AN ACCELERATED INEXACT PROXIMAL POINT METHOD FOR SOLVING NONCONVEX-CONCAVE MIN-MAX PROBLEMS

WEIWEI KONG* AND RENATO D.C. MONTEIRO*

Abstract. This paper presents a quadratic-penalty type method for solving linearly-constrained composite nonconvex-concave min-max problems. The method consists of solving a sequence of penalty subproblems which, due to the min-max structure of the problem, are potentially nonsmooth but can be approximated by smooth composite nonconvex minimization problems. Each of these penalty subproblems is then solved by applying an accelerated inexact proximal point method to its corresponding smooth composite nonconvex approximation. Iteration complexity bounds for obtaining approximate stationary points of the linearly-constrained composite nonconvex-concave min-max problem are also established. Finally, numerical results are given to demonstrate the efficiency of the proposed method.

Key words. quadratic penalty method, composite nonconvex problem, iteration-complexity, inexact proximal point method, first-order accelerated gradient method, minimax problem.

AMS subject classifications. 47J22, 90C26, 90C30, 90C47, 90C60, 65K10.

1. Introduction.

The first goal of this paper is to present and study the complexity of accelerated inexact proximal point smoothing (AIPP-S) methods for approximately solving the (potentially nonsmooth) min-max composite nonconvex optimization (CNO) problem

$$\min_x \left\{ \hat{p}(x) := h(x) + \max_y \Phi(x,y) \right\} \tag{1.1}$$

where h is a "simple" proper lower-semicontinuous convex function and Φ is a function that satisfies the following assumptions: (i) for every $x \in \operatorname{dom} h$, the function $-\Phi(x,\cdot)$ is proper lower-semicontinuous convex and $\operatorname{dom}[-\Phi(x,\cdot)] = Y$ for some compact set Y; and (ii) for every $y \in Y$, the function $\Phi(\cdot,y)$ is nonconvex differentiable on $\operatorname{dom} h$, its gradient is uniformly (with respect to y) Lipschitz continuous on $\operatorname{dom} h$, and $\Phi(\cdot,y)$ has a uniform (with respect to y) lower curvature on $\operatorname{dom} h$ (see (3.4)). The function $\hat{p}$ is then the sum of a simple (potentially nonsmooth) convex function h and the pointwise supremum of differentiable nonconvex functions which is generally a (complicated) nonsmooth nonconvex function.

When Y is a singleton, the max term in (1.1) becomes smooth and (1.1) reduces to a smooth CNO problem for which many algorithms have been developed in the literature. In particular, accelerated inexact proximal points (AIPP) methods, i.e. methods which use an accelerated composite gradient variant to approximately solve the generated sequence of prox subproblems, have been developed for it (see, for example, [2, 6]). When Y is not a singleton, (1.1) can no longer be directly solved by an AIPP method due to the nonsmoothness of the max term. The AIPP-S methods developed in this paper are based instead on a perturbed version of (1.1) in which the max term in (1.1) is replaced by a smooth approximation and the resulting smooth CNO problem is solved by an AIPP method.

We first develop an AIPP-S variant that computes an approximate solution involving the directional derivative of $\hat{p}$. More specifically, given a tolerance $\delta > 0$, it

*School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, GA, 30332-0205. (E-mails: wkong37@gatech.edu & monteiro@isye.gatech.edu). The works of these authors were partially supported by ONR Grant N00014-18-1-2077.

is shown that this variant computes a point $x \in \operatorname{dom} h$ such that there exists another point $\hat{x} \in \operatorname{dom} h$ satisfying

$$\inf_{\|d\| \le 1} \hat{p}'(\hat{x}; d) \ge -\delta, \quad \|\hat{x} - x\| \le \delta \tag{1.2}$$

in at most $\mathcal{O}(\delta^{-3})$ gradient and proximal subproblem evaluations. Next, we develop an AIPP-S variant that computes an approximate solution involving a saddle-point formulation of (1.1). More specifically, given a tolerance pair $(\rho_x, \rho_y) \in \mathbb{R}^2_{++}$, it is shown that this variant computes a quadruple $(\bar{u}, \bar{v}, \bar{x}, \bar{y})$ satisfying

$$\begin{pmatrix} \bar{u} \\ \bar{v} \end{pmatrix} \in \begin{pmatrix} \nabla_x \Phi(\bar{x}, \bar{y}) \\ 0 \end{pmatrix} + \begin{pmatrix} \partial h(\bar{x}) \\ \partial \left[-\Phi(\bar{x}, \cdot) \right] (\bar{y}) \end{pmatrix}, \quad \|\bar{u}\| \le \rho_x, \quad \|\bar{v}\| \le \rho_y \tag{1.3}$$

in at most $\mathcal{O}(\rho_x^{-2} \rho_y^{-1/2})$ gradient and proximal subproblem evaluations.

The second goal of this paper is to develop a quadratic-penalty AIPP-S (QP-AIPP-S) method to approximately solve a linearly constrained version of (1.1), namely

$$\min_x \{\hat{p}(x) : \mathcal{A}x = b\} \tag{1.4}$$

where p is as in (1.1), $\mathcal{A}$ is a linear operator, and b is in the range of $\mathcal{A}$. The method is a penalty-type method which approximately solves a sequence of penalty subproblems of the form

$$\min_x \left\{ \hat{p}(x) + \frac{c}{2} \|\mathcal{A}x - b\|^2 \right\} \tag{1.5}$$

for an increasing sequence of positive penalty parameters c. Similar to the approach used for the first goal of this paper, the method considers a perturbed variant of (1.5) in which the objective function is replaced by a smooth approximation and the resulting problem is solved by the quadratic-penalty AIPP (QP-AIPP) method proposed in [6]. For a given tolerance triple $(\rho_x, \rho_y, \eta) \in \mathbb{R}^3_{++}$, it is shown that the method computes a quintuple $(\bar{u}, \bar{v}, \bar{x}, \bar{y}, \bar{r})$ satisfying

$$\begin{pmatrix} \bar{u} \\ \bar{v} \end{pmatrix} \in \begin{pmatrix} \nabla_x \Phi(\bar{x}, \bar{y}) + \mathcal{A}^* \bar{r} \\ 0 \end{pmatrix} + \begin{pmatrix} \partial h(\bar{x}) \\ \partial \left[-\Phi(\bar{x}, \cdot) \right] (\bar{y}) \end{pmatrix}, \tag{1.6}$$

$$\|\bar{u}\| \le \rho_x, \quad \|\bar{v}\| \le \rho_y, \quad \|\mathcal{A}\bar{x} - b\| \le \eta.$$

in at most $\mathcal{O}(\rho_x^{-2} \rho_y^{-1/2} + \rho_x^{-2} \eta^{-1})$ gradient and proximal subproblem evaluations.

It is worth mentioning that all of the above complexities are obtained under the mild assumption that the optimal value in each of the respective optimization problems, namely (1.1) and (1.4), be bounded below. Moreover, it is neither assumed that $\operatorname{dom} h$ be bounded nor that (1.1) or (1.4) has an optimal solution.

Related Works. Since the case when $\Phi(\cdot, \cdot)$ in (1.1) is convex-concave has been well-studied in the literature (see, for example, [1, 4, 5, 9, 10, 11, 14]), we will make no more mention of it here. Instead, we will focus on papers that consider (1.1) where $\Phi(\cdot, y)$ is differentiable nonconvex for every $y \in Y$ and there are mild conditions on $\Phi(x, \cdot)$ for every $x \in \operatorname{dom} h$. The method in [13] considers (1.1) under the assumption that Φ be differentiable everywhere and the gradients $\nabla_x \Phi(\cdot, y)$ and $\nabla_y \Phi(x, \cdot)$ be Lipschitz everywhere for every (x, y). The method in [12] considers a perturbed variant

of (1.1) and a smoothing method similar to our proposed AIPP-S methods. However, their method does not solve the perturbed problem using an accelerated method unlike the approach taken in this paper and requires $\operatorname{dom} h$ be bounded. Each of the methods in [12, 13] consider notions of approximate solutions that are different from (1.2) and (1.3), making a comparison between these methods and the one presented in this paper not straightforward. We instead defer this discussion to Section 6 where it is shown that the AIPP-S method is more efficient when a common termination criterion is used.

Organization of the paper. Subsection 1.1 presents notation and some basic definitions that are used in this paper. Section 2 is divided into two subsections. The first one reviews an AIPP method studied in [6], its key iteration complexities, for solving a class of smooth CNO problems. The second one presents a QP-AIPP method, and its iteration complexity, for solving a class of linear-constrained smooth CNO problems. Section 3 is also divided into two subsections. The first one precisely states the problem of interest, its assumptions, and the various definitions of approximate solutions to this problem. The second one presents the AIPP-S method for solving the problem of interest and the complexity analysis for two instances of this method. Section 4 presents a method for solving a linearly-constrained variant of the problem of interest. Section 5 presents computational results. Section 6 presents concluding remarks. Finally, several appendices at the end of this paper contain proofs of technical results needed in our presentation.

1.1. Notation and basic definitions. This subsection provides some basic notation and definitions.

The set of real numbers is denoted by $\mathbb{R}$. The set of non-negative real numbers and the set of positive real numbers is denoted by $\mathbb{R}_+$ and $\mathbb{R}_{++}$ respectively. The set of natural numbers is denoted by $\mathbb{N}$. For $t > 0$, define $\log_1^+(t) := \max\{1, \log(t)\}$. Let $\mathbb{R}^n$ denote a real–valued n–dimensional Euclidean space with standard norm $\|\cdot\|$. Given a linear operator $A : \mathbb{R}^n \mapsto \mathbb{R}^p$, the operator norm of A is denoted by $\|A\| := \sup\{\|Az\|/\|z\| : z \in \mathbb{R}^n, z \neq 0\}$.

The following notation and definitions are for a general complete inner product space $\mathcal{Z}$, whose inner product and its associated induced norm are denoted by $\langle \cdot, \cdot \rangle$ and $\|\cdot\|$ respectively.. Let $\psi : \mathcal{Z} \mapsto (-\infty, \infty]$ be given. The effective domain of ψ is denoted as $\operatorname{dom} \psi := \{z \in \mathcal{Z} : \psi(z) < \infty\}$ and ψ is said to be proper if $\operatorname{dom} \psi \neq \emptyset$. For $\varepsilon \geq 0$, the *ε-subdifferential* of ψ at $z \in \operatorname{dom} \psi$ is denoted by

$$\partial_\varepsilon \psi(z) := \{w \in \mathbb{R}^n : \psi(z') \geq \psi(z) + \langle w, z' - z \rangle - \varepsilon, \forall z' \in \mathcal{Z}\}, \tag{1.7}$$

and we denote $\partial \psi \equiv \partial_0 \psi$. The set of proper, lower semi-continuous, convex functions $\psi : \mathcal{Z} \mapsto (-\infty, \infty]$ is denoted as $\overline{\operatorname{Conv}}(\mathcal{Z})$. The *directional derivative* of ψ at $z \in \mathcal{Z}$ in the direction $d \in \mathcal{Z}$ is denoted by

$$\psi'(z; d) := \lim_{t \to 0} \frac{\psi(z + td) - \psi(z)}{t}.$$

It is well-known that if ψ is differentiable at $z \in \operatorname{dom} \psi$ then for a given direction $d \in \mathcal{Z}$ we have $\psi'(z; d) = \langle \nabla \psi(z), d \rangle$.

For a given $Z \subseteq \mathcal{Z}$, the indicator function of Z, denoted by δ_Z, is defined as $\delta_Z(z) = 0$ if $z \in Z$ and $\delta_Z(z) = \infty$ if $z \notin Z$.

2. AIPP methods for nonconvex optimization. As mentioned in the introduction, our interest in this paper is in the development of a smoothing method for

solving (1.1) (resp. (1.4)) that applies the AIPP (resp. QP-AIPP) method of [6] to solve a perturbed version of it. Hence, this section, which contains two subsections, begins by reviewing the AIPP method in its first subsection and then the QP-AIPP method in its second one.

2.1. AIPP method. This subsection describes the AIPP method studied in [6], and its corresponding iteration complexity result, for solving a class of smooth CNO problems.

We first describe the problem that the AIPP method is intended to solve. Let $\mathcal{X}$ be a finite-dimensional inner product and consider the smooth CNO problem

$$\phi_* := \inf_{x\in\mathcal{X}} [\phi(x) := f(x) + h(x)] \tag{2.1}$$

where $h : \mathcal{X} \mapsto (-\infty, \infty]$ and real-valued function f satisfy the following assumptions:

(P1) $h \in \overline{\mathrm{Conv}}(\mathcal{Z})$ and f is differentiable on $\mathrm{dom}\, h$;

(P2) for some $M \geq m > 0$ the function f satisfies

$$-\frac{m}{2}\|x' - x\|^2 \leq f(x') - [f(x) + \langle \nabla f(x), x' - x\rangle], \tag{2.2}$$

$$\|\nabla f(x') - \nabla f(x)\| \leq M\|x' - x\|, \tag{2.3}$$

for any $x, x' \in \mathrm{dom}\, h$;

(P3) ϕ_* defined in (2.1) is finite.

It is well-known that a necessary condition for $x^* \in \mathrm{dom}\, h$ to be a local minimum of (2.1) is that x^* is a stationary point of ϕ, i.e. $0 \in \nabla f(x^*) + \partial h(x^*)$.

For the purpose of discussing the complexity results of this subsection, we consider the following notion of approximate solution of (2.1): given a tolerance $\bar\rho > 0$, a pair $(\bar x, \bar u) \in \mathrm{dom}\, h \times \mathcal{X}$ is said to be a $\bar\rho$–approximate solution of (2.1) if

$$\bar u \in \nabla f(\bar x) + \partial h(\bar x), \quad \|\bar u\| \leq \bar\rho. \tag{2.4}$$

We now describe a method, namely the AIPP method of [6], that is able to generate an approximate solution as in (2.4).

AIPP method

Input: a scalar pair $(m, M) \in \mathbb{R}^2_{++}$ satisfying (2.2), a function pair (f, h), scalars $\lambda \in (0, 1/(2m)]$ and $\sigma \in (0, 1)$, an initial point $x_0 \in \mathrm{dom}\, h$, and a tolerance $\bar\rho > 0$;

Output: a pair $(\bar x, \bar u) \in \mathrm{dom}\, h \times \mathcal{X}$ satisfying (2.4);

(0) set $k = 1$ and define

$$\hat\rho = \frac{\bar\rho}{4}, \quad \hat\varepsilon = \frac{\bar\rho^2}{32(M + \lambda^{-1})}, \quad M_\lambda := M + \lambda^{-1};$$

(1) perform at least $\lceil 6\sqrt{2\lambda M + 1}\rceil$ iterations of the ACG method in Appendix A starting from x_{k-1}, and with inputs

$$(\mu, L) = (-\lambda m + 1/2, \lambda M + 1/2),$$

$$(\psi_s, \psi_n) = \left(\lambda f + \frac{1}{4}\|\cdot - x_{k-1}\|^2, \lambda h + \frac{1}{4}\|\cdot - x_{k-1}\|^2\right),$$

and x_0 in order to obtain a triple $(x, u, \varepsilon) \in \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}_+$ satisfying

$$u \in \partial_\varepsilon \left(\lambda\phi + \frac{1}{2}\|\cdot - x_{k-1}\|^2 \right)(x), \quad \|u\|^2 + 2\varepsilon \le \sigma\|x_{k-1} - x + u\|^2; \tag{2.5}$$

(2) if the residual

$$\|x_{k-1} - x + u\| \le \frac{\lambda\hat{\rho}}{5}, \tag{2.6}$$

then go to (3); otherwise set $(x_k, \tilde{u}_k, \tilde{\varepsilon}_k) = (x, u, \varepsilon)$, increment $k = k + 1$ and go to (1);

(3) restart the previous call to the ACG method in step 1 to find a triple $(\tilde{x}, \tilde{u}, \tilde{\varepsilon})$ such that $\tilde{\varepsilon} \le \hat{\varepsilon}\lambda$ and $(x, u, \varepsilon) = (\tilde{x}, \tilde{u}, \tilde{\varepsilon})$ satisfies (2.5);

(4) compute

$$\bar{x} := \operatorname*{argmin}_{x' \in \mathcal{X}} \left\{ \langle \nabla f(x), x' - x\rangle + h(x') + \frac{M_\lambda}{2}\|x' - x\|^2 \right\}, \tag{2.7}$$

$$\bar{u} := M_\lambda(x - \bar{x}) + \nabla f(\bar{x}) - \nabla f(x), \tag{2.8}$$

and output the pair $(\bar{x}, \bar{u})$.

We now make three remarks about the above AIPP method. First, at the k^{th} iteration of the method, its step 1 invokes an accelerated composite gradient (ACG) method, whose description is given in Appendix A, for (approximately) solve the strongly convex proximal subproblem

$$\min_x \left\{ \lambda\phi(x) + \frac{1}{2}\|x - x_{k-1}\|^2 \right\} \tag{2.9}$$

according to (2.5). Second, only the inequality in (2.5) needs to be verified as every ACG iterate (x, u, ε) satisfies the inclusion in (2.5). Third, note that (2.3) implies that the gradient of the function ψ_s defined in step 1 of the AIPP method is $(\lambda M + 1/2)$–Lipschitz continuous. As a consequence, Lemma 8 with $L = \lambda M + 1/2$ implies that the triple (x, u, ε) in step 1 of any iteration of the AIPP method can be obtained in $\mathcal{O}(\sqrt{\lambda M + 1})$ ACG iterations.

It is worth mentioning that the above method differs slightly from the one presented in [6] in that it adds step 4 in order to directly output a $\bar{\rho}$–approximate solution as in (2.4). The justification for the latter claim follows from [6, Corollary 14], which also implies the following complexity result.

PROPOSITION 1. *The AIPP method terminates with a $\bar{\rho}$–approximate solution of (2.1) in at most*

$$\mathcal{O}\left(\sqrt{\lambda M + 1} \left[\frac{R(\phi; \lambda)}{\lambda^2 \bar{\rho}^2} + \log_1^+(\lambda M) \right] \right) \tag{2.10}$$

ACG iterations where

$$R(\phi; \lambda) = \inf_{x'} \left\{ \frac{1}{2}\|x_0 - x'\|^2 + \lambda\left[\phi(x') - \phi_*\right] \right\}. \tag{2.11}$$

It is easy to see that the quantity $R(\phi; \lambda)$ in (2.10) admits the upper bound

$$R(\phi; \lambda) \le \min\left\{ \frac{1}{2}d_0^2, \lambda\left[\phi(x_0) - \phi_*\right] \right\} \tag{2.12}$$

where

$$d_0 := \inf \{\|x_0 - x_*\| : x_* \text{ is an optimal solution of (2.1)}\}.$$

It is also worth mentioning that a more practical variant of the above AIPP method, called the relaxed AIPP (R-AIPP) method, is presented in [7]. Like the AIPP method, the R-AIPP similarly: (i) invokes at each of its (outer) iteration an ACG method to (approximately) solve the proximal subproblem (2.9); and (ii) outputs a $\bar{\rho}$–approximate solution of (2.1). However, the R-AIPP method is more computationally efficient due to three key practical improvements over the AIPP method, namely: (i) it allows the stepsize λ to be significantly larger than the $1/(2m)$ upper bound in the AIPP method; (ii) it uses a weaker ACG termination criterion compared to the one in (2.5); and (iii) it does not prespecify the number of ACG iterations as the AIPP method does in its step 1. Even though the proposed smoothing method of this paper invokes the AIPP method for solving a perturbed version of (1.1), the computational results use the more computationally efficient R-AIPP method (in place of the AIPP method) in a modified smoothing method called the R-AIPP smoothing (R-AIPP-S) method.

2.2. Quadratic penalty AIPP method. This subsection describes the QP-AIPP method studied in [6], and its corresponding iteration complexity, for solving linearly-constrained smooth CNO problems.

We begin by describing the problem that the QP-AIPP method intends to solve. Let $\mathcal{X}$ and $\mathcal{U}$ be finite-dimensional inner product spaces and consider the linearly-constrained smooth CNO problem

$$\hat{\phi}_* := \inf_x \{\phi(x) := f(x) + h(x) : \mathcal{A}x = b\} \tag{2.13}$$

where $h : \mathcal{X} \mapsto (-\infty, \infty]$ and a real-valued function f satisfy assumptions (P1)–(P3), the operator $\mathcal{A} : \mathcal{X} \mapsto \mathcal{U}$ is linear, $b \in \mathcal{U}$, and the following additional assumptions hold:

(Q1) $\mathcal{A} \not\equiv 0$ and $\mathcal{F} := \{x \in \operatorname{dom} h : \mathcal{A}x = b\} \neq \emptyset$;

(Q2) there exists $\hat{c} \geq 0$ such that $\hat{\phi}_{\hat{c}} > -\infty$ where

$$\hat{\phi}_c := \inf_x \left\{\phi_c(x) := \phi(x) + \frac{c}{2}\|\mathcal{A}x - b\|^2\right\}, \quad \forall c \geq 0. \tag{2.14}$$

Similar to problem (2.1), it is well-known that a necessary condition for $x^* \in \operatorname{dom} h$ to be a local minimum of (2.13) is that x^* satisfies $0 \in \nabla f(x^*) + \partial h(x^*) + \mathcal{A}^* r^*$ for some $r^* \in \mathcal{U}$.

Our interest in this subsection is in finding an approximate solution of (2.13) in the following sense: given a tolerance pair $(\bar{\rho}, \bar{\eta}) \in \mathbb{R}^2_{++}$, a triple $(\bar{x}, \bar{u}, \bar{r}) \in \mathcal{X} \times \mathcal{X} \times \mathcal{U}$ is said to be a $(\bar{\rho}, \bar{\eta})$–approximate solution of (2.13) if

$$\bar{u} \in \nabla f(\bar{x}) + \partial h(\bar{x}) + \mathcal{A}^* \bar{r}, \quad \|\bar{u}\| \leq \bar{\rho}, \quad \|\mathcal{A}\bar{x} - b\| \leq \bar{\eta}. \tag{2.15}$$

The QP-AIPP method below provides one way of obtaining an approximate solution of (2.13) as in (2.15) using similar arguments as in Subsection 2.1. Its main idea is to invoke the AIPP method to solve subproblems of the form

$$\min_x \left\{f(x) + h(x) + \frac{c}{2}\|\mathcal{A}x - b\|^2\right\}, \tag{2.16}$$

for increasing values of c.

Quadratic penalty AIPP (QP-AIPP) method

Input: a scalar pair $(m, M) \in \mathbb{R}^2_{++}$ satisfying (2.2), a function pair (f, h), a scalar $\sigma \in (0, 1)$, a scalar $\hat{c}$ satisfying assumption (Q2), an initial point $x_0 \in \operatorname{dom} h$, and a tolerance pair $(\bar{\rho}, \bar{\eta}) \in \mathbb{R}^2_{++}$;

Output: a triple $(\bar{x}, \bar{u}, \bar{r}) \in \operatorname{dom} h \times \mathcal{X} \times \mathcal{U}$ satisfying (2.15);

(0) set $\lambda = 1/(2m)$ and $c = \hat{c} + M/\|\mathcal{A}\|^2$;

(1) define the quantities

$$M_c := M + c\|\mathcal{A}\|^2, \quad f_c := f + \frac{c}{2}\|\mathcal{A}(\cdot) - b\|^2, \quad \phi_c = f_c + h, \tag{2.17}$$

and apply the AIPP method with inputs (m, M_c), (f_c, h), λ, σ, x_0, and $\bar{\rho}$ to obtain a $\bar{\rho}$–approximate solution $(\bar{x}, \bar{u})$ of (2.16);

(2) if $\|\mathcal{A}\bar{x} - b\|_{\mathcal{U}} > \bar{\eta}$ then set $c = 2c$ and go to (1); otherwise, set $\bar{r} = c\,(\mathcal{A}\bar{x} - b)$ and output the triple $(\bar{x}, \bar{u}, \bar{r})$.

We now give two remarks about the above method. First, it straightforward to see that QP-AIPP method terminates due to the results in [6, Section 4]. Second, in view of the second remark following Proposition 1 with $(\phi, M) = (\phi_c, M_c)$, it is easy to see that the number of ACG iterations executed in step 1 at any iteration of the method is

$$\mathcal{O}\left(\sqrt{\lambda M_c + 1}\left[\frac{R(\phi_c; \lambda)}{\lambda^2\bar{\rho}^2} + \log_1^+(\lambda M_c)\right]\right) \tag{2.18}$$

and that the pair $(\bar{x}, \bar{u})$ computed in step 2 satisfies the inclusion and the first inequality in (2.15).

We now focus on the iteration complexity of the QP-AIPP method. Before proceeding, we first define the useful quantity

$$R_c(\phi; \lambda) := \inf_{x'}\left\{\frac{1}{2}\|x_0 - x'\|^2 + \lambda\left[\phi(x') - \hat{\phi}_c\right] : x' \in \mathcal{F}\right\}, \tag{2.19}$$

for every $c \geq \hat{c}$, where ϕ_c is as defined in (2.14). The quantity in (2.19) plays an analogous role as (2.11) in (2.10) and, due to [6, Lemma 16], it also admits the upper bound

$$R_c(\phi; \lambda) \leq R_{\hat{c}}(\phi; \lambda) \leq \min\left\{\frac{1}{2}\hat{d}_0^2, \lambda\left[\hat{\phi}_* - \hat{\phi}_{\hat{c}}\right]\right\} \tag{2.20}$$

where $\hat{\phi}_*$ is as defined in (2.13) and

$$\hat{d}_0 := \inf\{\|x_0 - x_*\| : x_* \text{ is an optimal solution of (2.13)}\}.$$

We now state the iteration complexity of the QP-AIPP method, whose proof can be found in [6, Theorem 18].

PROPOSITION 2. *Let a constant $\hat{c}$ as in assumption (Q2), scalar $\sigma \in (0, 1)$, curvature pair $(m, M) \in \mathbb{R}^2_{++}$, and a tolerance pair $(\bar{\rho}, \bar{\eta}) \in \mathbb{R}^2_+$ be given. Moreover, define $\lambda := 1/(2m)$ and*

$$T_{\bar{\eta}} := \frac{2R_{\hat{c}}(\phi; \lambda)}{\bar{\eta}^2(1 - \sigma)\lambda} + \hat{c}, \quad \Xi := M + T_{\bar{\eta}}\|\mathcal{A}\|^2. \tag{2.21}$$

Then, the QP-AIPP method outputs a triple $(\bar{u}, \bar{x}, \bar{r})$ *satisfying* (2.15) *in at most*

$$\mathcal{O}\left(\sqrt{\lambda\Xi + 1}\left[\frac{R_{\hat{c}}(\phi;\lambda)}{\lambda^2\bar{\rho}^2} + \log_1^+(\lambda\Xi)\right]\right) \tag{2.22}$$

ACG iterations.

Similar to the remark at the end of Subsection 2.1, it is worth mentioning that a more practical variant of the above QP-AIPP method, called the relaxed QP-AIPP (R-QP-AIPP) method, is presented in [7]. This variant is more computationally efficient compared to the QP-AIPP method due to two key practical improvements, namely: (i) it replaces the AIPP method in step 1 of QP-AIPP method with the more practical R-AIPP method of [7] (see the last paragraph of Subsection 2.1); and (ii) it applies a warm–start strategy which chooses the input x_0 to the AIPP call for solving the next quadratic penalty subproblem as the output $\bar{x}$ from the AIPP call for solving the current one.

On the other hand, it is worth noting that there are key theoretical differences between the QP-AIPP and R-QP-AIPP methods. More specifically: (i) the iteration complexity of the latter method is established under the assumption that $\operatorname{dom} h$ be bounded, while no such assumption is needed for the iteration complexity of the former method; and (ii) the latter method replaces assumption (Q2) with the assumption that $\inf_x \phi(x)$ be finite. Hence, due to the limitations of the latter method, we decided to present the smoothing method of Section 4 for solving (1.4) in terms of the QP-AIPP method.

3. AIPP smoothing method.

The main goal of this section is to precisely describe the problem of interest in this paper and to describe ways of finding approximate solutions of this problem. It contains two subsections. The first subsection describes the problem of interest as well as several notions of approximate solutions for it. The second subsection details two ways of finding these approximate solutions.

3.1. The problem of interest.

Let $\mathcal{X}$ and $\mathcal{Y}$ be finite dimensional inner product spaces and let $X \subseteq \mathcal{X}$ and $Y \subseteq \mathcal{Y}$ be nonempty convex sets. Moreover, define

$$Z := X \times Y. \tag{3.1}$$

Given a real-valued function $\hat{\Phi} : Z \mapsto \mathbb{R}$, our problem of interest in this section is the min-max problem

$$\hat{p}_* := \min_{x\in X} \max_{y\in Y} \hat{\Phi}(x, y). \tag{3.2}$$

It is assumed that $\hat{\Phi}$ is endowed with a nonconvex composite structure on the space $\mathcal{X}$ which consists of the existence of a real valued function Φ whose domain contains Z and a function $h \in \overline{\operatorname{Conv}}(\mathcal{X})$ satisfying

$$\begin{aligned} &\operatorname{dom} h = X, \\ &\hat{\Phi}(x, y) = \Phi(x, y) + h(x) \quad \forall (x, y) \in Z, \end{aligned} \tag{3.3}$$

and the following three additional conditions:

(A1) $-\Phi(x, \cdot) \in \overline{\operatorname{Conv}}(\mathcal{Y})$ and $\operatorname{dom}[-\Phi(x, \cdot)] = Y$ for every $x \in X$;

(A2) $\Phi(\cdot, y)$ is continuously differentiable on X for every $y \in Y$;

(A3) there exist scalars $(L_x, L_y) \in \mathbb{R}^2_{++}$, and $m \in (0, L_x]$ such that

$$\Phi(x,y) - [\Phi(x',y) + \langle \nabla_x \Phi(x',y), x - x' \rangle] \geq -\frac{m}{2}\|x - x'\|^2, \tag{3.4}$$

$$\|\nabla_x \Phi(x,y) - \nabla_x \Phi(x',y')\| \leq L_x \|x - x'\| + L_y \|y - y'\|, \tag{3.5}$$

for every $x, x' \in X$ and $y, y' \in Y$.

It is also assumed that (3.2) satisfies:

(A4) Y is a closed set whose diameter $D_y := \sup\{\|y - y'\| : y, y' \in Y\}$ is finite;

(A5) $\hat{p}_*$ defined in (3.3) is finite.

We now make three remarks about the above assumptions. First, the composite structure (3.3) implies that (3.2) is equivalent to the (possibly nonsmooth) CNO problem

$$\min_x \{\hat{p}(x) := p(x) + h(x)\} \tag{3.6}$$

where p is given by

$$p(x) := \max_{y \in Y} \Phi(x,y) \quad \forall x \in X \tag{3.7}$$

and hence $\hat{p} = \max_{y \in Y} \hat{\Phi}(x,y)$. Second, it is well-known that (3.5) implies that

$$\Phi(x',y) - [\Phi(x,y) + \langle \nabla_x \Phi(x,y), x' - x \rangle] \leq \frac{L_x}{2}\|x' - x\|^2, \quad \forall (x', x, y) \in X \times X \times Y \tag{3.8}$$

Third, equation (3.4) implies that, for any $y \in Y$, the function $\Phi(\cdot, y) + m\|\cdot\|^2/2$ is convex, and hence $p + m\|\cdot\|^2/2$ is as well. Note that while $\hat{p}$ is generally nonconvex and nonsmooth, it also has the nice property that $\hat{p} + m\|\cdot\|^2/2$ is convex.

Even though we are only interested in the case where $m > 0$, it is worth discussing the case in which $m = 0$, and hence $\hat{p}$ is convex. First, finding an optimal solution of (3.2) is equivalent to finding a point $x^* \in X$ such that

$$\inf_{\|d\| \leq 1} \hat{p}'(x^*; d) \geq 0. \tag{3.9}$$

Second, it is well-known that (3.2) is related to the saddle-point problem which consists of finding a pair $(x^*, y^*) \in Z$ such that

$$\hat{\Phi}(x^*, y) \leq \hat{\Phi}(x^*, y^*) \leq \hat{\Phi}(x, y^*) \quad \forall (x,y) \in Z. \tag{3.10}$$

More specifically, (x^*, y^*) satisfies (3.10) if and only if x^* is an optimal solution of (3.2), y^* is an optimal solution of the dual of (3.2), and there is no duality gap between the two problems. Using the composite structure described above for $\hat{\Phi}$, it follows that (x^*, y^*) satisfies (3.10) if and only if

$$\begin{pmatrix} 0 \\ 0 \end{pmatrix} \in \begin{pmatrix} \nabla_x \Phi(x^*, y^*) \\ 0 \end{pmatrix} + \begin{pmatrix} \partial h(x^*) \\ \partial\left[-\Phi(x^*, \cdot)\right](y^*) \end{pmatrix}. \tag{3.11}$$

We will now discuss the case in which $m > 0$ in light of the remarks made in the previous paragraph. First, (3.9) is only a necessary condition for $x^* \in X$ to be an optimal solution of (3.2). Second, (3.11) is also only a necessary condition for (3.10) to hold. Finally, the problem of solving either a relaxed version of (3.10) or the problem of finding a near optimal solution of (3.2) is difficult in general. Hence, in this

paper we will only examine the problems of computing approximate solutions to (3.9) and (3.11). More specifically, we consider the following two notions of approximate stationary points. First, given a tolerance $\delta > 0$, a point $\hat{x} \in X$ is said to be a δ–directional-stationary point of (3.2) if $\hat{x}$ satisfies the first inequality in (1.2), which corresponds to an approximate solution of (3.9). Second, for a given tolerance pair $(\rho_x, \rho_y) \in \mathbb{R}^2_{++}$, a quadruple $(\bar{u}, \bar{v}, \bar{x}, \bar{y}) \in \mathcal{X} \times \mathcal{Y} \times X \times Y$ is said to be a (ρ_x, ρ_y)–saddle-stationary point of (3.2) if it satisfies (1.3), which corresponds to an approximate solution of (3.11).

Observe that (1.2) is generally hard to verify for a given point $\bar{x} \in X$. This is primarily because the definition requires us to check an infinite number of directional derivatives for a (potentially) nonsmooth function at points $\hat{x}$ near $\bar{x}$. In contrast, the definition of an approximate saddle-stationary point is generally easier to verify because the quantities $\|\bar{u}\|$ and $\|\bar{v}\|$ can be measured directly and the inclusions in (1.3) are easy to verify when h and $\Phi(x, \cdot)$, for every $x \in X$, are simple enough.

We are now ready to briefly discuss some approaches for finding approximate stationary points of (3.6). One approach is to apply a proximal descent type method directly to problem (3.6), but this would lead to subproblems with nonsmooth convex composite functions. A second approach is based on first applying a smoothing method to (3.6) and then using a prox-convexifying descent method such as the one in [6] to solve the perturbed smooth problem. An advantage of the second approach, which is the one pursued in this paper, is that it generates subproblems with smooth convex composite objective functions. The details of the latter approach are described in the next subsection.

3.2. AIPP smoothing approach. This subsection describes two ways of finding approximate stationary points of (3.6). More specifically, the first one described in Proposition 5 considers approximate solutions as in (1.2) and the second one described in Proposition 6 considers approximate solutions as in (1.3). Both ways consider a smooth approximation of (3.6) obtained by using a smoothing method similar to that used in [11], and then invoke the AIPP method described in Section 2.1 to solve the perturbed smooth problem.

We start this subsection by describing the aforementioned smoothing method. The main idea is to apply the AIPP method described in Section 2 to the minimization problem

$$\min_x \{\hat{p}_\xi(x) := p_\xi(x) + h(x)\} \tag{3.12}$$

where $\xi > 0$ and p_ξ is defined as

$$p_\xi(x) := \max_{y \in Y} \left\{ \Phi_\xi(x, y) := \Phi(x, y) - \frac{1}{2\xi}\|y - y_0\|^2 \right\} \quad \forall x \in X, \tag{3.13}$$

for some $y_0 \in Y$. The difference between (3.12) and (3.6) is that the function p in (3.7) is replaced by the function $p_\xi : X \mapsto \mathbb{R}$ which approximates p.

In order for this approach to be valid, we need to establish that (3.12) is a problem that can be solved by the AIPP method. As $h \in \overline{\text{Conv}}(\mathcal{X})$, it is sufficient to show that p_ξ satisfies assumption (P2) in Subsection 2.1. This is done in the following results which also give additional properties about the functions p_ξ and $\hat{p}_\xi$ as in (3.13) and (3.12), respectively, and the optimal solution of (3.13) as a function of x.

PROPOSITION 3. *Let $\xi > 0$ be given and assume that Φ is a real-valued function satisfying conditions (A1)–(A3) and whose domain contains Z. Let p_ξ and Φ_ξ be as defined in (3.13) and define*

$$y_\xi(x) := \operatorname*{argmax}_{y' \in Y} \Phi_\xi(x, y') \quad \forall x \in X. \tag{3.14}$$

Then, the following properties hold:

(a) y_ξ is Q_ξ–Lipschitz on X where

$$Q_\xi := \xi L_y + \sqrt{\xi(L_x + m)}; \tag{3.15}$$

(b) p_ξ is continuously differentiable on X and $\nabla p_\xi(x) = \nabla_x \Phi(x, y_\xi(x))$ for every $x \in X$;

(c) ∇p_ξ is L_ξ–Lipschitz on X where

$$L_\xi := L_y Q_\xi + L_x \le \left(L_y \sqrt{\xi} + \sqrt{L_x} \right)^2; \tag{3.16}$$

(d) for every $x, x' \in X$, we have

$$p_\xi(x) - [p_\xi(x') + \langle \nabla p_\xi(x'), x - x' \rangle] \ge -\frac{m}{2} \|x - x'\|^2; \tag{3.17}$$

Proof. The inequality in (3.16) follows from (a), the fact that $m \le L_x$, and the bound

$$L_\xi = L_y \left[\xi L_y + \sqrt{\xi(L_x + m)} \right] + L_x \le \xi L_y^2 + 2\sqrt{\xi L_x} + L_x = \left(L_y \sqrt{\xi} + \sqrt{L_x} \right)^2.$$

The other conclusions of (a)–(c) follow from Proposition 9 and Proposition 10 in Appendix B with $(\Psi, \psi) = (\Phi_\xi, p_\xi)$. We now show that the conclusion of (d) is true. Indeed, if we consider (3.4) at $(y, x') = (y_\xi(x'), x')$, the definition of Φ_ξ, and use the definition of ∇p_ξ in (b), then

$$\begin{aligned} &-\frac{m}{2} \|x - x'\|^2 \le \Phi(x', y_\xi(x)) - [\Phi(x, y_\xi(x)) + \langle \nabla_x \Phi(x, y_\xi(x)), x' - x \rangle] \\ &= \Phi_\xi(x', y_\xi(x)) - [p_\xi(x) + \langle \nabla p_\xi(x), x' - x \rangle] \le p_\xi(x') - [p_\xi(x) + \langle \nabla p_\xi(x), x' - x \rangle], \end{aligned}$$

where the last inequality follows from the optimality of y. ∎

LEMMA 4. *For every $\xi > 0$ and $\lambda \ge 0$ we have*

$$-\infty < \hat{p}(x) - \frac{D_y^2}{2\xi} \le \hat{p}_\xi(x) \le \hat{p}(x) \tag{3.18}$$

as well as

$$R(\hat{p}_\xi; \lambda) \le R(\hat{p}; \lambda) + \frac{\lambda D_y^2}{2\xi}. \tag{3.19}$$

Proof. (a) We first observe that for every $y_0 \in Y$ we have

$$\Phi(x, y) + h(x) \ge \Phi(x, y) - \frac{1}{2\xi} \|y - y_0\|^2 + h(x) \ge \Phi(x, y) + h(x) - \frac{D_y^2}{2\xi} \quad \forall (x, y) \in Z.$$

Hence, taking the supremum of the above quantities over $y \in Y$, using the definitions of $\hat{p}, \hat{p}_\xi, \Phi_\xi, p_\xi$, and assumption (A5) gives

$$-\infty < \hat{p}(x) - \frac{D_y^2}{2\xi} \leq \hat{p}_\xi(x) \leq \hat{p}(x) \quad \forall x \in X$$

which is the first set of inequalities. It now follows that

$$\hat{p}_\xi(x) - \inf_{x'} \hat{p}_\xi(x') \leq \hat{p}(x) - \inf_{x'} \hat{p}(x') + \frac{D_y^2}{2\xi}, \quad \forall x \in X. \tag{3.20}$$

Multiplying the above expression by $(1-\sigma)\lambda$ and adding the quantity $\|x_0 - x\|^2/2$ yields the inequality

$$\begin{aligned} &\frac{1}{2}\|x_0 - x\|^2 + (1-\sigma)\lambda \left[\hat{p}_\xi(x) - \inf_{x'} \hat{p}_\xi(x')\right] \\ &\leq \frac{1}{2}\|x_0 - x\|^2 + (1-\sigma)\lambda \left[\hat{p}(x) - \inf_{\tilde{x}} \hat{p}(x')\right] + (1-\sigma)\frac{\lambda D_y^2}{2\xi} \quad \forall x \in X, \end{aligned} \tag{3.21}$$

Taking the infimum of the above expression, and using the definition of $R(\cdot;\cdot)$ in (2.11) yields the conclusion of the lemma. ∎

We now make two remarks about Proposition 3. First, the Lipschitz constants of p_ξ and ∇p_ξ depend on the value of ξ while the lower curvature constant m in (3.17) does not. Second, in view of the fact that the AIPP method is applied to the smoothed problem (3.12) and the complexity bound (2.10) with $\phi = \hat{p}_\xi$, the quantity $R(\hat{p}_\xi; \lambda)$ naturally appears in the complexity bound for the method. The bound (3.19) can then be used to express the final bound in terms of $R(\hat{p}; \lambda)$, and hence in terms of the data of our problem of interest in this subsection (see the proofs of Proposition 5 and 6).

For the remainder of this section, we assume that subproblems of the form in (3.13) and (A.1) with $\psi_n = h$ are easily solvable for any $(x_0, y_0) \in \mathcal{X} \times \mathcal{Y}$ and $(\lambda, \xi) \in \mathbb{R}^2_{++}$. Note that (A.1) is needed as an oracle in the AIPP method while (3.13) is needed to compute p_ξ at various points in X.

We are now ready to state a smoothing approximation method for finding approximate stationary points of (3.2). It is stated in a incomplete manner in the sense that it does not specify how the approximation parameter ξ and the tolerance ρ used in its step 2 are chosen. Two specific instances of this method with different choices of ξ and ρ will be considered afterwards in Propositions 5 and 6 which describe the iteration-complexities for finding approximate solutions of (3.6) in the sense of (1.2) and (1.3), respectively.

AIPP smoothing (AIPP-S) method

Input: a triple $(m, L_x, L_y) \in \mathbb{R}^3_{++}$ satisfying assumption (A3), scalars $\lambda \in (0, 1/(2m)]$ and $\sigma \in (0,1)$, a smoothing constant $\xi > 0$, an initial point $(x_0, y_0) \in Z$, and a tolerance $\rho > 0$;

Output: a pair $(x, u) \in X \times \mathcal{X}$;

(0) set L_ξ as in (3.16) and define p_ξ as in (3.13);

(1) apply the AIPP method with inputs (m, L_ξ), (p_ξ, h), λ, σ, x_0, and ρ to obtain a pair (x, u) satisfying

$$u \in \nabla p_\xi(x) + \partial h(x), \quad \|u\| \leq \rho; \tag{3.22}$$

(2) output the pair (x, u).

Some remarks about the above method are in order. First, the AIPP method invoked in step 2 terminates due to [6, Theorem 13]. Second, since the AIPP-S method is a one-pass method (as opposed to an iterative method), the complexity of the AIPP-S method is essentially that of the AIPP method. Third, similar to the smoothing method of [11] which assumes $m = 0$, the AIPP-S method is also a smoothing method for the case in which $m > 0$. On the other hand, in contrast to the algorithm of [11] which uses an ACG variant, AIPP-S invokes the AIPP method to solve (3.12) due to its nonconvexity.

It is not clear how the pair output by the AIPP-S method is related to the definitions of approximate stationary points described in either (1.2) or (1.3). For the remainder of this subsection, our goal will be to show that a careful selection of the parameter ξ and the tolerance ρ will allow the AIPP-S method to generate approximate stationary points in the sense of (1.2) and (1.3).

We start by presenting a result showing how the AIPP-S method is able to generate a point that is *near* a δ–directional-stationary point, i.e. one satisfying (1.2).

PROPOSITION 5. *Let a tolerance $\delta > 0$ be given and consider the AIPP-S method with input parameter ξ and tolerance ρ satisfying*

$$\xi \geq \frac{D_y^2}{\delta^2} \max\left\{32m, \frac{8}{m}\right\}, \quad \rho = \frac{\delta}{2}. \tag{3.23}$$

Then, the following statements hold:

(a) the AIPP-S method performs

$$\mathcal{O}\left(\Omega_\xi \left[\frac{R(\hat{p};\lambda)}{\lambda^2\delta^2} + \frac{D_y^2}{\lambda\xi\delta^2} + \log_1^+(\Omega_\xi)\right]\right) \tag{3.24}$$

gradient and subproblem evaluations where $R(\cdot;\cdot)$ is as defined in (2.11), and

$$\Omega_\xi := 1 + \sqrt{\lambda}\left(\sqrt{\xi}L_y + \sqrt{L_x}\right); \tag{3.25}$$

(b) there exists a δ–directional-stationary point $\hat{x}$ (see the first inequality in (1.2)) satisfying $\|\hat{x} - x\| \leq \delta$ where x is the first argument in the pair output by the AIPP-S method.

Proof. (a) Let us first observe from (3.16) that

(3.26)

$$\sqrt{\lambda L_\xi + 1} \leq 1 + \sqrt{\lambda L_\xi} \leq 1 + \sqrt{\lambda\left(\xi^{1/2}L_y + L_x^{1/2}\right)^2} \leq 1 + \sqrt{\lambda\xi L_y^2} + \sqrt{\lambda L_x} = \Omega_\xi.$$

The complexity in (3.24) now follows from using (2.10) with $M = L_\xi$, Lemma 4 (in particular (3.19)), and the inequality in (3.26).

(b) Let (x, u) be the ρ–prox-approximate solution of (3.12) generated by the AIPP-S method (see step 2) with ξ and $\bar{\rho}$ satisfying (3.23). Define the quantities

$$\hat{q}_\xi := \hat{p}_\xi + \frac{1}{2\lambda}\|\cdot - x\|^2, \quad \hat{q} := \hat{p} + \frac{1}{2\lambda}\|\cdot - x\|^2, \quad \omega := \frac{D_y^2}{2\xi}, \tag{3.27}$$

and observe that Lemma 4 (in particular (3.18)), implies that

$$\hat{q} - \omega = \hat{q} - \frac{D_y^2}{2\xi} \leq \hat{q}_\xi \leq \hat{q}. \tag{3.28}$$

Also, (3.22) together with the fact that $p_\xi + \|\cdot - x\|^2/(2\lambda)$ is convex and differentiable implies that

$$u \in \nabla p_\xi(x) + \partial h(x) = \partial \left(p_\xi + \frac{1}{2\lambda}\|\cdot - x\|^2 \right)(x) + \partial h(x) = \partial \hat{q}_\xi(x) \tag{3.29}$$

Combining (3.28) with (3.29), we conclude that, for every $x' \in X$, we have

$$\hat{q}(x') \geq \hat{q}_\xi(x') \geq \hat{q}_\xi(x) + \langle u, x' - x\rangle \geq \hat{q}(x) + \langle u, x' - x\rangle - \omega,$$

which implies that $u \in \partial_\omega \hat{q}(x)$. Hence, the triple (x, u, ω) satisfies the inclusion in (C.1) with $(\phi, \varepsilon, x^-) = (\hat{p}, \omega, x)$. Since $\hat{q}$ is m–strongly convex from assumption (A3), we invoke Lemma 11 with $(\phi, \varepsilon, x^-, \mu) = (\hat{p}, \omega, x, m)$ to obtain the existence of a point $\hat{x} \in X$ satisfying

$$\inf_{\|d\|\leq 1} \hat{p}'(\hat{x}; d) \geq -\rho - 2\sqrt{2m\omega}, \quad \|\hat{x} - x\| \leq \sqrt{\frac{2\omega}{m}}. \tag{3.30}$$

The conclusion follows by observing the value of ω in (3.27) with the choice of ξ in (3.23), the fact that $\|u\| \leq \delta/2$ from (3.23), and applying these observations to (3.30). ∎

We now give four remarks about the above result. First, recall that $R(\hat{p}; \lambda)$ in the complexity (3.24) can be majorized by the rightmost quantity in (2.12). Second, Proposition 5(b) states that, while x not a stationary point itself, it is near a δ–directional-stationary point $\hat{x}$. Third, under the assumption that $\lambda = 1/(2m)$ and (3.23) is satisfied as equality, the complexity of the AIPP-S method reduces to

$$\mathcal{O}\left(m^{3/2} \cdot R(\hat{p}; \lambda) \cdot \left[\frac{L_x^{1/2}}{\delta^2} + \frac{L_y D_y}{\delta^3} \right] \right) \tag{3.31}$$

under the reasonable assumption that the $\mathcal{O}(\delta^{-2} + \delta^{-3})$ term in (3.24) dominates the other $\mathcal{O}(\delta^{-1})$ terms. Fourth, when Y is a singleton, it is easy to see that (3.6) becomes a special instance of (2.1), the AIPP-S method becomes equivalent to the AIPP method of Subsection 2.1, and the complexity in (3.31) reduces to

$$\mathcal{O}\left(\frac{m^{3/2} L_x^{1/2} R(\hat{p}; \lambda)}{\delta^2} \right). \tag{3.32}$$

In view of the last remark, the $\mathcal{O}(\delta^{-3})$ term in (3.31) is attributed to the (possible) nonsmoothness in (3.6).

Next, we present a result showing that the AIPP-S method is able to generate a (ρ_x, ρ_y)–saddle-stationary point, i.e. one satisfying (1.3).

PROPOSITION 6. *For a given tolerance pair $(\rho_x, \rho_y) \in \mathbb{R}^2_{++}$, let (x, u) be the pair output by the AIPP-S method with input parameter ξ and tolerance ρ satisfying*

$$\xi \geq \frac{D_y}{\rho_y}, \quad \rho = \rho_x. \tag{3.33}$$

Moreover, define

$$(\bar{u}, \bar{v}) := \left(u, \frac{y_0 - y_\xi(x)}{\xi}\right), \quad (\bar{x}, \bar{y}) := (x, y_\xi(x)), \tag{3.34}$$

where y_ξ is as in (3.14). Then, the following statements hold:

(a) the AIPP-S method performs

$$\mathcal{O}\left(\Omega_\xi \left[\frac{R(\hat{p};\lambda)}{\lambda^2 \rho_x^2} + \frac{D_y^2}{\lambda \xi \rho_x^2} + \log_1^+(\Omega_\xi)\right]\right) \tag{3.35}$$

gradient and subproblem evaluations where $R(\cdot;\cdot)$ and Ω_ξ are as in (2.11) and (3.25), respectively;

(b) the quadruple $(\bar{u}, \bar{v}, \bar{x}, \bar{y})$ is a (ρ_x, ρ_y)–saddle-approximate stationary point of (3.2).

Proof. (a) The complexity in (3.35) follows from the discussion after Proposition 1 with variables $(\phi, M) = (\hat{p}_\xi, L_\xi)$, the inequality (3.26), and Lemma 4 (in particular (3.19)).

(b) It follows from the definitions of p_ξ, tolerance ρ, and $(\bar{y}, \bar{u})$ in (3.13), (3.33), and (3.34), respectively, Proposition 3(b), and the inclusion in (3.22) that $\|\bar{u}\| \le \rho_x$ and

$$\bar{u} \in \nabla p_\xi(\bar{x}) + \partial h(\bar{x}) = \nabla_x \Phi(\bar{x}, y_\xi(\bar{x})) + \partial h(\bar{x}) = \nabla_x \Phi(\bar{x}, \bar{y}) + \partial h(\bar{x}).$$

Hence, we conclude that the top block and the upper bound on $\|\bar{u}\|$ in (1.3) hold. Next, the optimality condition of $\bar{y} = y_\xi(\bar{x})$ as a solution to (3.13) and the definition of $\bar{v}$ in in (3.13) give

$$0 \in \partial\left[-\Phi(\bar{x}, \cdot)\right](\bar{y}) + \frac{\bar{y} - y_0}{\xi} = \partial\left[-\Phi(\bar{x}, \cdot)\right](\bar{y}) - \bar{v} \tag{3.36}$$

Moreover, the definition of ξ implies

$$\|\bar{v}\| = \frac{\|\bar{y} - y_0\|}{\xi} \le \frac{D_y}{D_y/\rho_y} = \rho_y. \tag{3.37}$$

Hence, combining (3.36) and (3.37), we conclude that the bottom block and the upper bound on $\|\bar{v}\|$ in (1.3) hold. ∎

We now make three remarks about Proposition 6. First, recall that $R(\hat{p};\lambda)$ in the complexity (3.35) can be majorized by the rightmost quantity in (2.12). Second, under the assumption that $\lambda = 1/(2m)$ and (3.33) is satisfied as equality, the complexity of AIPP-S method reduces to

$$\mathcal{O}\left(m^{3/2} \cdot R(\hat{p};\lambda) \cdot \left[\frac{L_x^{1/2}}{\rho_x^2} + \frac{L_y D_y^{1/2}}{\rho_x^2 \rho_y^{1/2}}\right]\right) \tag{3.38}$$

under the reasonable assumption that the $\mathcal{O}(\rho_x^{-2} + \rho_x^{-2}\rho_y^{-1/2})$ term in (3.35) dominates the other terms. Third, recall from the last remark following the previous proposition that when Y is a singleton, (3.6) becomes a special instance of (2.1) and the AIPP-S method becomes equivalent to the AIPP method of Subsection 2.1. It similarly follows that the complexity in (3.38) reduces to

$$\mathcal{O}\left(\frac{m^{3/2} L_x^{1/2} R(\hat{p};\lambda)}{\rho_x^2}\right) \tag{3.39}$$

and, in view of this remark, the $\mathcal{O}(\rho_x^{-2}\rho_y^{-1/2})$ term in (3.38) is attributed to the (possible) nonsmoothness in (3.6).

4. Quadratic penalty AIPP-S method. This section studies a linearly constrained variant of problem 3.6, namely problem (1.4). More specifically, it discusses a notion of an approximate solution of (1.4) as well as an algorithm, named the QP-AIPP-S method, that can obtain such a solution.

Let $\mathcal{U}$ be a finite inner product space and let $\mathcal{X}, \mathcal{Y}$ and X, Y, Z be as defined in Subsection 3.1. Our problem of interest in this section is problem (1.4) where it is assumed $\hat{\Phi}$ has the nonconvex composite structure given in (3.3) and problem (1.4) satisfies assumptions (A1)–(A5) of Subsection 3.2. Moreover, it is assumed that conditions (Q1)–(Q2) of Subsection 2.2 hold with $\phi = p + h$ where p is given in (3.7).

We start by noting that (1.4) is the primal problem for the saddle function $\Psi : \mathcal{X} \times Y \times \mathcal{U} \to \mathbb{R}$ defined as

$$\Psi(x, y, r) := \Phi(x, y) + h(x) + \langle r, \mathcal{A}x - b\rangle \quad \forall (x, y, r) \in \mathcal{X} \times Y \times \mathcal{U}. \tag{4.1}$$

It is easy to see that a necessary condition for a triple $(\bar{x}, \bar{y}, \bar{r}) \in \mathcal{X} \times Y \times \mathcal{U}$ to be a saddle point of (4.1) is that (1.6) holds with $\rho_x = \rho_y = \eta = 0$. Clearly (1.6) is a relaxation of the latter necessary condition and a quintuple $(\bar{u}, \bar{v}, \bar{x}, \bar{y}, \bar{r}) \in \mathcal{X} \times \mathcal{Y} \times X \times Y \times \mathcal{U}$ satisfying it is referred to as a (ρ_x, ρ_y, η)–saddle-stationary point of (1.4). In this section, we will describe and study the complexity of an algorithm that obtains a saddle-stationary point of (1.4), which is based on the QP-AIPP method of Subsection 2.2.

We will now briefly outline aforementioned algorithm. First, we consider the smooth approximation of (1.4) which arises by replacing its objective function by $\hat{p}_\xi$ defined in (3.12), namely

$$\min_x \{\hat{p}_\xi(x) : \mathcal{A}x = b\}. \tag{4.2}$$

We now observe that the definition of $\hat{p}_\xi$ implies that (4.2) is of the form given in (2.13) where $f = p_\xi$. Since Proposition 3 implies that assumptions (P1)–(P3) of Subsection 2.1 and assumptions (Q1)–(Q2) of Subsection 2.2 are satisfied with $(f, h, M) = (p_\xi, h, L_\xi)$, the QP-AIPP method of Subsection 2.2 is used to solve (4.2).

In view of its description in Subsection 2.2, the QP-AIPP applied to (4.2) consists of solving penalty subproblems of the form

$$\min_x \left\{\hat{p}_\xi(x) + \frac{c}{2}\|\mathcal{A}x - b\|^2\right\} \tag{4.3}$$

for increasing values of c using the AIPP method of Subsection 2.1. Note that in order to solve the above subproblems, the AIPP method requires that subproblems of the form (3.13) and (A.1) are easily solvable.

We are now ready to state the QP-AIPP smoothing method for finding an approximate saddle-stationary point of (1.4).

Quadratic penalty AIPP smoothing (QP-AIPP-S) method

Input: a triple $(m, L_x, L_y) \in \mathbb{R}^2_{++}$ satisfying assumption (A4), a scalar $\sigma \in (0, 1)$, a scalar $\hat{c}$ satisfying assumption (Q2), a smoothing constant $\xi \geq D_y/\rho_y$, an initial point $(x_0, y_0) \in Z$, and a tolerance triple $(\rho_x, \rho_y, \eta) \in \mathbb{R}^3_{++}$;

Output: a triple $(\bar{u}, \bar{v}, \bar{x}, \bar{y}, \bar{r})$ satisfying (1.6);

(0) set L_ξ as in (3.16) and define p_ξ as in (3.13);
(1) apply the QP-AIPP method of Subsection 2.2 with inputs(m, L_ξ), (p_ξ, h), σ, $\hat{c}$, x_0, and (ρ_x, η) to obtain a triple $(\bar{u}, \bar{x}, \bar{r})$ satisfying

$$\bar{u} \in \nabla p_\xi(\bar{x}) + \partial h(\bar{x}) + A^*\bar{r}, \quad \|\bar{u}\| \le \rho_x, \quad \|\mathcal{A}\bar{x} - b\|_{\mathcal{U}} \le \eta. \tag{4.4}$$

(2) define $(\bar{v}, \bar{y})$ as in (3.34) and output the quintuple $(\bar{u}, \bar{v}, \bar{x}, \bar{y}, \bar{r})$.

We now make two brief remarks about the above algorithm. First, the QP-AIPP method invoked in step 1 terminates due to the results in Subsection 2.2. Second, since the QP-AIPP-S method is a one-pass algorithm (as opposed to an iterative algorithm), the complexity of the QP-AIPP-S method is essentially that of the QP-AIPP method.

The next result states two key facts about the QP-AIPP-S method.

PROPOSITION 7. *Let a tolerance triple $(\rho_x, \rho_x, \eta) \in \mathbb{R}^3_{++}$ be given and let the quadruple $(\bar{u}, \bar{v}, \bar{x}, \bar{y}, \bar{r})$ be the output obtained by the QP-AIPP-S method. Then the following properties hold:*

(a) the QP-AIPP-S method terminates in

$$\mathcal{O}\left(\Omega_{\xi,\eta}\left[\frac{R_{\hat{c}}(\hat{p};\lambda)}{\lambda^2\rho_x^2} + \frac{D_y^2}{\lambda\xi\rho_x^2} + \log_1^+(\Omega_{\xi,\eta})\right]\right) \tag{4.5}$$

gradient evaluations and subproblem evaluations where

$$\Omega_{\xi,\eta} := 1 + \frac{\|\mathcal{A}\|D_y}{\eta\sqrt{\xi}} + \sqrt{\lambda\xi L_y^2} + \sqrt{\lambda L_x} + \frac{\sqrt{\|\mathcal{A}\|^2 R_{\hat{c}}(\hat{p};\lambda)}}{\eta} \tag{4.6}$$

and $R_{\hat{c}}(\cdot,\cdot)$ is as defined in (2.19);
(b) the quintuple $(\bar{u}, \bar{v}, \bar{x}, \bar{y}, \bar{r})$ is a (ρ_x, ρ_y, η)–saddle-stationary point.

Proof. (a) Using the same arguments as in Lemma 4, it is easy to see that

$$R_{\hat{c}}(\hat{p}_\xi;\lambda) \le R_{\hat{c}}(\hat{p};\lambda) + \frac{\lambda D_y^2}{2\xi}, \tag{4.7}$$

where $R_{\hat{c}}(\cdot;\cdot)$ is as defined in (2.19). The complexity given in (4.5) now follows from applying Proposition 2 with $(\phi, f, M) = (p, p_\xi, L_\xi)$, using the bound in (3.26), and (4.7).

(b) It follows from Proposition 3(b), the definition of $\bar{y}$ in step 2 of the algorithm, and the inclusion in (4.4) that the quintuple $(\bar{u}, \bar{v}, \bar{x}, \bar{y}, \bar{r})$ satisfies the inclusions in (1.6), using similar arguments as in Proposition 6(b). Moreover, the inequalities in (1.6) follow from the inequalities in (4.4) and similar arguments as in Proposition 6(b). ∎

We now make three remarks about the above complexity bound. First, recall that $R_{\hat{c}}(p;\lambda)$ in the complexity (7) can be majorized by the rightmost quantity in (2.20). Second, under the assumption that $\xi = D_y/\rho_y$ and $\lambda = 1/(2m)$, the complexity of the QP-AIPP-S method reduces to

$$\mathcal{O}\left(m^{3/2} \cdot R_{\hat{c}}(\hat{p};\lambda) \cdot \left[\frac{L_x^{1/2}}{\rho_x^2} + \frac{L_y D_y^{1/2}}{\rho_y^{1/2}\rho_x^2} + \frac{m^{1/2}\|\mathcal{A}\| \cdot R_{\hat{c}}^{1/2}(p;\lambda)}{\eta\rho_x^2}\right]\right), \tag{4.8}$$

under the reasonable assumption that the $\mathcal{O}(\rho_x^{-2} + \eta^{-1}\rho_x^{-2} + \rho_y^{-1/2}\rho_x^{-2})$ term in (4.5) dominates the other terms. Third, when Y is a singleton, it is easy to see that (1.4) becomes a special instance of the smooth, linearly-constrained composite problem (2.13), the QP-AIPP-S of this subsection becomes equivalent to the QP-AIPP method of Subsection 2.2, and the complexity in (4.8) reduces to

$$\mathcal{O}\left(m^{3/2} \cdot R_{\hat{c}}(\hat{p};\lambda) \cdot \left[\frac{L_x^{1/2}}{\rho_x^2} + \frac{m^{1/2}\|\mathcal{A}\| \cdot R_{\hat{c}}^{1/2}(p;\lambda)}{\eta\rho_x^2}\right]\right). \tag{4.9}$$

In view of the last remark, the $\mathcal{O}(\rho_x^{-2}\rho_y^{-1/2})$ term in (4.8) is attributed to the (possible) nonsmoothness in (1.4).

Let us now conclude this section with a remark about the formulation in (4.3). It is easy to see that problem (4.3) can be equivalently reformulated as

$$\min_x \{\hat{p}_{c,\xi}(x) := p_{c,\xi}(x) + h(x)\}, \tag{4.10}$$

where $p_{c,\xi} : X \mapsto \mathbb{R}$ is defined as

$$p_{c,\xi}(x) := \max_{y\in Y, r\in\mathcal{U}} \left\{\Psi(x,y,r) - \frac{1}{2c}\|r\|^2 - \frac{1}{2\xi}\|y - y_0\|^2\right\} \quad \forall x \in X. \tag{4.11}$$

Moreover, problem (4.10) is similar to (3.12) in the sense that a smoothing procedure is applied to the underlying saddle function. On the other hand, observe that we cannot directly apply the smoothing method developed in Subsection 3.2 to (4.10) as the set $\mathcal{U}$ is generally unbounded. One approach that avoids this problem is to invoke the AIPP method of Subsection 2.1 to solve a sequence subproblems of the form (4.10) for increasing values of c. However, in view of the equivalence of (4.3) and (4.10), this is exactly the approach taken by the QP-AIPP-S of this section.

5. Numerical experiments. This section presents numerical results that illustrate the computational efficiency of the R-AIPP-S method mentioned in the last paragraph of Subsection 2.1. It contains three subsections. Each subsection presents computational results for a specific unconstrained nonconvex min-max optimization problem class.

Each unconstrained problem considered in this section is of the form in (3.2) and is such that the computation of the function y_ξ in (3.14) is easy. Moreover, for a given initial point $x_0 \in \operatorname{dom} h$, three algorithms are run for each problem instance until a quadruple $(\bar{u}, \bar{v}, \bar{x}, \bar{y})$ satisfying

$$\begin{aligned} \begin{pmatrix} \bar{u} \\ \bar{v} \end{pmatrix} &\in \begin{pmatrix} \nabla_x \Phi(\bar{x},\bar{y}) \\ 0 \end{pmatrix} + \begin{pmatrix} \partial h(\bar{x}) \\ \partial\left[-\Phi(\bar{x},\cdot)\right](\bar{y}) \end{pmatrix}, \\ \frac{\|\bar{u}\|}{\|\nabla p_\xi(z_0)\| + 1} &\le \rho_x, \quad \|\bar{v}\| \le \rho_y, \end{aligned} \tag{5.1}$$

is obtained, where $\xi = D_y/\rho_y$.

We first describe the three nonconvex-concave min-max methods that are being compared in this section, namely: (i) the R-AIPP-S method; (ii) the accelerated gradient smoothing (AG-S) method; and (iii) the projected gradient step framework (PGSF). Both the AG-S and R-AIPP-S methods are modifications of the AIPP-S method which, instead of using the AIPP method in its step 1, use the AG and R-AIPP methods, respectively. The PGSF is a simplified variant of Algorithm 2 of [12,

Subsection 4.1] which explicitly evaluates the argmax function $\alpha^*(\cdot)$ in [12, Section 4] instead of applying an accelerated gradient to estimate its evaluation.

The AG method is implemented as described in Algorithm 2 of [3]. We now describe the details of our implementation of the R-AIPP method. We assume that the reader is familiar with an iteration of the R-AIPP method of [7] and its various parameters, e.g. $\lambda, \lambda_k, \theta, \tau$, etc. A single iteration of our R-AIPP implementation is the same as an iteration of the R-AIPP method of [7] with $\theta = 4$, except that λ_k and τ are updated differently at the end of the iteration. In order to describe the update rule for λ_k, we first define an iteration $k \geq 1$ to be "good" if the intermediate parameter λ has not been halved in step 1 or 2 of the R-AIPP method for any iteration $j \leq k$. The update rule for λ_k at the end of the k^{th} iteration is then

$$\lambda_k = \begin{cases} \min\{2\lambda, 100/m\}, & \text{if iteration } k \text{ is "good"} \\ \lambda, & \text{otherwise,} \end{cases}$$

with λ_0 set to be $1/m$. The update rule for τ at the end of the k^{th} iteration is

$$\tau = \begin{cases} 1.5\tau_{\text{prev}}/\pi_k, & \text{if } \pi_k > 1.5, \\ 1.2\tau_{\text{prev}}/\pi_k, & \text{if } \pi_k < 1.2, \\ \tau_{\text{prev}}, & \text{otherwise,} \end{cases} \tag{5.2}$$

where τ_{prev} denotes the value of τ at the end of the $(k-1)^{\text{th}}$ iteration, $\pi_k := (\lambda_k \|\hat{v}_k\|)/\|v_k + z_{k-1} - z_k\|$, and τ_{prev} is set to be $10(\lambda_0 M + 1)$ at the first iteration. Also, like the other implementations of the R-AIPP method in [7], the R-AIPP implementation in this section adaptively estimates the constant $\widetilde{M}$ used in every iteration of the R-ACG subroutine that is called in its step 1.

It worth noting the update rules for λ_k and τ in the previous paragraph are different from the implementations of the R-AIPP method in [7]. More specifically, in [7] the iterates λ_k are simply set to be λ and τ is set to be τ_{prev} at every iteration $k \geq 1$. Moreover, the parameter τ in the experiments of [7] was chosen according to the problem class being examined and, hence, the update of τ in (5.2) has the advantage that it dynamically adjusts to the geometry of a problem instance.

We also state some additional details about the numerical experiments. First, each algorithm is run with a time limit of 4000 seconds. If an algorithm does not terminate with a solution for a particular problem instance, we do not report any details about its iteration count or function value at the point of termination and the runtime for that instance is marked with a [*] symbol. Second, the iterations listed in the tables of this section include even those that are extraneously performed due to an improper choice of an input parameter, e.g., λ_0 and $\widetilde{M}$. Third, the bold numbers in each of the computational tables in this section highlight the algorithm that performed the most efficiently in terms of iteration count or total runtime. Moreover, each of tables contain a column labeled $\hat{p}_\xi(\bar{x})$ that contains the smallest obtained value of the smoothed function in (3.12), across all of the tested algorithms. Fourth, the description of y_ξ and justification of the constants m, L_x, and L_y for each of the considered optimization problems are given in Appendix E. Fifth, y_0 is chosen to be 0 for all of the experiments. Finally, all algorithms described at the beginning of this section are implemented in MATLAB 2019a and are run on Linux 64-bit machines each containing Xeon E5520 processors and at least 8 GB of memory.

5.1. Maximum of a finite number of nonconvex functions. This subsection presents computational results for a minmax quadratic vector problem, which is

based on a similar problem in [7].

We first describe the problem. Given a dimension triple $(n, l, k) \in \mathbb{N}^3$, a set of parameters $\{(\alpha_i, \beta_i)\}_{i=1}^k \subseteq \mathbb{R}^2_{++}$, a set of vectors $\{d_i\}_{i=1}^k \subseteq \mathbb{R}^l$, a set of diagonal matrices $\{D_i\}_{i=1}^k \subseteq \mathbb{R}^{n\times n}$, and matrices $\{C_i\}_{i=1}^k \subseteq \mathbb{R}^{l\times n}$ and $\{B_i\}_{i=1}^k \subseteq \mathbb{R}^{n\times n}$, the problem of interest is the quadratic vector minmax (QVM) problem

$$\min_{x\in\mathbb{R}^n} \max_{y\in\mathbb{R}^k} \left\{ \delta_{\Delta^n}(x) + \sum_{i=1}^k y_i g_i(x) : y \in \Delta^k \right\},$$

where, for every index $1 \le i \le k$, integer $p \in \mathbb{N}$, and $x \in \mathbb{R}^n$,

$$g_i(x) := \frac{\alpha_i}{2}\|C_i x - d_i\|^2 - \frac{\beta_i}{2}\|D_i B_i x\|^2, \quad \Delta^p := \left\{ z \in \mathbb{R}^p_+ : \sum_{i=1}^p z_i = 1, z \ge 0 \right\}. \tag{5.3}$$

We now describe the experiment parameters for the instances considered. First, the dimensions are set to be $(n, l, k) = (200, 10, 5)$ and only 5.0% of the entries of the submatrices B_i and C_i are nonzero. Second, the entries of B_i, C_i, and d_i (resp., D_i) are generated by sampling from the uniform distribution $\mathcal{U}[0, 1]$ (resp., $\mathcal{U}[1, 1000]$). Third, the initial starting point is $z_0 = I_n/n$, where I_n is the n–dimensional identity matrix. Fourth, with respect to the termination criterion (5.1), the inputs, for every $(x, y) \in \mathbb{R}^n \times \mathbb{R}^k$, are

$$\Phi(x, y) = \sum_{i=1}^k y_i g_i(x), \quad h(x) = \delta_{\Delta^n}(x), \quad \rho_x = 10^{-2}, \quad \rho_y = 10^{-1}, \quad Y = \Delta^k. \tag{5.4}$$

Fifth, each problem instance considered is based on a specific curvature pair $(m, M) \in \mathbb{R}^2_{++}$ satisfying $m \le M$, for which each scalar pair $(\alpha_i, \beta_i) \in \mathbb{R}^2_{++}$ is selected so that

$$M = \lambda_{\max}(\nabla^2 g_i), \quad -m = \lambda_{\min}(\nabla^2 g_i). \tag{5.5}$$

Finally, the Lipschitz and curvature constants selected are

$$m = m, \quad L_x = M, \quad L_y = M\sqrt{k} + \|P\|, \tag{5.6}$$

where P is an n–by–k matrix whose i^{th} column is equal to $\alpha_i C_i^T d_i$.

We now present the results.

M	m	$\hat{p}_\xi(\bar{x})$	Iteration Count			Runtime		
			R-AIPP-S	AG-S	PGSF	R-AIPP-S	AG	PGSF
10^0	10^0	2.85E-01	**23**	294	1591	**0.66**	5.72	22.60
10^1	10^0	2.88E+00	**86**	1371	14815	**1.37**	25.96	209.62
10^2	10^0	2.85E+01	**217**	6270	150493	**3.35**	118.32	2122.93
10^3	10^0	2.85E+02	**1417**	28989	-	**21.58**	546.25	4000.00*

Table 5.1: Iteration counts and runtimes for QVM problems.

5.2. Truncated robust regression. This subsection presents computational results for the robust regression problem in [13].

It is worth mentioning that [13] also presents a min-max algorithm for obtaining a stationary point as in (5.1). However, its iteration complexity, which is $\mathcal{O}(\rho^{-6})$

when $\rho = \rho_x = \rho_y$, is significantly worse than the other algorithms considered in this section and, hence, we choose not to include this algorithm in our benchmarks.

We now describe the problem. Given a dimension pair $(n, k) \in \mathbb{N}^2$, a set of n data points $\{(a_j, b_j)\}_{i=1}^n \subseteq \mathbb{R}^k \times \{1, -1\}$ and a parameter $\alpha > 0$, the problem of interest is the truncated robust regression (TRR) problem

$$\min_{x\in\mathbb{R}^k} \max_{y\in\mathbb{R}^n} \left\{ \sum_{j=1}^n y_j(\phi_\alpha \circ \ell_j)(x) : y \in \Delta^n \right\}$$

where Δ^n is as in (5.3) with $p = n$ and, for every $(\alpha, t, x) \in \mathbb{R}_{++} \times \mathbb{R}_{++} \times \mathbb{R}^k$,

$$\phi_\alpha(t) := \alpha \log\left(1 + \frac{t}{\alpha}\right), \quad \ell_j(x) := \log\left(1 + e^{-b_j\langle a_j, x\rangle}\right).$$

We now describe the experiment parameters for the instances considered. First, α is set to 10 and the data points $\{(a_i, b_i)\}$ are taken from different datasets in the LIBSVM library[1] for which each problem instance is based off of (see the "data name" column in the table below, which corresponds to a particular LIBSVM dataset). Second, the initial starting point is $z_0 = 0$. Third, with respect to the termination criterion (5.1), the inputs, for every $(x, y) \in \mathbb{R}^k \times \mathbb{R}^n$, are

$$\Phi(x, y) = \sum_{j=1}^n y_j(\phi_\alpha \circ \ell_j)(x), \quad h(x) = 0, \quad \rho_x = 10^{-5}, \quad \rho_y = 10^{-3}, \quad Y = \Delta^n.$$

Finally, the Lipschitz and curvature constants selected are

$$m = L_x = \frac{1}{\alpha} \max_{1\le j\le n} \|a_j\|^2, \quad L_y = \sqrt{\sum_{j=1}^n \|a_j\|^2}. \tag{5.7}$$

We now present the results.

data name	$\hat{p}_\xi(\bar{x})$	Iteration Count			Runtime		
		R-AIPP-S	AG-S	PGSF	R-AIPP-S	AG	PGSF
heart	6.70E-01	**425**	1747	6409	**6.37**	15.54	32.76
diabetes	6.70E-01	**852**	1642	3718	**8.61**	24.12	52.77
ionosphere	6.70E-01	**1197**	8328	54481	**8.26**	63.82	320.72
sonar	6.70E-01	**45350**	96209	-	**461.52**	580.37	4000.00*
breast-cancer	1.11E-03	**46097**	-	-	**476.59**	4000.00*	4000.00*

Table 5.2: Iteration counts and runtimes for TRR problems

5.3. Power control in the presence of a jammer. This subsection presents computational results for the power control problem in [8].

It is worth mentioning that [8] also presents a min-max algorithm for obtaining stationary points for the aforementioned problem. However, its termination criterion and notion of stationarity are significantly different than what is being considered in this paper and, hence, we choose not to include the algorithm of [8] in our benchmarks.

[1] See https://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/binary.html.

We now describe the problem. Given a dimension pair $(N, K) \in \mathbb{N}^2$, a pair of parameters $(\sigma, R) \in \mathbb{R}^2_{++}$, a 3D tensor $\mathcal{A} \in \mathbb{R}^{K\times K\times N}_+$, and a matrix $B \in \mathbb{R}^{K\times N}_+$, the problem of interest is the power control (PC) problem

$$\min_{X\in\mathbb{R}^{K\times N}} \max_{y\in\mathbb{R}^N} \left\{ \sum_{k=1}^{K}\sum_{n=1}^{N} f_{k,n}(X,y) : 0 \le X \le R, 0 \le y \le \frac{N}{2}, \right\},$$

where, for every $(X, y) \in \mathbb{R}^{K\times N} \times \mathbb{R}^N$,

$$f_{k,n}(X,y) := -\log\left(1 + \frac{\mathcal{A}_{k,k,n}X_{k,n}}{\sigma^2 + B_{k,n}y_n + \sum_{j=1,j\neq k}^{K} \mathcal{A}_{j,k,n}X_{j,n}}\right).$$

We now describe the experiment parameters for the instances considered. First, the scalar parameters are set to be $(\sigma, R) = (1/\sqrt{2}, K^{1/K})$ and the quantities $\mathcal{A}$ and B are set to be the squared moduli of the entries of two Gaussian sampled complex–valued matrices $\mathcal{H} \in \mathbb{C}^{K\times K\times N}$ and $P \in \mathbb{C}^{K\times N}$. More precisely, the entries of $\mathcal{H}$ and P are sampled from the standard complex Gaussian distribution $\mathcal{CN}(0, 1)$ and

$$\mathcal{A}_{j,k,n} = |\mathcal{H}_{j,k,n}|^2, \quad B_{k,n} = |P_{k,n}|^2 \quad \forall (j,k,n).$$

Second, the initial starting point is $z_0 = 0$. Third, with respect to the termination criterion (5.1), the inputs, for every $(X, y) \in \mathbb{R}^{K\times N} \times \mathbb{R}^N$, are

$$\Phi(X,y) = \sum_{k=1}^{K}\sum_{n=1}^{N} f_{k,n}(X,y), \quad h(X) = \delta_{Q_R^{K\times N}}(X),$$

$$\rho_x = 10^{-1}, \quad \rho_y = 10^{-1}, \quad Y = Q_{N/2}^{N\times 1}.$$

where $Q_T^{U\times V} := \{z \in \mathbb{R}^{p\times q} : 0 \le z \le T\}$ for every $T > 0$ and $(U, V) \in \mathbb{N}^2$. Fourth, each problem instance considered is based on a specific dimension pair (N, K). Finally, the Lipschitz and curvature constants selected are

(5.8)
$$m = L_x = \frac{2}{\min\{\sigma^4, \sigma^6\}} \max_{\substack{1\le k\le K\\ 1\le n\le N}} \sum_{j=1}^{K} \mathcal{A}_{k,j,n}^2, \quad L_y = \frac{2}{\min\{\sigma^4, \sigma^6\}} \max_{\substack{1\le k\le K\\ 1\le n\le N}} \sum_{j=1}^{K} B_{j,n}\mathcal{A}_{k,j,n}.$$

We now present the results.

N	K	$\hat{p}_\xi(\bar{x})$	Iteration Count			Runtime		
			R-AIPP-S	AG-S	PGSF	R-AIPP-S	AG	PGSF
5	5	-3.64E+00	**37**	322832	-	**0.96**	2371.27	4000.00*
10	10	-2.82E+00	**54**	33399	-	**0.75**	293.60	4000.00*
25	25	-4.52E+00	**183**	-	-	**9.44**	4000.00*	4000.00*
50	50	-4.58E+00	**566**	-	-	**40.89**	4000.00*	4000.00*

Table 5.3: Iteration counts and runtimes for PC problems.

6. Concluding Remarks. This section makes some concluding remarks about the results obtained in Section 3.

Section 3 contains two variants of the AIPP-S method and analyzed their complexities with respect to two termination criteria, namely (1.2) and (1.3). We now

briefly deal with another termination criterion considered in [12], namely: finding a point $x \in X$ satisfying, for some $\delta > 0$,

$$\inf_{\|d\| \le 1} (\hat{p}_\xi)'(x; d) \ge -\delta, \quad \xi = \Theta\left(\delta^{-1}\right), \tag{6.1}$$

Assuming that (A1)–(A5) of Section 3 hold, the function h in (1.1) is identically 0, and $\Phi(x', \cdot)$ is differentiable and its gradient is uniformly (with respect to x') Lipschitz continuous for every $x' \in \mathcal{X}$, the algorithm in [12] finds a point $x \in X$ satisfying (6.1) in $\mathcal{O}(\delta^{-3})$ gradient and proximal subproblem evaluations. We now show that a specific instance of the AIPP-S method generates a point $\hat{x}$ satisfying (6.1) in $\mathcal{O}(\delta^{-2.5})$ gradient and proximal subproblem evaluations. Indeed, consider the instance of the AIPP-S method with inputs λ, ξ, and ρ given by

$$\lambda = \frac{1}{2m}, \quad \xi = \Theta\left(\delta^{-1}\right), \quad \rho = \delta, \tag{6.2}$$

and observe that a similar argument as in Proposition 5(a) shows that it performs $\mathcal{O}(\delta^{-2.5})$ gradient and proximal subproblem evaluations. Moreover, Lemma 12 with $\phi = \hat{p}_\xi$ describes how the output of this instance yields a computable point $x \in X$ satisfying (6.1). It is worth emphasizing that, in contrast to [12], the AIPP-S method requires neither that h be identically 0 nor that $\Phi(x', \cdot)$ be differentiable and its gradient be uniformly (with respect to x') Lipschitz continuous.

Appendix A. This appendix contains a description and a result about an ACG variant used in the analysis of [6].

Part of the input of the ACG variant, which is described below, consists of a pair of functions (ψ_s, ψ_n) satisfying:

- $\psi_n \in \overline{\text{Conv}}(\mathcal{Z})$ is μ–strongly convex for some $\mu \ge 0$;
- ψ_s is a convex differentiable function on $\operatorname{dom} \psi_n$ whose gradient is L–Lipschitz continuous for some $L > 0$.

ACG method

Input: a scalar pair $(\mu, L) \in \mathbb{R}^2_{++}$, a function pair (ψ_n, ψ_s), and an initial point $x_0 \in \operatorname{dom} h$;

(0) set $y_0 = x_0$, $A_0 = 0$, $\Gamma_0 \equiv 0$ and $j = 0$;

(1) compute

$$A_{j+1} = A_j + \frac{\mu A_j + 1 + \sqrt{(\mu A_j + 1)^2 + 4L(\mu A_j + 1)A_j}}{2L},$$

$$\tilde{x}_j = \frac{A_j}{A_{j+1}} x_j + \frac{A_{j+1} - A_j}{A_{j+1}} y_j,$$

$$\Gamma_{j+1} = \frac{A_j}{A_{j+1}} \Gamma_j + \frac{A_{j+1} - A_j}{A_{j+1}} \left[\psi_s(\tilde{x}_j) + \langle \nabla \psi_s(\tilde{x}_j), \cdot - \tilde{x}_j \rangle\right],$$

$$y_{j+1} = \operatorname*{argmin}_y \left\{ \Gamma_{j+1}(y) + \psi_n(y) + \frac{1}{2A_{j+1}} \|y - y_0\|^2 \right\},$$

$$x_{j+1} = \frac{A_j}{A_{j+1}} x_j + \frac{A_{j+1} - A_j}{A_{j+1}} y_{j+1};$$

(2) compute

$$u_{j+1} = \frac{y_0 - y_{j+1}}{A_{j+1}},$$

$$\eta_{j+1} = \psi(x_{j+1}) - \Gamma_{j+1}(y_{j+1}) - \psi_n(y_{j+1}) - \langle u_{j+1}, x_{j+1} - y_{j+1}\rangle;$$

(3) increment $j = j + 1$ and go to (1).

We observe that a single iteration of the ACG method requires the evaluation of two distinct oracles, namely: (i) the evaluation of the functions ψ_n, ψ_s, $\nabla\psi_s$ at any point in $\operatorname{dom}\psi_n$; and (ii) the computation of the exact solution of subproblems of the form

$$\min_{x'}\left\{\psi_n(x) + \frac{1}{2\lambda}\|x - a\|^2\right\} \tag{A.1}$$

for any $a \in \mathcal{X}$.

The following result, whose proof is given in [6, Lemma 9], is used to establish the iteration complexity of obtaining the triple (x, u, ε) in step 1 of the AIPP method of Subsection 2.1.

LEMMA 8. *Let $\{(A_j, x_j, u_j, \eta_j)\}$ be the sequence generated by the ACG method. Then, for any $\sigma > 0$, the ACG method obtains a triple (x, u, η) satisfying*

$$u \in \partial_\eta(\psi_s + \psi_n)(x) \quad \|u\|^2 + 2\eta \le \sigma\|x_0 - x + u\|^2 \tag{A.2}$$

in at most $\left\lceil 2\sqrt{2L}(1 + \sqrt{\sigma})/\sqrt{\sigma}\right\rceil$ iterations.

Appendix B. This appendix contains results about functions that can described as the maximum of a family of differentiable functions.

PROPOSITION 9. *Suppose Ψ is a real-valued function that satisfies assumptions (A1)–(A2) with $\Phi = \Psi$ and that $Y \subseteq \mathcal{Y}$ is bounded. Moreover, suppose $\nabla_x\Psi(\cdot,\cdot)$ is continuous on Z and, for every $x \in X$, the function $\Psi(x,\cdot)$ is μ–strongly concave on Y for some $\mu > 0$. Defining the functions*

$$\psi(x) := \max_{y' \in \mathcal{Y}}\{\Psi(x, y') : y' \in Y\} \quad \forall x \in X, \tag{B.1}$$

$$y(x) := \operatorname*{argmax}_{y' \in \mathcal{Y}}\{\Psi(x, y') : y' \in Y\} \quad \forall x \in X, \tag{B.2}$$

we have that the following properties hold:

(a) y is continuous on X;

(b) ψ is continuously differentiable on X and $\nabla\psi(x) = \nabla_x\Psi(x, y(x))$ for every $x \in X$.

Proof. (a) Let $x, \tilde{x} \in X$ be given and denote $(y, \tilde{y}) = (y(x), y(\tilde{x}))$. Using the compactness of Y and the assumption that $\nabla_x\Psi(\cdot,\cdot)$ is continuous, it is easy to see that

$$\limsup_{\tilde{x}\to x}\|\nabla_x\Psi(\tilde{x}, y) - \nabla_x\Psi(\tilde{x}, \tilde{y})\| \le K \tag{B.3}$$

where

$$K = K(x) := \max_{y,y' \in Y}\|\nabla_x\Psi(x, y) - \nabla_x\Psi(x, y')\| < \infty. \tag{B.4}$$

Next, define the function $\alpha : X \mapsto \mathbb{R}$ by

$$\alpha(u) := \Psi(u, y) - \Psi(u, \tilde{y}) \quad \forall u \in X. \tag{B.5}$$

Remark that the optimality conditions of y and $\tilde{y}$ imply that

$$\alpha(x) \geq \frac{\mu}{2}\|y-\tilde{y}\|^2, \quad -\alpha(\tilde{x}) \geq \frac{\mu}{2}\|y-\tilde{y}\|^2. \tag{B.6}$$

Adding the two above inequalities and using the Mean Value Theorem on the functions $\Psi(\cdot, y)$ and $\Psi(\cdot, \tilde{y})$, along with the Cauchy-Schwarz inequality, we conclude that there exists $\hat{x} \in [x, \tilde{x}]$ such that

$$\begin{aligned} \mu\|y-\tilde{y}\|^2 &\leq \alpha(x) - \alpha(\tilde{x}) = \langle \nabla_x \Psi(\hat{x}, y) - \nabla_x \Psi(\hat{x}, \tilde{y}), x - \tilde{x}\rangle \\ &\leq \|\nabla_x \Psi(\hat{x}, y) - \nabla_x \Psi(\hat{x}, \tilde{y})\|\|x-\tilde{x}\| \end{aligned} \tag{B.7}$$

The conclusion of (a) now follows from combining the above with (B.3) and (B.4).

(b) Let $x, \tilde{x} \in X$ be given and denote $(y, \tilde{y}) = (y(x), y(\tilde{x}))$. Observe that the optimality of $\tilde{y}$ and the Mean Value Theorem on the function $\Psi(\cdot, y)$ yield, for some $\hat{x}_\ell \in [\tilde{x}, x]$, the lower bound

$$\begin{aligned} &\psi(\tilde{x}) - \psi(x) - \langle \nabla_x \Psi(x, y), \tilde{x} - x\rangle \geq \Psi(\tilde{x}, y) - \Psi(x, y) - \langle \nabla_x \Psi(x, y), \tilde{x} - x\rangle \\ &= \langle \nabla_x \Psi(\hat{x}_\ell, y) - \nabla_x \Psi(x, y), \tilde{x} - x\rangle \geq -\|\tilde{x} - x\|\|\nabla_x \Psi(\hat{x}_\ell, y) - \nabla_x \Psi(x, y)\| \end{aligned}$$

Since $\nabla_x \Psi(\cdot, y)$ is continuous, we have that $\nabla_x \Psi(\hat{x}_\ell, y) \to \nabla_x \Psi(x, y)$ as $\tilde{x} \to x$. Hence,

$$\psi(\tilde{x}) - \psi(x) - \langle \nabla_x \Psi(x, y), \tilde{x} - x\rangle \geq o(\|\tilde{x} - x\|). \tag{B.8}$$

Conversely, observe that the optimality of y and the Mean Value Theorem on the function $\Psi(\cdot, \tilde{y})$ yield, for some $\hat{x}_u \in [\tilde{x}, x]$, the upper bound

$$\begin{aligned} &\psi(\tilde{x}) - \psi(x) - \langle \nabla_x \Psi(x, y), \tilde{x} - x\rangle \leq \Psi(\tilde{x}, \tilde{y}) - \Psi(x, \tilde{y}) - \langle \nabla_x \Psi(x, y), \tilde{x} - x\rangle \\ &= \langle \nabla_x \Psi(\hat{x}_u, \tilde{y}) - \nabla_x \Psi(x, y), \tilde{x} - x\rangle \leq \|\tilde{x} - x\|\|\nabla_x \Psi(\hat{x}_u, \tilde{y}) - \nabla_x \Psi(x, y)\|. \end{aligned}$$

Since $\nabla_x \Psi(\cdot, \cdot)$ is assumed to be continuous and $y(\cdot)$ is continuous from part (a), we have that $\nabla_x \Psi(\hat{x}_u, \tilde{y}) \to \nabla_x \Psi(x, y)$ as $\tilde{x} \to x$. Hence,

$$\psi(\tilde{x}) - \psi(x) - \langle \nabla \Psi(x, y), \tilde{x} - x\rangle \leq o(\|\tilde{x} - x\|). \tag{B.9}$$

Combining (B.8) and (B.9) now gives the conclusion of (b). ∎

PROPOSITION 10. *Suppose Ψ is a real-valued function that satisfies the assumptions of Proposition 9 and let ψ and y be as defined in (B.1) and (B.2), respectively. If Ψ also satisfies assumption (A3) with $\Phi = \Psi$ then the following additional properties hold:*

(a) y is Q_μ–Lipschitz on X where

$$Q_\mu := \frac{L_y}{\mu} + \sqrt{\frac{L_x + m}{\mu}}; \tag{B.10}$$

(b) $\nabla\psi$ is L_μ–Lipschitz on X where

$$L_\mu := L_y Q_\mu + L_x. \tag{B.11}$$

Proof. (a) Let $x, \tilde{x} \in X$ be given and denote $(y, \tilde{y}) = (y(x), y(\tilde{x}))$. Let α be as defined in (B.5) and observe that (B.6) still holds. Using (B.6), (3.4), (3.5), (3.8), and the Cauchy-Schwarz inequality, we conclude that

$$\begin{aligned}\mu\|y-\tilde{y}\|^2 \le \alpha(x)-\alpha(\tilde{x}) &\le \langle \nabla_x\Psi(x,y)-\nabla_x\Psi(x,\tilde{y}), x-\tilde{x}\rangle + \frac{L_x+m}{2}\|x-\tilde{x}\|^2 \\ &\le \|\nabla_x\Psi(x,y)-\nabla_x\Psi(x,\tilde{y})\|\|x-\tilde{x}\| + \frac{L_x+m}{2}\|x-\tilde{x}\|^2 \\ &\le L_y\|y-\tilde{y}\|\|x-\tilde{x}\| + \frac{L_x+m}{2}\|x-\tilde{x}\|^2.\end{aligned}$$

Considering the above as a quadratic inequality in $\|\tilde{y}-y\|$ yields the bound

$$\begin{aligned}\|y-\tilde{y}\| &\le \frac{1}{2\mu}\left[L_y\|x-\tilde{x}\| + \sqrt{L_y^2\|x-\tilde{x}\|^2 + 4\mu(L_x+m)\|x-\tilde{x}\|^2}\right] \\ &\le \left[\frac{L_y}{\mu} + \sqrt{\frac{L_x+m}{\mu}}\right]\|x-\tilde{x}\| = Q_\mu\|x-\tilde{x}\|\end{aligned}$$

which is the conclusion of (a).

(b) Let $x, \tilde{x} \in X$ be given and denote $(y, \tilde{y}) = (y(x), y(\tilde{x}))$. Using part (a) and (3.5) we have that

$$\begin{aligned}\|\nabla\psi(x)-\nabla\psi(\tilde{x})\| &= \|\nabla_x\Phi(x,y)-\nabla_x\Phi(\tilde{x},\tilde{y})\| \\ &\le \|\nabla_x\Phi(x,y)-\nabla_x\Phi(x,\tilde{y})\| + \|\nabla_x\Phi(x,\tilde{y})-\nabla_x\Phi(\tilde{x},\tilde{y})\| \\ &\le L_y\|y-\tilde{y}\| + L_x\|x-\tilde{x}\| \le (L_yQ_\mu + L_x)\|x-\tilde{x}\| = L_\mu\|x-\tilde{x}\|,\end{aligned}$$

which is the conclusion of (b). ∎

Appendix C. This appendix contains a result that relates approximate solutions of a function ϕ with lower curvature and the directional derivatives of ϕ. It is worth mentioning that this result does not require the differentiability of ϕ.

LEMMA 11. *Let a proper closed function $\phi : \mathcal{X} \mapsto (-\infty, \infty]$ and assume that $\phi + \|\cdot\|^2/2\lambda$ is μ-strongly convex for some scalars $\mu, \lambda > 0$. If a quadruple $(x^-, x, u, \varepsilon) \in \mathcal{X} \times \operatorname{dom}\phi \times \mathcal{X} \times \mathbb{R}_+$ together with λ satisfy*

$$u \in \partial_\varepsilon\left(\phi + \frac{1}{2\lambda}\|\cdot - x^-\|^2\right)(x), \tag{C.1}$$

then the point $\hat{x} \in \operatorname{dom}\phi$ given by

$$\hat{x} := \operatorname*{argmin}_{x'}\left\{\phi_\lambda(x') := \phi(x') + \frac{1}{2\lambda}\|x'-x^-\|^2 - \langle u, x'\rangle\right\} \tag{C.2}$$

satisfies

$$\inf_{\|d\|\le 1}\phi'(\hat{x};d) \ge -\frac{\|x^- - x + \lambda u\|}{\lambda} - \sqrt{\frac{2\varepsilon}{\mu\lambda^2}}, \quad \|\hat{x}-x\| \le \sqrt{\frac{2\varepsilon}{\mu}}. \tag{C.3}$$

Proof. We first observe that (C.1) implies that

$$\phi_\lambda(x') \ge \phi_\lambda(x) - \hat{\varepsilon} \quad \forall x' \in \mathcal{X}. \tag{C.4}$$

Remark that (C.4) at $x' = \hat{x}$, the optimality of $\hat{x}$, and the μ–strong convexity of ϕ_λ imply that

$$\frac{\mu}{2}\|\hat{x} - x\|^2 \leq \phi_\lambda(x) - \phi_\lambda(\hat{x}) \leq \hat{\varepsilon}$$

from which we conclude that $\|\hat{x} - x\| \leq \sqrt{2\hat{\varepsilon}/\mu}$. On the other hand, using the definition of ϕ_λ, we obtain

$$\begin{aligned} 0 \leq \inf_{\|d\|\leq 1} \phi'_\lambda(\hat{x}; d) &= \inf_{\|d\|\leq 1} \phi'(\hat{x}; d) - \frac{1}{\lambda}\left\langle d, \lambda u + x^- - \hat{x}\right\rangle \\ &\leq \inf_{\|d\|\leq 1} \phi'(\hat{x}; d) + \frac{\|x^- - \hat{x} + \lambda u\|}{\lambda} \\ &\leq \inf_{\|d\|\leq 1} \phi'(\hat{x}; d) + \frac{\|x^- - \hat{x} + \lambda u\|}{\lambda} + \frac{1}{\lambda}\|x - \hat{x}\|. \end{aligned} \tag{C.5}$$

Combining (C.5) with the bound on $\|\hat{x} - x\|$ now yields the result. ∎

Appendix D. This appendix contains a result that relates approximate solutions of a smooth composite function ϕ with lower curvature and the directional derivatives of ϕ.

LEMMA 12. *Let ϕ be a function with the composite structure given in (2.1) and such that the function pair (f, h) in (2.1) also satisfies assumptions (P1)–(P2) of Subsection 2.1. Moreover, given a tolerance $\rho > 0$, let (x, u) be a ρ–approximate solution of (2.1). We then have that*

$$\inf_{\|d\|\leq 1} \phi'(x; d) \geq -\rho. \tag{D.1}$$

Proof. Let (x, u) be a ρ–approximate solution of (2.1), or equivalently,

$$u \in \nabla f(x) + \partial h(x), \quad \|u\| \leq \rho.$$

Using the definition of $\phi'(\cdot, \cdot)$ together with the above, we directly conclude that

$$\begin{aligned} \inf_{\|d\|\leq 1} \phi'(x; d) &= \inf_{\|d\|\leq 1} \left[\langle \nabla f(x), d\rangle + \sup_{s\in\partial h(x)} \langle s, d\rangle\right] = \inf_{\|d\|\leq 1} \sup_{\tilde{u}\in\nabla f(x)+\partial h(x)} \langle \tilde{u}, d\rangle \\ &= \sup_{\tilde{u}\in\nabla f(x)+\partial h(x)} \inf_{\|d\|\leq 1} \langle \tilde{u}, d\rangle = \sup_{u\in\nabla f(x)+\partial h(x)} -\|\tilde{u}\| \geq -\|u\| \geq -\rho. \end{aligned}$$

∎

Observe that the conclusions of Lemmas 11 and 12 are similar in that the quantities being bounded are the same. However, they differ in that the former does not assume that ϕ has a smooth composite structure while the latter does and, as a consequence, shows that $\hat{x}$ can be obtained by a single evaluation of the resolvent of h.

Appendix E. This appendix presents the description of y_ξ and justification for the constants m, L_x, and L_y for each of the optimization problems in Section 5.

Maximum of a finite number of nonconvex functions. Since $Y = \Delta^k$, it is easy to verify that

$$y_\xi(x) = \operatorname*{argmax}_{y'} \left\{\|y' - \xi g_i(x)\| : y' \in \Delta^k\right\} \quad \forall x \in \mathbb{R}^n.$$

For the validity of the constants m, L_x, and L_y, we first define, for every $1 \leq i \leq k$, the quantities

$$P_i = \alpha_i C_i^T d_i, \quad Q_i^x := \alpha_i C_i^T C_i x - \beta_i B_i^T D_i^T D_i B_i x \quad \forall x \in \mathbb{R}^n,$$

and observe that $\nabla_x \Phi(x, y) = \sum_{i=1}^k (Q_i^x + P_i) y_i$. Now, using the fact that $y \in \Delta^k$, (5.5), and defining $N_i := \alpha_i C_i^T C_i - \beta_i B_i^T D_i^T D_i B_i$, we then have that

$$\lambda_{\max}(\nabla^2_{xx} \Phi) \leq \sum_{i=1}^k y_i \lambda_{\max}(N_i) = \sum_{i=1}^k y_i \lambda_{\max}(\nabla^2 g_i) \leq M = L_x,$$

$$\lambda_{\min}(\nabla^2_{xx} \Phi) \geq \sum_{i=1}^k y_i \lambda_{\min}(N_i) = \sum_{i=1}^k y_i \lambda_{\min}(\nabla^2 g_i) \geq -m \geq -L_x,$$

and hence we conclude that the choice of m and L_x in (5.6) is valid. On the other hand, using the fact that $\|x\| \leq 1$ for every $x \in \Delta^n$ and (5.5), we then have that for every $y, y' \in Y$,

$$\|\nabla_x \Phi(x, y) - \nabla_x \Phi(x, y')\| = \left\| \sum_{i=1}^k (Q_i^x + P_i)(y_i - y_i') \right\|$$

$$\leq \left(\sqrt{\sum_{i=1}^k M^2 \|x\|^2} + \|P\| \right) \|y - y'\| \leq L_y \|y - y'\|,$$

where P is a an n–by–k matrix whose i^{th} column is $\alpha_i C_i^T d_i$, and hence we conclude that the choice of L_y in (5.6) is valid.

Truncated robust regression. Since $Y = \Delta^n$, it is easy to verify that

$$y_\xi(x) = \operatorname*{argmax}_{y'} \{\|y' - \xi g_i(x)\| : y' \in \Delta^n\} \quad \forall x \in \mathbb{R}^k.$$

For the validity of the constants m, L_x, and L_y, we first define for every $1 \leq i \leq k$ the function

$$\tau_j(x) := \left[e^{-b_j \langle a_j, x \rangle}\right] \left[1 + e^{-b_j \langle a_j, x \rangle}\right]^{-1} [\alpha + \ell_j(x)]^{-1} \quad \forall x \in \mathbb{R}^k,$$

and observe that $\nabla_x \Phi(x, y) = -\alpha \sum_{j=1}^n [y_j b_j \tau_j(x)] a_j$ and also that

$$\sup_{x \in \mathbb{R}^k} |\tau_j(x)| \leq \alpha^{-1}, \tag{E.1}$$

for every $1 \leq j \leq n$. Now, using the fact that $y \in \Delta^n$, the bound (E.1), and the Mean Value Theorem applied to τ_j, we have that for every $x, x' \in \mathbb{R}^k$,

$$\|\nabla_x \Phi(x, y) - \nabla_x \Phi(x', y)\| \leq \alpha \sum_{j=1}^n y_j \|a_j [\tau_j(x) - \tau_j(x')]\|$$

$$\leq \alpha \max_j (\|a_j [\tau_j(x) - \tau_j(x')]\|) = \alpha \max_{1 \leq j \leq n} [\|a_j\| \cdot |\tau_j(x) - \tau_j(x')|]$$

$$\leq \alpha \max_{1 \leq j \leq n} \left[\|a_j\| \sup_{x \in \mathbb{R}^k} \|\nabla \tau_j(x)\| \|x - x'\| \right] = \alpha \max_{1 \leq j \leq n} \left[\|a_j\|^2 \sup_{x \in \mathbb{R}^k} \left| \frac{\tau_j(z)}{\alpha + \ell_j(z)} \right| \right] \|x - x'\|$$

$$\leq \frac{1}{\alpha} \max_{1\leq j\leq n} \|a_j\|^2 \|x - x'\| = L_x \|x - x'\|,$$

and hence we conclude that the choice of $m = L_x$ in (5.7) is valid. On the other hand, using the bound (E.1), we have that for every $y, y' \in \mathbb{R}^n$,

$$\|\nabla_x \Phi(x, y) - \nabla_x \Phi(x, y')\| = \alpha \left\| \sum_{j=1}^{n} b_j \tau_j(x) a_j [y_j - y'_j] \right\| \leq L_y \|y - y'\|,$$

and hence we conclude that the choice of L_y in (5.7) is valid.

Power control in the presence of a jammer. For every $1 \leq k \leq K$ and $1 \leq n \leq N$, we first define the quantities

$$S^-_{k,n}(X, y) := \sigma^2 + B_{k,n} y_n + \sum_{j=1, j\neq k}^{K} \mathcal{A}_{j,k,n} X_{j,n}, \quad S_{k,n}(X, y) := \mathcal{A}_{k,k,n} X_{k,n} + S^-_{k,n},$$

as well as

$$T_{j,n}(X, y) := \left[S^-_{j,n}(X, y) + S_{j,n}(X, y)\right] / \left[S_{j,n}(X, y) S^-_{j,n}(X, y)\right]^2,$$

for every $(X, y) \in \mathbb{R}^{K\times N} \times \mathbb{R}^N$. Observe now that

$$\frac{\partial \Phi}{\partial y_n}(X, y) = \frac{B_{k,n}}{S_{k,n}(X, y) S^-_{k,n}(X, y)} \quad \forall n \in \{1, ..., N\}. \tag{E.2}$$

The form in (E.2) implies that $\nabla_y \Phi(X, y)$ is a separable function in y where each component is a monotonically decreasing function in its argument. Hence, since $Y = Q^{N\times 1}_{N/2}$, the computation of y_ξ reduces to an N–dimensional bisection search on the functions

$$F_n(y; \xi) = \left[\sum_{k=1}^{K} \frac{B_{k,n}}{S_{k,n}(X, y) S^-_{k,n}(X, y)}\right] - \frac{y_n}{\xi} \quad \forall n \in \{1, ..., N\}.$$

For the validity of the constants m, L_x, and L_y, we first observe that, for every $1 \leq k \leq K$ and $1 \leq n \leq N$ and also $(X, y) \in \mathbb{R}^{K\times N} \times \mathbb{R}^N$, we have

$$\frac{\partial \Phi}{\partial X_{k,n}}(X, y) = -\frac{A_{k,k,n}}{S_{k,n}(X, y)} + \sum_{j=1, j\neq k}^{K} \frac{A_{k,j,n}}{S_{j,n}(X, y) S^-_{j,n}(X, y)} \quad \forall (X, y) \in \mathbb{R}^{K\times N} \times \mathbb{R}^N.$$

Using the Mean Value Theorem with respect to $X_{k,n}$ on $\partial\Phi/\partial X_{k,n}$, we have that for every $X, X' \in \mathbb{R}^{K\times N}$,

$$\left|\frac{\partial}{\partial X_{k,n}} f(X, y) - \frac{\partial}{\partial X_{k,n}} f(X', y)\right| \leq \sup_{(X,y)\in\mathbb{R}^{K\times N}\times\mathbb{R}^N} \left|\frac{\partial^2}{\partial X^2_{k,n}} f(X, y)\right| |X_{k,n} - X'_{k,n}|$$

$$= \sup_{(X,y)\in\mathbb{R}^{K\times N}\times\mathbb{R}^N} \left|\frac{\mathcal{A}^2_{k,k,n}}{S_{k,n}(X, y)} - \sum_{j=1, j\neq k}^{K} \mathcal{A}^2_{k,j,n} T_{j,n}(X, y)\right| |X_{k,n} - X'_{k,n}|$$

$$\leq \frac{2\sum_{j=1}^{K} \mathcal{A}^2_{k,j,n}}{\min\{\sigma^4, \sigma^6\}} |X_{k,n} - X'_{k,n}| \leq L_x |X_{k,n} - X'_{k,n}|,$$

and hence we conclude that the choice of L_x in (5.8) is valid. On the other hand, using the Mean Value Theorem with respect to y_n on $\partial\Phi/\partial X_{k,n}$, we have that for every $y, y' \in \mathbb{R}^{K\times N}$,

$$\begin{aligned}
&\left|\frac{\partial}{\partial X_{k,n}} f(X,y) - \frac{\partial}{\partial X_{k,n}} f(X',y)\right| \leq \sup_{(X,y)\in\mathbb{R}^{K\times N}\times\mathbb{R}^N} \left|\frac{\partial^2}{\partial y_n X_{k,n}} f(X,y)\right| |y_n - y_n'| \\
&= \sup_{(X,y)\in\mathbb{R}^{K\times N}\times\mathbb{R}^N} \left|\frac{B_{k,n}\mathcal{A}_{k,k,n}}{S_{k,n}(X,y)} - \sum_{j=1,j\neq k}^{K} B_{k,n}\mathcal{A}_{k,j,n}T_{j,n}(X,y)\right| |y_n - y_n'| \\
&\leq \frac{2\sum_{j=1}^{K} B_{k,n}\mathcal{A}_{k,j,n}}{\min\{\sigma^4,\sigma^6\}} |y_n - y_n'| \leq L_y |y_n - y_n'|,
\end{aligned}$$

and hence we conclude that the choice of L_y in (5.8) is valid.

REFERENCES

[1] K. Arrow, L. Hurwicz, and H. Uzawa, *Studies in linear and non-linear programming*, Cambridge Univ. Press, 1958.

[2] Y. Carmon, J. Duchi, O. Hinder, and A. Sidford, *Accelerated methods for non-convex optimization*, Available on arXiv:1611.00756, (2017).

[3] S. Ghadimi and G. Lan, *Accelerated gradient methods for nonconvex nonlinear and stochastic programming*, Math. Program., 156 (2016), pp. 59–99.

[4] Y. He and R. Monteiro, *An accelerated HPE-type algorithm for a class of composite convex-concave saddle-point problems*, SIAM J. Optim., 26 (2016), pp. 29–56.

[5] O. Kolossoski and R. Monteiro, *An accelerated non-euclidean hybrid proximal extragradient-type algorithm for convex-concave saddle-point problems*, Optim. Methods Softw., 32 (2017), pp. 1244–1272.

[6] W. Kong, J. Melo, and R. Monteiro, *Complexity of a quadratic penalty accelerated inexact proximal point method for solving linearly constrained nonconvex composite programs*, SIAM Journal on Optimization, 29 (2019), pp. 2566–2593.

[7] W. Kong, J. G. Melo, and R. D. C. Monteiro, *An efficient adaptive accelerated inexact proximal point method for solving linearly constrained nonconvex composite problems*, arXiv e-prints, (2018), arXiv:1812.06352, p. arXiv:1812.06352, https://arxiv.org/abs/1812.06352.

[8] S. Lu, I. Tsaknakis, and M. Hong, *Block alternating optimization for non-convex min-max problems: Algorithms and applications in signal processing and communications*, ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), (2019), pp. 4754–4758.

[9] A. Nemirovski, *Prox-method with rate of convergence $O(1/t)$ for variational inequalities with lipschitz continuous monotone operators and smooth convex-concave saddle point problems*, SIAM J. Optim., 15 (2004), pp. 229–251.

[10] A. Nemirovski and D. Yudin, *Cesari convergence of the gradient method of approximating saddle points of convex-concave functions*, in Dokl. Akad. Nauk, vol. 239, Russian Academy of Sciences, 1978, pp. 1056–1059.

[11] Y. Nesterov, *Smooth minimization of non-smooth functions*, Math. Program., 103 (2005), pp. 127–152.

[12] M. Nouiehed, M. Sanjabi, T. Huang, J. Lee, and M. Razaviyayn, *Solving a class of non-convex min-max games using iterative first order methods*, in Advances in Neural Information Processing Systems, 2019, pp. 14905–14916.

[13] H. Rafique, M. Liu, Q. Lin, and T. Yang, *Non-convex min-max optimization: Provable algorithms and applications in machine learning*, arXiv e-prints, (2018), arXiv:1810.02060, https://arxiv.org/abs/1810.02060.

[14] R. Rockafellar, *Convex Analysis*, Princeton University Press, Princeton, 1970.