq \mathbf{0}}{\operatorname{argmax}} \ \boldsymbol{\lambda}^\top (2\boldsymbol{f}(\boldsymbol{x}^k) - \boldsymbol{f}(\boldsymbol{x}^{k-1})) - \frac{1}{2\beta} \|\boldsymbol{\lambda} - \boldsymbol{\lambda}^k\|_2^2 = [\boldsymbol{\lambda}^k + \beta(2\boldsymbol{f}(\boldsymbol{x}^k) - \boldsymbol{f}(\boldsymbol{x}^{k-1}))]_+.$$

Unfortunately, even the corrected scheme would not work in our situation, at least not efficiently. The reason is that each component $f_m(\boldsymbol{x}^k)$ of the vector $\boldsymbol{f}(\boldsymbol{x}^k)$ is a partial maximum of $g_m(\boldsymbol{x}^k, \boldsymbol{z}_m)$ with respect to its second argument $\boldsymbol{z}_m$. This prohibits efficient evaluation of f_m or its gradient. We circumvent this issue by approximating $\boldsymbol{f}$ in both $\boldsymbol{\lambda}$- and $\boldsymbol{x}$-updates. Specifically, to approximate the $\boldsymbol{\lambda}$-update, we replace $\boldsymbol{f}(\boldsymbol{x}^k)$ by

$$\boldsymbol{g}(\boldsymbol{x}^k, \boldsymbol{z}^k) = (g_1(\boldsymbol{x}^k, \boldsymbol{z}_1^k), \ldots, g_M(\boldsymbol{x}^k, \boldsymbol{z}_M^k)),$$

where each $\boldsymbol{z}_m^k$ is an approximate maximizer of $g_m(\boldsymbol{x}^k, \cdot)$ over $\mathcal{Z}_m$ satisfying the condition

$$f_m(\boldsymbol{x}^k) = \max_{\boldsymbol{z}_m \in \mathcal{Z}_m} g_m(\boldsymbol{x}^k,\ \boldsymbol{z}_m) \leq g_m(\boldsymbol{x}^k,\ \boldsymbol{z}_m^k) + \theta$$

for some prescribed $\theta > 0$.

For the $\boldsymbol{x}$-update, by noting that it is equivalent to the min-max problem

$$\min_{\boldsymbol{x}\in\mathcal{X}} \max_{\boldsymbol{z}\in\mathcal{Z}} \left\{ f_0(\boldsymbol{x}) + \sum_{m\in[M]} \lambda_m^{k+1} g_m(\boldsymbol{x}, \boldsymbol{z}_m) + \frac{1}{2\alpha}\|\boldsymbol{x} - \boldsymbol{x}^k\|_2^2 \right\},$$

we invoke a min-max algorithm to solve it to a custom-made stopping condition below.

Definition 1 (Strong Approximate Saddle Point)**.** *Consider a function $\mathcal{F} : \mathcal{U} \times \mathcal{V} \to \mathbb{R}$ such that $\mathcal{F}(\cdot, \boldsymbol{v})$ is σ-strongly convex on $\mathcal{U}$ for any $\boldsymbol{v} \in \mathcal{V}$, where $\sigma > 0$ is some constant independent of $\boldsymbol{v}$, and $\mathcal{F}(\boldsymbol{u}, \cdot)$ is concave on $\mathcal{V}$ for any $\boldsymbol{u} \in \mathcal{U}$. For any $\nu > 0$, a pair $(\tilde{\boldsymbol{u}}, \tilde{\boldsymbol{v}}) \in \mathcal{U} \times \mathcal{V}$ is said to be a strong ν-approximate saddle point of $\mathcal{F}$ if*

$$\mathcal{F}(\tilde{\boldsymbol{u}}, \boldsymbol{v}) \leq \mathcal{F}(\boldsymbol{u}, \tilde{\boldsymbol{v}}) - \frac{\sigma}{2}\|\boldsymbol{u} - \tilde{\boldsymbol{u}}\|_2^2 + \nu \quad \forall(\boldsymbol{u}, \boldsymbol{v}) \in \mathcal{U} \times \mathcal{V}.$$

Definition 1 is stronger than the standard notion of approximate saddle point concept, $\mathcal{F}(\tilde{\boldsymbol{u}}, \boldsymbol{v}) - \mathcal{F}(\boldsymbol{u}, \tilde{\boldsymbol{v}}) \leq \nu$. The following proposition guarantees the existence of strong approximate saddle points under mild assumptions.

Proposition 1. *Let $\mathcal{U}$ and $\mathcal{V}$ be non-empty compact convex sets and $\mathcal{F} : \mathcal{U} \times \mathcal{V} \to \mathbb{R}$ be a function such that $\mathcal{F}(\cdot, \boldsymbol{v})$ is σ-strongly convex on $\mathcal{U}$ for any $\boldsymbol{v} \in \mathcal{V}$, where $\sigma > 0$ is some constant independent of $\boldsymbol{v}$, and $\mathcal{F}(\boldsymbol{u}, \cdot)$ is concave on $\mathcal{V}$ for any $\boldsymbol{u} \in \mathcal{U}$. Then, for any $\nu > 0$, $\mathcal{F}$ has a strong ν-approximate saddle point.*

The outer algorithm is formally presented in Algorithm 1, and its convergence analysis will be presented in Section 3.1. It should be pointed out that although the alternating proximal algorithm for saddle-point problems has been studied, the convergence rate for its outer algorithm does not readily follow from existing results but requires certain new ideas; see the discussion after Theorem 1 for details.

Here, we offer an observation that can improve the practical performance of the outer algorithm. In the $\boldsymbol{\lambda}$-update, we do not necessarily need to compute the approximate maximizer $\boldsymbol{z}_m^k$ for every $m \in [M]$. Indeed, if $\lambda_m^k > 0$, then by definition, $\tilde{\boldsymbol{z}}_m^k$ approximately maximizes the function $g_m(\boldsymbol{x}^k, \cdot)$ over $\mathcal{Z}_m$ and hence, with a careful choice of θ and ν, is precisely the $\boldsymbol{z}_m^k$ that we need to compute at the beginning of the next iteration.

```
Algorithm 1: Outer Algorithm.
  Input : $K \geq 1$, $\theta > 0$, $\nu > 0$, $\alpha > 0$, $\beta > 0$, $\boldsymbol{\lambda}^0 = \mathbf{0}$ and $\boldsymbol{x}^0 \in \mathcal{X}$.
  for $k = 0, 1, \ldots, K-1$ do
      ($\boldsymbol{\lambda}$-update) For $m \in [M]$, find $\boldsymbol{z}_m^k \in \mathcal{Z}_m$ satisfying $f_m(\boldsymbol{x}^k) - g_m(\boldsymbol{x}^k,\ \boldsymbol{z}_m^k) \leq \theta$. Set
          $\boldsymbol{\lambda}^{k+1} = [\boldsymbol{\lambda}^k + \beta(2\,\boldsymbol{g}(\boldsymbol{x}^k, \boldsymbol{z}^k) - \boldsymbol{g}(\boldsymbol{x}^{k-1}, \boldsymbol{z}^{k-1}))]_+,$
        where $\boldsymbol{x}^{-1} = \boldsymbol{x}^0$ and $\boldsymbol{z}^{-1} = \boldsymbol{z}^0$ for the 0-th iteration.
      ($\boldsymbol{x}$-update) Compute a strong $\nu$-approximate saddle point $(\boldsymbol{x}^{k+1}, \tilde{\boldsymbol{z}}^{k+1}) \in \mathcal{X} \times \mathcal{Z}$
        of the problem
          $\min_{\boldsymbol{x} \in \mathcal{X}} \max_{\boldsymbol{z} \in \mathcal{Z}} \left\{ f_0(\boldsymbol{x}) + \sum_{m \in [M]} \lambda_m^{k+1} g_m(\boldsymbol{x}, \boldsymbol{z}_m) + \frac{1}{2\alpha} \|\boldsymbol{x} - \boldsymbol{x}^k\|_2^2 \right\}.$    (Inner)
  end
  Output: $\bar{\boldsymbol{x}}^K = \frac{1}{K} \sum_{k \in [K]} \boldsymbol{x}^k$.
```

2.2 Inner Algorithm

The $\boldsymbol{x}$-update step in the outer algorithm (see step 3 of Algorithm 1) is the inner saddle-point problem, Inner. A distinctive property of Inner is a non-linear and non-smooth coupling term between the two variables $\boldsymbol{x}$ and $\boldsymbol{z}$. This should be contrasted with the relatively more common assumption that the coupling term is bilinear, *i.e.*, $\boldsymbol{x}^\top \boldsymbol{Q} \boldsymbol{z}$ for some matrix $\boldsymbol{Q}$. Another feature of the Inner problem is that its objective function is strongly convex in $\boldsymbol{x}$. Saddle-point problems of this specific form have not been well studied.

Due to the generality of the coupling term, algorithmic options are limited. We adopt the following variant of the subgradient ascent descent algorithm, which is known to enjoy a better convergence rate in the smooth case:

$$\boldsymbol{z}_{t+1} = \mathrm{Proj}_{\mathcal{Z}} \left(\boldsymbol{z}_t - \delta(2\boldsymbol{\zeta}_t - \boldsymbol{\zeta}_{t-1}) \right),$$

where $\boldsymbol{\zeta}_t \in \partial_{\boldsymbol{z}}(-f_0 - (\boldsymbol{\lambda}^{k+1})^\top \boldsymbol{g})(\boldsymbol{x}_t, \boldsymbol{z}_t)$ and $\mathrm{Proj}_{\mathcal{U}}(\cdot)$ denotes the projection onto a close convex set $\mathcal{U}$. As the maximization over $\boldsymbol{z}$ in the Inner problem is decomposable, the modified subgradient step can be executed by updating each $\boldsymbol{z}_m$ separately: that is, for each $m \in [M]$,

$$\boldsymbol{z}_{t+1,m} = \mathrm{Proj}_{\mathcal{Z}_m} \left(\boldsymbol{z}_{t,m} - \delta\, \lambda_m^{k+1} \, (2\boldsymbol{\zeta}_{t,m} - \boldsymbol{\zeta}_{t-1,m}) \right),$$

Algorithm 2: Inner Algorithm.

```
Input  : $T \geq 1$, $\delta > 0$, $\gamma > 0$, $\boldsymbol{x}_0 \in \mathcal{X}$, $\boldsymbol{z}_0 \in \mathcal{Z}$, $\alpha > 0$, $\boldsymbol{x}^k \in \mathcal{X}$ and $\boldsymbol{\lambda}^{k+1} \geq \mathbf{0}$.
for $t = 0, \dots, T-1$ do
    ($\boldsymbol{z}$-update) For $m \in [M]$, compute $\boldsymbol{\zeta}_{t,m} \in \partial_{\boldsymbol{z}_m}(-g_m)(\boldsymbol{x}_t, \boldsymbol{z}_{t,m})$. Set
        $$\boldsymbol{z}_{t+1,m} = \operatorname{Proj}_{\mathcal{Z}_m}\left(\boldsymbol{z}_{t,m} - \delta\, \lambda_m^{k+1}\, (2\boldsymbol{\zeta}_{t,m} - \boldsymbol{\zeta}_{t-1,m})\right),$$
     where $\boldsymbol{\zeta}_{-1,m} = \boldsymbol{\zeta}_{0,m}$ for the 0-th iteration.
    ($\boldsymbol{x}$-update) For $m \in [M]$, compute $\boldsymbol{\xi}_{t,0} \in \partial f_0(\boldsymbol{x}_t)$ and $\boldsymbol{\xi}_{t,m} \in \partial_{\boldsymbol{x}} g_m(\boldsymbol{x}_t, \boldsymbol{z}_{t+1,m})$. Set
     $\boldsymbol{\xi}_t = \boldsymbol{\xi}_{t,0} + \lambda_1^{k+1}\, \boldsymbol{\xi}_{t,1} + \cdots + \lambda_M^{k+1}\, \boldsymbol{\xi}_{t,M}$ and
        $$\boldsymbol{x}_{t+1} = \operatorname{Proj}_{\mathcal{X}}\left(\frac{\alpha\gamma}{\alpha+\gamma}\left(\frac{1}{\alpha}\boldsymbol{x}^k + \frac{1}{\gamma}\boldsymbol{x}_t - \boldsymbol{\xi}_t\right)\right).$$
end
Output: $\bar{\boldsymbol{x}}_T = \frac{1}{T}\sum_{t\in[T]} \boldsymbol{x}_t$ and $\bar{\boldsymbol{z}}_T = \frac{1}{T}\sum_{t\in[T]} \boldsymbol{z}_t$.
```

where $\boldsymbol{\zeta}_{t,m} \in \partial_{\boldsymbol{z}_m}(-g_m)(\boldsymbol{x}_t, \boldsymbol{z}_{t,m})$.

For the $\boldsymbol{x}$-update (the subgradient descent step), we slightly tweak the standard subgradient ascent descent framework to exploit the strong convexity. Specifically, in the $\boldsymbol{x}$-update, we linearize only the non-smooth part but retain the strongly convex quadratic term:

$$\begin{aligned}
\boldsymbol{x}_{t+1} &= \operatorname*{argmin}_{\boldsymbol{x}\in\mathcal{X}} \left\{\boldsymbol{\xi}_t^\top(\boldsymbol{x} - \boldsymbol{x}_t) + \frac{1}{2\alpha}\|\boldsymbol{x} - \boldsymbol{x}^{k-1}\|_2^2 + \frac{1}{2\gamma}\|\boldsymbol{x} - \boldsymbol{x}_t\|_2^2\right\} \\
&= \operatorname{Proj}_{\mathcal{X}}\left(\frac{\alpha\gamma}{\alpha+\gamma}\left(\frac{1}{\alpha}\boldsymbol{x}^{k-1} + \frac{1}{\gamma}\boldsymbol{x}_t - \boldsymbol{\xi}_t\right)\right),
\end{aligned}$$

where $\boldsymbol{\xi}_t \in \partial_{\boldsymbol{x}}(f_0(\boldsymbol{x}_t) + (\boldsymbol{\lambda}^{k+1})^\top \boldsymbol{g}(\boldsymbol{x}_t, \boldsymbol{z}_{t+1}))$. This tweak allows for a larger step size γ and improves practical performance. Note that by the subdifferential sum rule, the desired subgradient $\boldsymbol{\xi}_t$ can be obtained by $\boldsymbol{\xi}_t = \boldsymbol{\xi}_{t,0} + \lambda_1^{k+1}\, \boldsymbol{\xi}_{t,1} + \cdots + \lambda_M^{k+1}\, \boldsymbol{\xi}_{t,M}$, where $\boldsymbol{\xi}_{t,0} \in \partial f_0(\boldsymbol{x}_t)$ and $\boldsymbol{\xi}_{t,m} \in \partial_{\boldsymbol{x}} g_m(\boldsymbol{x}_t, \boldsymbol{z}_{t+1,m})$ for $m \in [M]$. The inner algorithm is formally presented in Algorithm 2, and its convergence analysis will be presented in Section 3.2.

Although the inner algorithm and its theoretical results are developed and presented in relation to Inner, they are actually applicable to general saddle-point problems of the form

$$\min_{\boldsymbol{u}\in\mathcal{U}} \max_{\boldsymbol{v}\in\mathcal{V}} \mathcal{F}(\boldsymbol{u}, \boldsymbol{v}),$$

where the objective function $\mathcal{F}(\boldsymbol{u}, \boldsymbol{v})$ is strongly convex in $\boldsymbol{u}$, concave in $\boldsymbol{v}$ and has a general non-linear and non-smooth coupling term, and where the feasible regions $\mathcal{U}$ and $\mathcal{V}$ are non-empty closed convex sets. In fact, our proofs for the convergence results are presented in this general setting; see Appendix C. However, for the convenience of RO theorists and practitioners, we customize the presentation of the inner algorithm to Inner in the main text.

We also remark that our proposed algorithm ProM^3 for RO relies only on the projection and subgradient oracles for the sets and functions, respectively, that define the Robust problem, and can be easily implemented. Contrary to SGSP (Postek and Shtern 2021), we do not need to pre-compute a Slater point or an upper bound of the dual optimal solution. Numerical experiments in Section 5 show its promising performance on large-scale instances, in comparison with the reformulation approach, cutting-plane method, and the first-order methods developed in Postek and Shtern (2021) and Ho-Nguyen and Kılınç-Karzan (2018).

3 Convergence Analysis

This section determines the oracle complexity of the proposed algorithm ProM^3. To this end, we first theoretically analyze the convergence behavior of the outer and inner algorithms.

We collect the assumptions needed for our theoretical development. Assumption 1 below is standard in the RO literature (see, *e.g.*, Ben-Tal et al. 2009).

Assumption 1. *The following conditions hold.*

(i) (Compactness and Convexity of Sets) *The sets* $\mathcal{Z}_1 \subseteq \mathbb{R}^{J_1}, \ldots, \mathcal{Z}_M \subseteq \mathbb{R}^{J_M}$ *and* $\mathcal{X} \subseteq \mathbb{R}^N$ *are non-empty, compact and convex.*

(ii) (Convexity of Functions) *The function* $f_0 : \mathbb{R}^N \to \mathbb{R} \cup \{+\infty\}$ *is convex, with* $\mathcal{X} \subseteq \operatorname{dom}(f_0)$*. For any* $m \in [M]$*, the function* $g_m : \mathbb{R}^N \times \mathbb{R}^{J_m} \to \mathbb{R} \cup \{\pm\infty\}$ *satisfies that* $g_m(\cdot, \boldsymbol{z}_m)$ *is convex on* $\mathbb{R}^N$ *for any* $\boldsymbol{z}_m \in \mathcal{Z}_m$ *and* $g_m(\boldsymbol{x}, \cdot)$ *is concave on* $\mathbb{R}^{J_m}$ *for any* $\boldsymbol{x} \in \mathcal{X}$*, with* $\mathcal{X} \times \mathcal{Z}_m \subseteq \operatorname{dom}(g_m)$*. Here,* $\operatorname{dom}(\cdot)$ *denotes the domain.*

(iii) (Existence of Slater Points) *There exists* $\bar{\boldsymbol{x}} \in \mathcal{X}$ *such that* $\max_{\boldsymbol{z}_m \in \mathcal{Z}_m} g_m(\bar{\boldsymbol{x}}, \boldsymbol{z}_m) < 0$ *for any* $m \in [M]$.

(*iv*) (Existence of Optimal Solutions) *The* ROBUST *problem has an optimal solution.*

An immediate consequence of Assumption 1 is the following equivalence.

Proposition 2. *Suppose that Assumption 1 holds. Then the* ROBUST *problem is equivalent to the* MAX-MIN-MAX *problem in the sense that their optimal values are equal and that for any optimal solution* $(\boldsymbol{\lambda}^\star, \boldsymbol{x}^\star, \boldsymbol{z}^\star)$ *to* MAX-MIN-MAX*,* $\boldsymbol{x}^\star$ *is an optimal solution to* ROBUST.

We also need the following assumption concerning the subgradient of the functions $g_1, \ldots, g_m$ and f_0, which is customary in the literature of subgradient-type algorithms.

Assumption 2 (Uniformly Bounded Subdifferentials)**.** *The function* f_0 *is subdifferentiable on* $\mathcal{X}$*. For any* $m \in [M]$*,* $\boldsymbol{x} \in \mathcal{X}$ *and* $\boldsymbol{z}_m \in \mathcal{Z}_m$*, the functions* $g_m(\cdot, \boldsymbol{z}_m)$ *and* $-g_m(\boldsymbol{x}, \cdot)$ *are subdifferentiable on* $\mathcal{X}$ *and* $\mathcal{Z}_m$*, respectively. There exist constants* $D_0, D_1, \ldots, D_M, E_1, \ldots, E_M > 0$ *such that for any* $m \in [M]$*,* $\boldsymbol{z}_m \in \mathcal{Z}_m$ *and* $\boldsymbol{x} \in \mathcal{X}$*,*

$$\begin{aligned} \|\boldsymbol{\xi}_0\|_2 &\leq D_0 & \forall \boldsymbol{\xi}_0 &\in \partial f_0(\boldsymbol{x}), \\ \|\boldsymbol{\xi}_m\|_2 &\leq D_m & \forall \boldsymbol{\xi}_m &\in \partial_{\boldsymbol{x}} g_m(\boldsymbol{x}, \boldsymbol{z}_m), \\ \|\boldsymbol{\zeta}_m\|_2 &\leq E_m & \forall \boldsymbol{\zeta}_m &\in \partial_{\boldsymbol{z}_m}(-g_m)(\boldsymbol{x}, \boldsymbol{z}_m). \end{aligned}$$

Recall that a function is subdifferentiable at a point if its subdifferential (*i.e.*, the set of subgradients) at that point is non-empty (Rockafellar 1970, Section 23). It is well-known that any convex function has a bounded non-empty subdifferential at any point in the interior of its domain (Rockafellar 1970, Theorem 23.4). Therefore, in view of Assumption 1(*ii*), Assumption 2 can only be violated on the boundary and hence very mild. Indeed, if $\mathcal{X} \subseteq \text{int}(\text{dom}(f_0))$ and $\mathcal{X} \times \mathcal{Z}_m \subseteq \text{int}(\text{dom}(g_m))$ for all $m \in [M]$ (*e.g.*, when f_0 and $g_1, \ldots, g_m$ are real-valued everywhere), where $\text{int}(\cdot)$ denotes the interior, then Assumption 2 holds.

3.1 Outer Convergence Rate

Our first main theoretical result concerns the convergence rate of the outer algorithm.

Theorem 1. *Suppose that Assumptions 1 and 2 hold. Consider Algorithm 1 with* $\theta = \nu = \frac{1}{K}$*,*

$\alpha \leq \frac{1}{\sqrt{\sum_{m\in[M]} D_m^2}}$ *and* $\beta \leq \frac{1}{2\sqrt{\sum_{m\in[M]} D_m^2}}$. *Then, the output* $\bar{\boldsymbol{x}}^K$ *satisfies*

$$f_0(\bar{\boldsymbol{x}}^K) - f_0(\boldsymbol{x}^\star) \leq \frac{C_{o,1}}{K} \quad \text{and} \quad \max_{m\in[M]} \left[f_m(\bar{\boldsymbol{x}}^K)\right]_+ \leq \frac{C_{o,2}}{K}$$

for some constants $C_{o,1}, C_{o,2} > 0$.

Although the outer algorithm shares a similar blueprint of the alternating proximal algorithm, Theorem 1 does not follow directly from existing studies but requires certain new ideas. First, even though alternating proximal algorithms with inaccurate updates have been studied in a number of works, the forms of inaccuracy assumed in those papers do not cover that of our outer algorithm. Second, existing theoretical works on alternating proximal algorithms primarily focus on bounding the saddle gap $\mathcal{L}(\bar{\boldsymbol{\lambda}}^K, \boldsymbol{x}^\star) - \mathcal{L}(\boldsymbol{\lambda}^\star, \bar{\boldsymbol{x}}^K)$; see Chambolle and Pock (2011, 2016). We, however, care more about the optimality gap and constraint violation, since our ultimate goal is to solve the Robust problem.

3.2 Inner Convergence Rate

The following theorem asserts that under Assumptions 1 and 2, the inner algorithm finds a strong ν-approximate saddle point in $\mathcal{O}(\nu^{-2})$ iterations.

Theorem 2. *Suppose that Assumptions 1 and 2 hold. There exists a constant* $C_{i,1} > 0$ *such that if* $T \geq \frac{C_{i,1}}{\nu^2}$ *and* $\gamma, \delta \leq \frac{1}{\sqrt{T}}$, *then the output* $(\bar{\boldsymbol{x}}_T, \bar{\boldsymbol{z}}_T)$ *of Algorithm 2 is a strong* ν*-approximate saddle point.*

To present our next result, we introduce the following smoothness conditions on the functions $f_0, g_1, \ldots, g_M$.

Assumption 3. *The function* f_0 *is differentiable*[1] *on* $\mathcal{X}$. *For any* $m \in [M]$, *the function* g_m *is differentiable on* $\mathcal{X} \times \mathcal{Z}_m$. *There exist constants* $D_0', D_1', \ldots, D_M', E_{1,1}', E_{1,2}', \ldots, E_{M,1}', E_{M,2}' >$

[1] A function is differentiable on a *non-open* set $\mathcal{S}$ if it is differentiable on an open set containing $\mathcal{S}$.

0 *such that for any* $m \in [M]$, $\boldsymbol{z}_m, \boldsymbol{z}'_m \in \mathcal{Z}_m$ *and* $\boldsymbol{x}, \boldsymbol{x}' \in \mathcal{X}$, *we have*

$$\begin{aligned} \|\nabla f_0(\boldsymbol{x}) - \nabla f_0(\boldsymbol{x}')\|_2 &\leq D'_0 \, \|\boldsymbol{x} - \boldsymbol{x}'\|_2, \\ \|\nabla_{\boldsymbol{x}} g_m(\boldsymbol{x}, \boldsymbol{z}_m) - \nabla_{\boldsymbol{x}} g_m(\boldsymbol{x}', \boldsymbol{z}_m)\|_2 &\leq D'_m \, \|\boldsymbol{x} - \boldsymbol{x}'\|_2, \\ \|\nabla_{\boldsymbol{z}_m} g_m(\boldsymbol{x}, \boldsymbol{z}_m) - \nabla_{\boldsymbol{z}_m} g_m(\boldsymbol{x}', \boldsymbol{z}'_m)\|_2 &\leq E'_{m,1} \, \|\boldsymbol{x} - \boldsymbol{x}'\|_2 + E'_{m,2} \, \|\boldsymbol{z}_m - \boldsymbol{z}'_m\|_2. \end{aligned}$$

Under Assumption 1, it can be readily shown that Assumption 3 implies Assumption 2. The theorem below asserts that the iteration complexity of the inner algorithm can be improved to $\mathcal{O}(\nu^{-1})$ if we replace Assumption 2 by Assumption 3.

Theorem 3. *Suppose that Assumptions 1 and 3 hold. There exists a constant* $C_{i,2} > 0$ *such that if* $T \geq \frac{C_{i,2}}{\nu}$,

$$\gamma \leq \frac{1}{D'_0 + \sum\limits_{m \in [M]} \lambda_m^{k+1} D'_m + \sqrt{2 \sum\limits_{m \in [M]} \left(\lambda_m^{k+1} E'_{m,1}\right)^2}},$$

and

$$\delta \leq \frac{1}{2\sqrt{2} \left(\max\limits_{m \in [M]} \lambda_m^{k+1} E'_{m,2} \right) + \sqrt{2 \sum\limits_{m \in [M]} \left(\lambda_m^{k+1} E'_{m,1}\right)^2}},$$

then the output $(\bar{\boldsymbol{x}}_T, \bar{\boldsymbol{z}}_T)$ *of Algorithm 2 is a strong* ν*-approximate saddle point.*

3.3 Oracle Complexity

Since our approach assumes access to the subgradient oracles of the functions $f_0, g_1, \ldots, g_M$ as well as the projection oracles of the sets $\mathcal{X}, \mathcal{Z}_1, \ldots, \mathcal{Z}_M$, a faithful and popular measure of computational efficiency would be the so-called *oracle complexity*—the total number of calls of these oracles—for achieving an ε-approximate optimal solution. Here we recall that for any $\varepsilon > 0$, a point $\boldsymbol{x}$ is an ε-approximate optimal solution to Robust if

$$f_0(\boldsymbol{x}) - f_0(\boldsymbol{x}^\star) \leq \varepsilon \quad \text{and} \quad \max_{m \in [M]} \, [f_m(\boldsymbol{x})]_+ \leq \varepsilon.$$

The next theorem presents the oracle complexity of ProM^3 by combining the convergence rate results for the outer and inner algorithms.

Theorem 4. *Suppose that Assumptions 1 and 2 hold. Then, for any $\varepsilon > 0$, the oracle complexity of* ProM^3 *for achieving an ε-approximate optimal solution to* Robust *is $\mathcal{O}(\varepsilon^{-3})$.*

Under similar assumptions as in Theorem 4, Postek and Shtern (2021) proved that the oracle complexity of the algorithm SGSP is $\mathcal{O}(\varepsilon^{-2})$. However, due to the use of perspective transformation, SGSP relies on subgradient and projection oracles that are generally more computationally expensive. Thus, our oracle complexity results are not directly comparable to that of Postek and Shtern (2021).

The oracle complexity can be improved if Assumption 2 is replaced by Assumption 3.

Theorem 5. *Suppose that Assumptions 1 and 3 hold. Then, for any $\varepsilon > 0$, the oracle complexity of* ProM^3 *for achieving an ε-approximate optimal solution to* Robust *is $\mathcal{O}(\varepsilon^{-2})$.*

Under similar assumptions to Theorem 5, a first-order method is developed in Ho-Nguyen and Kılınç-Karzan (2018, 2019) via online convex optimization and proved to enjoy the oracle complexity $\mathcal{O}(\varepsilon^{-1}\log\frac{1}{\varepsilon})$. Nevertheless, the authors considered only low- to medium-dimensional instances in their numerical experiments. In Section 5, we demonstrate that when compared against the first-order methods in Postek and Shtern (2021) and Ho-Nguyen and Kılınç-Karzan (2018, 2019) our algorithm ProM^3 is substantially more stable and efficient.

4 Extension to Projection-Unfriendly Uncertainty Sets

For some applications of RO, the uncertainty sets $\mathcal{Z}_m$ take the form of an intersection and do not admit an easy projection. Directly invoking our ProM^3 in Section 2 to such RO problems could be inefficient. Below we extend our first-order algorithm ProM^3 to RO problems with uncertainty sets of the form

$$\mathcal{Z}_m = \widetilde{\mathcal{Z}}_m \cap \left(\bigcap_{i\in[I_m]} \mathcal{Z}_{m,i} \right), \tag{3}$$

where for each $i \in [I_m]$, the set $\mathcal{Z}_{m,i} = \{\boldsymbol{z}_m \in \mathbb{R}^{J_m} \mid h_{m,i}(\boldsymbol{z}_m) \leq 0\}$ is defined by some function $h_{m,i}$.

To present the extended ProM^3 and its analysis, we make the following assumption: an adaptation of Assumption 1 to the projection-unfriendly setting.

Assumption 4. *The following conditions hold.*

(i) (Compactness and Convexity of Sets) *The sets* $\widetilde{\mathcal{Z}}_1 \subseteq \mathbb{R}^{J_1}, \ldots, \widetilde{\mathcal{Z}}_M \subseteq \mathbb{R}^{J_M}$ *and* $\mathcal{X} \subseteq \mathbb{R}^N$ *are non-empty, compact and convex.*

(ii) (Convexity of Functions) *The function* $f_0 : \mathbb{R}^N \to \mathbb{R} \cup \{+\infty\}$ *is convex, with* $\mathcal{X} \subseteq \operatorname{dom}(f_0)$*. For any* $m \in [M]$*, the function* $g_m : \mathbb{R}^N \times \mathbb{R}^{J_m} \to \mathbb{R} \cup \{\pm\infty\}$ *satisfies that* $g_m(\cdot, \boldsymbol{z}_m)$ *is convex on* $\mathbb{R}^N$ *for any* $\boldsymbol{z}_m \in \widetilde{\mathcal{Z}}_m$ *and* $g_m(\boldsymbol{x}, \cdot)$ *is concave on* $\mathbb{R}^{J_m}$ *for any* $\boldsymbol{x} \in \mathcal{X}$*, with* $\mathcal{X} \times \widetilde{\mathcal{Z}}_m \subseteq \operatorname{dom}(g_m)$*. For any* $m \in [M]$ *and* $i \in [I_m]$*, the function* $h_{m,i} : \mathbb{R}^{J_m} \to \mathbb{R} \cup \{+\infty\}$ *is convex, with* $\widetilde{\mathcal{Z}}_m \subseteq \operatorname{dom}(h_{m,i})$*.*

(iii) (Existence of Slater Points) *There exists* $\bar{\boldsymbol{x}} \in \mathcal{X}$ *such that* $\max_{\boldsymbol{z}_m \in \widetilde{\mathcal{Z}}_m} g_m(\bar{\boldsymbol{x}}, \boldsymbol{z}_m) < 0$ *for any* $m \in [M]$*. For any* $m \in [M]$*, there exists* $\bar{\boldsymbol{z}}_m \in \widetilde{\mathcal{Z}}_m$ *such that* $h_{m,i}(\bar{\boldsymbol{z}}_m) < 0$ *for any* $i \in [I_m]$*.*

(iv) (Existence of Optimal Solutions) *The* Robust *problem has an optimal solution.*

We also need an adaptation of Assumption 2.

Assumption 5 (Uniformly Bounded Subdifferentials)**.** *The function* f_0 *is subdifferentiable on* $\mathcal{X}$*. For any* $m \in [M]$*,* $\boldsymbol{x} \in \mathcal{X}$ *and* $\boldsymbol{z}_m \in \widetilde{\mathcal{Z}}_m$*, the functions* $g_m(\cdot, \boldsymbol{z}_m)$ *and* $-g_m(\boldsymbol{x}, \cdot)$ *are subdifferentiable on* $\mathcal{X}$ *and* $\widetilde{\mathcal{Z}}_m$*, respectively. For any* $m \in [M]$ *and* $i \in [I_m]$*, the function* $h_{m,i}$ *is subdifferentiable on* $\widetilde{\mathcal{Z}}_m$*. There exist constants* $D_0, D_1, \ldots, D_M, E_1, \ldots, E_M, F_1, \ldots, F_M > 0$ *such that for any* $m \in [M]$*,* $i \in [I_m]$*,* $\boldsymbol{z}_m \in \widetilde{\mathcal{Z}}_m$ *and* $\boldsymbol{x} \in \mathcal{X}$*, we have*

$$\begin{aligned}
\|\boldsymbol{\xi}_0\|_2 &\le D_0 && \forall \boldsymbol{\xi}_0 \in \partial f_0(\boldsymbol{x}),\\
\|\boldsymbol{\xi}_m\|_2 &\le D_m && \forall \boldsymbol{\xi}_m \in \partial_{\boldsymbol{x}} g_m(\boldsymbol{x}, \boldsymbol{z}_m),\\
\|\boldsymbol{\zeta}_m\|_2 &\le E_m && \forall \boldsymbol{\zeta}_m \in \partial_{\boldsymbol{z}_m}(-g_m)(\boldsymbol{x}, \boldsymbol{z}_m),\\
\|\boldsymbol{\eta}_{m,i}\|_2 &\le F_m && \forall \boldsymbol{\eta}_{m,i} \in \partial h_{m,i}(\boldsymbol{z}_m).
\end{aligned}$$

Similarly, the oracle complexity of the extended ProM^3 can be improved when the gradients of the constituent functions satisfy certain Lipschitz property.

Assumption 6. *The function* f_0 *is differentiable on* $\mathcal{X}$*. For any* $m \in [M]$*, the function* g_m *is differentiable on* $\mathcal{X} \times \widetilde{\mathcal{Z}}_m$*. For any* $m \in [M]$ *and* $i \in [I_m]$*, the function* $h_{m,i}$ *is differentiable* *on $\widetilde{\mathcal{Z}}_m$. There exist constants $D'_0, D'_1, \ldots, D'_M, E'_{1,1}, E'_{1,2}, \ldots, E'_{M,1}, E'_{M,2}, F'_1, \ldots, F'_M > 0$ such that for any $m \in [M]$, $i \in [I_m]$, $\boldsymbol{z}_m, \boldsymbol{z}'_m \in \widetilde{\mathcal{Z}}_m$ and $\boldsymbol{x}, \boldsymbol{x}' \in \mathcal{X}$, we have*

$$\begin{aligned}
\|\nabla f_0(\boldsymbol{x}) - \nabla f_0(\boldsymbol{x}')\|_2 &\leq D'_0 \, \|\boldsymbol{x} - \boldsymbol{x}'\|_2, \\
\|\nabla_{\boldsymbol{x}} g_m(\boldsymbol{x}, \boldsymbol{z}_m) - \nabla_{\boldsymbol{x}} g_m(\boldsymbol{x}', \boldsymbol{z}_m)\|_2 &\leq D'_m \, \|\boldsymbol{x} - \boldsymbol{x}'\|_2, \\
\|\nabla_{\boldsymbol{z}_m} g_m(\boldsymbol{x}, \boldsymbol{z}_m) - \nabla_{\boldsymbol{z}_m} g_m(\boldsymbol{x}', \boldsymbol{z}'_m)\|_2 &\leq E'_{m,1} \, \|\boldsymbol{x} - \boldsymbol{x}'\|_2 + E'_{m,2} \, \|\boldsymbol{z}_m - \boldsymbol{z}'_m\|_2, \\
\|\nabla h_{m,i}(\boldsymbol{z}_m) - \nabla h_{m,i}(\boldsymbol{z}'_m)\|_2 &\leq F'_m \, \|\boldsymbol{z}_m - \boldsymbol{z}'_m\|_2.
\end{aligned}$$

Note that for any $m \in [M]$, $g_m(\cdot, \bar{\boldsymbol{z}}_m)$ is continuous on $\mathcal{X}$ because it is real-valued and convex on $\mathcal{X}$. By the compactness of $\mathcal{X}$, there exist constants $G_1, \ldots, G_M < 0$ such that

$$g_m(\boldsymbol{x}, \bar{\boldsymbol{z}}_m) \geq G_m \qquad \forall \boldsymbol{x} \in \mathcal{X}, \; m \in [M].$$

For simplicity, we denote $\boldsymbol{h}_m(\boldsymbol{z}_m) = (h_{m,1}(\boldsymbol{z}_m), \ldots, h_{m,I_m}(\boldsymbol{z}_m))$, $\widetilde{\mathcal{Z}} = \widetilde{\mathcal{Z}}_1 \times \cdots \times \widetilde{\mathcal{Z}}_M$ and $\mathcal{M} = [0, a_1]^{I_1} \times \cdots \times [0, a_M]^{I_M}$, where

$$a_m = \frac{G_m}{\max_{i \in [I_m]} \{h_{m,i}(\bar{\boldsymbol{z}}_m)\}}.$$

Our extension of ProM^3 to projection-unfriendly uncertainty sets of the form (3) is based on the following proposition.

Proposition 3. *Consider the* Robust *problem with uncertainty sets $\mathcal{Z}_m$ as in* (3)*. Suppose that Assumption 4 holds. Then, the* Robust *problem is equivalent to the problem*

$$\begin{aligned}
\min \quad & \tilde{f}_0(\tilde{\boldsymbol{x}}) \\
\text{s.t.} \quad & \max_{\boldsymbol{z}_m \in \widetilde{\mathcal{Z}}_m} \tilde{g}_m(\tilde{\boldsymbol{x}}, \boldsymbol{z}_m) \leq 0 \quad \forall m \in [M] \\
& \tilde{\boldsymbol{x}} = (\boldsymbol{x}, \boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_M) \in \widetilde{\mathcal{X}},
\end{aligned} \qquad (\widetilde{\text{Robust}})$$

where $\tilde{f}_0(\tilde{\boldsymbol{x}}) = f_0(\boldsymbol{x})$, $\widetilde{\mathcal{X}} = \mathcal{X} \times \mathcal{M}$ and $\tilde{g}_m(\tilde{\boldsymbol{x}}, \boldsymbol{z}_m) = g_m(\boldsymbol{x}, \boldsymbol{z}_m) - \boldsymbol{\mu}_m^\top \boldsymbol{h}_m(\boldsymbol{z}_m)$ for all $m \in [M]$.

The $\widetilde{\text{Robust}}$ problem is obtained by penalizing the constraints $\boldsymbol{h}_m(\boldsymbol{z}_m) \leq \boldsymbol{0}$ in the m-th embedded problem to its objective. It is expected to be equivalent to the original Robust problem under suitable assumptions if we do not restrict the dual variable $\boldsymbol{\mu}_m$ from above,

Algorithm 3: Extended Outer Algorithm.

```
Input : $K \geq 1$, $\beta > 0$, $\alpha > 0$, $\nu > 0$, $\theta > 0$, $\boldsymbol{\lambda}^0 = \mathbf{0}$, $\boldsymbol{x}^0 \in \mathcal{X}$ and $\boldsymbol{\mu}^0 \in \mathcal{M}$.
for $k = 0, 1, \ldots, K-1$ do
    ($\boldsymbol{\lambda}$-update) For all $m \in [M]$, find $\boldsymbol{z}_m^k \in \widetilde{\mathcal{Z}}_m$ satisfying
        $\max_{\boldsymbol{z}_m \in \widetilde{\mathcal{Z}}_m} g_m(\boldsymbol{x}^k, \boldsymbol{z}_m) - (\boldsymbol{\mu}_m^k)^\top \boldsymbol{h}_m(\boldsymbol{z}) \leq g_m(\boldsymbol{x}^k,\ \boldsymbol{z}_m^k) - (\boldsymbol{\mu}_m^k)^\top \boldsymbol{h}_m(\boldsymbol{z}_m^k) + \theta,$
    and set
        $\lambda_m^{k+1} = \left[\lambda_m^k + \beta\left(2\, g_m(\boldsymbol{x}^k, \boldsymbol{z}_m^k) - 2(\boldsymbol{\mu}_m^k)^\top \boldsymbol{h}_m(\boldsymbol{z}_m^k) - g_m(\boldsymbol{x}^{k-1}, \boldsymbol{z}_m^{k-1}) + (\boldsymbol{\mu}_m^{k-1})^\top \boldsymbol{h}_m(\boldsymbol{z}_m^{k-1})\right)\right]_+,$
    where $\boldsymbol{x}^{-1} = \boldsymbol{x}^0$ and $\boldsymbol{z}^{-1} = \boldsymbol{z}^0$ for the 0-th iteration.
    ($\tilde{\boldsymbol{x}}$-update) Compute a strong $\nu$-approximate saddle point $((\boldsymbol{x}^{k+1}, \boldsymbol{\mu}^{k+1}), \tilde{\boldsymbol{z}}^{k+1})$ of
    the following problem via the Extended Inner Algorithm (Algorithm 4):
        $\min_{\boldsymbol{x} \in \mathcal{X},\, \boldsymbol{\mu} \in \mathcal{M}} \max_{\boldsymbol{z} \in \widetilde{\mathcal{Z}}} \left\{ f_0(\boldsymbol{x}) + \sum_{m \in [M]} \lambda_m^{k+1}\left(g_m(\boldsymbol{x}, \boldsymbol{z}_m) - \boldsymbol{\mu}_m^\top \boldsymbol{h}_m(\boldsymbol{z}_m)\right) + \frac{1}{2\alpha}\|\boldsymbol{x} - \boldsymbol{x}^k\|_2^2 + \frac{1}{2\alpha}\|\boldsymbol{\mu} - \boldsymbol{\mu}^k\|_2^2 \right\},$    ($\widetilde{\textsc{Inner}}$)
end
Output: $\bar{\boldsymbol{x}}^K = \frac{1}{K}\sum_{k \in [K]} \boldsymbol{x}^k$.
```

i.e., if a_m in the definition of $\mathcal{M}$ is replaced by $+\infty$ for all $m \in [M]$. Nevertheless, for our algorithmic framework and theoretical results in Section 2 to be applicable, we need to introduce an upper bound a_m for each $\boldsymbol{\mu}_m$ to make the feasible region $\widetilde{\mathcal{X}}$ compact. What is perhaps less trivial is that the equivalence remains after restricting the dual variable.

Applying our framework to the $\widetilde{\textsc{Robust}}$ problem, we obtain an extension of ProM^3 for RO problems with projection-unfriendly uncertainty sets of the form (3). To facilitate easy usage, we present the extended outer and inner algorithms fully in terms of the basic constituent functions and sets in Algorithms 3 and 4, respectively. The extended ProM^3 enjoys the following complexity result.

Theorem 6. *Consider the* Robust *problem with uncertainty sets $\mathcal{Z}_m$ as in* (3)*. Suppose that Assumptions 4 and 5 hold. Then, for any $\varepsilon > 0$, the oracle complexity of extended* ProM^3 *(Algorithms 3 and 4 combined) for achieving an ε-approximate optimal solution to*

```
Algorithm 4: Extended Inner Algorithm.
Input : $T \geq 1$, $\delta > 0$, $\gamma > 0$, $\boldsymbol{x}_0, \boldsymbol{x}^k \in \mathcal{X}$, $\boldsymbol{\mu}_0, \boldsymbol{\mu}^k \in \mathcal{M}$, $\boldsymbol{z}_0 \in \widetilde{\mathcal{Z}}$ and $\boldsymbol{\lambda}^{k+1} \geq \mathbf{0}$.
for $t = 0, \dots, T-1$ do
    ($\boldsymbol{z}$-update) Compute $\boldsymbol{\zeta}_{t,m} \in \partial_{\boldsymbol{z}_m}(-g_m)(\boldsymbol{x}_t, \boldsymbol{z}_{t,m})$ and $\boldsymbol{\eta}_{t,m,i} \in \partial h_{m,i}(\boldsymbol{z}_{t,m})$ for all $m \in [M]$ and $i \in [I_m]$. Set
        $\boldsymbol{z}_{t+1,m} = \mathrm{Proj}_{\widetilde{\mathcal{Z}}_m}\left(\boldsymbol{z}_{t,m} - \delta\, \lambda_m^{k+1}\left(2\boldsymbol{\zeta}_{t,m} - \boldsymbol{\zeta}_{t-1,m} + \sum_{i=1}^{I_m} \mu_{m,i}(2\boldsymbol{\eta}_{t,m,i} - \boldsymbol{\eta}_{t-1,m,i})\right)\right),$
    where $\boldsymbol{\zeta}_{-1,m} = \boldsymbol{\zeta}_{0,m}$ and $\boldsymbol{\eta}_{-1,m,i} = \boldsymbol{\eta}_{0,m,i}$ for the 0-th iteration.
    ($\tilde{\boldsymbol{x}}$-update) Compute $\boldsymbol{\xi}_{t,0} \in \partial f_0(\boldsymbol{x}_t)$ and $\boldsymbol{\xi}_{t,m} \in \partial_{\boldsymbol{x}} g(\boldsymbol{x}_t, \boldsymbol{z}_{t+1,m})$ for all $m \in [M]$.
    Set $\boldsymbol{\xi}_t = \boldsymbol{\xi}_{t,0} + \lambda_1^{k+1}\,\boldsymbol{\xi}_{t,1} + \cdots + \lambda_M^{k+1}\,\boldsymbol{\xi}_{t,M}$,
        $\boldsymbol{x}_{t+1} = \mathrm{Proj}_{\mathcal{X}}\left(\frac{\alpha\gamma}{\alpha+\gamma}\left(\frac{1}{\alpha}\boldsymbol{x}^k + \frac{1}{\gamma}\boldsymbol{x}_t - \boldsymbol{\xi}_t\right)\right)$ and
        $\boldsymbol{\mu}_{t+1,m} = \mathrm{Proj}_{[0,a_m]^{I_m}}\left(\frac{\alpha\gamma}{\alpha+\gamma}\left(\frac{1}{\alpha}\boldsymbol{\mu}_m^k + \frac{1}{\gamma}\boldsymbol{\mu}_{t,m} + \lambda_m^{k+1}\boldsymbol{h}_m(\boldsymbol{z}_{t+1,m})\right)\right).$
end
Output: $(\bar{\boldsymbol{x}}_T, \bar{\boldsymbol{\mu}}_T) = (\frac{1}{T}\sum_{t\in[T]} \boldsymbol{x}_t, \frac{1}{T}\sum_{t\in[T]} \boldsymbol{\mu}_t)$ and $\bar{\boldsymbol{z}}_T = \frac{1}{T}\sum_{t\in[T]} \boldsymbol{z}_t$.
```

ROBUST *is* $\mathcal{O}(\varepsilon^{-3})$. *If Assumption 5 is replaced by Assumption 6, then the oracle complexity is strengthened to* $\mathcal{O}(\varepsilon^{-2})$.

A major part of the proof of Theorem 6 is to verify that problem $\widetilde{\text{ROBUST}}$ satisfies all the assumptions required by our results in Section 3 (*i.e.*, Assumptions 1-3).

5 Numerical Experiments

This section explores the practical performance of the proposed algorithm ProM^3 through extensive numerical experiments. We compare its performance with the reformulation approach, the cutting-plane method, and two recently developed first-order methods by Ho-Nguyen and Kılınç-Karzan (2018) and Postek and Shtern (2021). We use the legends "CP" for the cutting-plane method, "OCO" for the first-order method in Ho-Nguyen and Kılınç-Karzan (2018), and "SGSP" for the first-order method in Postek and Shtern (2021). For the reformulation approach, we use Ref-ε, where ε is the stopping accuracy and set to the usual accuracy level of first-order methods, 10^{-4} or 10^{-5}, for a fair comparison. Furthermore, when we calculate the time for the reformulation approach, we calculate only the solver time but exclude the modeling and compilation time due to the interfacing. All algorithms are implemented in Python, and all experiments are performed on a 2.6GHz laptop with 16GB memory.

5.1 Robust QCQP

Our first experiment concerns the following robust quadratically constrained quadratic program appeared in Ho-Nguyen and Kılınç-Karzan (2018), Postek and Shtern (2021):

$$\begin{array}{ll} \min & \displaystyle\max_{\boldsymbol{z}_0 \in \mathcal{Z}_0} g_0(\boldsymbol{x}, \boldsymbol{z}_0) \\ \text{s.t.} & \displaystyle\max_{\boldsymbol{z}_m \in \mathcal{Z}_m} g_m(\boldsymbol{x}, \boldsymbol{z}_m) \leq 0 \quad \forall m \in [M] \\ & \boldsymbol{x} \in \mathcal{X}, \end{array} \tag{4}$$

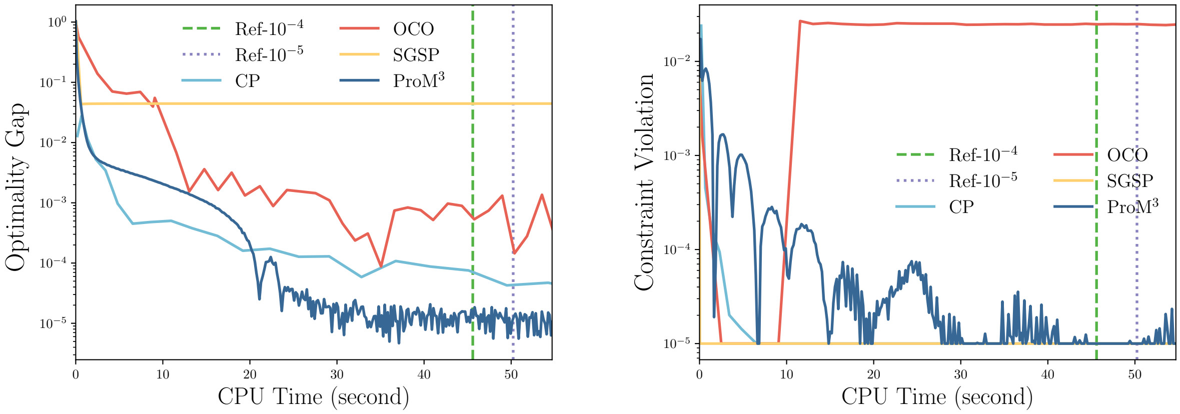
where $\mathcal{X} = \{\boldsymbol{x} \in \mathbb{R}^N \mid \|\boldsymbol{x}\|_2 \leq 1\}$ and for $m \in \{0\} \cup [M]$, $\mathcal{Z}_m = \{\boldsymbol{z}_m \in \mathbb{R}^{J_m} \mid \|\boldsymbol{z}_m\|_2 \leq 1\}$,

$$g_m(\boldsymbol{x}, \boldsymbol{z}_m) = \left\| \left(\boldsymbol{P}_{m0} + \sum_{j=1}^{J_m} \boldsymbol{P}_{mj} z_{mj} \right) \boldsymbol{x} \right\|_2^2 + \boldsymbol{b}_m^\top \boldsymbol{x} + c_m.$$

Here, $\boldsymbol{b}_m \in \mathbb{R}^N$, $c_m \in \mathbb{R}$, $\boldsymbol{P}_{mj} \in \mathbb{R}^{P \times N}$ and z_{mj} is the j-th entry of $\boldsymbol{z}_m$ for all $j \in [J_m]$. We generate the problem data $\boldsymbol{P}_{mj}$, $\boldsymbol{b}_m$ and c_m in the same manner as in Postek and Shtern (2021). For all m and j, the entries of $\boldsymbol{P}_{mj}$ and $\boldsymbol{b}_m$ are i.i.d. uniform random variables on $[-1, 1]$, normalized via $\boldsymbol{P}_{mj} \leftarrow \boldsymbol{P}_{mj} / \|[\boldsymbol{P}_{m0}^\top \cdots \boldsymbol{P}_{mK}^\top]^\top\|_2$ and $\boldsymbol{b}_m \leftarrow \boldsymbol{b}_m / \|\boldsymbol{b}_m\|_2$. We fix $c_m = -0.05$ to ensure that problem (4) has a Slater point. Note that the each $g_m(\boldsymbol{x}, \boldsymbol{z}_m)$ is convex in $\boldsymbol{x}$ but not concave in $\boldsymbol{z}_m$. Nonetheless, by using the techniques in Ho-Nguyen and Kılınç-Karzan (2018), Postek and Shtern (2021), we can transform problem (4) into an instance of Robust satisfying all required assumptions. Note also that the reformulation in this case is a semidefinite program (Ben-Tal and Nemirovski 1998, Theorem 3.2).

We test the algorithms on three problem dimensions: $(M, N, P, J_m) = (3, 1500, 30, 30)$, $(M, N, P, J_m) = (40, 1500, 30, 30)$ and $(M, N, P, J_m) = (4, 8000, 50, 50)$, and the corresponding results are plotted in Figures 1-3, with the optimality gap $|f_0(\boldsymbol{x}) - f_0(\boldsymbol{x}^\star)|$ and the constraint violation $\max_{m \in [M]} [f_m(\boldsymbol{x})]_+$ shown on the left and right panels, respectively. To

Figure 1: $(M, N, P, J_m) = (3, 1500, 30, 30)$.



compute the optimality gap, for the first two cases, we take the solution returned by the reformulation approach with the default accuracy 10^{-12} as the "true" optimum $\boldsymbol{x}^\star$. However, for the last case, the reformulation and cutting-plane methods do not work due to memory issues. In this case, we take the solution returned by our algorithm ProM^3 as $\boldsymbol{x}^\star$, as it achieves a much higher accuracy than the other two first-order methods. Since we cannot keep track of the iterations of the solver in the reformulation approach, we only indicate its total time as a vertical line.

From Figures 1 and 2, our algorithm ProM^3 achieves the optimality gap 10^{-4} and 10^{-5} much faster than the reformulation approach and cutting-plane method, while the other two competing first-order methods get stuck at the level of 10^{-1} to 10^{-2}. In terms of constraint violation, all the tested algorithms reach feasibility in a reasonable amount of time. Figure 3 shows the result for the highest dimensional and the most challenging case, which is beyond the reach of the reformulation approach and cutting-plane method. Our algorithm ProM^3 again considerably outperforms the two other first-order methods.

5.2 Robust Log-Sum-Exponential Constraint

We then consider a more involved RO problem with a highly non-linear embedded optimization problem. This should be contrasted with the robust QCQP example in Section 5.1, where the embedded problem's objective function and feasible region (uncertainty set) are

Figure 2: $(M, N, P, J_m) = (40, 1500, 30, 30)$.

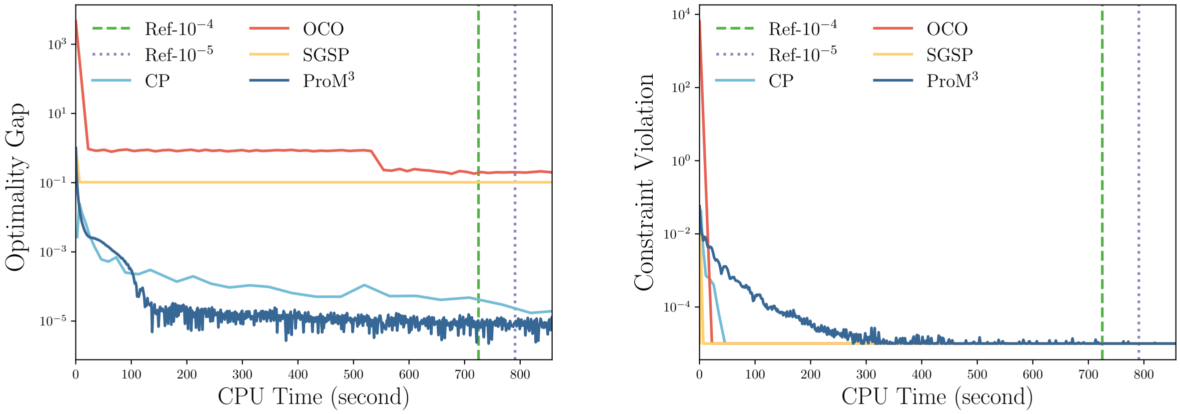
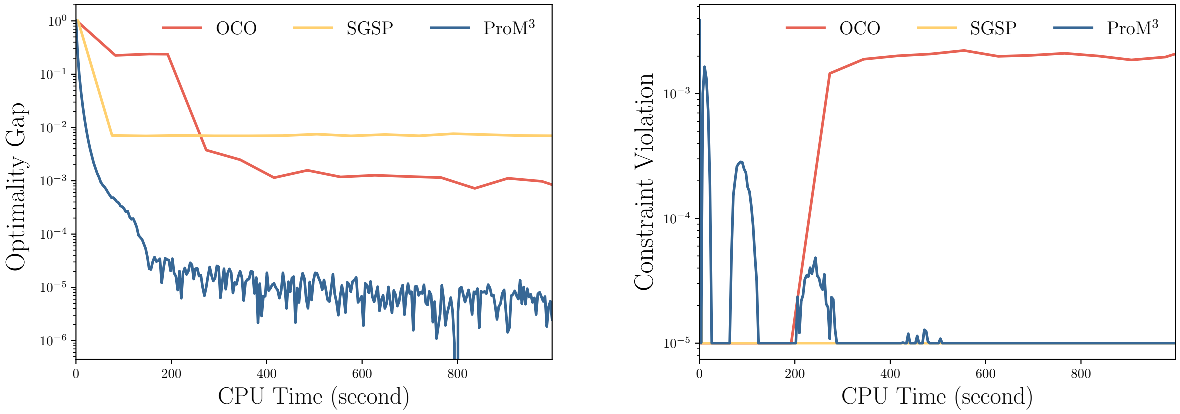


Figure 3: $(M, N, P, J_m) = (40, 8000, 50, 50)$.



both quadratic. Specifically, we consider the problem

$$\begin{aligned} \min \quad & \boldsymbol{c}^\top \boldsymbol{x} \\ \text{s.t.} \quad & \max_{\boldsymbol{z}_m \in \mathcal{Z}_m} g_m(\boldsymbol{x}, \boldsymbol{z}_m) \le 0 \quad \forall m \in [M] \\ & \boldsymbol{x} \in \mathcal{X}, \end{aligned} \tag{5}$$

where $\mathcal{X} = \{\boldsymbol{x} \mid -\boldsymbol{1} \le \boldsymbol{x} \le \boldsymbol{1}\}$ and for each $m \in [M]$, $\mathcal{Z}_m = \{\boldsymbol{z}_m \in \mathbb{R}^{J_m} \mid \boldsymbol{l} \le \boldsymbol{z}_m \le \boldsymbol{u}\}$ and

$$g_m(\boldsymbol{x}, \boldsymbol{z}_m) = \boldsymbol{x}^\top \boldsymbol{A}_m \boldsymbol{z}_m - d_m + \log\left(\boldsymbol{z}_{m,1} + \sum_{j=2}^{J_m} \boldsymbol{z}_{m,j} \exp\left(\boldsymbol{b}_{m,j}^\top \boldsymbol{x}\right) \right).$$

Figure 4: $(M, N, J_m) = (5, 200, 1000)$.

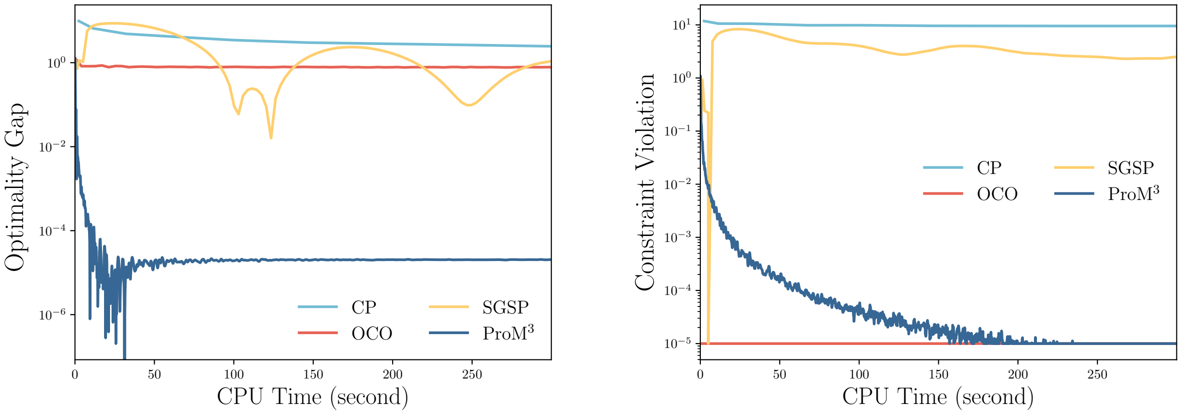
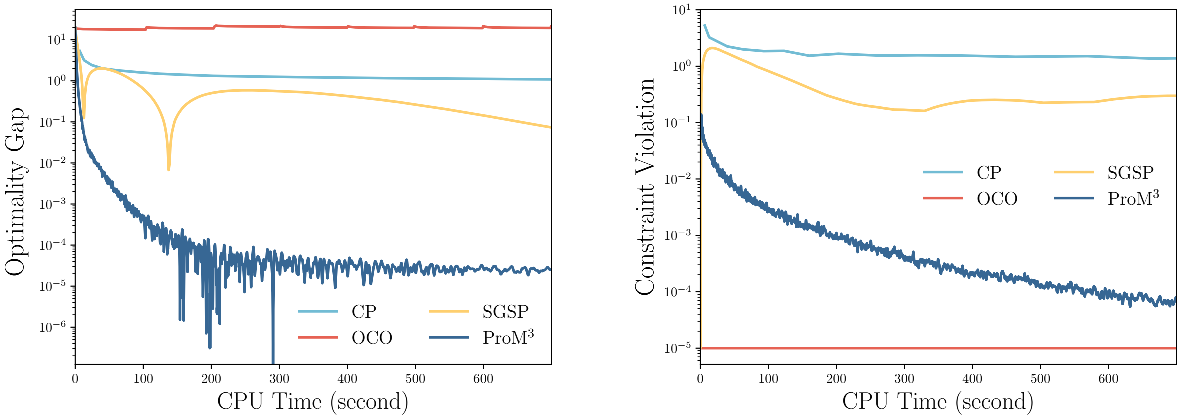


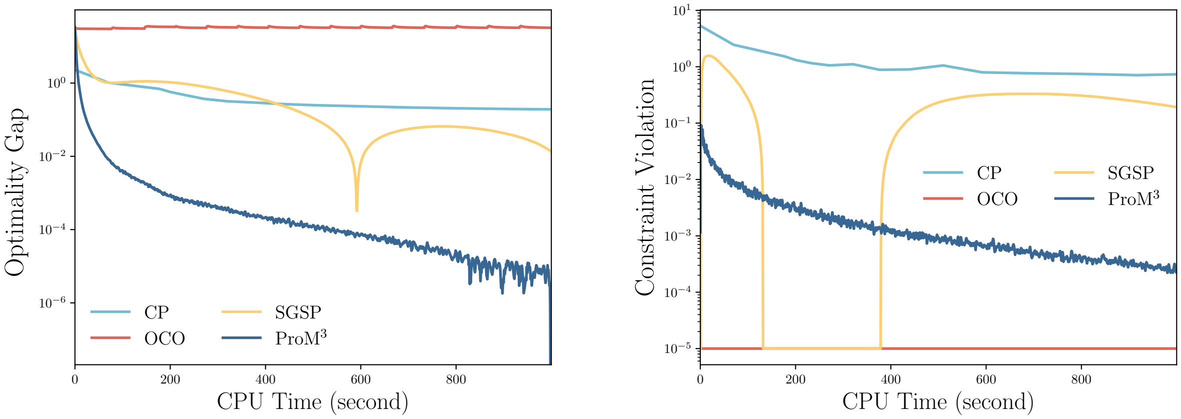
Figure 5: $(M, N, J_m) = (5, 1000, 200)$.



Such functions g_m are called log-sum-exponential functions, and RO problems with robust log-sum-exponential constraints have been considered in (Ben-Tal et al. 2015a, Example 28) and Bertsimas and Hertog (2022). The RO problem (5) is computationally very challenging since the robust log-sum-exponential constraint exhibits high non-linearity. To the best of our knowledge, there is no tractable convex reformulation for the RO problem (5) (Ben-Tal et al. 2015a, Example 28). We therefore do not compare with the reformulation approach in this experiment. Nevertheless, all other methods can be applied. This also shows that the scope of the reformulation approach is in general more restricted.

The data are generated as follows. We set $\boldsymbol{l} = 0.001$ and $\boldsymbol{u} = 1$, and the vector $\boldsymbol{c}$ is also generated randomly with i.i.d. standard Gaussian entries. For each $m \in [M]$, denoting $\boldsymbol{B}_m =$

Figure 6: $(M, N, J_m) = (3, 2000, 200)$.



$[\boldsymbol{b}_{m,2}^\top; \boldsymbol{b}_{m,3}^\top; \cdots; \boldsymbol{b}_{m,J_m}^\top]$, we first generate $\boldsymbol{B}_m$ with i.i.d. standard Gaussian entries and then normalize it by $\boldsymbol{B}_m \leftarrow \boldsymbol{B}_m/\|\boldsymbol{B}_m\|_2$. The matrix $\boldsymbol{A}_m$ is generated in the same manner as $\boldsymbol{B}_m$. We also set $d_m = \max_{\boldsymbol{z}_m \in \mathcal{Z}_m} (\boldsymbol{u}/\|\boldsymbol{u}\|_2)^\top \boldsymbol{A}_m \boldsymbol{z}_m + \log\left(\boldsymbol{z}_{m,1} + \sum_{j=2}^{J_m} \boldsymbol{z}_{m,j} \exp\left(\boldsymbol{b}_{m,j}^\top (\boldsymbol{u}/\|\boldsymbol{u}\|_2)\right)\right)$, where $\boldsymbol{u}$ is a random vector with entries being i.i.d. uniform random variables on $[0, 1]$.

We test the algorithms on three problem dimensions: $(M, N, J_m) = (5, 200, 1000)$, $(M, N, J_m) = (5, 1000, 200)$ and $(M, N, J_m) = (3, 2000, 200)$, and the corresponding results are plotted in Figures 4-6. In this experiment, we take the solution returned by our algorithm ProM^3 as the true optimal solution since it achieves a much higher accuracy than the competing methods, as indicated in Section 5.1.

In all these three cases, our algorithm ProM^3 converges much faster in terms of optimality gap than all other algorithms. Also, ProM^3 can achieve the accuracy level 10^{-5} to 10^{-6}, whereas all other methods get stuck at the level of 10^{-1}. In terms of constraint violation, ProM^3 converges to the feasible region stably and efficiently, whereas the cutting-plane method and SGSP struggle at the level of 10^{0}- 10^{-1}. The sequence of iterates generated by OCO seems to be feasible over the course of execution. This experiment shows that our algorithm is suitable also for highly non-linear RO problems.

5.3 Distributionally Robust Newsvendor Problem

Finally, we investigate the numerical performance of our extend ProM^3 for tackling RO problems with projection-unfriendly uncertainty sets. To this end, we consider a multiproduct newsvendor problem that aims at minimizing the total ordering cost subject to a distributionally robust profit-risk constraint. Let there be M products. For each product $m \in [M]$, the random demand d_m has N possible outcomes, $d_m^1, \ldots, d_m^N$, and the corresponding probabilities are $z_m^1, \ldots, z_m^N$. We also denote by c_m, v_m, s_m and t_m the unit purchase cost, the unit selling price, the unit salvage value and the unit storage cost, respectively. If we purchase x_m units of product m, under the demand outcome d_m^n, the profit is

$$r(x_m, d_m^n) = v_m \min\{d_m^n, x_m\} + s_m(x_m - d_m^n)_+ - t_m(d_m^n - x_m)_+ - c_m x_m. \tag{6}$$

The conditional value-at-risk (CVaR) of the loss $-r(x_m, d_m^n)$ at the quantile level $\kappa \in [0, 1)$ is given by

$$\inf_{\tau \in \mathbb{R}} \left\{ \frac{\mathbb{E}_{d_m \sim \boldsymbol{z}_m}[[\tau - r(x_m, d_m)]_+]}{1 - \kappa} - \tau \right\},$$

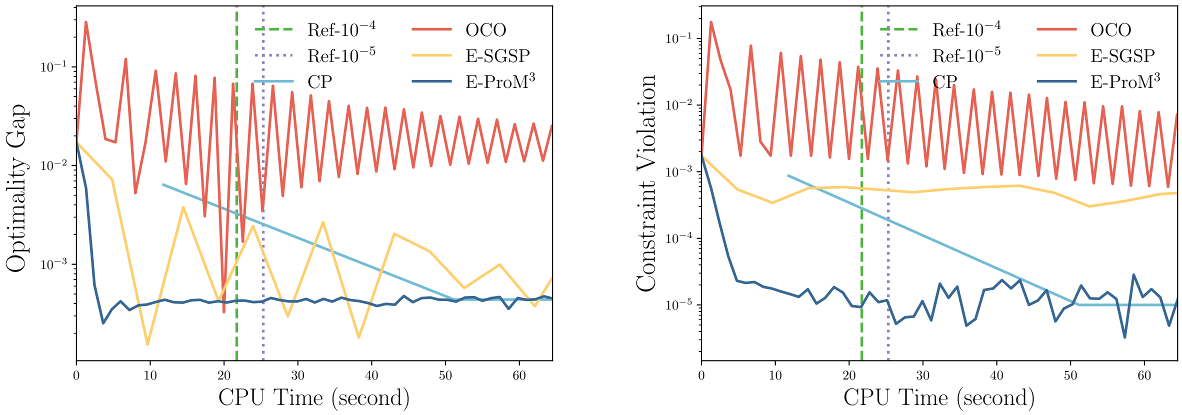
which represents the average of the loss $-r(x_m, d_m)$ over its $(1-\kappa)$-tail region. Following the literature on distributionally robust optimization (Ben-Tal et al. 2013, Mohajerin Esfahani and Kuhn 2018, Yue et al. 2022, Gao and Kleywegt 2023), we assume that the probability distribution $\boldsymbol{z}_m$ of the random demand is not precisely known but lies in an ambiguity set $\mathcal{Z}_m$. It is then natural to consider the following formulation, which minimizes the purchasing cost subject to the distributionally robust loss CVaR constraint:

$$\begin{aligned} \min \quad & \boldsymbol{c}^\top \boldsymbol{x} \\ \text{s.t.} \quad & \max_{\boldsymbol{z}_m \in \mathcal{Z}_m} \left\{ \frac{\mathbb{E}_{d_m \sim \boldsymbol{z}_m}[[\tau_m - r(x_m, d_m)]_+]}{1 - \kappa} - \tau_m \right\} \leq \rho_m \quad \forall m \in [M] \\ & \boldsymbol{x} \in [0, 1]^M, \ \boldsymbol{\tau} \in \mathbb{R}^M, \end{aligned} \tag{7}$$

where $\rho_m > 0$ is a prescribed risk threshold and $r(x_m, d_m)$ is defined in (6). The ambiguity sets $\mathcal{Z}_m$ are taken as an intersection of the probability simplex in $\mathbb{R}^N$ and an Euclidean ball centered at the empirical distribution, *i.e.*, each outcome d_m^n takes probability $1/N$. Formulation (7) is thus an instance of RO problems with projection unfriendly uncertainty sets of the form (3). The other problem parameters are generated randomly.

Similarly to Sections 5.1 and 5.2, we compare our algorithm, the extended ProM^3, with the cutting-plane method, the reformulation approach and two first-order methods from

Figure 7: $(M, N) = (30, 5000)$.



the papers Ho-Nguyen and Kılınç-Karzan (2018) and Postek and Shtern (2021). However, the paper Postek and Shtern (2021) also developed an extension of SGSP for RO problems having uncertainty sets of the form (3). For fairness, we will invoke the extended SGSP in this experiment. The legends "E-ProM^3" and "E-SGSP" are adopted to represent the extended ProM^3 and extended SGSP, respectively. For the other competing algorithms, the legends are the same as before. The result of a typical instance with dimension $(M, N) = (30, 5000)$ is plotted in Figure 7.

The experiment result indicates that our algorithm is more efficient than the competing algorithms, reaching 10^{-3} for optimality gap and 10^{-4} for constraint violation in a few seconds. The cutting-plane method can also reach the same level of accuracy, for a much longer computational time though. In fact, in this experiment, the cutting-plane method iterated only twice during the 5x seconds of execution. The missing part of the "CP" curve at the beginning is due to the fact that the cutting-plane method does not require an initial point $\boldsymbol{x}^0$ and an $\boldsymbol{x}$-iterate is generated only after the first optimization step is completed.

6 Conclusion

Based on a novel max-min-max perspective, this paper devised an iterative algorithm, ProM^3, for solving RO problems. The algorithm ProM^3 operates directly on the model functions and sets through their gradient and projection oracles, respectively. Such a feature is highly desirable, as it allows for easy exploitation of useful problem structures, and makes the algorithm particularly suitable for contemporary large-scale decision problems. Theoretically, we proved that ProM^3 enjoys strong convergence guarantees. We also extended our algorithm to RO problems with projection-unfriendly uncertainty sets. Numerical results under different challenging regimes (high-dimensional, highly constrained and/or highly non-linear) demonstrated the promising performance of ProM^3 and its extension.

References

Ben-Tal, Aharon, Dick den Hertog, Anja De Waegenaere, Bertrand Melenberg, Gijs Rennen. 2013. Robust solutions of optimization problems affected by uncertain probabilities. *Management Science* **59**(2) 341–357.

Ben-Tal, Aharon, Dick den Hertog, Jean-Philippe Vial. 2015a. Deriving robust counterparts of nonlinear uncertain inequalities. *Mathematical Programming* **149**(1-2) 265–299.

Ben-Tal, Aharon, Laurent El Ghaoui, Arkadi Nemirovski. 2009. *Robust Optimization*, vol. 28. Princeton University Press.

Ben-Tal, Aharon, Elad Hazan, Tomer Koren, Shie Mannor. 2015b. Oracle-based robust optimization via online learning. *Operations Research* **63**(3) 628–638.

Ben-Tal, Aharon, Arkadi Nemirovski. 1998. Robust convex optimization. *Mathematics of Operations Research* **23**(4) 769–805.

Bertsekas, Dimitri. 1999. *Nonlinear Programming*. Athena Scientific.

Bertsimas, Dimitris, Iain Dunning, Miles Lubin. 2016. Reformulation versus cutting-planes for robust optimization: A computational study. *Computational Management Science* **13** 195–217.

Bertsimas, Dimitris, Dick den Hertog. 2022. *Robust and adaptive optimization*. Dynamic Ideas LLC.

Bertsimas, Dimitris, Shimrit Shtern, Bradley Sturt. 2022. Two-stage sample robust optimization. *Operations Research* **70**(1) 624–640.

Bertsimas, Dimitris, Shimrit Shtern, Bradley Sturt. 2023. A data-driven approach to multistage stochastic linear optimization. *Management Science* **69**(1) 51–74.

Boyd, Stephen, Lieven Vandenberghe. 2004. *Convex optimization*. Cambridge university press.

Chambolle, Antonin, Thomas Pock. 2011. A first-order primal-dual algorithm for convex problems with applications to imaging. *Journal of Mathematical Imaging and Vision* **40** 120–145.

Chambolle, Antonin, Thomas Pock. 2016. On the ergodic convergence rates of a first-order primal-dual algorithm. *Mathematical Programming* **159**(1-2) 253–287.

Gao, Rui, Anton Kleywegt. 2023. Distributionally robust stochastic optimization with Wasserstein distance. *Mathematics of Operations Research* **48**(2) 603–655.

Gregory, Christine, Ken Darby-Dowman, Gautam Mitra. 2011. Robust optimization and portfolio selection: The cost of robustness. *European Journal of Operational Research* **212**(2) 417–428.

Hazan, Elad. 2022. *Introduction to online convex optimization*. MIT Press.

Ho-Nguyen, Nam, Fatma Kılınç-Karzan. 2018. Online first-order framework for robust convex optimization. *Operations Research* **66**(6) 1670–1692.

Ho-Nguyen, Nam, Fatma Kılınç-Karzan. 2019. Exploiting problem structure in optimization under uncertainty via online convex optimization. *Mathematical Programming* **177**(1-2) 113–147.

Kelley, James. 1960. The cutting-plane method for solving convex programs. *Journal of the Society for Industrial and Applied Mathematics* **8**(4) 703–712.

Liu, Huikang, Man-Chung Yue, Anthony Man-Cho So. 2023. A unified approach to synchronization problems over subgroups of the orthogonal group. *Applied and Computational Harmonic Analysis* .

Meng, Fanwen, Jin Qi, Meilin Zhang, James Ang, Singfat Chu, Melvyn Sim. 2015. A robust optimization model for managing elective admission in a public hospital. *Operations Research* **63**(6) 1452–1467.

Mitchell, John. 2009. Cutting plane methods and subgradient methods. *Decision Technologies and Applications*. INFORMS, 34–61.

Mohajerin Esfahani, Peyman, Daniel Kuhn. 2018. Data-driven distributionally robust optimization using the Wasserstein metric: Performance guarantees and tractable reformulations. *Mathematical Programming* **171**(1-2) 115–166.

Mutapcic, Almir, Stephen Boyd. 2009. Cutting-set methods for robust convex optimization with pessimizing oracles. *Optimization Methods & Software* **24**(3) 381–406.

Nesterov, Yurii, Arkadii Nemirovskii. 1994. *Interior-Point Polynomial Algorithms in Convex Programming*. SIAM.

Polak, Elijah, Johannes Royset. 2003. Algorithms for finite and semi-infinite min-max-min problems using adaptive smoothing techniques. *Journal of Optimization Theory and Applications* **119** 421–457.

Postek, Krzysztof, Shimrit Shtern. 2021. First-order algorithms for robust optimization problems via convex-concave saddle-point Lagrangian reformulation. *arXiv preprint arXiv:2101.02669*.

Rockafellar, Tyrrell. 1970. *Convex Analysis*. Princeton University Press.

Singla, Manisha, Debdas Ghosh, KK Shukla. 2020. A survey of robust optimization based machine learning with special reference to support vector machines. *International Journal of Machine Learning and Cybernetics* **11**(7) 1359–1385.

Toh, Kim-Chuan, Sangwoon Yun. 2010. An accelerated proximal gradient algorithm for nuclear norm regularized linear least squares problems. *Pacific Journal of Optimization* **6**(615-640) 15.

Wiesemann, Wolfram, Daniel Kuhn, Melvyn Sim. 2014. Distributionally robust convex optimization. *Operations Research* **62**(6) 1358–1376.

Wu, Sissi Xiaoxiao, Man-Chung Yue, Anthony Man-Cho So, Wing-Kin Ma. 2017. SDR approximation bounds for the robust multicast beamforming problem with interference temperature constraints. *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 4054–4058.

Xie, Weijun. 2020. Tractable reformulations of two-stage distributionally robust linear programs over the type-∞ Wasserstein ball. *Operations Research Letters* **48**(4) 513–523.

Yue, Man-Chung, Daniel Kuhn, Wolfram Wiesemann. 2022. On linear optimization over Wasserstein balls. *Mathematical Programming* **195**(1) 1107–1122.

A Auxiliary Results

Proof of Proposition 1. By Corollary 37.6.2 in Rockafellar (1970), $\mathcal{F}$ has a saddle point $(\tilde{\boldsymbol{u}}, \tilde{\boldsymbol{v}}) \in \mathcal{U} \times \mathcal{V}$. By definition,

$$\mathcal{F}(\tilde{\boldsymbol{u}}, \boldsymbol{v}) \leq \mathcal{F}(\tilde{\boldsymbol{u}}, \tilde{\boldsymbol{v}}) \leq \mathcal{F}(\boldsymbol{u}, \tilde{\boldsymbol{v}}) \quad \forall (\boldsymbol{u}, \boldsymbol{v}) \in \mathcal{U} \times \mathcal{V}, \tag{8}$$

which implies in particular that $\mathbf{0} \in \partial_{\boldsymbol{u}} \mathcal{F}(\tilde{\boldsymbol{u}}, \tilde{\boldsymbol{v}})$. By strong convexity of $\mathcal{F}(\cdot, \tilde{\boldsymbol{v}})$, we then have

$$\mathcal{F}(\boldsymbol{u}, \tilde{\boldsymbol{v}}) \geq \mathcal{F}(\tilde{\boldsymbol{u}}, \tilde{\boldsymbol{v}}) + \frac{\sigma}{2} \|\boldsymbol{u} - \tilde{\boldsymbol{u}}\|_2^2 \quad \forall \boldsymbol{u} \in \mathcal{U}.$$

This, together with (8), implies that

$$\mathcal{F}(\tilde{\boldsymbol{u}}, \boldsymbol{v}) \leq \mathcal{F}(\boldsymbol{u}, \tilde{\boldsymbol{v}}) - \frac{\sigma}{2} \|\boldsymbol{u} - \tilde{\boldsymbol{u}}\|_2^2 \quad \forall (\boldsymbol{u}, \boldsymbol{v}) \in \mathcal{U} \times \mathcal{V},$$

concluding the proof. □

Proof of Proposition 2. Each f_m is convex since it is a point-wise supremum of a family of convex functions. Then, by Assumption 1(*iii*)-(*iv*) as well as Corollary 28.2.1 and Theorem 28.3 in Rockafellar (1970), the max-min problem

$$\max_{\boldsymbol{\lambda} \geq \mathbf{0}} \min_{\boldsymbol{x} \in \mathcal{X}} \left\{ f_0(\boldsymbol{x}) + \sum_{m \in [M]} \lambda_m f_m(\boldsymbol{x}) \right\}$$

admits a saddle point solution, and it is equivalent to the Robust problem in the sense that the saddle value equals the optimal value of the Robust problem and that for any saddle point $(\boldsymbol{\lambda}, \boldsymbol{x})$ solving the max-min problem, $\boldsymbol{x}$ is an optimal solution to the Robust problem. Unfolding the definitions of $\mathcal{K}$ and f_m completes the proof. □

B Outer Convergence Analysis

We first prove that the function $\boldsymbol{f}$ is Lipschitz continuous on $\mathcal{X}$.

Proposition 4. *Suppose that Assumption 1(i)-(ii) and Assumption 2 hold. Then,*

$$\|\boldsymbol{f}(\boldsymbol{x}) - \boldsymbol{f}(\boldsymbol{y})\|_2 \leq \mathrm{Lip}_{\boldsymbol{f}} \|\boldsymbol{x} - \boldsymbol{y}\|_2 \quad \forall \boldsymbol{x}, \boldsymbol{y} \in \mathcal{X},$$

where $\mathrm{Lip}_{\boldsymbol{f}} = \sqrt{\sum_{m \in [M]} D_m^2}$.

Proof of Proposition 4. By the definition of f_m, Assumption 1(i)-(ii) and Danskin's theorem (see, *e.g.*, Bertsekas 1999), we have that for any $m \in [M]$ and $\boldsymbol{x} \in \mathcal{X}$,

$$\partial f_m(\boldsymbol{x}) = \mathbf{conv}\left(\{\boldsymbol{\xi}_m \mid \boldsymbol{\xi}_m \in \partial_{\boldsymbol{x}} g_m(\boldsymbol{x}, \boldsymbol{z}_m^\star),\ g_m(\boldsymbol{x}, \boldsymbol{z}_m^\star) = f_m(\boldsymbol{x})\}\right),$$

where $\mathbf{conv}(\cdot)$ denotes the convex hull. From Assumption 2, any element in $\partial f_m(\boldsymbol{x})$ has its norm bounded by D_m. By convexity, for any $\boldsymbol{x}, \boldsymbol{y} \in \mathcal{X}$ and $\hat{\boldsymbol{\xi}}_m \in \partial f_m(\boldsymbol{x})$,

$$f_m(\boldsymbol{x}) - f_m(\boldsymbol{y}) \leq \hat{\boldsymbol{\xi}}_m^\top (\boldsymbol{x} - \boldsymbol{y}) \leq D_m \|\boldsymbol{x} - \boldsymbol{y}\|_2,$$

which implies

$$\|\boldsymbol{f}(\boldsymbol{x}) - \boldsymbol{f}(\boldsymbol{y})\|_2 = \sqrt{\sum_{m \in [M]} |f_m(\boldsymbol{x}) - f_m(\boldsymbol{y})|^2} \leq \sqrt{\sum_{m \in [M]} D_m^2 \|\boldsymbol{x} - \boldsymbol{y}\|_2^2} = \sqrt{\sum_{m \in [M]} D_m^2} \|\boldsymbol{x} - \boldsymbol{y}\|_2,$$

completing the proof. □

We then prove a technical lemma regarding the saddle-point gap.

Lemma 1. *Suppose that Assumption 1(i)-(ii) and Assumption 2 hold. Consider the sequence* $\{(\boldsymbol{\lambda}^k, \boldsymbol{x}^k, \boldsymbol{z}^k)\}_{k \in [K]}$ *generated by Algorithm 1 with* $\theta = \frac{1}{K}$, $\nu = \frac{1}{K}$, $\alpha \leq \frac{1}{\mathrm{Lip}_{\boldsymbol{f}}}$ *and* $\beta \leq \frac{1}{2\mathrm{Lip}_{\boldsymbol{f}}}$. *Then, for any* $(\boldsymbol{\lambda}, \boldsymbol{x}) \in \mathbb{R}_+^M \times \mathcal{X}$, *it holds that*

$$\sum_{k \in [K]} \left(\mathcal{L}(\boldsymbol{\lambda}, \boldsymbol{x}^k) - \mathcal{L}(\boldsymbol{\lambda}^k, \boldsymbol{x})\right) \leq \frac{\|\boldsymbol{\lambda}\|_2^2}{2\beta} + \frac{\|\boldsymbol{x} - \boldsymbol{x}^0\|_2^2}{2\alpha} - \frac{\|\boldsymbol{\lambda} - \boldsymbol{\lambda}^K\|_2^2}{4\beta} + \frac{3\sqrt{M}}{K} \sum_{k \in [K]} \|\boldsymbol{\lambda}^k\|_2 + \left(3\sqrt{M}\|\boldsymbol{\lambda}\|_2 + 1\right).$$

Proof of Lemma 1. Fix any $(\boldsymbol{\lambda}, \boldsymbol{x}) \in \mathbb{R}_+^M \times \mathcal{X}$. Since $\mathcal{L}(\boldsymbol{\lambda}, \boldsymbol{x}) = f_0(\boldsymbol{x}) + \boldsymbol{\lambda}^\top \boldsymbol{f}(\boldsymbol{x})$,

$$\begin{aligned} \sum_{k\in[K]} \Big(\mathcal{L}(\boldsymbol{\lambda}, \boldsymbol{x}^k) - \mathcal{L}(\boldsymbol{\lambda}^k, \boldsymbol{x})\Big) &= \sum_{k=0}^{K-1} \Big(\mathcal{L}(\boldsymbol{\lambda}, \boldsymbol{x}^{k+1}) - \mathcal{L}(\boldsymbol{\lambda}^{k+1}, \boldsymbol{x})\Big) \\ = \sum_{k=0}^{K-1} \Big(\mathcal{L}(\boldsymbol{\lambda}^{k+1}, \boldsymbol{x}^{k+1}) - \mathcal{L}(\boldsymbol{\lambda}^{k+1}, \boldsymbol{x}) &+ \mathcal{L}(\boldsymbol{\lambda}, \boldsymbol{x}^{k+1}) - \mathcal{L}(\boldsymbol{\lambda}^{k+1}, \boldsymbol{x}^{k+1})\Big) \qquad (9) \\ = \sum_{k=0}^{K-1} \Big(\mathcal{L}(\boldsymbol{\lambda}^{k+1}, \boldsymbol{x}^{k+1}) - \mathcal{L}(\boldsymbol{\lambda}^{k+1}, \boldsymbol{x}) &+ (\boldsymbol{\lambda} - \boldsymbol{\lambda}^{k+1})^\top \boldsymbol{f}(\boldsymbol{x}^{k+1})\Big). \end{aligned}$$

Next, for any $\boldsymbol{z} \in \mathcal{Z}$, we have

$$\begin{aligned} &\mathcal{K}(\boldsymbol{\lambda}^{k+1}, \boldsymbol{x}^{k+1}, \boldsymbol{z}) - \mathcal{L}(\boldsymbol{\lambda}^{k+1}, \boldsymbol{x}) \le \mathcal{K}(\boldsymbol{\lambda}^{k+1}, \boldsymbol{x}^{k+1}, \boldsymbol{z}) - \mathcal{K}(\boldsymbol{\lambda}^{k+1}, \boldsymbol{x}, \tilde{\boldsymbol{z}}^{k+1}) \\ &= f_0(\boldsymbol{x}^{k+1}) + {\boldsymbol{\lambda}^{k+1}}^\top \boldsymbol{g}(\boldsymbol{x}^{k+1}, \boldsymbol{z}) + \frac{1}{2\alpha}\|\boldsymbol{x}^{k+1} - \boldsymbol{x}^k\|_2^2 - f_0(\boldsymbol{x}) - {\boldsymbol{\lambda}^{k+1}}^\top \boldsymbol{g}(\boldsymbol{x}, \tilde{\boldsymbol{z}}^{k+1}) - \frac{1}{2\alpha}\|\boldsymbol{x} - \boldsymbol{x}^k\|_2^2 \\ &\qquad + \frac{1}{2\alpha}\|\boldsymbol{x} - \boldsymbol{x}^k\|_2^2 - \frac{1}{2\alpha}\|\boldsymbol{x}^{k+1} - \boldsymbol{x}^k\|_2^2 \\ &\le \frac{1}{2\alpha}\Big(\|\boldsymbol{x} - \boldsymbol{x}^k\|_2^2 - \|\boldsymbol{x} - \boldsymbol{x}^{k+1}\|_2^2 - \|\boldsymbol{x}^k - \boldsymbol{x}^{k+1}\|_2^2\Big) + \nu, \end{aligned}$$

where the first inequality (resp., equality) follows from the definition of $\mathcal{L}$ (resp., $\mathcal{K}$) and the second inequality from the fact that $(\boldsymbol{x}^{k+1}, \tilde{\boldsymbol{z}}^{k+1})$ is a strong ν-approximate saddle point of the Inner problem (see Algorithm 1). Maximizing the left-hand side w.r.t. $\boldsymbol{z}$ over $\mathcal{Z}$ yields

$$\mathcal{L}(\boldsymbol{\lambda}^{k+1}, \boldsymbol{x}^{k+1}) - \mathcal{L}(\boldsymbol{\lambda}^{k+1}, \boldsymbol{x}) \le \frac{1}{2\alpha}\Big(\|\boldsymbol{x} - \boldsymbol{x}^k\|_2^2 - \|\boldsymbol{x} - \boldsymbol{x}^{k+1}\|_2^2 - \|\boldsymbol{x}^k - \boldsymbol{x}^{k+1}\|_2^2\Big) + \nu. \qquad (10)$$

Also, since $\boldsymbol{\lambda}^{k+1} = \operatorname{argmax}_{\boldsymbol{\lambda}\ge\mathbf{0}} \boldsymbol{\lambda}^\top\Big(2\boldsymbol{g}(\boldsymbol{x}^k, \boldsymbol{z}^k) - \boldsymbol{g}(\boldsymbol{x}^{k-1}, \boldsymbol{z}^{k-1})\Big) - \frac{1}{2\beta}\|\boldsymbol{\lambda} - \boldsymbol{\lambda}^k\|_2^2$, it follows from the optimality of $\boldsymbol{\lambda}^{k+1}$ and the strong concavity that

$$\frac{1}{2\beta}\Big(\|\boldsymbol{\lambda} - \boldsymbol{\lambda}^k\|_2^2 - \|\boldsymbol{\lambda} - \boldsymbol{\lambda}^{k+1}\|_2^2 - \|\boldsymbol{\lambda}^k - \boldsymbol{\lambda}^{k+1}\|_2^2\Big) - (\boldsymbol{\lambda} - \boldsymbol{\lambda}^{k+1})^\top(2\boldsymbol{g}(\boldsymbol{x}^k, \boldsymbol{z}^k) - \boldsymbol{g}(\boldsymbol{x}^{k-1}, \boldsymbol{z}^{k-1})) \ge 0.$$

Hence,

$$\begin{aligned} (\boldsymbol{\lambda} - \boldsymbol{\lambda}^{k+1})^\top \boldsymbol{f}(\boldsymbol{x}^{k+1}) \le &\frac{1}{2\beta}\Big(\|\boldsymbol{\lambda} - \boldsymbol{\lambda}^k\|_2^2 - \|\boldsymbol{\lambda} - \boldsymbol{\lambda}^{k+1}\|_2^2 - \|\boldsymbol{\lambda}^k - \boldsymbol{\lambda}^{k+1}\|_2^2\Big) \\ &+ (\boldsymbol{\lambda} - \boldsymbol{\lambda}^{k+1})^\top\Big(\boldsymbol{f}(\boldsymbol{x}^{k+1}) - (2\boldsymbol{g}(\boldsymbol{x}^k, \boldsymbol{z}^k) - \boldsymbol{g}(\boldsymbol{x}^{k-1}, \boldsymbol{z}^{k-1}))\Big). \end{aligned} \qquad (11)$$

The last term on the right-hand side of (11) satisfies that

$$\begin{aligned}
&(\boldsymbol{\lambda}-\boldsymbol{\lambda}^{k+1})^\top \Big(\boldsymbol{f}(\boldsymbol{x}^{k+1})-(2\boldsymbol{g}(\boldsymbol{x}^k,\boldsymbol{z}^k)-\boldsymbol{g}(\boldsymbol{x}^{k-1},\boldsymbol{z}^{k-1}))\Big)\\
\le\ &(\boldsymbol{\lambda}-\boldsymbol{\lambda}^{k+1})^\top(\boldsymbol{f}(\boldsymbol{x}^{k+1})-\boldsymbol{f}(\boldsymbol{x}^k))-(\boldsymbol{\lambda}-\boldsymbol{\lambda}^k)^\top(\boldsymbol{f}(\boldsymbol{x}^k)-\boldsymbol{f}(\boldsymbol{x}^{k-1}))\\
&+3\sqrt{M}\theta(\|\boldsymbol{\lambda}\|_2+\|\boldsymbol{\lambda}^{k+1}\|_2)+(\boldsymbol{\lambda}^{k+1}-\boldsymbol{\lambda}^k)^\top(\boldsymbol{f}(\boldsymbol{x}^k)-\boldsymbol{f}(\boldsymbol{x}^{k-1})) \qquad (12)\\
\le\ &(\boldsymbol{\lambda}-\boldsymbol{\lambda}^{k+1})^\top(\boldsymbol{f}(\boldsymbol{x}^{k+1})-\boldsymbol{f}(\boldsymbol{x}^k))-(\boldsymbol{\lambda}-\boldsymbol{\lambda}^k)^\top(\boldsymbol{f}(\boldsymbol{x}^k)-\boldsymbol{f}(\boldsymbol{x}^{k-1}))\\
&+3\sqrt{M}\theta(\|\boldsymbol{\lambda}\|_2+\|\boldsymbol{\lambda}^{k+1}\|_2)+\frac{\mathrm{Lip}_{\boldsymbol{f}}}{2}\|\boldsymbol{\lambda}^{k+1}-\boldsymbol{\lambda}^k\|_2^2+\frac{\mathrm{Lip}_{\boldsymbol{f}}}{2}\|\boldsymbol{x}^k-\boldsymbol{x}^{k-1}\|_2^2,
\end{aligned}$$

where the first inequality follows from the fact that

$$\|\boldsymbol{f}(\boldsymbol{x}^k)-\boldsymbol{g}(\boldsymbol{x}^k,\boldsymbol{z}^k)\|_2=\sqrt{\sum_{m\in[M]}(f_m(\boldsymbol{x}^k)-g_m(\boldsymbol{x}^k,\boldsymbol{z}_m^k))^2}\le\sqrt{M}\theta \quad \forall k\ge 0,$$

and the second inequality from Proposition 4. Substituting (12) into (11), we get

$$\begin{aligned}
(\boldsymbol{\lambda}-\boldsymbol{\lambda}^{k+1})^\top\boldsymbol{f}(\boldsymbol{x}^{k+1})\le\ &\frac{1}{2\beta}\Big(\|\boldsymbol{\lambda}-\boldsymbol{\lambda}^k\|_2^2-\|\boldsymbol{\lambda}-\boldsymbol{\lambda}^{k+1}\|_2^2-\|\boldsymbol{\lambda}^k-\boldsymbol{\lambda}^{k+1}\|_2^2\Big)\\
&+(\boldsymbol{\lambda}-\boldsymbol{\lambda}^{k+1})^\top(\boldsymbol{f}(\boldsymbol{x}^{k+1})-\boldsymbol{f}(\boldsymbol{x}^k))-(\boldsymbol{\lambda}-\boldsymbol{\lambda}^k)^\top(\boldsymbol{f}(\boldsymbol{x}^k)-\boldsymbol{f}(\boldsymbol{x}^{k-1}))\\
&+3\sqrt{M}\theta(\|\boldsymbol{\lambda}\|_2+\|\boldsymbol{\lambda}^{k+1}\|_2)+\frac{\mathrm{Lip}_{\boldsymbol{f}}}{2}\|\boldsymbol{\lambda}^{k+1}-\boldsymbol{\lambda}^k\|_2^2+\frac{\mathrm{Lip}_{\boldsymbol{f}}}{2}\|\boldsymbol{x}^k-\boldsymbol{x}^{k-1}\|_2^2,
\end{aligned}$$

Then, substituting the above inequality and inequality (10) into (9), we obtain

$$
\begin{aligned}
&\sum_{k\in[K]} \Big(\mathcal{L}(\boldsymbol{\lambda}, \boldsymbol{x}^k) - \mathcal{L}(\boldsymbol{\lambda}^k, \boldsymbol{x})\Big) \\
\le\ & \frac{\|\boldsymbol{x}-\boldsymbol{x}^0\|_2^2}{2\alpha} + \frac{\|\boldsymbol{\lambda}\|_2^2}{2\beta} - \frac{\|\boldsymbol{x}-\boldsymbol{x}^K\|_2^2}{2\alpha} - \frac{\|\boldsymbol{\lambda}-\boldsymbol{\lambda}^K\|_2^2}{2\beta} - \frac{\|\boldsymbol{x}^K-\boldsymbol{x}^{K-1}\|^2}{2\alpha} \\
& + (\boldsymbol{\lambda}-\boldsymbol{\lambda}^K)^\top(\boldsymbol{f}(\boldsymbol{x}^K) - \boldsymbol{f}(\boldsymbol{x}^{K-1})) + 3\sqrt{M}\theta \sum_{k\in[K]} \|\boldsymbol{\lambda}^k\|_2 + 3\sqrt{M}\theta K \|\boldsymbol{\lambda}\|_2 + \nu K \\
& - \left(\frac{1}{2\beta} - \frac{\mathrm{Lip}_{\boldsymbol{f}}}{2}\right) \sum_{k\in[K]} \|\boldsymbol{\lambda}^k - \boldsymbol{\lambda}^{k-1}\|_2^2 - \left(\frac{1}{2\alpha} - \frac{\mathrm{Lip}_{\boldsymbol{f}}}{2}\right) \sum_{k\in[K-1]} \|\boldsymbol{x}^k - \boldsymbol{x}^{k-1}\|_2^2 \\
\le\ & \frac{\|\boldsymbol{x}-\boldsymbol{x}^0\|_2^2}{2\alpha} + \frac{\|\boldsymbol{\lambda}\|_2^2}{2\beta} - \frac{\|\boldsymbol{x}-\boldsymbol{x}^K\|_2^2}{2\alpha} - \frac{\|\boldsymbol{\lambda}-\boldsymbol{\lambda}^K\|_2^2}{4\beta} + 3\sqrt{M}\theta \sum_{k\in[K]} \|\boldsymbol{\lambda}^k\|_2 + \Big(3\sqrt{M}\theta\|\boldsymbol{\lambda}\|_2 + \nu\Big) K, \\
& - \left(\frac{1}{4\beta} - \frac{\mathrm{Lip}_{\boldsymbol{f}}}{2}\right) \|\boldsymbol{\lambda}-\boldsymbol{\lambda}^K\|_2^2 - \left(\frac{1}{2\alpha} - \frac{\mathrm{Lip}_{\boldsymbol{f}}}{2}\right) \sum_{k\in[K]} \|\boldsymbol{x}^k - \boldsymbol{x}^{k-1}\|_2^2 - \left(\frac{1}{2\beta} - \frac{\mathrm{Lip}_{\boldsymbol{f}}}{2}\right) \sum_{k\in[K]} \|\boldsymbol{\lambda}^k - \boldsymbol{\lambda}^{k-1}\|_2^2 \\
\le\ & \frac{\|\boldsymbol{x}-\boldsymbol{x}^0\|_2^2}{2\alpha} + \frac{\|\boldsymbol{\lambda}\|_2^2}{2\beta} - \frac{\|\boldsymbol{\lambda}-\boldsymbol{\lambda}^K\|_2^2}{4\beta} + \frac{3\sqrt{M}}{K} \sum_{k\in[K]} \|\boldsymbol{\lambda}^k\|_2 + \Big(3\sqrt{M}\|\boldsymbol{\lambda}\|_2 + 1\Big),
\end{aligned}
$$

where the first inequality follows from telescoping and the initial conditions $\boldsymbol{\lambda}^0 = \mathbf{0}$ and $\boldsymbol{x}^{-1} = \boldsymbol{x}^0$, and the second from the inequality

$$
(\boldsymbol{\lambda}-\boldsymbol{\lambda}^K)^\top(\boldsymbol{f}(\boldsymbol{x}^K) - \boldsymbol{f}(\boldsymbol{x}^{K-1})) \le \frac{\mathrm{Lip}_{\boldsymbol{f}}}{2}(\|\boldsymbol{\lambda}-\boldsymbol{\lambda}^K\|_2^2 + \|\boldsymbol{x}^K - \boldsymbol{x}^{K-1}\|_2^2),
$$

and the third from the choice of θ, ν, α and β. This completes the proof. □

The following lemma asserts that the iterator $\boldsymbol{\lambda}^k$ is upper bounded uniformly in k.

Lemma 2. *Suppose that Assumptions 1 and 2 hold. Then, $\mathcal{L}$ admits a saddle point. Moreover, the sequence $\{\boldsymbol{\lambda}^k\}_{k\in[K]}$ generated by the Algorithm 1 with $\theta = \frac{1}{K}$, $\nu = \frac{1}{K}$, $\alpha \le \frac{1}{\mathrm{Lip}_{\boldsymbol{f}}}$ and $\beta \le \frac{1}{2\mathrm{Lip}_{\boldsymbol{f}}}$ satisfies that for any $k \in [K]$,*

$$
\|\boldsymbol{\lambda}^k\|_2^2 \le 12\|\boldsymbol{\lambda}^\star\|_2^2 + \frac{8\beta}{\alpha}\|\boldsymbol{x}^\star - \boldsymbol{x}^0\|_2^2 + 48\beta\sqrt{M}\|\boldsymbol{\lambda}^\star\|_2 + \frac{576\beta M}{\mathrm{Lip}_{\boldsymbol{f}}} + 16\beta,
$$

where $(\boldsymbol{\lambda}^\star, \boldsymbol{x}^\star) \in \mathbb{R}_+^M \times \mathcal{X}$ is an arbitrary saddle point of $\mathcal{L}$.

Proof of Lemma 2. The existence of a saddle point of $\mathcal{L}$ follows from the proof of Proposition 2. By definition, for any $(\boldsymbol{\lambda}, \boldsymbol{x}) \in \mathbb{R}_+^M \times \mathcal{X}$, we have

$$\mathcal{L}(\boldsymbol{\lambda}, \boldsymbol{x}^\star) \leq \mathcal{L}(\boldsymbol{\lambda}^\star, \boldsymbol{x}^\star) \leq \mathcal{L}(\boldsymbol{\lambda}^\star, \boldsymbol{x}). \tag{13}$$

Taking $(\boldsymbol{\lambda}, \boldsymbol{x}) = (\boldsymbol{\lambda}^k, \boldsymbol{x}^k)$, we then have $\mathcal{L}(\boldsymbol{\lambda}^\star, \boldsymbol{x}^k) \geq \mathcal{L}(\boldsymbol{\lambda}^k, \boldsymbol{x}^\star)$ $\forall k \in [K]$, which implies that

$$\frac{\|\boldsymbol{\lambda}^\star - \boldsymbol{\lambda}^{k'}\|_2^2}{4\beta} \leq \sum_{k \in [k']} \Big(\mathcal{L}(\boldsymbol{\lambda}^\star, \boldsymbol{x}^k) - \mathcal{L}(\boldsymbol{\lambda}^k, \boldsymbol{x}^\star) \Big) + \frac{\|\boldsymbol{\lambda}^\star - \boldsymbol{\lambda}^{k'}\|_2^2}{4\beta} \ \forall k' \in [K]. \tag{14}$$

Using Lemma 1 with $(\boldsymbol{\lambda}, \boldsymbol{x}) = (\boldsymbol{\lambda}^\star, \boldsymbol{x}^\star)$ and (14), we obtain

$$\begin{aligned}
\frac{\|\boldsymbol{\lambda}^\star - \boldsymbol{\lambda}^{k'}\|_2^2}{4\beta} &\leq \frac{\|\boldsymbol{\lambda}^\star\|_2^2}{2\beta} + \frac{\|\boldsymbol{x}^\star - \boldsymbol{x}^0\|_2^2}{2\alpha} + 3\sqrt{M}\theta \sum_{k \in [k']} \|\boldsymbol{\lambda}^k\|_2 + 3\sqrt{M}\|\boldsymbol{\lambda}^\star\|_2 + 1 \\
&\leq \frac{\|\boldsymbol{\lambda}^\star\|_2^2}{2\beta} + \frac{\|\boldsymbol{x}^\star - \boldsymbol{x}^0\|_2^2}{2\alpha} + \sum_{k \in [k']} \left(\frac{\|\boldsymbol{\lambda}^k\|_2^2 \mathrm{Lip}_{\boldsymbol{f}}}{16K} + \frac{36KM\theta^2}{\mathrm{Lip}_{\boldsymbol{f}}} \right) + 3\sqrt{M}\|\boldsymbol{\lambda}^\star\|_2 + 1 \\
&\leq \frac{\|\boldsymbol{\lambda}^\star\|_2^2}{2\beta} + \frac{\|\boldsymbol{x}^\star - \boldsymbol{x}^0\|_2^2}{2\alpha} + \frac{\mathrm{Lip}_{\boldsymbol{f}}}{16K} \sum_{k \in [k']} \|\boldsymbol{\lambda}^k\|_2^2 + \frac{36M}{\mathrm{Lip}_{\boldsymbol{f}}} + 3\sqrt{M}\|\boldsymbol{\lambda}^\star\|_2 + 1,
\end{aligned}$$

where the last inequality follows from the fact that $\theta \leq \frac{1}{K}$. Since $\beta \leq \frac{1}{2\mathrm{Lip}_{\boldsymbol{f}}}$,

$$\|\boldsymbol{\lambda}^\star - \boldsymbol{\lambda}^{k'}\|_2^2 \leq 2\|\boldsymbol{\lambda}^\star\|_2^2 + \frac{2\beta\|\boldsymbol{x}^\star - \boldsymbol{x}^0\|_2^2}{\alpha} + \frac{1}{8K} \sum_{k \in [k']} \|\boldsymbol{\lambda}^k\|_2^2 + \frac{144\beta M}{\mathrm{Lip}_{\boldsymbol{f}}} + 12\beta\sqrt{M}\|\boldsymbol{\lambda}^\star\|_2 + 4\beta.$$

For simplicity, we denote

$$\Gamma = 12\|\boldsymbol{\lambda}^\star\|_2^2 + \frac{8\beta}{\alpha}\|\boldsymbol{x}^\star - \boldsymbol{x}^0\|_2^2 + 48\beta\sqrt{M}\|\boldsymbol{\lambda}^\star\|_2 + \frac{576\beta M}{\mathrm{Lip}_{\boldsymbol{f}}} + 16\beta. \tag{15}$$

Then, the above inequality implies that for any $k' \in [K]$,

$$\|\boldsymbol{\lambda}^\star - \boldsymbol{\lambda}^{k'}\|_2^2 \leq \frac{1}{4}\Gamma - \|\boldsymbol{\lambda}^\star\|_2^2 + \frac{1}{8K} \sum_{k \in [K]} \|\boldsymbol{\lambda}^k\|_2^2.$$

This, together with the fact that $\|\boldsymbol{\lambda}^{k'}\|_2^2 \leq 2\|\boldsymbol{\lambda}^\star - \boldsymbol{\lambda}^{k'}\|_2^2 + 2\|\boldsymbol{\lambda}^\star\|_2^2$ for all $k' \in [K]$, yields

$$\|\boldsymbol{\lambda}^{k'}\|_2^2 \leq \frac{1}{2}\Gamma + \frac{1}{4K} \sum_{k \in [K]} \|\boldsymbol{\lambda}^k\|_2^2 \quad \forall k' \in [K]. \tag{16}$$

Next, we prove by induction the desired result that $\|\boldsymbol{\lambda}^{k'}\|_2 \le \Gamma$ for any $k' \in [K]$. For $k' = 1$, by (16), we have

$$\|\boldsymbol{\lambda}^1\|_2^2 \le \frac{1}{2}\Gamma + \frac{1}{4}\|\boldsymbol{\lambda}^1\|_2^2 \le \frac{1}{2}\Gamma + \frac{1}{2}\|\boldsymbol{\lambda}^1\|_2^2,$$

which implies

$$\|\boldsymbol{\lambda}^1\|_2^2 \le \Gamma.$$

Next, assume $\|\boldsymbol{\lambda}^{k'}\|_2 \le \Gamma$ holds for some $k' \in [1, K)$. By (16) we then have

$$\begin{aligned}\|\boldsymbol{\lambda}^{k'+1}\|_2^2 &\le \frac{1}{2}\Gamma + \frac{1}{4K}\sum_{k=1}^{k'+1}\|\boldsymbol{\lambda}^k\|_2^2 = \frac{1}{2}\Gamma + \frac{1}{4}\|\boldsymbol{\lambda}^{k'+1}\|_2^2 + \frac{1}{4K}\sum_{k=1}^{k'}\|\boldsymbol{\lambda}^k\|_2^2 \\ &\le \frac{1}{2}\Gamma + \frac{1}{4}\|\boldsymbol{\lambda}^{k'+1}\|_2^2 + \frac{1}{4K}\sum_{k=1}^{k'}\Gamma \le \frac{3}{4}\Gamma + \frac{1}{4}\|\boldsymbol{\lambda}^{k'+1}\|_2^2,\end{aligned}$$

which concludes our proof. □

We are now ready to prove Theorem 1.

Proof of Theorem 1. To bound $f_0(\bar{\boldsymbol{x}}^K) - f_0(\boldsymbol{x}^\star)$, we let $\bar{\boldsymbol{\lambda}}^K = \frac{1}{K}\sum_{k\in[K]}\boldsymbol{\lambda}^k$. By the convexity of $\mathcal{L}(\boldsymbol{\lambda}, \cdot)$ and $-\mathcal{L}(\cdot, \boldsymbol{x})$, and Lemmas 1 and 2, we have that for any $(\boldsymbol{\lambda}, \boldsymbol{x}) \in \mathbb{R}_+^M \times \mathcal{X}$,

$$\mathcal{L}(\boldsymbol{\lambda}, \bar{\boldsymbol{x}}^K) - \mathcal{L}(\bar{\boldsymbol{\lambda}}^K, \boldsymbol{x}) \le \frac{1}{K}\left(\frac{\|\boldsymbol{\lambda}\|_2^2}{2\beta} + \frac{\|\boldsymbol{x} - \boldsymbol{x}^0\|_2^2}{2\alpha} + 3\sqrt{M\Gamma} + 3\sqrt{M}\|\boldsymbol{\lambda}\|_2 + 1\right), \tag{17}$$

where Γ is the constant defined in (15). Then,

$$\begin{aligned}&f_0(\bar{\boldsymbol{x}}^K) - f_0(\boldsymbol{x}^\star) = \mathcal{L}(\boldsymbol{0}, \bar{\boldsymbol{x}}^K) - \mathcal{L}(\boldsymbol{\lambda}^\star, \boldsymbol{x}^\star) \le \mathcal{L}(\boldsymbol{0}, \bar{\boldsymbol{x}}^K) - \mathcal{L}(\bar{\boldsymbol{\lambda}}, \boldsymbol{x}^\star) \\ &\le \frac{1}{K}\left(\frac{\|\boldsymbol{x}^\star - \boldsymbol{x}^0\|_2^2}{2\alpha} + 3\sqrt{M\Gamma} + 1\right),\end{aligned}$$

where the equality follows from the definition of $\mathcal{L}$, the first inequality from (13), and the second inequality from (17).

Next, we bound $\max_{m\in[M]}[f_m(\bar{\boldsymbol{x}}^K)]_+$. By the definition of $\mathcal{L}$ and (13), we have

$$f_0(\bar{\boldsymbol{x}}^K) - f_0(\boldsymbol{x}^\star) \ge -\sum_{m\in[M]}\lambda_m^\star f_m(\bar{\boldsymbol{x}}^K) \ge -\sum_{m\in[M]}\lambda_m^\star[f_m(\bar{\boldsymbol{x}}^K)]_+. \tag{18}$$

Let $\hat{\boldsymbol{\lambda}}$ be the vector defined by $\hat{\lambda}_m = 1 + \lambda_m^\star$ if $f_m(\bar{\boldsymbol{x}}^K) > 0$, and $\hat{\lambda}_m = 0$ otherwise, for any $m \in [M]$. By the definition of $\mathcal{L}$, (13) and (17), we have

$$\begin{aligned} &f_0(\bar{\boldsymbol{x}}^K) - f_0(\boldsymbol{x}^\star) + \sum_{m\in[M]} \hat{\lambda}_m f_m(\bar{\boldsymbol{x}}^K) = \mathcal{L}(\hat{\boldsymbol{\lambda}}, \bar{\boldsymbol{x}}^K) - \mathcal{L}(\boldsymbol{\lambda}^\star, \boldsymbol{x}^\star) \\ \le\, &\mathcal{L}(\hat{\boldsymbol{\lambda}}, \bar{\boldsymbol{x}}^K) - \mathcal{L}(\bar{\boldsymbol{\lambda}}^K, \boldsymbol{x}^\star) \le \frac{1}{K}\left(\frac{\|\hat{\boldsymbol{\lambda}}\|_2^2}{2\beta} + \frac{\|\boldsymbol{x}^\star - \boldsymbol{x}^0\|_2^2}{2\alpha} + 3\sqrt{M}\Gamma + 3\sqrt{M}\|\hat{\boldsymbol{\lambda}}\|_2 + 1\right), \end{aligned}$$

which, together with (18), yields

$$\begin{aligned} &\max_{m\in[M]} [f_m(\bar{\boldsymbol{x}}^K)]_+ \le \sum_{m\in[M]} [f_m(\bar{\boldsymbol{x}}^K)]_+ + \sum_{m\in[M]} \lambda_m^\star [f_m(\bar{\boldsymbol{x}}^K)]_+ + f_0(\bar{\boldsymbol{x}}^K) - f_0(\boldsymbol{x}^\star) \\ = &f_0(\bar{\boldsymbol{x}}^K) - f_0(\boldsymbol{x}^\star) + \sum_{m\in[M]} \hat{\lambda}_m f_m(\bar{\boldsymbol{x}}^K) \le \frac{1}{K}\left(\frac{\|\hat{\boldsymbol{\lambda}}\|_2^2}{2\beta} + \frac{\|\boldsymbol{x}^\star - \boldsymbol{x}^0\|_2^2}{2\alpha} + 3\sqrt{M}\Gamma + 3\sqrt{M}\|\hat{\boldsymbol{\lambda}}\|_2 + 1\right). \end{aligned}$$

This completes the proof. □

C Inner Convergence Analysis

To analyze the Inner problem, we consider general saddle-point problems of the form

$$\min_{\boldsymbol{u}\in\mathcal{U}} \max_{\boldsymbol{v}\in\mathcal{V}} \hat{\mathcal{F}}(\boldsymbol{u}, \boldsymbol{v}) + \frac{\sigma}{2}\|\boldsymbol{u} - \hat{\boldsymbol{u}}\|_2^2, \tag{19}$$

where $\hat{\mathcal{F}}(\boldsymbol{u}, \boldsymbol{v})$ is convex in $\boldsymbol{u}$ and concave in $\boldsymbol{v}$ (possibly non-linear, non-smooth), $\mathcal{U}$ and $\mathcal{V}$ are non-empty compact convex sets, and $\hat{\boldsymbol{u}} \in \mathcal{U}$. We solve it by the following algorithm.

Proposition 5. *Suppose that $\mathcal{U}$ and $\mathcal{V}$ are nonempty compact convex sets, that $\hat{\mathcal{F}}(\cdot, \boldsymbol{v})$ is convex on $\mathcal{U}$ for any $\boldsymbol{v} \in \mathcal{V}$ and $\hat{\mathcal{F}}(\boldsymbol{u}, \cdot)$ is concave on $\mathcal{V}$ for any $\boldsymbol{u} \in \mathcal{U}$, and that there exist $C_1, C_2 > 0$ such that $\|\boldsymbol{\xi}\|_2 \le C_1$ and $\|\boldsymbol{\zeta}\|_2 \le C_2$ for any $(\boldsymbol{u}, \boldsymbol{v}) \in \mathcal{U} \times \mathcal{V}$, $\boldsymbol{\xi} \in \partial_{\boldsymbol{u}} \hat{\mathcal{F}}(\boldsymbol{u}, \boldsymbol{v})$ and $\boldsymbol{\zeta} \in \partial_{\boldsymbol{v}}(-\hat{\mathcal{F}})(\boldsymbol{u}, \boldsymbol{v})$. Then, the output $(\bar{\boldsymbol{u}}_T, \bar{\boldsymbol{v}}_T)$ of Algorithm 5 with $\gamma, \delta \le \frac{1}{\sqrt{T}}$ is a strong $\mathcal{O}(T^{-1/2})$-approximate saddle point of problem* (19).

Algorithm 5: Modified Projected Subgradient Ascent Descent.

```
Input  : $T \geq 1$, $\delta > 0$, $\gamma > 0$, $\boldsymbol{u}_0 \in \mathcal{U}$, $\boldsymbol{v}_0 \in \mathcal{V}$.
for $t = 0, \ldots, T-1$ do
    ($\boldsymbol{v}$-update) Compute $\boldsymbol{\zeta}_t \in \partial_{\boldsymbol{v}}(-\hat{\mathcal{F}})(\boldsymbol{u}_t, \boldsymbol{v}_t)$. Set
        $\boldsymbol{v}_{t+1} = \mathrm{Proj}_{\mathcal{V}}\left(\boldsymbol{v}_t - \delta\left(2\boldsymbol{\zeta}_t - \boldsymbol{\zeta}_{t-1}\right)\right).$
    ($\boldsymbol{u}$-update) Compute $\boldsymbol{\xi}_t \in \partial_{\boldsymbol{u}}\hat{\mathcal{F}}(\boldsymbol{u}_t, \boldsymbol{v}_{t+1})$. Set
        $\boldsymbol{u}_{t+1} = \mathrm{Proj}_{\mathcal{U}}\left(\frac{1}{1+\gamma\sigma}\left(\gamma\sigma\hat{\boldsymbol{u}} + \boldsymbol{u}_t - \gamma\boldsymbol{\xi}_t\right)\right).$
end
Output: $\bar{\boldsymbol{u}}_T = \frac{1}{T}\sum_{t\in[T]} \boldsymbol{u}_t$ and $\bar{\boldsymbol{v}}_T = \frac{1}{T}\sum_{t\in[T]} \boldsymbol{v}_t$.
```

Proof of Proposition 5. We first note that

$$\begin{aligned}\hat{\mathcal{F}}(\boldsymbol{u}_{t+1}, \boldsymbol{v}) - \hat{\mathcal{F}}(\boldsymbol{u}, \boldsymbol{v}_{t+1}) = \hat{\mathcal{F}}(\boldsymbol{u}_{t+1}, \boldsymbol{v}) - \hat{\mathcal{F}}(\boldsymbol{u}_{t+1}, \boldsymbol{v}_{t+1}) + \hat{\mathcal{F}}(\boldsymbol{u}_{t+1}, \boldsymbol{v}_{t+1}) - \hat{\mathcal{F}}(\boldsymbol{u}_t, \boldsymbol{v}_{t+1}) \\ + \hat{\mathcal{F}}(\boldsymbol{u}_t, \boldsymbol{v}_{t+1}) - \hat{\mathcal{F}}(\boldsymbol{u}, \boldsymbol{v}_{t+1}).\end{aligned} \tag{20}$$

Since $\hat{\mathcal{F}}(\cdot, \boldsymbol{v})$ is convex on $\mathcal{U}$ for any $\boldsymbol{v} \in \mathcal{V}$ and $\hat{\mathcal{F}}(\boldsymbol{u}, \cdot)$ is concave on $\mathcal{V}$ for any $\boldsymbol{u} \in \mathcal{U}$, for any $\boldsymbol{\xi}'_t \in \partial_{\boldsymbol{u}}\hat{\mathcal{F}}(\boldsymbol{u}_{t+1}, \boldsymbol{v}_{t+1})$ and $\boldsymbol{\zeta}_{t+1} \in \partial_{\boldsymbol{v}}(-\hat{\mathcal{F}})(\boldsymbol{u}_{t+1}, \boldsymbol{v}_{t+1})$, we get the three inequalities

$$\begin{aligned}\hat{\mathcal{F}}(\boldsymbol{u}_{t+1}, \boldsymbol{v}) - \hat{\mathcal{F}}(\boldsymbol{u}_{t+1}, \boldsymbol{v}_{t+1}) &\leq -(\boldsymbol{v} - \boldsymbol{v}_{t+1})^\top \boldsymbol{\zeta}_{t+1} \\ \hat{\mathcal{F}}(\boldsymbol{u}_{t+1}, \boldsymbol{v}_{t+1}) - \hat{\mathcal{F}}(\boldsymbol{u}_t, \boldsymbol{v}_{t+1}) &\leq -(\boldsymbol{u}_t - \boldsymbol{u}_{t+1})^\top \boldsymbol{\xi}'_t \\ \hat{\mathcal{F}}(\boldsymbol{u}_t, \boldsymbol{v}_{t+1}) - \hat{\mathcal{F}}(\boldsymbol{u}, \boldsymbol{v}_{t+1}) &\leq -(\boldsymbol{u} - \boldsymbol{u}_t)^\top \boldsymbol{\xi}_t\end{aligned} \tag{21}$$

Noting that

$$\begin{aligned}\boldsymbol{u}_{t+1} &= \operatorname*{argmin}_{\boldsymbol{u}\in\mathcal{U}} \ \boldsymbol{\xi}_t^\top(\boldsymbol{u} - \boldsymbol{u}_t) + \tfrac{\sigma}{2}\|\boldsymbol{u} - \hat{\boldsymbol{u}}\|_2^2 + \tfrac{1}{2\gamma}\|\boldsymbol{u} - \boldsymbol{u}_t\|_2^2 \\ \boldsymbol{v}_{t+1} &= \operatorname*{argmin}_{\boldsymbol{v}\in\mathcal{V}} \ (2\boldsymbol{\zeta}_t - \boldsymbol{\zeta}_{t-1})^\top(\boldsymbol{v} - \boldsymbol{v}_t) + \tfrac{1}{2\delta}\|\boldsymbol{v} - \boldsymbol{v}_t\|_2^2\end{aligned}$$

and using their optimality conditions, we get the two inequalities

$$\begin{aligned}0 \leq &(\boldsymbol{u}-\boldsymbol{u}_{t+1})^\top \boldsymbol{\xi}_t + \frac{1}{2\gamma}(\|\boldsymbol{u}-\boldsymbol{u}_t\|_2^2 - \|\boldsymbol{u}-\boldsymbol{u}_{t+1}\|_2^2 - \|\boldsymbol{u}_{t+1}-\boldsymbol{u}_t\|_2^2)\\ &+ \frac{\sigma}{2}(\|\boldsymbol{u}-\hat{\boldsymbol{u}}\|_2^2 - \|\boldsymbol{u}-\boldsymbol{u}_{t+1}\|_2^2 - \|\boldsymbol{u}_{t+1}-\hat{\boldsymbol{u}}\|_2^2) \quad \text{and}\\ 0 \leq &(\boldsymbol{v}-\boldsymbol{v}_{t+1})^\top \left(2\boldsymbol{\zeta}_t - \boldsymbol{\zeta}_{t-1}\right) + \frac{1}{2\delta}\|\boldsymbol{v}-\boldsymbol{v}_t\|_2^2 - \frac{1}{2\delta}\|\boldsymbol{v}-\boldsymbol{v}_{t+1}\|_2^2 - \frac{1}{2\delta}\|\boldsymbol{v}_{t+1}, \boldsymbol{v}_t\|_2^2.\end{aligned} \tag{22}$$

Combining (20), (21) and (22) yields

$$\begin{aligned}\hat{\mathcal{F}}(\boldsymbol{u}_{t+1}, \boldsymbol{v}) - \hat{\mathcal{F}}(\boldsymbol{u}, \boldsymbol{v}_{t+1}) \leq &\frac{1}{2\gamma}\|\boldsymbol{u}-\boldsymbol{u}_t\|_2^2 - \frac{1}{2\gamma}\|\boldsymbol{u}-\boldsymbol{u}_{t+1}\|_2^2 - \frac{1}{2\gamma}\|\boldsymbol{u}_{t+1}-\boldsymbol{u}_t\|_2^2\\ &+ \frac{1}{2\delta}\|\boldsymbol{v}-\boldsymbol{v}_t\|_2^2 - \frac{1}{2\delta}\|\boldsymbol{v}-\boldsymbol{v}_{t+1}\|_2^2 - \frac{1}{2\delta}\|\boldsymbol{v}_{t+1}-\boldsymbol{v}_t\|_2^2\\ &+ \frac{\sigma}{2}\|\boldsymbol{u}-\hat{\boldsymbol{u}}\|_2^2 - \frac{\sigma}{2}\|\boldsymbol{u}-\boldsymbol{u}_{t+1}\|_2^2 - \frac{\sigma}{2}\|\boldsymbol{u}_{t+1}-\hat{\boldsymbol{u}}\|_2^2\\ &+ (\boldsymbol{v}-\boldsymbol{v}_{t+1})^\top \left(2\boldsymbol{\zeta}_t - \boldsymbol{\zeta}_{t+1} - \boldsymbol{\zeta}_{t-1}\right) + (\boldsymbol{u}_t - \boldsymbol{u}_{t+1})^\top(\boldsymbol{\xi}_t - \boldsymbol{\xi}_t').\end{aligned} \tag{23}$$

Using the bounds on the subgradients and the fact

$$(\boldsymbol{v}_{t+1}-\boldsymbol{v}_t)^\top(\boldsymbol{\zeta}_{t-1}-\boldsymbol{\zeta}_t) \leq \frac{\delta}{2}\|\boldsymbol{\zeta}_{t-1}-\boldsymbol{\zeta}_t\|_2^2 + \frac{1}{2\delta}\|\boldsymbol{v}_{t+1}-\boldsymbol{v}_t\|_2^2 \leq 2\delta C_2^2 + \frac{1}{2\delta}\|\boldsymbol{v}_{t+1}-\boldsymbol{v}_t\|_2^2,$$

we obtain the two inequalities

$$\begin{aligned}(\boldsymbol{u}_t - \boldsymbol{u}_{t+1})^\top(\boldsymbol{\xi}_t - \boldsymbol{\xi}_t') &\leq \frac{1}{2\gamma}\|\boldsymbol{u}_t - \boldsymbol{u}_{t+1}\|_2^2 + 2\gamma C_1^2 \quad \text{and}\\ (\boldsymbol{v}-\boldsymbol{v}_{t+1})^\top \left(2\boldsymbol{\zeta}_t - \boldsymbol{\zeta}_{t+1} - \boldsymbol{\zeta}_{t-1}\right) &\leq (\boldsymbol{v}-\boldsymbol{v}_{t+1})^\top(\boldsymbol{\zeta}_t - \boldsymbol{\zeta}_{t+1}) - (\boldsymbol{v}-\boldsymbol{v}_t)^\top(\boldsymbol{\zeta}_{t-1}-\boldsymbol{\zeta}_t)\\ &\quad + 2\delta C_2^2 + \frac{1}{2\delta}\|\boldsymbol{v}_{t+1}-\boldsymbol{v}_t\|_2^2,\end{aligned} \tag{24}$$

Substituting (24) into (23), we get

$$\begin{aligned}&\hat{\mathcal{F}}(\boldsymbol{u}_{t+1}, \boldsymbol{v}) - \hat{\mathcal{F}}(\boldsymbol{u}, \boldsymbol{v}_{t+1}) + \frac{\sigma}{2}(\|\boldsymbol{u}_{t+1}-\hat{\boldsymbol{u}}\|_2^2 + \|\boldsymbol{u}_{t+1}-\boldsymbol{u}\|_2^2)\\ \leq &\frac{\sigma}{2}\|\boldsymbol{u}-\hat{\boldsymbol{u}}\|_2^2 + \frac{1}{2\gamma}\|\boldsymbol{u}-\boldsymbol{u}_t\|_2^2 + \frac{1}{2\delta}\|\boldsymbol{v}-\boldsymbol{v}_t\|_2^2 - \frac{1}{2\gamma}\|\boldsymbol{u}-\boldsymbol{u}_{t+1}\|_2^2 - \frac{1}{2\delta}\|\boldsymbol{v}-\boldsymbol{v}_{t+1}\|_2^2\\ &+ (\boldsymbol{v}-\boldsymbol{v}_{t+1})^\top(\boldsymbol{\zeta}_t - \boldsymbol{\zeta}_{t+1}) - (\boldsymbol{v}-\boldsymbol{v}_t)^\top(\boldsymbol{\zeta}_{t-1}-\boldsymbol{\zeta}_t) + 2\gamma C_1^2 + 2\delta C_2^2.\end{aligned}$$

Summing over $t = 0, \cdots, T-1$ and using the convexity of $\hat{\mathcal{F}}(\cdot, \boldsymbol{v})$, $-\hat{\mathcal{F}}(\boldsymbol{u}, \cdot)$ and $\|\cdot\|_2^2$, we have that for any $(\boldsymbol{u}, \boldsymbol{v}) \in \mathcal{U} \times \mathcal{V}$,

$$\begin{aligned}\hat{\mathcal{F}}(\bar{\boldsymbol{u}}_T, \boldsymbol{v}) - \hat{\mathcal{F}}(\boldsymbol{u}, \bar{\boldsymbol{v}}_T) \le \frac{\|\boldsymbol{u} - \boldsymbol{u}_0\|_2^2}{2\gamma T} + \frac{\|\boldsymbol{v} - \boldsymbol{v}_0\|_2^2}{2\delta T} + 2C_1^2\gamma + 2C_2^2\delta \\ + \frac{\sigma}{2}(\|\boldsymbol{u} - \hat{\boldsymbol{u}}\|_2^2 - \|\bar{\boldsymbol{u}}_T - \hat{\boldsymbol{u}}\|_2^2 - \|\boldsymbol{u} - \bar{\boldsymbol{u}}_T\|_2^2) \\ \le \mathcal{O}(T^{-1/2}) + \frac{\sigma}{2}(\|\boldsymbol{u} - \hat{\boldsymbol{u}}\|_2^2 - \|\bar{\boldsymbol{u}}_T - \hat{\boldsymbol{u}}\|_2^2 - \|\boldsymbol{u} - \bar{\boldsymbol{u}}_T\|_2^2),\end{aligned}$$

where the first inequality follows from the definitions of $\bar{\boldsymbol{u}}_T$ and $\bar{\boldsymbol{v}}_T$, the bounds on subgradients, and the inequality

$$(\boldsymbol{v} - \boldsymbol{v}_T)^\top(\boldsymbol{\zeta}_{T-1} - \boldsymbol{\zeta}_T) \le \frac{\delta}{2}\|\boldsymbol{\zeta}_{T-1} - \boldsymbol{\zeta}_T\|_2^2 + \frac{1}{2\delta}\|\boldsymbol{v} - \boldsymbol{v}_T\|_2^2 \le 2C_2^2\delta + \frac{1}{2\delta}\|\boldsymbol{v} - \boldsymbol{v}_T\|_2^2, \tag{25}$$

and the second inequality follows from $\gamma = \delta = T^{-1/2}$. This completes the proof. □

Proposition 6. *Instate the conditions of Proposition 5. Suppose furthermore the existence of* $\mathrm{Lip}_{\boldsymbol{uu}}, \mathrm{Lip}_{\boldsymbol{vv}}, \mathrm{Lip}_{\boldsymbol{vu}} > 0$ *such that for any* $\boldsymbol{u}, \boldsymbol{u}' \in \mathcal{U}$, $\boldsymbol{v}, \boldsymbol{v}' \in \mathcal{V}$,

$$\begin{aligned}&\|\nabla_{\boldsymbol{u}}\hat{\mathcal{F}}(\boldsymbol{u}, \boldsymbol{v}) - \nabla_{\boldsymbol{u}}\hat{\mathcal{F}}(\boldsymbol{u}', \boldsymbol{v})\|_2 \le \mathrm{Lip}_{\boldsymbol{uu}}\|\boldsymbol{u} - \boldsymbol{u}'\|_2 \quad \textit{and} \\ &\|\nabla_{\boldsymbol{v}}\hat{\mathcal{F}}(\boldsymbol{u}, \boldsymbol{v}) - \nabla_{\boldsymbol{v}}\hat{\mathcal{F}}(\boldsymbol{u}', \tilde{\boldsymbol{v}})\|_2 \le \mathrm{Lip}_{\boldsymbol{vv}}\|\boldsymbol{v} - \boldsymbol{v}'\|_2 + \mathrm{Lip}_{\boldsymbol{vu}}\|\boldsymbol{u} - \boldsymbol{u}'\|_2.\end{aligned} \tag{26}$$

Then, the output $(\bar{\boldsymbol{u}}_T, \bar{\boldsymbol{v}}_T)$ *of Algorithm 5 with* $\gamma \le \frac{1}{\mathrm{Lip}_{\boldsymbol{uu}} + \mathrm{Lip}_{\boldsymbol{vu}}}$ *and* $\delta \le \frac{1}{2\mathrm{Lip}_{\boldsymbol{vv}} + \mathrm{Lip}_{\boldsymbol{vu}}}$ *is a strong* $\mathcal{O}(T^{-1})$*-approximate saddle point of problem* (19).

Proof of Proposition 6. Using the Lipschitz property of the gradient of $\hat{\mathcal{F}}$, we can improve the second inequality in (21) to

$$\hat{\mathcal{F}}(\boldsymbol{u}_{t+1}, \boldsymbol{v}_{t+1}) - \hat{\mathcal{F}}(\boldsymbol{u}_t, \boldsymbol{v}_{t+1}) \le (\boldsymbol{u}_{t+1} - \boldsymbol{u}_t)^\top \nabla_{\boldsymbol{u}}\hat{\mathcal{F}}(\boldsymbol{u}_t, \boldsymbol{v}_{t+1}) + \frac{\mathrm{Lip}_{\boldsymbol{uu}}}{2}\|\boldsymbol{u}_t - \boldsymbol{u}_{t+1}\|_2^2,$$

the two inequalities in (24) to

$$\begin{aligned}&(\boldsymbol{u}_t - \boldsymbol{u}_{t+1})^\top(\boldsymbol{\xi}_t - \boldsymbol{\xi}_t') = 0 \quad \text{and} \\ &(\boldsymbol{v} - \boldsymbol{v}_{t+1})^\top(2\boldsymbol{\zeta}_t - \boldsymbol{\zeta}_{t+1} - \boldsymbol{\zeta}_{t-1}) \le (\boldsymbol{v} - \boldsymbol{v}_{t+1})^\top(\boldsymbol{\zeta}_t - \boldsymbol{\zeta}_{t+1}) - (\boldsymbol{v} - \boldsymbol{v}_t)^\top(\boldsymbol{\zeta}_{t-1} - \boldsymbol{\zeta}_t) \\ &\qquad + \frac{\mathrm{Lip}_{\boldsymbol{vv}} + \mathrm{Lip}_{\boldsymbol{vu}}}{2}\|\boldsymbol{v}_t - \boldsymbol{v}_{t+1}\|_2^2 + \frac{\mathrm{Lip}_{\boldsymbol{vv}}}{2}\|\boldsymbol{v}_t - \boldsymbol{v}_{t-1}\|_2^2 + \frac{\mathrm{Lip}_{\boldsymbol{vu}}}{2}\|\boldsymbol{u}_t - \boldsymbol{u}_{t-1}\|_2^2,\end{aligned}$$

and inequality (25) to

$$(\boldsymbol{v}-\boldsymbol{v}_T)^\top(\boldsymbol{\zeta}_{T-1}-\boldsymbol{\zeta}_T) \le \frac{\mathrm{Lip}_{\boldsymbol{vv}}+\mathrm{Lip}_{\boldsymbol{vu}}}{2}\|\boldsymbol{v}-\boldsymbol{v}_T\|_2^2+\frac{\mathrm{Lip}_{\boldsymbol{vv}}}{2}\|\boldsymbol{v}_T-\boldsymbol{v}_{T-1}\|_2^2+\frac{\mathrm{Lip}_{\boldsymbol{vu}}}{2}\|\boldsymbol{u}_T-\boldsymbol{u}_{T-1}\|_2^2.$$

Following the proof of Proposition 5 but using the above improved inequalities, we get

$$\begin{aligned}
&\hat{\mathcal{F}}(\bar{\boldsymbol{u}}_T,\boldsymbol{v})-\hat{\mathcal{F}}(\boldsymbol{u},\bar{\boldsymbol{v}}_T)+\frac{\sigma}{2}\left(\|\bar{\boldsymbol{u}}_T-\hat{\boldsymbol{u}}\|_2^2+\|\boldsymbol{u}-\bar{\boldsymbol{u}}_T\|_2^2-\|\boldsymbol{u}-\hat{\boldsymbol{u}}\|_2^2\right)\\
\le&\frac{1}{2T\gamma}\|\boldsymbol{u}-\boldsymbol{u}_0\|_2^2+\frac{1}{2\delta T}\|\boldsymbol{v}-\boldsymbol{v}_0\|_2^2-\frac{1}{2T\gamma}\|\boldsymbol{u}-\boldsymbol{u}_T\|_2^2-\frac{1}{T}\left(\frac{1}{2\gamma}-\frac{\mathrm{Lip}_{\boldsymbol{uu}}+\mathrm{Lip}_{\boldsymbol{vu}}}{2}\right)\sum_{t=1}^{T}\|\boldsymbol{u}_t-\boldsymbol{u}_{t-1}\|_2^2\\
&-\frac{1}{T}\left(\frac{1}{2\delta}-\frac{2\mathrm{Lip}_{\boldsymbol{vv}}+\mathrm{Lip}_{\boldsymbol{vu}}}{2}\right)\sum_{t=1}^{T}\|\boldsymbol{v}_t-\boldsymbol{v}_{t-1}\|_2^2-\frac{1}{T}\left(\frac{1}{2\delta}-\frac{\mathrm{Lip}_{\boldsymbol{vv}}+\mathrm{Lip}_{\boldsymbol{vu}}}{2}\right)\|\boldsymbol{v}_{T-1}-\boldsymbol{v}_T\|_2^2\\
\le&\,\mathcal{O}(T^{-1}),
\end{aligned}$$

where the second inequality follows from $\gamma \le \frac{1}{\mathrm{Lip}_{\boldsymbol{uu}}+\mathrm{Lip}_{\boldsymbol{vu}}}$ and $\delta \le \frac{1}{2\mathrm{Lip}_{\boldsymbol{vv}}+\mathrm{Lip}_{\boldsymbol{vu}}}$. □

We are now ready to prove Theorem 2.

Proof of Theorem 2. The Inner problem is a special case of problem (19), with $\sigma = 1/\alpha$, $\mathcal{U}=\mathcal{X}$, $\mathcal{V}=\mathcal{Z}$ and

$$\hat{\mathcal{F}}(\boldsymbol{x},\boldsymbol{z}) = f_0(\boldsymbol{x}) + \sum_{m\in[M]} \lambda_m^{k+1} g_m(\boldsymbol{x},\boldsymbol{z}_m). \tag{27}$$

When applied to the Inner problem, Algorithm 2 reduces to Algorithm 5. By Assumption 1, we see that $\hat{\mathcal{F}}(\cdot,\boldsymbol{z})$ is convex on $\mathcal{X}$ for any $\boldsymbol{z}\in\mathcal{Z}$ and $\hat{\mathcal{F}}(\boldsymbol{x},\cdot)$ is concave on $\mathcal{Z}$ for any $\boldsymbol{x}\in\mathcal{X}$. Also, by Assumption 2, we have that the subdifferentials $\partial_{\boldsymbol{x}}\hat{\mathcal{F}}(\boldsymbol{x},\boldsymbol{z})$ and $\partial_{\boldsymbol{z}}(-\hat{\mathcal{F}})(\boldsymbol{x},\boldsymbol{z})$ are both uniformly bounded on $\mathcal{X}\times\mathcal{Z}$. The desired result thus follows from Proposition 5. □

Theorem 3 can be proved in a similar vein.

Proof of Theorem 3. The proof is similar to that of Theorem 2, except that we need to verify the Lipschitz property (26) and determine the step-sizes. Recall that for the Inner problem,

$\hat{\mathcal{F}}(\boldsymbol{x}, \boldsymbol{z})$ is given by (27). Therefore, by Assumption 3,

$$\begin{aligned}
&\|\nabla_{\boldsymbol{x}} \hat{\mathcal{F}}(\boldsymbol{x}, \boldsymbol{z}) - \nabla_{\boldsymbol{x}} \hat{\mathcal{F}}(\boldsymbol{x}', \boldsymbol{z})\|_2 \\
= &\|\nabla_{\boldsymbol{x}} f_0(\boldsymbol{x}) - \nabla_{\boldsymbol{x}} f_0(\boldsymbol{x}')\|_2 + \sum_{m \in [M]} \lambda_m^{k+1} \|\nabla_{\boldsymbol{x}} g_m(\boldsymbol{x}, \boldsymbol{z}_m) - \nabla_{\boldsymbol{x}} g_m(\boldsymbol{x}', \boldsymbol{z}_m)\|_2 \\
\leq &\left(D_0' + \sum_{m \in [M]} \lambda_m^{k+1} D_m' \right) \|\boldsymbol{x} - \boldsymbol{x}'\|_2,
\end{aligned}$$

which implies $\mathrm{Lip}_{\boldsymbol{uu}} = (D_0' + \sum_{m \in [M]} \lambda_m^{k+1} D_m')$.

Next, by Assumption 3,

$$\begin{aligned}
&\|\nabla_{\boldsymbol{z}} \hat{\mathcal{F}}(\boldsymbol{x}, \boldsymbol{z}) - \nabla_{\boldsymbol{z}} \hat{\mathcal{F}}(\boldsymbol{x}', \boldsymbol{z}')\|_2 \\
= &\sqrt{\sum_{m \in [M]} {\lambda_m^{k+1}}^2 \|\nabla_{\boldsymbol{z}_m} g_m(\boldsymbol{x}, \boldsymbol{z}_m) - \nabla_{\boldsymbol{z}_m} g_m(\boldsymbol{x}', \boldsymbol{z}_m')\|_2^2} \\
\leq &\sqrt{\sum_{m \in [M]} {\lambda_m^{k+1}}^2 \left(E_{m,1}' \|\boldsymbol{x} - \boldsymbol{x}'\|_2 + E_{m,2}' \|\boldsymbol{z}_m - \boldsymbol{z}_m'\|_2 \right)^2} \\
\leq &\sqrt{2 \sum_{m \in [M]} {\lambda_m^{k+1}}^2 {E_{m,1}'}^2} \|\boldsymbol{x} - \boldsymbol{x}'\|_2 + \sqrt{2 \max_{m \in [M]} {\lambda_m^{k+1}}^2 {E_{m,2}'}^2} \|\boldsymbol{z} - \boldsymbol{z}'\|_2,
\end{aligned}$$

which implies $\mathrm{Lip}_{\boldsymbol{vu}} = \sqrt{2 \sum_{m \in [M]} {\lambda_m^{k+1}}^2 {E_{m,1}'}^2}$ and $\mathrm{Lip}_{\boldsymbol{vv}} = \sqrt{2 \max_{m \in [M]} {\lambda_m^{k+1}}^2 {E_{m,2}'}^2}$. The desired result thus follows from Proposition 6. □

D Oracle Complexity

Proof of Theorem 4. Note that the outer algorithm does not directly rely on the projection or subgradient oracles, but only through the invocation of the inner algorithm. Also, each iteration of the inner algorithm requires at most a constant number of calls to the oracles, independent of ε. Therefore, to prove the oracle complexity of ProM^3, it suffices to count the aggregated number of inner iterations.

Consider Algorithms 1 and 2 with $K = \mathcal{O}(\varepsilon^{-1})$ and other algorithmic parameters chosen as in Theorems 1 and 2. By Theorem 1, ProM^3 produces an ε-approximate optimal solution to the Robust problem in $\mathcal{O}(\varepsilon^{-1})$ outer iterations. So, we need to invoke the inner algorithm

$\mathcal{O}(\varepsilon^{-1})$ times. Each invocation needs to compute a strong ε-approximate saddle point, which by Theorem 2, requires $\mathcal{O}(\varepsilon^{-2})$ inner iterations. We thus conclude that the oracle complexity is $\mathcal{O}(\varepsilon^{-1})\mathcal{O}(\varepsilon^{-2}) = \mathcal{O}(\varepsilon^{-3})$. □

Proof of Theorem 5. The proof is the same as that of Theorem 4 (except that we use Theorem 3 instead of Theorem 2) and hence omitted. □

E Proofs for Extended ProM^3

Proof of Proposition 3. Consider the Robust problem with projection-unfriendly uncertainty sets (3). For any fixed $m \in [M]$ and $\boldsymbol{x} \in \mathcal{X}$, the embedded maximization problem in the robust constraint reads

$$\begin{array}{rll} \max & g_m(\boldsymbol{x}, \boldsymbol{z}_m) & \\ \text{s.t.} & h_{m,i}(\boldsymbol{z}_m) \leq 0 \quad \forall i \in [I_m] & \\ & \boldsymbol{z}_m \in \widetilde{\mathcal{Z}}_m. & \end{array} \tag{28}$$

By Assumption 4(i)-(iii), we have that $\widetilde{\mathcal{Z}}_m$ is a non-empty, compact and convex set, that $g_m(\boldsymbol{x}, \cdot)$ is concave and $h_{m,i}$ is convex for all $i \in [I_m]$, and that a Slater point for problem (28) exists. Therefore, strong duality holds and problem (28) is equivalent to its Lagrangian dual

$$\min_{\boldsymbol{\mu}_m \in \mathbb{R}_+^{I_m}} \max_{\boldsymbol{z}_m \in \widetilde{\mathcal{Z}}_m} g_m(\boldsymbol{x}, \boldsymbol{z}_m) - \boldsymbol{\mu}_m^\top \boldsymbol{h}_m(\boldsymbol{z}_m) \tag{29}$$

Replacing the embedded problem by problem (29), we see that the Robust problem is equivalent to

$$\begin{array}{rl} \min & f_0(\boldsymbol{x}) \\ \text{s.t.} & \displaystyle\max_{\boldsymbol{z}_m \in \widetilde{\mathcal{Z}}_m} g_m(\boldsymbol{x}, \boldsymbol{z}_m) - \boldsymbol{\mu}_m^\top \boldsymbol{h}_m(\boldsymbol{z}_m) \leq 0 \quad \forall m \in [M] \\ & \boldsymbol{x} \in \mathcal{X}, \ \boldsymbol{\mu}_m \in \mathbb{R}_+^{I_m} \quad \forall m \in [M]. \end{array}$$

It remains to prove that any optimal solution $(\boldsymbol{x}^\star, \boldsymbol{\mu}^\star) \in \mathcal{X} \times \mathbb{R}_+^{I_1 + \cdots + I_M}$ satisfies that

$$\mu_{m,i}^\star \leq \frac{G_m}{\max_{i \in [I_m]}\{h_{m,i}(\bar{\boldsymbol{z}}_m)\}} \quad \forall i \in [I_m], \, m \in [M].$$

To do so, let $(\boldsymbol{x}^\star, \boldsymbol{\mu}^\star) \in \mathcal{X} \times \mathbb{R}_+^{I_1+\cdots+I_M}$ be any optimal solution. Then, for any $m \in [M]$,

$$\max_{\boldsymbol{z}_m \in \widetilde{\mathcal{Z}}_m} g_m(\boldsymbol{x}^\star, \boldsymbol{z}_m) - {\boldsymbol{\mu}_m^\star}^\top \boldsymbol{h}_m(\boldsymbol{z}_m) \leq 0,$$

which implies that

$$g_m(\boldsymbol{x}^\star, \bar{\boldsymbol{z}}_m) \leq {\boldsymbol{\mu}_m^\star}^\top \boldsymbol{h}_m(\bar{\boldsymbol{z}}_m).$$

Noting that $\mu^\star_{m,i}\, h_{m,i}(\bar{\boldsymbol{z}}_m) \leq 0$ for all $i \in [I_m]$, we have

$$\mu^\star_{m,i}\, h_{m,i}(\bar{\boldsymbol{z}}_m) \geq {\boldsymbol{\mu}_m^\star}^\top \boldsymbol{h}_m(\bar{\boldsymbol{z}}_m) \geq g_m(\boldsymbol{x}^\star, \bar{\boldsymbol{z}}_m) \geq G_m.$$

By Assumption 4(iii), $h_{m,i}(\bar{\boldsymbol{z}}_m) < 0$ for all $i \in [I_m]$. Therefore, for any $i \in [I_m]$,

$$\mu^\star_{m,i} \leq \frac{G_m}{h_{m,i}(\bar{\boldsymbol{z}}_m)} \leq \frac{G_m}{\max_{i \in [I_m]}\{h_{m,i}(\bar{\boldsymbol{z}}_m)\}}.$$

This completes the proof. □

Proof of Theorem 6. The extended ProM^3 is the algorithm obtained by applying ProM^3 (from Section 2) to the problem $\widetilde{\textsc{Robust}}$. It suffices is to verify that problem $\widetilde{\textsc{Robust}}$ satisfies Assumptions 1, 2 and 3 in the sense that these assumptions hold when the data $(f_0, g_1, \ldots, g_M, \mathcal{X}, \mathcal{Z}_1, \ldots, \mathcal{Z}_M)$ is replaced by $(\tilde{f}_0, \tilde{g}_1, \ldots, \tilde{g}_M, \widetilde{\mathcal{X}}, \widetilde{\mathcal{Z}}_1, \ldots, \widetilde{\mathcal{Z}}_M)$.

Assumption 1(*i*): The non-emptiness, compactness and convexity of the sets $\widetilde{\mathcal{Z}}_1, \ldots, \widetilde{\mathcal{Z}}_M$ follow directly from Assumption 4(i). Recall that $\widetilde{\mathcal{X}} = \mathcal{X} \times \mathcal{M}$, where $\mathcal{M} = [0, a_1]^{I_1} \times \cdots \times [0, a_M]^{I_M}$. Assumption 4($i$) therefore implies also that $\widetilde{\mathcal{X}}$ is non-empty, compact and convex.

Assumption 1(*ii*): The objective function $\tilde{f}_0(\tilde{\boldsymbol{x}}) = f_0(\boldsymbol{x})$ is obviously convex and finite-valued on $\widetilde{\mathcal{X}} = \mathcal{X} \times \mathcal{M}$, since f_0 is convex and finite-valued on $\mathcal{X}$ by Assumption 4(ii). Also by Assumption 4(ii), $g_m(\cdot, \boldsymbol{z}_m)$ is convex for any $\boldsymbol{z}_m \in \widetilde{\mathcal{Z}}_m$ and $g_m(\boldsymbol{x}, \cdot)$ is concave for any $\boldsymbol{x} \in \mathcal{X}$, and $h_{m,i}$ is convex and finite-valued on $\widetilde{\mathcal{Z}}_m$ for all $i \in [I_m]$ and $m \in [M]$. Together with the boundedness of $\mathcal{M}$, this implies that the function $\tilde{g}_m(\tilde{\boldsymbol{x}}, \boldsymbol{z}_m) = g_m(\boldsymbol{x}, \boldsymbol{z}_m) - \boldsymbol{\mu}_m^\top \boldsymbol{h}_m(\boldsymbol{z}_m)$ is finite-valued on $\widetilde{\mathcal{X}} \times \widetilde{\mathcal{Z}}_m$ and satisfies that $\tilde{g}_m(\cdot, \boldsymbol{z}_m)$ is convex for any $\boldsymbol{z} \in \widetilde{\mathcal{Z}}_m$ and $\tilde{g}_m(\tilde{\boldsymbol{x}}, \cdot)$ is concave for any $\tilde{\boldsymbol{x}} = (\boldsymbol{x}, \boldsymbol{\mu}) \in \widetilde{\mathcal{X}}$.

Assumption 1(*iii*): By Assumption 4(iii), there exists $\bar{\boldsymbol{x}}$ such that $\displaystyle\max_{\boldsymbol{z}_m \in \widetilde{\mathcal{Z}}_m} g_m(\bar{\boldsymbol{x}}, \boldsymbol{z}_m) < 0$

for any $m \in [M]$. Therefore, $\max_{\boldsymbol{z}_m \in \widetilde{\mathcal{Z}}_m} \tilde{g}_m((\bar{\boldsymbol{x}}, \boldsymbol{0}), \boldsymbol{z}_m) = \max_{\boldsymbol{z}_m \in \widetilde{\mathcal{Z}}_m} g_m(\bar{\boldsymbol{x}}, \boldsymbol{z}_m) < 0$.

Assumption 1(*iv*): It follows directly from Assumption 4(*iv*).

Assumption 2: By Assumption 5, we have that $\tilde{f}_0(\tilde{\boldsymbol{x}}) = f_0(\boldsymbol{x})$ is subdifferentiable on $\widetilde{\mathcal{X}} = \mathcal{X} \times \mathcal{M}$. Recall that $\tilde{g}_m(\tilde{\boldsymbol{x}}, \boldsymbol{z}_m) = g_m(\boldsymbol{x}, \boldsymbol{z}_m) - \boldsymbol{\mu}_m^\top \boldsymbol{h}_m(\boldsymbol{z}_m)$ for any $m \in [M]$. By Assumption 5, we have that $\tilde{g}_m(\tilde{\boldsymbol{x}}, \boldsymbol{z}_m)$ is subdifferentiable in $\tilde{\boldsymbol{x}} = (\boldsymbol{x}, \boldsymbol{\mu})$ on $\widetilde{\mathcal{X}}$ and $-\tilde{g}_m(\tilde{\boldsymbol{x}}, \boldsymbol{z}_m)$ subdifferentiable in $\boldsymbol{z}_m$ on $\widetilde{\mathcal{Z}}_m$.

We then bound the subdifferentials. First, any subgradient of $\tilde{f}_0$ at $\tilde{\boldsymbol{x}} \in \widetilde{\mathcal{X}}$ is of the form $\tilde{\boldsymbol{\xi}}_0 = (\boldsymbol{\xi}_0, \boldsymbol{0})$ for some $\boldsymbol{\xi}_0 \in \partial f_0(\boldsymbol{x})$. So, by Assumption 5, the subdifferential $\partial \tilde{f}_0(\tilde{\boldsymbol{x}})$ is uniformly bounded by $\tilde{D}_0 = D_0$. Next, for any $m \in [M]$ and $\boldsymbol{z}_m \in \widetilde{\mathcal{Z}}_m$, any subgradient of $\tilde{g}_m(\cdot, \boldsymbol{z}_m)$ is of the form $\tilde{\boldsymbol{\xi}}_m = (\boldsymbol{\xi}_m, \boldsymbol{0}, -\boldsymbol{h}_m(\boldsymbol{z}_m), \boldsymbol{0})$ for some $\boldsymbol{\xi}_m \in \partial_{\boldsymbol{x}} g_m(\boldsymbol{x}, \boldsymbol{z}_m)$. Since H_m is real-valued and convex on $\widetilde{\mathcal{Z}}_m$, there exists a constant $H_m > 0$ such that $\|\boldsymbol{h}_m(\boldsymbol{z}_m)\| \le H_m$ for all $\boldsymbol{z}_m \in \mathcal{Z}_m$. Noting that $\|\tilde{\boldsymbol{\xi}}_m\| \le \sqrt{\|\boldsymbol{\xi}_m\|^2 + \|\boldsymbol{h}_m(\boldsymbol{z}_m)\|^2}$, by Assumption 5, the subdifferential $\partial_{\tilde{\boldsymbol{x}}} \tilde{g}_m(\tilde{\boldsymbol{x}}, \boldsymbol{z}_m)$ is uniformly bounded by $\tilde{D}_m = \sqrt{D_m^2 + H_m^2}$. Finally, for any $m \in [M]$ and $\tilde{\boldsymbol{x}} = (\boldsymbol{x}, \boldsymbol{\mu}) \in \widetilde{\mathcal{X}}$, any subgradient of $(-\tilde{g}_m)(\tilde{\boldsymbol{x}}, \cdot)$ is of the form $\tilde{\boldsymbol{\zeta}}_m = \boldsymbol{\zeta}_m - \sum_{i \in [I_m]} \mu_{m,i} \boldsymbol{\eta}_{m,i}$ for some $\boldsymbol{\eta}_{m,1} \in \partial h_{m,1}(\boldsymbol{z}_m), \ldots, \boldsymbol{\eta}_{m,I_m} \in \partial h_{m,I_m}(\boldsymbol{z}_m)$ and $\boldsymbol{\zeta}_m \in \partial_{\boldsymbol{z}_m}(-g_m)(\boldsymbol{x}, \boldsymbol{z}_m)$. Noting that $\|\tilde{\boldsymbol{\zeta}}_m\| \le \|\boldsymbol{\zeta}_m\| + \sum_{i \in [I_m]} \mu_{m,i} \|\boldsymbol{\eta}_{m,i}\|$, by Assumption 5, the subdifferential $\partial_{\boldsymbol{z}_m}(-\tilde{g}_m)(\tilde{\boldsymbol{x}}, \boldsymbol{z}_m)$ is uniformly bounded by $\tilde{E}_m = E_m + a_m I_m F_m$.

Assumption 3: Since $\tilde{f}_0(\tilde{\boldsymbol{x}}) = f_0(\boldsymbol{x})$ for any $\tilde{\boldsymbol{x}} \in \widetilde{\mathcal{X}}$, it follows from Assumption 6 that $\nabla \tilde{f}_0$ is Lipschitz continuous on $\widetilde{\mathcal{X}}$. For any $m \in [M]$, $\boldsymbol{z} \in \widetilde{\mathcal{Z}}_m$ and $\tilde{\boldsymbol{x}} = (\boldsymbol{x}, \boldsymbol{\mu}) \in \widetilde{\mathcal{X}}$, $\nabla_{\tilde{\boldsymbol{x}}} \tilde{g}_m(\tilde{\boldsymbol{x}}, \boldsymbol{z}_m) = (\nabla_{\boldsymbol{x}} g_m(\boldsymbol{x}, \boldsymbol{z}_m), \boldsymbol{0}, -\boldsymbol{h}_m(\boldsymbol{z}_m), \boldsymbol{0})$ and $\nabla_{\boldsymbol{z}_m} \tilde{g}_m(\tilde{\boldsymbol{x}}, \boldsymbol{z}_m) = \nabla_{\boldsymbol{z}_m} g_m(\boldsymbol{x}, \boldsymbol{z}_m) - \sum_{i \in [I_m]} \mu_{m,i} \nabla h_{m,i}(\boldsymbol{z}_m)$. By Assumption 6, $\nabla_{\tilde{\boldsymbol{x}}} \tilde{g}_m(\cdot, \boldsymbol{z}_m)$ is Lipschitz continuous on $\widetilde{\mathcal{X}}$ for any fixed $\boldsymbol{z}_m \in \widetilde{\mathcal{Z}}_m$, and $\nabla_{\boldsymbol{z}_m} \tilde{g}_m$ is jointly Lipschitz continuous on $\widetilde{X} \times \widetilde{\mathcal{Z}}_m$.

The proof is completed. $\square$