

REDUCING THE COMPLEXITY OF TWO CLASSES OF
OPTIMIZATION PROBLEMS BY INEXACT ACCELERATED
PROXIMAL GRADIENT METHOD*QIHANG LIN[†] AND YANGYANG XU[‡]

Abstract. We propose a double-loop inexact accelerated proximal gradient (APG) method for a strongly convex composite optimization problem with two smooth components of different smoothness constants and computational costs. Compared to APG, the inexact APG can reduce the time complexity for finding a near-stationary point when one smooth component has higher computational cost but a smaller smoothness constant than the other. The strongly convex composite optimization problem with this property arises from subproblems of a regularized augmented Lagrangian method for affine-constrained composite convex optimization and also from the smooth approximation for bilinear saddle-point structured nonsmooth convex optimization. We show that the inexact APG method can be applied to these two problems and reduce the time complexity for finding a near-stationary solution. Numerical experiments demonstrate significantly higher efficiency of our methods over an optimal primal-dual first-order method by Hamedani and Aybat [*SIAM J. Optim.*, 31 (2021), pp. 1299–1329] and the gradient sliding method by Lan, Ouyang, and Zhou [arXiv2101.00143, 2021].

Key words. first-order method, constrained optimization, saddle-point nonsmooth optimization

MSC codes. 65K05, 68Q25, 90C30, 90C60

DOI. 10.1137/22M1469584

1. Introduction. We consider *composite optimization* in the form of

$$(1.1) \quad F^* = \min_{\mathbf{x} \in \mathbb{R}^n} \{F(\mathbf{x}) = g(\mathbf{x}) + H(\mathbf{x})\} \text{ with } H(\mathbf{x}) = h(\mathbf{x}) + r(\mathbf{x}),$$

where g is L_g -smooth and μ -strongly convex with $\mu \geq 0$, h is convex and L_h -smooth, and r is closed convex with an easy proximal mapping and an easy projection onto $\partial r(\cdot)$. This problem arises in many applications, e.g., sparse regression [65, 82], multitask learning [15], matrix completion [8], and sparse inverse covariance estimation [16].

Besides (1.1) itself, we also study its application in the numerical schemes to solve two classes of convex problems. One is affine-constrained composite optimization,

$$(1.2) \quad \min_{\mathbf{x} \in \mathbb{R}^n} f(\mathbf{x}) + r(\mathbf{x}) \text{ s.t. } \mathbf{Ax} = \mathbf{b},$$

and the other is bilinear saddle-point structured nonsmooth optimization,

$$(1.3) \quad \min_{\mathbf{x} \in \mathbb{R}^n} \left\{ f(\mathbf{x}) + r(\mathbf{x}) + \max_{\mathbf{y} \in \mathbb{R}^m} [\langle \mathbf{y}, \mathbf{Ax} \rangle - \phi(\mathbf{y})] \right\}.$$

* Received by the editors January 6, 2022; accepted for publication (in revised form) September 30, 2022; published electronically January 19, 2023.

<https://doi.org/10.1137/22M1469584>

Funding: This work is partly supported by NSF grants DMS-2053493 and DMS-2208394 and by ONR award N00014-22-1-2573.

[†] Department of Business Analytics, University of Iowa, Iowa City, IA 52242 USA (qihanglin@uiowa.edu).

[‡] Department of Mathematical Sciences, Rensselaer Polytechnic Institute, Troy, NY 12180 USA (xuy21@rpi.edu).

In both problems, we assume that f is L_f -smooth μ -strongly convex with $\mu \geq 0$, while r is similar to that in (1.1). Also, ϕ is closed convex, has a bounded domain, and admits an easy proximal mapping. For simplicity, we only consider equality constraints in (1.2) in this section, but we will consider *both equality and inequality constraints* in the main body of the paper as shown in (5.1). The applications of (1.2) can be found in linearly constrained LASSO problems [17, 30] and shape-restricted nonparametric regression [11], and problem (1.3) arises in overlapping group LASSO [10, 29, 81], fused LASSO [10, 66], and robust principal component analysis [7].

Most of the existing works target an ε -optimal solution of (1.1), namely, a solution $\bar{\mathbf{x}}$ satisfying $F(\bar{\mathbf{x}}) - F^* \leq \varepsilon$. In contrast, we aim at finding an ε -stationary solution of (1.1), namely, a solution $\bar{\mathbf{x}}$ satisfying $\|\xi\| \leq \varepsilon$ for some $\xi \in \partial F(\bar{\mathbf{x}})$. It is easy to obtain an $O(\sqrt{\varepsilon})$ -stationary solution (see (3.15) below) from an ε -optimal solution, which, however, may not be a near-stationary solution. For example, $\bar{x} = \varepsilon$ is an ε -optimal solution of $\min_x |x|$, but it is not a near-stationary solution for any $\varepsilon > 0$. On the contrary, an ε -stationary point $\bar{\mathbf{x}}$ of (1.1) is also an $\varepsilon\|\bar{\mathbf{x}} - \mathbf{x}^*\|$ -optimal solution by $F(\bar{\mathbf{x}}) - F(\mathbf{x}^*) \leq \langle \xi, \bar{\mathbf{x}} - \mathbf{x}^* \rangle \leq \|\xi\| \cdot \|\bar{\mathbf{x}} - \mathbf{x}^*\|$ for any $\xi \in \partial F(\bar{\mathbf{x}})$, where \mathbf{x}^* is one minimizer. In addition, an ε -stationary solution can be verified in practice more easily than an ε -optimal solution. For this reason, we also focus on computing ε -stationary solutions (defined later in Definitions 5.1 and (6.1) of (1.2) and (1.3)).

1.1. Composite subproblems/approximation. Both (1.2) and (1.3) can be solved by numerical procedures that solve instances of (1.1) as we discuss below.

We consider solving (1.2) by an *inexact regularized augmented Lagrangian method* (iRALM), which performs the following update in the k th main iteration:

$$(1.4) \quad \mathbf{x}^{(k+1)} \approx \arg \min_{\mathbf{x} \in \mathbb{R}^n} f(\mathbf{x}) + r(\mathbf{x}) + \langle \boldsymbol{\lambda}^{(k)}, \mathbf{A}\mathbf{x} - \mathbf{b} \rangle + \frac{\beta_k}{2} \|\mathbf{A}\mathbf{x} - \mathbf{b}\|^2 + \frac{\rho_k}{2} \|\mathbf{x} - \mathbf{x}^{(k)}\|^2.$$

Here $\mathbf{x}^{(k)}$ is the main iterate, $\boldsymbol{\lambda}^{(k)}$ is the Lagrange multiplier, $\beta_k > 0$ is a penalty parameter, and $\rho_k > 0$ is a regularization parameter. It is easy to see that the problem in (1.4) is an instance of (1.1) with

$$(1.5) \quad g(\mathbf{x}) = f(\mathbf{x}) + \frac{\rho_k}{2} \|\mathbf{x} - \mathbf{x}^{(k)}\|^2 \quad \text{and} \quad h(\mathbf{x}) = \langle \boldsymbol{\lambda}^{(k)}, \mathbf{A}\mathbf{x} - \mathbf{b} \rangle + \frac{\beta_k}{2} \|\mathbf{A}\mathbf{x} - \mathbf{b}\|^2$$

and the smoothness constants are $L_g = L_f + \rho_k$ and $L_h = \beta_k \|\mathbf{A}\|^2$.

For (1.3), we use the smoothing technique by [54], which approximates (1.3) by

$$(1.6) \quad \min_{\mathbf{x} \in \mathbb{R}^n} \left\{ f(\mathbf{x}) + r(\mathbf{x}) + \max_{\mathbf{y} \in \mathbb{R}^m} \left[\langle \mathbf{y}, \mathbf{A}\mathbf{x} \rangle - \phi(\mathbf{y}) - \frac{\rho}{2} \|\mathbf{y} - \mathbf{y}^{(0)}\|^2 \right] \right\}$$

and solves (1.6) using a smooth optimization method. Here, $\rho > 0$ is a smoothing parameter, and $\mathbf{y}^{(0)} \in \text{dom}(\phi)$. Again, we can view (1.6) as an instance of (1.1) with

$$(1.7) \quad g(\mathbf{x}) = f(\mathbf{x}) \quad \text{and} \quad h(\mathbf{x}) = \max_{\mathbf{y} \in \mathbb{R}^m} \left[\langle \mathbf{y}, \mathbf{A}\mathbf{x} \rangle - \phi(\mathbf{y}) - \frac{\rho}{2} \|\mathbf{y} - \mathbf{y}^{(0)}\|^2 \right]$$

and the smoothness constants $L_g = L_f$ and $L_h = \|\mathbf{A}\|^2/\rho$.

We consider solving (1.2) and (1.3) by gradient-based methods which only need to query $(f, \nabla f)$ and $(\mathbf{A}(\cdot), \mathbf{A}^\top(\cdot))$ and use the proximal mappings of r and ϕ . We are interested in the *oracle complexity* of the studied methods, which is defined as the numbers of queries that the methods make to $(f, \nabla f)$ and $(\mathbf{A}(\cdot), \mathbf{A}^\top(\cdot))$, denoted by Q_f and $Q_{\mathbf{A}}$, respectively, until an ε -stationary point is found. Similarly, we define the

oracle complexity of a method for (1.1) as the numbers of queries it makes to $(g, \nabla g)$ and $(h, \nabla h)$, denoted by Q_g and Q_h , respectively, until an ε -stationary point is found. In contrast to oracle complexity, we define the *time complexity* or *cost* of a numerical procedure as the total number of arithmetic operations it performs. Additionally, we focus on a practical scenario where the time complexity for querying $(f, \nabla f)$ is significantly higher than $(\mathbf{A}(\cdot), \mathbf{A}^\top(\cdot))$. This scenario arises from many applications in statistics and machine learning, e.g., linearly constrained LASSO problems [17, 30], where querying $(f, \nabla f)$ requires processing a large amount of data while querying $(\mathbf{A}(\cdot), \mathbf{A}^\top(\cdot))$ does not involve any data and can be relatively easy.

1.2. Contributions. Our main contribution is to show that when the time complexity of querying $(f, \nabla f)$ is significantly higher than $(\mathbf{A}(\cdot), \mathbf{A}^\top(\cdot))$, the known time complexity in the literature for finding ε -stationary points of (1.2) and (1.3) can be further reduced if we solve (1.4) and (1.6) using an *inexact accelerated proximal gradient* (iAPG) method, which queries $(f, \nabla f)$ significantly fewer than $(\mathbf{A}(\cdot), \mathbf{A}^\top(\cdot))$.

Our iAPG is a double-loop variant of the APG [2, 52, 54, 55, 70]. When applied to (1.1), the APG treats $G := g + h$ as a whole and solves (1.1) by

$$(1.8) \quad \mathbf{x}^{(k+1)} = \arg \min_{\mathbf{x} \in \mathbb{R}^n} \left\langle \nabla G(\mathbf{y}^{(k)}), \mathbf{x} - \mathbf{y}^{(k)} \right\rangle + \frac{1}{2\eta_k} \|\mathbf{x} - \mathbf{y}^{(k)}\|^2 + r(\mathbf{x}) \text{ for } k \geq 0,$$

where $\mathbf{y}^{(k)} \in \mathbb{R}^n$ is an auxiliary iterate and $\eta_k > 0$ is a step length parameter. By the assumption made on r , (1.8) can be solved easily, e.g., in a closed form. When $\mu > 0$, it is known (see, e.g., [52]) that the APG finds an ε -optimal solution for (1.1) with $Q_g = Q_h = O(\sqrt{\frac{L_g + L_h}{\mu}} \ln(\frac{1}{\varepsilon}))$. However, according to the instantiations in (1.5) and (1.7), querying $(g, \nabla g)$ has significantly higher time complexity than $(h, \nabla h)$ in both instances since the former requires querying $(f, \nabla f)$ while the latter only requires querying $(\mathbf{A}(\cdot), \mathbf{A}^\top(\cdot))$. Given that, a potential strategy to reduce the time complexity for solving (1.1), and thus (1.2) and (1.3), is to query $(g, \nabla g)$ and $(h, \nabla h)$ in different frequencies so as to reduce Q_g , even if doing so may slightly increase Q_h .

To implement this strategy, one technique is to separate g and h by solving the following *proximal mapping* subproblem in the k th iteration:

$$(1.9) \quad \mathbf{x}^{(k+1)} = \arg \min_{\mathbf{x} \in \mathbb{R}^n} \left\langle \nabla g(\mathbf{y}^{(k)}), \mathbf{x} - \mathbf{y}^{(k)} \right\rangle + \frac{1}{2\eta_k} \|\mathbf{x} - \mathbf{y}^{(k)}\|^2 + h(\mathbf{x}) + r(\mathbf{x}).$$

Unlike (1.8), (1.9) typically cannot be solved explicitly. A practical solution is to use an iterative method to solve (1.9) inexactly to a certain precision. This requires a double-loop implementation. Note that (1.9) itself is a strongly convex instance of (1.1) and thus can be solved inexactly by the APG in oracle complexity with logarithmic dependency on the precision. By choosing an appropriate precision for solving (1.9) in each iteration, we show that, when $\mu > 0$, our iAPG can find an ε -stationary solution of (1.1) with oracle complexity¹

$$(1.10) \quad Q_g = O\left(\sqrt{\frac{L_g}{\mu}} \ln\left(\frac{1}{\varepsilon}\right)\right) \text{ and } Q_h = \tilde{O}\left(\sqrt{\frac{L_g + L_h}{\mu}} \ln\left(\frac{1}{\varepsilon}\right)\right).$$

The iAPG has lower time complexity than the APG when L_h is significantly larger than L_g and querying $(g, \nabla g)$ is much more costly than $(h, \nabla h)$.

¹Here and in the rest of the paper, \tilde{O} suppresses some logarithmic terms.

According to (1.5), the iAPG has lower time complexity than the APG for solving (1.4) when β_k is much larger than ρ_k , which is indeed the case in the iRALM. As a consequence, we show that the iRALM, in which (1.4) is solved by the iAPG, finds an ε -stationary point of (1.2) with oracle complexity²

$$(1.11) \quad Q_f = O\left(\sqrt{\frac{L_f}{\mu}} \ln^2\left(\frac{1}{\varepsilon}\right)\right) \text{ and } Q_{\mathbf{A}} = \tilde{O}\left(\sqrt{\frac{L_f}{\mu}} \ln\left(\frac{1}{\varepsilon}\right) + \frac{\|\mathbf{A}\|}{\sqrt{\mu\varepsilon}}\right).$$

Without the affine constraint $\mathbf{A}\mathbf{x} = \mathbf{b}$, it is shown by [51, 52] that any gradient-based method has to query $(f, \nabla f)$ at least $\Omega(\sqrt{\frac{L_f}{\mu}} \ln(\frac{1}{\varepsilon}))$ times to find an ε -optimal point of (1.2). With $\mathbf{A}\mathbf{x} = \mathbf{b}$, it is shown by [57] that any gradient-based method needs to query $(\mathbf{A}(\cdot), \mathbf{A}^\top(\cdot))$ at least $O(\frac{\|\mathbf{A}\|}{\sqrt{\mu\varepsilon}})$ times. In either case, the oracle complexity of the iRALM matches the corresponding lower bound up to logarithmic factors.

Similarly, according to (1.7), the iAPG has lower time complexity than the APG when ρ is small, which is true for the smoothing method. In fact, to obtain an ε -optimal point of (1.3) by solving (1.6), one needs to set $\rho = \Theta(\varepsilon)$. In this case, we show that when $\mu > 0$, the smoothing method, where (1.6) is solved by the iAPG, finds an ε -stationary point of (1.3) with the same oracle complexity as in (1.11). This complexity matches the lower bound [57] up to logarithmic factors.

Summary of contributions. We summarize our contributions mentioned above.

- We present an iAPG method for solving (1.1). It is a double-loop method where the inner iterations are terminated using a computable stopping criterion based on the stationarity measure of the solution. We prove the oracle complexity of the proposed iAPG is given in (1.10). When evaluating $(g, \nabla g)$ has significantly higher cost than $(h, \nabla h)$ but L_g is much smaller than L_h , the iAPG is superior to the APG for solving (1.1). Compared to the existing iAPGs, e.g., [35], our analysis focuses on the strongly convex case which has important applications in (1.2) and (1.3). Moreover, our method includes a line search scheme on the step length parameter to improve the practical performance while other iAPGs do not.
- Applying the proposed iAPG to the subproblems of the iRALM for (1.2), we derive in (1.11) the oracle complexity of the iRALM for finding an ε -stationary solution. This complexity is better than existing ones, e.g., [20, 73], when querying $(f, \nabla f)$ is significantly more costly than $(\mathbf{A}(\cdot), \mathbf{A}^\top(\cdot))$. The complexity in [41] is similar to ours.³ However, the inner loop of their method requires a predetermined number of iterations, which is often conservative and yields poor practical performance; see the numerical results in section 7. Additionally, we show that the iAPG combined with the smoothing technique [54] can find an ε -stationary solution of (1.3) with oracle complexity in (1.11), which is also better than existing ones.

1.3. Notation. $\mathbf{x} \odot \mathbf{y}$ denotes the componentwise product of two vectors \mathbf{x} and \mathbf{y} . For any number sequence $\{a_i\}_{i \geq 0}$, we define $\sum_{i=k_1}^{k_2} a_i = 0$ and $\prod_{i=k_1}^{k_2} a_i = 1$ if $k_1 > k_2$. The proximal mapping of a function r is $\mathbf{prox}_r(\mathbf{z}) := \arg \min_{\mathbf{x}} \{\frac{1}{2} \|\mathbf{x} - \mathbf{z}\|^2 + r(\mathbf{x})\}$. The distance of a point \mathbf{z} to a set \mathcal{S} is defined as $\mathbf{dist}(\mathbf{z}, \mathcal{S}) := \min_{\mathbf{x} \in \mathcal{S}} \|\mathbf{x} - \mathbf{z}\|$.

²The factor $\ln^2(\frac{1}{\varepsilon})$ in Q_f can be reduced to $\ln \frac{1}{\varepsilon}$ if $\beta_0 = \Theta(\frac{1}{\varepsilon})$ and $\rho_0 = \Theta(\varepsilon)$; see Remark 1.

³The complexity in [41] is lower than that in (1.11) by a logarithmic factor. However, [41] targets an ε -optimal solution which is hard to verify.

2. Literature review. The APG methods [2, 52, 54, 55, 70] are optimal gradient-based methods for (1.1). However, the APGs cannot be directly applied to (1.2) due to the affine constraints or to (1.3) due to the sophisticated nonsmooth term. The iAPG performs similar updates as an APG except that the proximal mapping subproblem (1.9) is solved inexactly by another optimization algorithm, making the iAPG a double-loop algorithm. Different iAPGs have been studied in the literature based on different inexactness criteria when solving the subproblems [4, 31, 34, 35, 63, 71].

2.1. Related iAPGs. The iAPG in [63] assumes that an ε_k -optimal solution of (1.9) can be found while the iAPG by [31] requires a solution of (1.9) that satisfies an inexact criterion based on the $O(\varepsilon_k^2/k^2)$ -subgradient of H . Both papers assume the summability of $\{\varepsilon_k\}$. They analyze the number of outer iterations for finding an ε -optimal solution of (1.1) but not the oracle complexity for solving (1.9). In contrast, we show the total oracle complexity for finding an ε -stationary solution of (1.1), which can be verified more easily than an ε -optimal solution.

The iAPG by [31] can be directly applied to (1.2) by viewing r in (1.1) as an indicator function of the constraint set of (1.2). This way, (1.9) becomes a quadratic program with affine constraints. Then, an inexact semismooth Newton-conjugate gradient method is applied to compute an inexact solution to (1.9) that approximately satisfies the primal-dual optimality conditions. However, they only analyze the number of outer iterations but not the total oracle complexity.

When (1.1) is convex but not strongly convex, the iAPG by [71] minimizes the duality gap of (1.9) using an APG method to find an approximate solution of (1.9) satisfying an inexact condition defined with the ε_k -subdifferential of H . Choosing $\varepsilon_k = 1/k^q$, it can find an ε -optimal solution of (1.1) with oracle complexity $Q_g = O(\frac{1}{\sqrt{\varepsilon}})$ and $Q_h = O(\frac{1}{\varepsilon^q})$ for q arbitrarily close to $\frac{3}{2}$. Under the same setting, the iAPG by [4] assumes an approximate solution to (1.9) that satisfies either an inexact relative rule or an inexact extra-step relative rule. With oracle complexity $Q_g = O(\frac{L_g^{2/3}}{\varepsilon^{2/3}})$, it finds a solution to (1.1) whose ε -subgradient has a norm no greater than ε , which is weaker than an ε -stationary point. They do not analyze the complexity for computing the inexact solution to (1.9) so Q_h is unknown.

The inner accelerated inexact composite gradient (IA-ICG) method and the doubly accelerated inexact composite gradient (DA-ICG) method proposed by [35] can be applied to (1.1). Both methods apply a relaxed accelerated gradient algorithm to find a solution of (1.9) satisfying two error inequalities (see Problem B in [35]). When (1.1) is convex but not strongly convex, the oracle complexities of the IA-ICG method and the DA-ICG method for finding an ε -stationary point of (1.1) are $Q_g = O(\frac{L_g}{\varepsilon^2})$, $Q_h = O(\frac{\sqrt{L_g+L_h}\sqrt{L_g}}{\varepsilon^2})$ and $Q_g = O(\frac{L_g^{2/3}}{\varepsilon^{2/3}})$, $Q_h = O(\frac{\sqrt{L_g+L_h}L_g^{1/6}}{\varepsilon^{2/3}})$, respectively, the latter of which is the best result in the literature.

In contrast to [4, 31, 35, 71], our work focuses on the case when (1.1) is strongly convex. Our result is the best in the literature and complements the results by [35]. Moreover, our main studies are the applications of the proposed iAPG in (1.2) and (1.3), which are not studied in [4, 31, 35, 71]. Additionally, our method includes a line search scheme for the step length parameter, which those works do not consider.

2.2. Related methods for solving (1.2). The augmented Lagrangian method (ALM) [27, 60, 61] and its variants [5, 21, 22, 23, 24, 25, 28, 32, 33, 39, 49, 58, 62, 73, 74, 75, 76] can be applied to (1.2). The methods in [22, 33] require an exact solution of ALM subproblems, i.e., (1.4) with $\rho_k = 0$, which is not practical for many applications. Inexact (regularized) ALMs are studied by [39, 49, 58, 75], where

(regularized) ALM subproblems are solved inexactly by APG. When $\mu = 0$, these methods have oracle complexity $Q_f = Q_{\mathbf{A}} = O(\frac{1}{\varepsilon})$, and when $\mu > 0$, the method by [75] has oracle complexity $Q_f = Q_{\mathbf{A}} = O(\frac{1}{\sqrt{\varepsilon}})$. An accelerated linearized ALM is studied by [73], where f in (1.2) is linearized in the ALM subproblem. If the augmented term is also linearized so that the subproblem can be solved exactly, the method by [73] has the same oracle complexity as [75] in both cases when $\mu = 0$ and when $\mu > 0$. If the augmented term is not linearized, the methods by [5, 23, 25, 73] only need $O(\frac{1}{\sqrt{\varepsilon}})$ iterations even when $\mu = 0$, but the ALM subproblem becomes challenging to solve exactly. The linearized ALM method is analyzed in a unified framework together with other variants of the ALM by [62] and is generalized for nonlinear constraints by [74]. The same complexity as [75] is achieved in [62, 74]. A cutting-plane based ALM is proposed by [76] that can find an ε -stationary point for (1.2) with oracle complexity $Q_f = Q_{\mathbf{A}} = \tilde{O}(\frac{m}{\sqrt{\varepsilon}})$ when $\mu = 0$ and $Q_f = Q_{\mathbf{A}} = \tilde{O}(m \ln(\frac{1}{\varepsilon}))$ when $\mu > 0$, where m is the number of constraints. Hence, its complexity is better than ours only when $m = o(\varepsilon^{-\frac{1}{2}})$. A method similar to ALM is studied in [46] for decentralized distributed optimization with the consensus constraint, which is a special case of the affine constraints in (1.2).

The (linearized) Bregman methods [79, 80] and their accelerated variants [28, 33] are equivalent to gradient-based methods applied to the Lagrangian dual problem of (1.2). Similar techniques are explored in [14, 18]. However, these methods require easy evaluation of the proximal mapping of f , which limits their applications. For (1.2) with a strongly convex but not necessarily smooth objective, a dual ε -optimal solution can be found by an accelerated Uzawa method [64] or an inexact ALM method [32] within $O(\frac{1}{\sqrt{\varepsilon}})$ main iterations. However, the method in [64] requires solving a Lagrangian subproblem exactly and is thus impractical for general f . Although the method by [32] only needs to solve ALM subproblems inexactly, the authors only analyze the total number of main iterations but not the overall oracle complexity.

Penalty methods [14, 18, 38, 44] are also classical approaches for (1.2), where the affine constraints are moved to the objective function through a penalty term and the unconstrained penalty problem is then solved by another optimization algorithm like the APG. The primal method in [14, 18] requires $r = 0$ and \mathbf{A} is positive semidefinite while the dual method in [14, 18] requires an easy evaluation of the convex conjugate function of f , which limits the applications. When $\mu = 0$, [38] shows that if the penalty parameter is large enough, the penalty method finds an $(\varepsilon, \varepsilon)$ -primal-dual solution of (1.2) (see Definition 1 in [38]) with oracle complexity $Q_f = Q_{\mathbf{A}} = O(\frac{1}{\varepsilon})$. The penalty method by [44] solves a sequence of unconstrained penalty problems with increasing penalty parameters and only performs one APG iteration on each penalty problem. It has oracle complexity $Q_f = Q_{\mathbf{A}} = O(\frac{1}{\varepsilon})$ when $\mu = 0$ and $Q_f = Q_{\mathbf{A}} = O(\frac{1}{\sqrt{\varepsilon}})$ when $\mu > 0$. The complexity results in [38, 44] are higher than ours in both cases. The penalty method has also been applied to distributed optimization problems in [45] with consensus constraint, which is a special affine constraint.

By Lagrange multipliers, constrained optimization can be formulated as a min-max problem to which the primal-dual methods [67, 68, 69, 72, 83], mostly based on smoothing technique [54], can be applied. However, the methods by [67, 69, 72] require a closed-form solution of $\mathbf{prox}_{\eta f}$ while the method by [83] requires a closed-form solution of the convex conjugate function of f , and thus they have limited applications. The authors of [68] extend the algorithm and analysis in [67] by allowing $\mathbf{prox}_{\eta f}$ to be evaluated inexactly. However, they do not include the oracle complexity for inexactly evaluating the proximal mapping in their complexity analysis.

2.3. Related methods for solving (1.3). Smoothing techniques [1, 3, 54] are a class of effective approaches for solving the structured problem (1.3). They construct close approximation of (1.3) by one or a sequence of smooth problems, which are then solved by smooth optimization methods such as the APG. When $\mu = 0$, the methods by [1, 3, 54] find an ε -optimal solution with complexity $Q_f = Q_A = O\left(\frac{\|A\|}{\varepsilon} + \sqrt{\frac{L_f}{\varepsilon}}\right)$. When $\mu > 0$, the adaptive smoothing method by [1] finds an ε -optimal solution with $Q_f = Q_A = O\left(\sqrt{\frac{L_f}{\mu}} \ln\left(\frac{1}{\varepsilon}\right) + \frac{\|A\|}{\sqrt{\mu\varepsilon}}\right)$, which is higher than our complexity given in (1.11) when the query to $(f, \nabla f)$ is significantly more costly than $(A(\cdot), A^\top(\cdot))$.

In the literature, (1.3) has also been studied as a bilinear saddle-point problem [6, 9, 12, 26, 53, 84, 85]. The methods in [6, 9, 53] require a closed form of the proximal mapping of $f + r$ and thus may not be applicable to (1.3). When $\mu = 0$, the methods by [12, 26, 84, 85] find an ε -saddle-point (see Definition 3.1 in [26]) or an ε -optimal solution with the same oracle complexity as the smoothing methods mentioned above. When $\mu > 0$, the method of [85] finds an ε -optimal solution with the same oracle complexity as the smoothing method [1]. Problem (1.3) has also been studied as a variational inequality [13, 50, 70]. In particular, when $\mu = 0$, the mirror-prox methods in [50, 70] find an ε -optimal solution of (1.3) with complexity $Q_f = Q_A = O\left(\frac{L_f + \|A\|}{\varepsilon}\right)$, which is later reduced to $Q_f = Q_A = O\left(\sqrt{\frac{L_f}{\varepsilon}} + \frac{\|A\|}{\varepsilon}\right)$ by [13].

For all the methods we discussed above for solving (1.2) and (1.3), the oracle complexity is essentially the number of iterations the algorithms perform to find the desired solution. Since all of those methods always evaluate both $(f, \nabla f)$ and $(A(\cdot), A^\top(\cdot))$ in each iteration, Q_f and Q_A are the same for them. When the evaluation cost of $(f, \nabla f)$ is significantly higher than that of $(A(\cdot), A^\top(\cdot))$, it will be more efficient to query $(f, \nabla f)$ less frequently than $(A(\cdot), A^\top(\cdot))$ without compromising the solution quality. This actually can be achieved using the gradient sliding techniques [36, 37, 40, 42, 43, 56], which compute the gradient of one (more expensive) component of the objective function once in each outer iteration and process the remaining components in each inner iteration. The iAPG in this paper utilizes a similar double-loop technique to differentiate the frequencies of evaluating $(f, \nabla f)$ and $(A(\cdot), A^\top(\cdot))$ and thus reduce Q_f . Although the idea behind the iAPG is similar to the gradient sliding techniques, such a technique has not been studied for problem (1.2) under an iRALM framework. Although (1.3) has been studied by [36, 40], we consider the case of $\mu > 0$, which is not covered in [36] and for which [40] needs to apply the sliding method for convex cases in multiple stages. Moreover, except [42], which terminates the inner loop based on a computable duality gap,⁴ the existing gradient sliding techniques must run the inner loop for a predetermined number of iterations which depends on some unknown parameters of the problem. On the contrary, we terminate our inner loop based on a computable stationarity measure, which makes our method more efficient in practice, as we demonstrate in section 7.

3. Inexact accelerated proximal gradient method with line search. In this section, we consider (1.1) where g is μ -strongly convex with⁵ $\mu > 0$ and L_g -smooth (i.e., ∇g is L_g -Lipschitz continuous), h is convex and L_h -smooth, and r is closed convex and allows easy computation of $\text{prox}_{\eta r}(\mathbf{z})$ and $\text{dist}(\mathbf{z}', \partial r(\mathbf{z}))$ for any $\mathbf{z}', \mathbf{z} \in \mathbb{R}^n$ and $\eta > 0$. We assume that $(g, \nabla g)$ is significantly more costly to query than $(h, \nabla h)$ and L_g is significantly smaller than L_h . We propose an iAPG for (1.1)

⁴The method in [42] is a conditional gradient method that assumes a linear optimization oracle, which is different from our setting.

⁵Results for the case of $\mu = 0$ can be found in the longer arXiv version [48].

Algorithm 1 $\tilde{\mathbf{x}}^{(k+1)} = \text{iAPG}(g, h, r, \mathbf{x}^{(0)}, \eta_{-1}, \gamma_0, \mu, \underline{L}, (\varepsilon_k)_{k \geq 0}, \varepsilon)$ for (1.1).

Inputs: The three components of (1.1): g, h and r , $\mathbf{x}^{(0)} \in \text{dom}(r)$, $\eta_{-1} \leq \frac{1}{\underline{L}}$, $\gamma_0 \in [\mu, \frac{1}{\eta_{-1}}]$, $\mu > 0$, $\underline{L} \in [\mu, L_g]$, $\varepsilon_k \geq 0$, $\gamma_{\text{dec}} \in (0, 1)$, $\gamma_{\text{inc}} \in [1, +\infty)$, $\forall k \geq 0$, $\varepsilon > 0$ $\tilde{\eta}_0 \leftarrow \eta_{-1}$, $\mathbf{z}^{(0)} \leftarrow \mathbf{x}^{(0)}$ and set global parameters $\gamma_{\text{dec}} \in (0, 1)$ and $\gamma_{\text{inc}} \in [1, +\infty)$

for $k = 0, 1, \dots$, **do**

$(\mathbf{x}^{(k+1)}, \gamma_{k+1}, \eta_k, \alpha_k) = \text{LineSearch}(\mathbf{x}^{(k)}, \mathbf{z}^{(k)}, \gamma_k, \eta_{k-1}, \mu, \underline{L}, \varepsilon_k)$. $\mathbf{x}^{(k)} + \frac{1}{\alpha_k} (\mathbf{x}^{(k+1)} - \mathbf{x}^{(k)})$. if $\text{dist}(\mathbf{0}, \partial F(\tilde{\mathbf{x}}^{(k+1)})) \leq \varepsilon$ then	$\mathbf{z}^{(k+1)} = \text{SeekStationary}(\mathbf{x}^{(k+1)}, \tilde{\eta}_k)$.
--	--

Return: $\tilde{\mathbf{x}}^{(k+1)}$

Algorithm 2 $(\mathbf{x}^{(k+1)}, \gamma_{k+1}, \eta_k, \alpha_k) = \text{LineSearch}(\mathbf{x}^{(k)}, \mathbf{z}^{(k)}, \gamma_k, \eta_{k-1}, \mu, \underline{L}, \varepsilon_k)$.

Inputs: $\mathbf{x}^{(k)}, \mathbf{z}^{(k)}, \gamma_k > 0, \eta_{k-1} > 0, \underline{L} \in [\mu, L_g], \varepsilon_k > 0$ $\eta_k \leftarrow \min \left\{ \frac{1}{\gamma_{\text{dec}} \underline{L}}, \frac{\gamma_{\text{inc}} \eta_{k-1}}{\gamma_{\text{dec}}} \right\}$

repeat

$\eta_k \leftarrow \gamma_{\text{dec}} \eta_k$; find $\alpha_k > 0$ and γ_{k+1} that satisfy $\gamma_{k+1} = \frac{\alpha_k^2}{\eta_k} = (1 - \alpha_k) \gamma_k + \alpha_k \mu$. Let $\mathbf{y}^{(k)} = \frac{1}{\alpha_k \gamma_k + \gamma_{k+1}} (\alpha_k \gamma_k \mathbf{z}^{(k)} + \gamma_{k+1} \mathbf{x}^{(k)})$; find $\mathbf{x}^{(k+1)}$ such that (3.2) $\text{dist} \left(\mathbf{0}, \nabla g(\mathbf{y}^{(k)}) + \frac{1}{\eta_k} (\mathbf{x}^{(k+1)} - \mathbf{y}^{(k)}) + \partial H(\mathbf{x}^{(k+1)}) \right) \leq \varepsilon_k$
--

until $g(\mathbf{x}^{(k+1)}) \leq g(\mathbf{y}^{(k)}) + \langle \nabla g(\mathbf{y}^{(k)}), \mathbf{x}^{(k+1)} - \mathbf{y}^{(k)} \rangle + \frac{1}{2\eta_k} \|\mathbf{x}^{(k+1)} - \mathbf{y}^{(k)}\|^2$

Return: $(\mathbf{x}^{(k+1)}, \gamma_{k+1}, \eta_k, \alpha_k)$

in Algorithm 1, which is a modification of the APG in [52, Algorithm 2.2.19], by including a line search procedure (in Algorithm 2) for the step length parameter η_k and solving the following proximal mapping subproblem inexactly:

$$(3.1) \quad \mathbf{x}^{(k+1)} \approx \mathbf{x}_*^{(k+1)}$$

$$:= \arg \min_{\mathbf{x} \in \mathbb{R}^n} \left\{ \Phi(\mathbf{x}; \mathbf{y}^{(k)}, \eta_k) := \langle \nabla g(\mathbf{y}^{(k)}), \mathbf{x} - \mathbf{y}^{(k)} \rangle + \frac{1}{2\eta_k} \|\mathbf{x} - \mathbf{y}^{(k)}\|^2 + h(\mathbf{x}) + r(\mathbf{x}) \right\}.$$

The APG requires $\mathbf{x}^{(k+1)} = \mathbf{x}_*^{(k+1)}$, while our iAPG only needs $\mathbf{x}^{(k+1)}$ to be an ε_k -stationary point, i.e., a point satisfying (3.2). Our line search procedure follows [47].

It can be shown that $\mathbf{x}^{(k+1)}$ produced by the iAPG is an ε -optimal solution of (1.1) if k is large enough and ε_k decreases to zero in an appropriate rate. To generate an ε -stationary solution of (1.1), we just need to perform a proximal gradient step from $\mathbf{x}^{(k+1)}$ using a separate step length $\tilde{\eta}_k$ that can also be searched by the standard scheme as in [2]. We present this procedure in Algorithm 3, where $G := g + h$.

3.1. Convergence analysis for iAPG. In this subsection, we analyze the convergence rate of the proposed iAPG. The analysis also applies to APG by setting $\varepsilon_k = 0$. The technical lemmas below are needed.

LEMMA 3.1. *Let $\{(\eta_k, \tilde{\eta}_k, \alpha_k, \gamma_k)\}$ be generated from Algorithm 1. It holds that*

$$(3.3) \quad \frac{\gamma_{\text{dec}}}{L_g} < \eta_k \leq \frac{1}{\underline{L}}, \quad \frac{\gamma_{\text{dec}}}{L_g + L_h} < \tilde{\eta}_k \leq \frac{1}{\underline{L}}, \quad \alpha_k \leq 1 \text{ and } \gamma_k \geq \mu, \quad \text{for any } k \geq 0.$$

Proof. From lines 2 and 4 of Algorithm 2, we have $\eta_k \leq \frac{1}{\underline{L}}$ in Algorithm 1. In addition, the condition in line 6 of Algorithm 2 will hold and Algorithm 2 will stop

Algorithm 3 $(\tilde{\mathbf{x}}^{(k+1)}, \tilde{\eta}_{k+1}) = \text{SeekStationary}(\mathbf{x}^{(k+1)}, \tilde{\eta}_k).$

Inputs: $\mathbf{x}^{(k+1)}, \tilde{\eta}_k > 0$ $\tilde{\eta}_{k+1} \leftarrow \frac{\tilde{\eta}_k}{\gamma_{\text{dec}}}$
repeat
 $\tilde{\eta}_{k+1} \leftarrow \gamma_{\text{dec}} \tilde{\eta}_{k+1} \text{ and } \tilde{\mathbf{x}}^{(k+1)} \leftarrow \text{prox}_{\tilde{\eta}_{k+1} r}(\mathbf{x}^{(k)} - \tilde{\eta}_{k+1} \nabla G(\mathbf{x}^{(k)})).$
until $G(\tilde{\mathbf{x}}^{(k+1)}) \leq G(\mathbf{x}^{(k)}) + \langle \nabla G(\mathbf{x}^{(k)}), \tilde{\mathbf{x}}^{(k+1)} - \mathbf{x}^{(k)} \rangle + \frac{1}{2\tilde{\eta}_{k+1}} \|\tilde{\mathbf{x}}^{(k+1)} - \mathbf{x}^{(k)}\|^2$
Return: $\tilde{\mathbf{x}}^{(k+1)}$ and $\tilde{\eta}_{k+1}.$

if $\eta_k \leq \frac{1}{L_g}$. Given line 4 of Algorithm 2, we have $\eta_k > \frac{\gamma_{\text{dec}}}{L_g}$ in Algorithm 1. Since $\tilde{\eta}_0 \leq \frac{1}{L}, \frac{\gamma_{\text{dec}}}{L_g + L_h} < \tilde{\eta}_k \leq \frac{1}{L}$ hold similarly.

Solving α_k from the equation in line 4 of Algorithm 2 gives

$$(3.4) \quad \alpha_k = \frac{-(\gamma_k - \mu) + \sqrt{(\gamma_k - \mu)^2 + 4\gamma_k/\eta_k}}{2/\eta_k} = \frac{2\gamma_k}{(\gamma_k - \mu) + \sqrt{(\gamma_k - \mu)^2 + 4\gamma_k/\eta_k}}.$$

Since $\mu \leq L \leq 1/\eta_k$, we have $(\gamma_k - \mu)^2 + 4\gamma_k/\eta_k \geq (\gamma_k + \mu)^2$. Thus it follows from (3.4) that $\alpha_k \leq 1 \forall k \geq 0$. Notice if $\gamma_k \geq \mu$, then $\gamma_{k+1} = (1 - \alpha_k)\gamma_k + \alpha_k\mu \geq \mu$. Since $\gamma_0 \geq \mu$, we have $\gamma_k \geq \mu \forall k \geq 0$ by induction. \square

LEMMA 3.2. *In any iteration of Algorithms 1, 2, and 3 will respectively perform at most $\log_{\gamma_{\text{dec}}} \frac{L\gamma_{\text{dec}}^2}{L_g}$ and $\log_{\gamma_{\text{dec}}} \frac{L\gamma_{\text{dec}}^2}{L_g + L_h}$ iterations. Moreover, if Algorithm 1 runs for t iterations, Algorithms 2 and 3 will perform at most $t + (\frac{\ln \gamma_{\text{inc}}}{\ln \gamma_{\text{dec}}^{-1}})t + \frac{1}{\ln \gamma_{\text{dec}}^{-1}} \ln(\frac{L_g \eta_{-1}}{\gamma_{\text{dec}}})$ and $t + 1 + \frac{1}{\ln \gamma_{\text{dec}}^{-1}} \ln(\frac{\tilde{\eta}_0(L_g + L_h)}{\gamma_{\text{dec}}})$ iterations in total, respectively.*

Proof. Let n_k and m_k be the numbers of iterations performed within Algorithms 2 and 3, respectively, in iteration k of Algorithm 1. When Algorithm 2 ends, we have $\eta_k = \gamma_{\text{dec}}^{n_k-1} \min\{\underline{L}^{-1}, \gamma_{\text{inc}} \eta_{k-1}\}$. Since $\frac{\gamma_{\text{dec}}}{L_g} < \eta_k$ and $\eta_{k-1} \leq \frac{1}{\underline{L}}$ by (3.3), we have $\frac{\gamma_{\text{dec}}}{L_g} < \gamma_{\text{dec}}^{n_k-1} \cdot \frac{1}{\underline{L}}$, which implies $n_k \leq \log_{\gamma_{\text{dec}}} \frac{L\gamma_{\text{dec}}^2}{L_g}$. Similarly, $m_k \leq \log_{\gamma_{\text{dec}}} \frac{L\gamma_{\text{dec}}^2}{L_g + L_h}$.

The second conclusion can be proved in the same way as Lemma 6 in [55]. In particular, when Algorithm 2 ends, we must have $\eta_k = \gamma_{\text{dec}}^{n_k-1} \min\{\underline{L}^{-1}, \gamma_{\text{inc}} \eta_{k-1}\} \leq \gamma_{\text{dec}}^{n_k-1} \gamma_{\text{inc}} \eta_{k-1}$, which means $n_k \leq 1 + (\frac{\ln \gamma_{\text{inc}}}{\ln \gamma_{\text{dec}}^{-1}}) + \frac{1}{\ln \gamma_{\text{dec}}^{-1}} \ln(\frac{\eta_{k-1}}{\eta_k})$ and thus

$$\sum_{k=0}^{t-1} n_k \leq t + \left(\frac{\ln \gamma_{\text{inc}}}{\ln \gamma_{\text{dec}}^{-1}} \right) t + \frac{1}{\ln \gamma_{\text{dec}}^{-1}} \ln \left(\frac{\eta_{-1}}{\eta_{t-1}} \right) \leq t + \left(\frac{\ln \gamma_{\text{inc}}}{\ln \gamma_{\text{dec}}^{-1}} \right) t + \frac{1}{\ln \gamma_{\text{dec}}^{-1}} \ln \left(\frac{L_g \eta_{-1}}{\gamma_{\text{dec}}} \right).$$

A similar argument can be used to bound $\sum_{k=0}^{t-1} m_k$. \square

LEMMA 3.3. *Let $\kappa = \frac{L_g}{\gamma_{\text{dec}} \mu}$ and α_k generated by Algorithm 1. Then $\alpha_k \geq \sqrt{\frac{1}{\kappa}} \forall k \geq 0$.*

Proof. Lemma 3.1 indicates $\gamma_{k+1} \geq \mu$. Hence, from (3.3) and the update of γ_{k+1} , it follows that $\alpha_k = \sqrt{\eta_k \gamma_{k+1}} \geq \sqrt{\frac{1}{\kappa}}$, and we obtain the desired results. \square

Next, we establish the relationship between two iterates in Algorithm 1.

PROPOSITION 3.4. *Let $\{(\mathbf{x}^{(k)}, \mathbf{z}^{(k)}, \alpha_k, \gamma_k)\}_{k \geq 0}$ be generated by Algorithm 1. Then*

$$(3.5) \quad \begin{aligned} & F(\mathbf{x}^{(k+1)}) - F^* + \frac{\gamma_{k+1}}{2} \|\mathbf{x}^* - \mathbf{z}^{(k+1)}\|^2 \\ & \leq (1 - \alpha_k) \left[F(\mathbf{x}^{(k)}) - F^* + \frac{\gamma_k}{2} \|\mathbf{x}^* - \mathbf{z}^{(k)}\|^2 \right] + \varepsilon_k \alpha_k \|\mathbf{x}^* - \mathbf{z}^{(k+1)}\| \quad \forall k \geq 0. \end{aligned}$$

Proof. Let \mathbf{x}^* be an optimal solution of (1.1) and $\hat{\mathbf{x}}^{(k)} = \alpha_k \mathbf{x}^* + (1 - \alpha_k) \mathbf{x}^{(k)}$. Then

$$(3.6) \quad \hat{\mathbf{x}}^{(k)} - \mathbf{y}^{(k)} = \alpha_k (\mathbf{x}^* - \mathbf{y}^{(k)}) + (1 - \alpha_k) (\mathbf{x}^{(k)} - \mathbf{y}^{(k)}).$$

By the update of $\mathbf{y}^{(k)}$ in Algorithm 2, we have $\mathbf{z}^{(k)} - \mathbf{y}^{(k)} = -\frac{\gamma_{k+1}}{\alpha_k \gamma_k} (\mathbf{x}^{(k)} - \mathbf{y}^{(k)})$. This together with (3.6) gives

$$(3.7) \quad \begin{aligned} \hat{\mathbf{x}}^{(k)} - \mathbf{y}^{(k)} &= \alpha_k (\mathbf{x}^* - \mathbf{y}^{(k)}) - \frac{\alpha_k (1 - \alpha_k) \gamma_k}{\gamma_{k+1}} (\mathbf{z}^{(k)} - \mathbf{y}^{(k)}) \\ &= \alpha_k \left[\mathbf{x}^* - \frac{(1 - \alpha_k) \gamma_k}{\gamma_{k+1}} \mathbf{z}^{(k)} - \frac{\alpha_k \mu}{\gamma_{k+1}} \mathbf{y}^{(k)} \right], \end{aligned}$$

where the last equality follows from the update of γ_{k+1} . According to (3.2), there exists $\mathbf{e}^{(k)} \in \mathbb{R}^n$ such that $\|\mathbf{e}^{(k)}\| \leq \varepsilon_k$ and $\mathbf{e}^{(k)} - \nabla g(\mathbf{y}^{(k)}) - \frac{1}{\eta_k} (\mathbf{x}^{(k+1)} - \mathbf{y}^{(k)}) \in \partial H(\mathbf{x}^{(k+1)})$. By the convexity of H , we have

$$H(\mathbf{x}^{(k+1)}) \leq H(\hat{\mathbf{x}}^{(k)}) + \langle \mathbf{e}^{(k)} - \nabla g(\mathbf{y}^{(k)}) - \frac{1}{\eta_k} (\mathbf{x}^{(k+1)} - \mathbf{y}^{(k)}), \mathbf{x}^{(k+1)} - \hat{\mathbf{x}}^{(k)} \rangle,$$

which, by the fact that $\langle \mathbf{u}, \mathbf{v} \rangle = \frac{1}{2} (\|\mathbf{u}\|^2 + \|\mathbf{v}\|^2 - \|\mathbf{u} - \mathbf{v}\|^2)$, implies

$$\begin{aligned} H(\mathbf{x}^{(k+1)}) &\leq H(\hat{\mathbf{x}}^{(k)}) + \left\langle \mathbf{e}^{(k)} - \nabla g(\mathbf{y}^{(k)}), \mathbf{x}^{(k+1)} - \hat{\mathbf{x}}^{(k)} \right\rangle \\ &\quad - \frac{1}{2\eta_k} \left(\|\mathbf{x}^{(k+1)} - \mathbf{y}^{(k)}\|^2 + \|\mathbf{x}^{(k+1)} - \hat{\mathbf{x}}^{(k)}\|^2 - \|\hat{\mathbf{x}}^{(k)} - \mathbf{y}^{(k)}\|^2 \right), \\ &\leq H(\hat{\mathbf{x}}^{(k)}) + \left\langle \nabla g(\mathbf{y}^{(k)}), \hat{\mathbf{x}}^{(k)} - \mathbf{x}^{(k+1)} \right\rangle + \varepsilon_k \|\mathbf{x}^{(k+1)} - \hat{\mathbf{x}}^{(k)}\| \\ &\quad - \frac{1}{2\eta_k} \left(\|\mathbf{x}^{(k+1)} - \mathbf{y}^{(k)}\|^2 + \|\mathbf{x}^{(k+1)} - \hat{\mathbf{x}}^{(k)}\|^2 - \|\hat{\mathbf{x}}^{(k)} - \mathbf{y}^{(k)}\|^2 \right). \end{aligned}$$

From the inequality above and the stopping condition of Algorithm 2, we have

$$\begin{aligned} F(\mathbf{x}^{(k+1)}) &\leq g(\mathbf{y}^{(k)}) + \langle \nabla g(\mathbf{y}^{(k)}), \mathbf{x}^{(k+1)} - \mathbf{y}^{(k)} \rangle + \frac{1}{2\eta_k} \|\mathbf{x}^{(k+1)} - \mathbf{y}^{(k)}\|^2 + H(\mathbf{x}^{(k+1)}) \\ &\leq g(\mathbf{y}^{(k)}) + \langle \nabla g(\mathbf{y}^{(k)}), \hat{\mathbf{x}}^{(k)} - \mathbf{y}^{(k)} \rangle + \frac{1}{2\eta_k} \|\hat{\mathbf{x}}^{(k)} - \mathbf{y}^{(k)}\|^2 + H(\hat{\mathbf{x}}^{(k)}) \\ &\quad - \frac{1}{2\eta_k} \|\hat{\mathbf{x}}^{(k)} - \mathbf{x}^{(k+1)}\|^2 + \varepsilon_k \|\mathbf{x}^{(k+1)} - \hat{\mathbf{x}}^{(k)}\|. \end{aligned}$$

Applying (3.6) to the above inequality, we have

$$\begin{aligned} F(\mathbf{x}^{(k+1)}) &\leq g(\mathbf{y}^{(k)}) + \langle \nabla g(\mathbf{y}^{(k)}), \alpha_k (\mathbf{x}^* - \mathbf{y}^{(k)}) + (1 - \alpha_k) (\mathbf{x}^{(k)} - \mathbf{y}^{(k)}) \rangle \\ &\quad + \frac{1}{2\eta_k} \|\hat{\mathbf{x}}^{(k)} - \mathbf{y}^{(k)}\|^2 \\ &\quad + H(\alpha_k \mathbf{x}^* + (1 - \alpha_k) \mathbf{x}^{(k)}) - \frac{1}{2\eta_k} \|\hat{\mathbf{x}}^{(k)} - \mathbf{x}^{(k+1)}\|^2 + \varepsilon_k \|\mathbf{x}^{(k+1)} - \hat{\mathbf{x}}^{(k)}\|. \end{aligned}$$

By the fact that $\alpha_k \in (0, 1]$ from Lemma 3.1, (3.7), and the convexity of H , we have

$$(3.8) \quad \begin{aligned} F(\mathbf{x}^{(k+1)}) &\leq (1 - \alpha_k) [g(\mathbf{y}^{(k)}) + \langle \nabla g(\mathbf{y}^{(k)}), \mathbf{x}^{(k)} - \mathbf{y}^{(k)} \rangle + H(\mathbf{x}^{(k)})] \\ &\quad + \alpha_k [g(\mathbf{y}^{(k)}) + \langle \nabla g(\mathbf{y}^{(k)}), \mathbf{x}^* - \mathbf{y}^{(k)} \rangle + H(\mathbf{x}^*)] + \varepsilon_k \|\mathbf{x}^{(k+1)} - \hat{\mathbf{x}}^{(k)}\| \\ &\quad + \frac{\alpha_k^2}{2\eta_k} \left\| \mathbf{x}^* - \frac{(1 - \alpha_k) \gamma_k}{\gamma_{k+1}} \mathbf{z}^{(k)} - \frac{\alpha_k \mu}{\gamma_{k+1}} \mathbf{y}^{(k)} \right\|^2 - \frac{1}{2\eta_k} \|\hat{\mathbf{x}}^{(k)} - \mathbf{x}^{(k+1)}\|^2. \end{aligned}$$

Since $\gamma_{k+1} = \alpha_k^2 / \eta_k = (1 - \alpha_k)\gamma_k + \alpha_k\mu$, we have from the convexity of $\|\cdot\|^2$ that

$$\begin{aligned} \frac{\alpha_k^2}{2\eta_k} \left\| \mathbf{x}^* - \frac{(1 - \alpha_k)\gamma_k}{\gamma_{k+1}} \mathbf{z}^{(k)} - \frac{\alpha_k\mu}{\gamma_{k+1}} \mathbf{y}^{(k)} \right\|^2 &= \frac{\gamma_{k+1}}{2} \left\| \mathbf{x}^* - \frac{(1 - \alpha_k)\gamma_k}{\gamma_{k+1}} \mathbf{z}^{(k)} - \frac{\alpha_k\mu}{\gamma_{k+1}} \mathbf{y}^{(k)} \right\|^2 \\ &\leq \frac{(1 - \alpha_k)\gamma_k}{2} \|\mathbf{x}^* - \mathbf{z}^{(k)}\|^2 + \frac{\alpha_k\mu}{2} \|\mathbf{x}^* - \mathbf{y}^{(k)}\|^2, \end{aligned}$$

which, together with (3.8) and the μ -strong convexity of g , implies

$$\begin{aligned} F(\mathbf{x}^{(k+1)}) &\leq (1 - \alpha_k) \left[g(\mathbf{y}^{(k)}) + \langle \nabla g(\mathbf{y}^{(k)}), \mathbf{x}^{(k)} - \mathbf{y}^{(k)} \rangle + H(\mathbf{x}^{(k)}) + \frac{\gamma_k}{2} \|\mathbf{x}^* - \mathbf{z}^{(k)}\|^2 \right] \\ &\quad + \alpha_k \left[g(\mathbf{y}^{(k)}) + \langle \nabla g(\mathbf{y}^{(k)}), \mathbf{x}^* - \mathbf{y}^{(k)} \rangle + H(\mathbf{x}^*) + \frac{\mu}{2} \|\mathbf{x}^* - \mathbf{y}^{(k)}\|^2 \right] \\ &\quad - \frac{1}{2\eta_k} \|\hat{\mathbf{x}}^{(k)} - \mathbf{x}^{(k+1)}\|^2 + \varepsilon_k \|\mathbf{x}^{(k+1)} - \hat{\mathbf{x}}^{(k)}\| \\ (3.9) \quad &\leq (1 - \alpha_k) \left[F(\mathbf{x}^{(k)}) + \frac{\gamma_k}{2} \|\mathbf{x}^* - \mathbf{z}^{(k)}\|^2 \right] + \alpha_k F(\mathbf{x}^*) - \frac{1}{2\eta_k} \|\hat{\mathbf{x}}^{(k)} - \mathbf{x}^{(k+1)}\|^2 \\ &\quad + \varepsilon_k \|\mathbf{x}^{(k+1)} - \hat{\mathbf{x}}^{(k)}\|. \end{aligned}$$

By the definitions of $\mathbf{z}^{(k+1)}$ and $\hat{\mathbf{x}}^{(k)}$, it holds that

$$(3.10) \quad \|\hat{\mathbf{x}}^{(k)} - \mathbf{x}^{(k+1)}\|^2 = \|\alpha_k \mathbf{x}^* + (1 - \alpha_k) \mathbf{x}^{(k)} - \mathbf{x}^{(k+1)}\|^2 = \alpha_k^2 \|\mathbf{x}^* - \mathbf{z}^{(k+1)}\|^2.$$

Apply (3.10) to (3.9) and use $\gamma_{k+1} = \alpha_k^2 / \eta_k$ to obtain the desired inequality. \square

We apply (3.5) to derive the convergence rate of Algorithm 1.

THEOREM 3.5. *For any $c \in [0, 1]$, Algorithm 1 guarantees that*

$$(3.11) \quad \psi_{k+1} \leq \prod_{j=0}^k (1 - c\alpha_j) \left(\psi_0 + \frac{\sqrt{\kappa}}{2(1-c)^2 L} \sum_{t=0}^k \frac{\varepsilon_t^2}{\prod_{j=0}^{t-1} (1 - c\alpha_j)} \right) \quad \text{for } k \geq 0,$$

where $\psi_k := F(\mathbf{x}^{(k)}) - F^* + (1 - (1 - c)\alpha_k) \frac{\gamma_k}{2} \|\mathbf{x}^* - \mathbf{z}^{(k)}\|^2$ and κ is defined in Lemma 3.3. In addition, when $\varepsilon_k = 0 \ \forall k$, Algorithm 1 guarantees that, for $k \geq 0$,

$$\begin{aligned} (3.12) \quad &F(\mathbf{x}^{(k+1)}) - F^* + \frac{\gamma_{k+1}}{2} \|\mathbf{x}^* - \mathbf{z}^{(k+1)}\|^2 \\ &\leq \left(1 - \frac{1}{\sqrt{\kappa}} \right)^{k+1} \left(F(\mathbf{x}^{(0)}) - F^* + \frac{\gamma_0}{2} \|\mathbf{x}^* - \mathbf{z}^{(0)}\|^2 \right). \end{aligned}$$

Proof. By the Young's inequality, we have that for any $c \in [0, 1]$,

$$\varepsilon_k \alpha_k \|\mathbf{x}^* - \mathbf{z}^{(k+1)}\| \leq \frac{(1 - c)\alpha_{k+1}\alpha_k^2}{2\eta_k} \|\mathbf{x}^* - \mathbf{z}^{(k+1)}\|^2 + \frac{\eta_k}{2(1 - c)\alpha_{k+1}} \varepsilon_k^2.$$

Recall $\gamma_{k+1} = \frac{\alpha_k^2}{\eta_k}$. Hence, we have from (3.5) that

$$\begin{aligned} &F(\mathbf{x}^{(k+1)}) - F^* + \frac{\gamma_{k+1}}{2} \|\mathbf{x}^* - \mathbf{z}^{(k+1)}\|^2 \\ &\leq (1 - \alpha_k) \left[F(\mathbf{x}^{(k)}) - F^* + \frac{\gamma_k}{2} \|\mathbf{x}^* - \mathbf{z}^{(k)}\|^2 \right] \\ &\quad + \frac{(1 - c)\alpha_{k+1}\gamma_{k+1}}{2} \|\mathbf{x}^* - \mathbf{z}^{(k+1)}\|^2 + \frac{\eta_k}{2(1 - c)\alpha_{k+1}} \varepsilon_k^2, \end{aligned}$$

which, after rearranging terms, is reduced to

$$(3.13) \quad \begin{aligned} & F(\mathbf{x}^{(k+1)}) - F^* + (1 - (1 - c)\alpha_{k+1}) \frac{\gamma_{k+1}}{2} \|\mathbf{x}^* - \mathbf{z}^{(k+1)}\|^2 \\ & \leq (1 - \alpha_k) \left[F(\mathbf{x}^{(k)}) - F^* + \frac{\gamma_k}{2} \|\mathbf{x}^* - \mathbf{z}^{(k)}\|^2 \right] + \frac{\eta_k}{2(1 - c)\alpha_{k+1}} \varepsilon_k^2. \end{aligned}$$

Then it follows from (3.13), the definition of ψ_k , and $F(\mathbf{x}^{(k)}) - F^* \geq 0$ that

$$(3.14) \quad \psi_{k+1} \leq \frac{1 - \alpha_k}{1 - (1 - c)\alpha_k} \psi_k + \frac{\eta_k}{2(1 - c)\alpha_{k+1}} \varepsilon_k^2 \leq (1 - c\alpha_k) \psi_k + \frac{\sqrt{\kappa}}{2(1 - c)\underline{L}} \varepsilon_k^2,$$

where the first inequality is because $\frac{1 - \alpha_k}{1 - (1 - c)\alpha_k} = 1 - \frac{c\alpha_k}{1 - (1 - c)\alpha_k} \leq 1 - c\alpha_k$ and the second inequality is by (3.3) and Lemma 3.3. Recursively applying (3.14) gives

$$\begin{aligned} \psi_{k+1} & \leq \prod_{j=0}^k (1 - c\alpha_j) \psi_0 + \frac{\sqrt{\kappa}}{2(1 - c)\underline{L}} \sum_{t=0}^k \left(\prod_{j=t+1}^k (1 - c\alpha_j) \right) \varepsilon_t^2 \\ & = \prod_{j=0}^k (1 - c\alpha_j) \left(\psi_0 + \frac{\sqrt{\kappa}}{2(1 - c)\underline{L}} \sum_{t=0}^k \frac{\varepsilon_t^2}{\prod_{j=0}^t (1 - c\alpha_j)} \right), \end{aligned}$$

which implies (3.11) because $\alpha_j \leq 1 \forall j \geq 0$.

When $\varepsilon_k = 0$, (3.12) can be derived by Lemma 3.3 and recursively using (3.5). \square

The result in (3.11) is similar to Propositions 2 and 4 in [63] but takes a different form. It will be later used to derive the oracle complexity of our iAPG. The result in (3.12) is exactly the convergence property of the APG [52] for a strongly convex case. Although (3.12) is not new, we still present it here because we need it later to analyze the complexity to obtain $\mathbf{x}^{(k+1)}$ in line 5 of Algorithm 2.

3.2. Complexity of APG for finding an ε -stationary point of (1.1). The oracle complexity of Algorithm 1 must include the complexity for finding $\mathbf{x}^{(k+1)}$ satisfying (3.2) in each iteration of Algorithm 2. Such an $\mathbf{x}^{(k+1)}$ can be found by approximately solving (3.1), which is an instance of (1.1) with the g , h , and r components being $\Phi(\cdot; \mathbf{y}^{(k)}, \eta_k) - r(\cdot)$, 0, and $r(\cdot)$, respectively. The assumption on r allows us to apply the exact APG method, i.e., Algorithm 1 with $\varepsilon_k = 0 \forall k \geq 0$ to (3.1) in order to find $\mathbf{x}^{(k+1)}$. The convergence of the objective gap by the exact APG method is characterized by (3.12). However, (3.2) requires $\mathbf{x}^{(k+1)}$ to be an ε_k -stationary solution of (1.1) instead of an ε_k -optimal solution. Hence, we first establish the complexity for the exact APG method to find an ε -stationary solution of (1.1). The analysis is standard in the literature and is included for the sake of completeness.

LEMMA 3.6. *Let $C_L = \frac{L_g + L_h}{\sqrt{\underline{L}}} + \sqrt{\frac{L_g + L_h}{\gamma_{\text{dec}}}}$, where \underline{L} and γ_{dec} are those in Algorithms 1 and 2. It holds that, for any $k \geq 0$,*

$$(3.15) \quad \text{dist}(\mathbf{0}, \partial F(\tilde{\mathbf{x}}^{(k+1)})) \leq C_L \sqrt{2(F(\mathbf{x}^{(k+1)}) - F^*)}.$$

Proof. When the stopping condition of Algorithm 3 holds, we have (cf. [78, Lemma 2.1]) $F(\mathbf{x}) - F(\tilde{\mathbf{x}}) \geq \frac{1}{2\bar{\eta}} \|\mathbf{x} - \tilde{\mathbf{x}}\|^2$, and thus

$$(3.16) \quad \|\tilde{\mathbf{x}}^{(k+1)} - \mathbf{x}^{(k+1)}\| \leq \sqrt{2\tilde{\eta}_{k+1}(F(\mathbf{x}^{(k+1)}) - F(\tilde{\mathbf{x}}^{(k+1)}))}.$$

Also, from the update of $\tilde{\mathbf{x}}$, we have $\mathbf{0} \in \nabla(g+h)(\mathbf{x}) + \frac{1}{\eta}(\tilde{\mathbf{x}} - \mathbf{x}) + \partial r(\tilde{\mathbf{x}})$, and thus $\text{dist}(\mathbf{0}, \partial F(\tilde{\mathbf{x}})) \leq \|\nabla(g+h)(\tilde{\mathbf{x}}) - \nabla(g+h)(\mathbf{x}) + \frac{1}{\eta}(\tilde{\mathbf{x}} - \mathbf{x})\| \leq (L_g + L_h + \frac{1}{\eta})\|\tilde{\mathbf{x}} - \mathbf{x}\|$. Hence, for $\tilde{\mathbf{x}}^{(k+1)}$ in Algorithm 1, it holds that

$$(3.17) \quad \begin{aligned} \text{dist}(\mathbf{0}, \partial F(\tilde{\mathbf{x}}^{(k+1)})) &\leq \left(L_g + L_h + \frac{1}{\tilde{\eta}_{k+1}} \right) \|\tilde{\mathbf{x}}^{(k+1)} - \mathbf{x}^{(k+1)}\| \\ &\stackrel{3.16}{\leq} \left(L_g + L_h + \frac{1}{\tilde{\eta}_{k+1}} \right) \sqrt{2\tilde{\eta}_{k+1}(F(\mathbf{x}^{(k+1)}) - F(\tilde{\mathbf{x}}^{(k+1)}))}. \end{aligned}$$

Applying (3.3) and the fact that $F(\tilde{\mathbf{x}}^{(k+1)}) \geq F^*$, we obtain the desired result. \square

By (3.12) and (3.15), we immediately have the following result.

THEOREM 3.7. *Let κ and C_L be defined in Lemmas 3.3 and 3.6. When $\varepsilon_k = 0$ for $k \geq 0$, Algorithm 1 returns $\tilde{\mathbf{x}}^{(k+1)}$ as an ε -stationary point of (1.1) with*

$$(3.18) \quad k+1 \leq 2\sqrt{\kappa} \ln \left(C_L \sqrt{2 \left(F(\mathbf{x}^{(0)}) - F^* + \frac{\gamma_0}{2} \|\mathbf{x}^* - \mathbf{z}^{(0)}\|^2 \right)} \frac{1}{\varepsilon} \right).$$

4. Oracle complexity of iAPG. In this section, we show the oracle complexity of Algorithm 1 for finding an ε -stationary solution of (1.1) in the strongly convex case. The complexity in the convex but not strongly convex case is not included due to space limits. For that result, we refer interested readers to [48].

4.1. Complexity for ensuring (3.2). We can find $\mathbf{x}^{(k+1)}$ satisfying (3.2) by calling the iAPG method (Algorithm 1) with the following inputs:

$$(4.1) \quad \mathbf{x}^{(k+1)} = \text{iAPG} \left(\Phi(\cdot; \mathbf{y}^{(k)}, \eta_k) - r(\cdot), 0, r(\cdot), \mathbf{x}^{(k)}, \eta_k, \eta_k^{-1}, \eta_k^{-1}, \eta_k^{-1}, (0)_{k \geq 0}, \varepsilon_k \right),$$

where Φ is defined in (3.1). Here we use $\mathbf{x}^{(k)}$ as the initial solution to compute $\mathbf{x}^{(k+1)}$ and the inputs in (4.1) are chosen based on the fact that $\Phi(\cdot; \mathbf{y}^{(k)}, \eta_k) - r(\cdot)$ is $1/\eta_k$ -strongly convex and $(1/\eta_k + L_h)$ -smooth. The complexity of finding $\mathbf{x}^{(k+1)}$ then follows from Theorem 3.7.

PROPOSITION 4.1 (complexity for ensuring (3.2)). *Let $\mathbf{x}_*^{(k+1)}$ and Φ be defined in (3.1). Suppose Algorithm 1 is applied to (3.1) with the inputs given in (4.1). Solution $\mathbf{x}^{(k+1)}$ satisfying (3.2) can be found after at most T_k queries to $(h, \nabla h)$, where*

$$(4.2) \quad T_k = O \left(\sqrt{1 + \frac{L_h}{L}} \ln \frac{\sqrt{L_g + L_h + L_h^2/L} \sqrt{\Phi(\mathbf{x}^{(k)}; \mathbf{y}^{(k)}, \eta_k) - \Phi(\mathbf{x}_*^{(k+1)}; \mathbf{y}^{(k)}, \eta_k)}}{\varepsilon_k} \right).$$

Proof. Recall that $\Phi(\cdot; \mathbf{y}^{(k)}, \eta_k) - r(\cdot)$ is $(\frac{1}{\eta_k} + L_h)$ -smooth and $\frac{1}{\eta_k}$ -strongly convex. From the strong convexity of Φ , it holds that

$$(4.3) \quad \frac{1}{2\eta_k} \|\mathbf{x}^{(k)} - \mathbf{x}_*^{(k+1)}\|^2 \leq \Phi \left(\mathbf{x}^{(k)}; \mathbf{y}^{(k)}, \eta_k \right) - \Phi \left(\mathbf{x}_*^{(k+1)}; \mathbf{y}^{(k)}, \eta_k \right).$$

By instantizing Theorem 3.7 on (3.1), Algorithm 1 with the inputs given in (4.1) must find $\mathbf{x}^{(k+1)}$ satisfying (3.2) in no more than t_k iterations with

(4.4)

$$\begin{aligned} t_k &\leq 2\sqrt{\frac{1+\eta_k L_h}{\gamma_{\text{dec}}}} \ln \frac{\left(\frac{1/\eta_k+L_h}{\sqrt{1/\eta_k}}+\sqrt{\frac{1/\eta_k+L_h}{\gamma_{\text{dec}}}}\right)\sqrt{2\Phi(\mathbf{x}^{(k)};\mathbf{y}^{(k)},\eta_k)-2\Phi(\mathbf{x}_*^{(k+1)};\mathbf{y}^{(k)},\eta_k)+\frac{1}{\eta_k}\|\mathbf{x}^{(k)}-\mathbf{x}_*^{(k+1)}\|^2}}{\varepsilon_k} \\ &= O\left(\sqrt{1+\frac{L_h}{L}} \ln \frac{\left(\sqrt{L_g}+\frac{L_h}{\sqrt{L}}+\sqrt{L_g+L_h}\right)\sqrt{\Phi(\mathbf{x}^{(k)};\mathbf{y}^{(k)},\eta_k)-\Phi(\mathbf{x}_*^{(k+1)};\mathbf{y}^{(k)},\eta_k)}}{\varepsilon_k}\right), \end{aligned}$$

where the second equation is because of (3.3) and (4.3) and uses the fact $\ln(1-a)^{-1} \geq a$ for any $a \in (0, 1)$. By instantizing Lemma 3.2 on (3.1) with the input given in (4.1), the total number of queries of $(h, \nabla h)$ must be no more than

$$T_k = 2\left(1 + \frac{\ln \gamma_{\text{inc}}}{\ln \gamma_{\text{dec}}^{-1}}\right)t_k + \frac{2}{\ln \gamma_{\text{dec}}^{-1}} \ln\left(\frac{1+\eta_k L_h}{\gamma_{\text{dec}}}\right) + 2t_k + 2 + \frac{2}{\ln \gamma_{\text{dec}}^{-1}} \ln\left(\frac{1+\eta_k L_h}{\gamma_{\text{dec}}}\right),$$

which, together with (4.4) and (3.3), implies the conclusion. \square

4.2. Oracle complexity in the strongly convex case. With Theorem 3.5 and Proposition 4.1, we establish the oracle complexity to produce an ε -stationary solution of (1.1) by specifying $\{\varepsilon_k\}_{k \geq 0}$ and bounding $\Phi(\mathbf{x}^{(k)};\mathbf{y}^{(k)},\eta_k)-\Phi(\mathbf{x}_*^{(k+1)};\mathbf{y}^{(k)},\eta_k)$. To do so, let $\varepsilon_0 > 0$ be any constant and define the following quantities:

$$(4.5) \quad \varepsilon_k = \frac{\varepsilon_0}{k+1} \sqrt{\prod_{j=0}^{k-1} (1-c\alpha_j)} \quad \forall k \geq 1,$$

$$(4.6) \quad S = \frac{\sqrt{\kappa}}{2(1-c)^2 L} \sum_{k=0}^{\infty} \frac{\varepsilon_k^2}{\prod_{j=0}^{k-1} (1-c\alpha_j)} = \frac{\sqrt{\kappa}}{2(1-c)^2 L} \sum_{k=0}^{\infty} \frac{\varepsilon_0^2}{(k+1)^2} < \infty,$$

$$(4.7) \quad \delta_k = \sqrt{\prod_{j=0}^{k-1} (1-c\alpha_j)} \sqrt{\frac{2(\psi_0 + S)}{\mu}} \quad \forall k \geq 0,$$

where $c \in [0, 1)$ is the same constant as that in Theorem 3.5 and κ is defined in Lemma 3.3. By (3.11), (4.5), and (4.6), we have

$$(4.8) \quad \psi_{k+1} \leq \prod_{j=0}^k (1-c\alpha_j) (\psi_0 + S) \quad \forall k \geq 0.$$

With these preparations, $\Phi(\mathbf{x}^{(k)};\mathbf{y}^{(k)},\eta_k)-\Phi(\mathbf{x}_*^{(k+1)};\mathbf{y}^{(k)},\eta_k)$ can be upper bounded.

LEMMA 4.2. *Suppose $\{\varepsilon_k\}_{k \geq 0}$ in Algorithm 1 are given in (4.5). Let $\mathbf{x}_*^{(k+1)}$ and Φ be defined by (3.1) and δ_k by (4.7) with $c \in [0, 1)$. Algorithm 1 guarantees that*

(4.9)

$$\Phi(\mathbf{x}^{(k)}; \mathbf{y}^{(k)}, \eta_k) - \Phi(\mathbf{x}_*^{(k+1)}; \mathbf{y}^{(k)}, \eta_k) \leq \begin{cases} \frac{1}{2\bar{L}} \text{dist}(\mathbf{0}, \partial F(\mathbf{x}^{(0)}))^2 & \text{if } k = 0, \\ \frac{1}{2\bar{L}} \left(\varepsilon_{k-1} + \frac{3L_g(\delta_k + \delta_{k-1})}{\gamma_{\text{dec}}\sqrt{c}} \right)^2 & \text{if } k \geq 1. \end{cases}$$

Proof. By $\mathbf{y}^{(0)} = \mathbf{x}^{(0)}$, we have $\Phi(\mathbf{x}^{(0)}; \mathbf{y}^{(0)}, \eta_0) = H(\mathbf{x}^{(0)})$. Also, it holds that

$$\Phi(\mathbf{x}; \mathbf{y}^{(0)}, \eta_0) \geq \left\langle \nabla g(\mathbf{x}^{(0)}), \mathbf{x} - \mathbf{x}^{(0)} \right\rangle + \frac{1}{2\eta_0} \|\mathbf{x} - \mathbf{x}^{(0)}\|^2 + H(\mathbf{x}^{(0)}) + \langle \boldsymbol{\xi}, \mathbf{x} - \mathbf{x}^{(0)} \rangle$$

for any $\boldsymbol{\xi} \in \partial H(\mathbf{x}^{(0)})$ and any \mathbf{x} , from the convexity of H . Since $\eta_0 \leq \frac{1}{\bar{L}}$, we have

$$\begin{aligned} (4.10) \quad & \Phi(\mathbf{x}^{(0)}; \mathbf{y}^{(0)}, \eta_0) - \Phi(\mathbf{x}_*^{(1)}; \mathbf{y}^{(0)}, \eta_0) \\ & \leq -\min_{\mathbf{x}} \left\{ \left\langle \nabla g(\mathbf{x}^{(0)}) + \boldsymbol{\xi}, \mathbf{x} - \mathbf{x}^{(0)} \right\rangle + \frac{1}{2\eta_0} \|\mathbf{x} - \mathbf{x}^{(0)}\|^2 \right\} \\ & = \frac{\eta_0}{2} \|\nabla g(\mathbf{x}^{(0)}) + \boldsymbol{\xi}\|^2 \leq \frac{1}{2\bar{L}} \|\nabla g(\mathbf{x}^{(0)}) + \boldsymbol{\xi}\|^2. \end{aligned}$$

Minimizing the right-hand side of (4.10) over $\boldsymbol{\xi} \in \partial H(\mathbf{x}^{(0)})$ gives (4.9) for $k = 0$.

Suppose $k \geq 1$. By the definition of ψ_k in Theorem 3.5 and the μ -strong convexity of F , we have

$$\psi_k \geq \frac{\mu}{2} \|\mathbf{x}^{(k)} - \mathbf{x}^*\|^2 + (1 - (1 - c)\alpha_k) \frac{\gamma_k}{2} \|\mathbf{z}^{(k)} - \mathbf{x}^*\|^2 \geq \frac{\mu}{2} \|\mathbf{x}^{(k)} - \mathbf{x}^*\|^2 + \frac{c\mu}{2} \|\mathbf{z}^{(k)} - \mathbf{x}^*\|^2,$$

where the second inequality is due to (3.3). This inequality implies, for any $k \geq 0$,

$$(4.11) \quad \max \left\{ \|\mathbf{x}^{(k)} - \mathbf{x}^*\|, \sqrt{c} \|\mathbf{z}^{(k)} - \mathbf{x}^*\| \right\} \leq \sqrt{\frac{2\psi_k}{\mu}} \leq \sqrt{\prod_{j=0}^{k-1} (1 - c\alpha_j)} \sqrt{\frac{2(\psi_0 + S)}{\mu}} = \delta_k,$$

where the second inequality is by (4.8) and the equality is by (4.7). Since $c \in (0, 1)$ and $\mathbf{y}^{(k)}$ is a convex combination of $\mathbf{x}^{(k)}$ and $\mathbf{z}^{(k)}$, it follows from (4.11) that

$$(4.12) \quad \|\mathbf{y}^{(k)} - \mathbf{x}^*\| \leq \frac{\delta_k}{\sqrt{c}} \quad \forall k \geq 0.$$

By (3.2), it holds that $\text{dist}(\mathbf{0}, \nabla g(\mathbf{y}^{(k-1)}) + \frac{1}{\eta_{k-1}}(\mathbf{x}^{(k)} - \mathbf{y}^{(k-1)}) + \partial H(\mathbf{x}^{(k)})) \leq \varepsilon_{k-1}$ for $k \geq 1$. Hence, by the definition of Φ in (3.1), we have

(4.13)

$$\begin{aligned} & \text{dist}(\mathbf{0}, \partial\Phi(\mathbf{x}^{(k)}; \mathbf{y}^{(k)}, \eta_k)) \\ & \leq \varepsilon_{k-1} + \|\nabla g(\mathbf{y}^{(k)}) - \nabla g(\mathbf{y}^{(k-1)})\| + \frac{1}{\eta_{k-1}} \|\mathbf{x}^{(k)} - \mathbf{y}^{(k-1)}\| + \frac{1}{\eta_k} \|\mathbf{x}^{(k)} - \mathbf{y}^{(k)}\| \\ & \leq \varepsilon_{k-1} + L_g \|\mathbf{y}^{(k)} - \mathbf{y}^{(k-1)}\| + \frac{L_g}{\gamma_{\text{dec}}} \|\mathbf{x}^{(k)} - \mathbf{y}^{(k-1)}\| + \frac{L_g}{\gamma_{\text{dec}}} \|\mathbf{x}^{(k)} - \mathbf{y}^{(k)}\| \\ & \leq \varepsilon_{k-1} + \frac{L_g(\delta_k + \delta_{k-1})}{\sqrt{c}} + \frac{L_g}{\gamma_{\text{dec}}} \left(\delta_k + \frac{\delta_{k-1}}{\sqrt{c}} \right) + \frac{L_g}{\gamma_{\text{dec}}} \left(\delta_k + \frac{\delta_k}{\sqrt{c}} \right), \end{aligned}$$

where the second inequality is by (3.3) and the third one by (4.11), (4.12), and the triangle inequality. In addition, by the strong convexity of $\Phi(\cdot; \mathbf{y}, \eta)$, it follows that

$$(4.14) \quad \Phi(\mathbf{x}^{(k)}; \mathbf{y}^{(k)}, \eta_k) - \Phi(\mathbf{x}_*^{(k+1)}; \mathbf{y}^{(k)}, \eta_k) \leq \frac{\eta_k}{2} \text{dist}(\mathbf{0}, \partial\Phi(\mathbf{x}^{(k)}; \mathbf{y}^{(k)}, \eta_k))^2 \quad \forall k \geq 1,$$

which, together with (4.13) and the facts that $c < 1$, $\gamma_{\text{dec}} \leq 1$, $\eta_k \leq \frac{1}{L}$, and $\delta_k \leq \delta_{k-1}$, gives the result in (4.9) for $k \geq 1$. This completes the proof. \square

Lemma 4.2 allows us to simplify (4.2) and obtain the following result.

THEOREM 4.3 (oracle complexity to obtain an ε -stationary solution). *Suppose $\{\varepsilon_k\}_{k \geq 0}$ in Algorithm 1 are given in (4.5). Also, suppose $\mathbf{x}^{(k+1)}$ is computed by applying Algorithm 1 to (3.1) with the inputs given in (4.1). Then for any $\varepsilon > 0$, Algorithm 1 with⁶ $\underline{L} = \Theta(L_g)$ can produce an ε -stationary solution of (1.1) by $K_{\text{crit}}^{\text{sc}}$ queries to $(g, \nabla g)$ and $T_{\text{crit}}^{\text{sc}}$ queries to $(h, \nabla h)$, where*

$$(4.15) \quad K_{\text{crit}}^{\text{sc}} = O\left(\sqrt{\kappa} \ln \frac{C_L^2(\psi_0 + S)}{\varepsilon^2} + \ln\left(\frac{L_g + L_h}{\mu}\right)\right),$$

$$(4.16) \quad T_{\text{crit}}^{\text{sc}} = O\left(\sqrt{\frac{L_g + L_h}{\mu}} \ln\left((1 + \frac{L_h}{L_g})C_\varepsilon\right) \cdot \ln \frac{C_L^2(\psi_0 + S)}{\varepsilon^2}\right).$$

Here, κ is defined in Lemma 3.3, ψ_0 in Theorem 3.5, C_L in Lemma 3.6, and

$$(4.17) \quad C_\varepsilon = \max \left\{ \frac{\text{dist}(\mathbf{0}, \partial F(\mathbf{x}^{(0)}))}{\varepsilon_0}, \frac{2}{\sqrt{1-c}} \right. \\ \left. + \frac{6L_g}{\varepsilon_0 \gamma_{\text{dec}} \sqrt{c(1-c)}} \sqrt{\frac{2(\psi_0 + S)}{\mu}} \left\lceil \sqrt{\kappa} \ln \frac{2(\psi_0 + S)C_L^2}{\varepsilon^2} \right\rceil \right\}.$$

Proof. Let K_1 be the smallest integer such that $\tilde{\mathbf{x}}^{(K_1)}$ is an ε -stationary point. It follows from the definition of ψ_k , (3.15), and (4.8) that

$$(4.18) \quad \text{dist}(\mathbf{0}, \partial F(\tilde{\mathbf{x}}^{(k+1)})) \leq C_L \sqrt{2 \prod_{j=0}^k (1 - c\alpha_j) (\psi_0 + S)} \quad \forall k \geq 0.$$

Let $K'_1 = \lceil \sqrt{\kappa} \ln \frac{2(\psi_0 + S)C_L^2}{\varepsilon^2} \rceil$. Since $\alpha_j \geq 1/\sqrt{\kappa}$ by Lemma 3.3, (4.18) implies $\prod_{j=0}^{K'_1-1} (1 - c\alpha_j) (\psi_0 + S) \leq (1 - c/\sqrt{\kappa})^{K'_1} (\psi_0 + S) \leq \frac{\varepsilon^2}{2C_L^2}$, which means $K_1 \leq K'_1$.

By Lemma 3.2, until an ε -stationary solution is found, the total numbers of iterations in Algorithms 2 and 3 are $(1 + \frac{\ln \gamma_{\text{inc}}}{\ln \gamma_{\text{dec}}})K_1 + \frac{1}{\ln \gamma_{\text{dec}}} \ln(\frac{L_g}{\mu \gamma_{\text{dec}}})$ and $K_1 + 1 + \frac{1}{\ln \gamma_{\text{dec}}^{-1}} \ln(\frac{L_g + L_h}{L_{\text{dec}}})$, respectively. Since $(g, \nabla g)$ is queried only twice in each iteration of Algorithms 2 and 3, the total number of queries to $(g, \nabla g)$ by Algorithm 1 is at most $(2 + \frac{\ln \gamma_{\text{inc}}}{\ln \gamma_{\text{dec}}})K_1 + 1 + \log_{\gamma_{\text{dec}}}(\frac{\mu L_g \gamma_{\text{dec}}}{L_g(L_g + L_h)})$, which implies (4.15) when $\underline{L} = \Theta(L_g)$ because $K_1 \leq K'_1$.

Next we bound the right-hand side of (4.2). By the definition of δ_k in (4.7) and the choice of ε_k in (4.5), we have

$$(4.19) \quad \frac{\varepsilon_{k-1} + \frac{3L_g(\delta_k + \delta_{k-1})}{\gamma_{\text{dec}}\sqrt{c}}}{\varepsilon_k} = \frac{(k+1)}{k\sqrt{1 - c\alpha_{k-1}}} + \frac{3L_g(k+1)}{\varepsilon_0 \gamma_{\text{dec}} \sqrt{c}} \sqrt{\frac{2(\psi_0 + S)}{\mu}} \\ + \frac{3L_g(k+1)}{\varepsilon_0 \gamma_{\text{dec}} \sqrt{c} \sqrt{1 - c\alpha_{k-1}}} \sqrt{\frac{2(\psi_0 + S)}{\mu}}$$

⁶We assume $\underline{L} = \Theta(L_g)$ just to simplify the results. The analysis holds for any $\underline{L} \leq L_g$.

$$\begin{aligned} &\leq \frac{2}{\sqrt{1-c}} + \frac{6L_g(k+1)}{\varepsilon_0\gamma_{\text{dec}}\sqrt{c(1-c)}}\sqrt{\frac{2(\psi_0+S)}{\mu}} \leq \frac{2}{\sqrt{1-c}} \\ &\quad + \frac{6L_gK'_1}{\varepsilon_0\gamma_{\text{dec}}\sqrt{c(1-c)}}\sqrt{\frac{2(\psi_0+S)}{\mu}} \end{aligned}$$

for any $1 \leq k \leq K'_1 - 1$, where the inequality comes from $\alpha_{k-1} \leq 1$. This implies

$$(4.20) \quad \Phi(\mathbf{x}^{(k)}; \mathbf{y}^{(k)}, \eta_k) - \Phi(\mathbf{x}_*^{(k+1)}; \mathbf{y}^{(k)}, \eta_k) \leq \frac{1}{2L}C_\varepsilon^2.$$

In iteration k of Algorithm 1, the query number of $(h, \nabla h)$ to compute $\mathbf{x}^{(k+1)}$ satisfying (3.2) is at most T_k given in (4.2), and Algorithm 2 will stop after at most $\log_{\gamma_{\text{dec}}} \frac{L\gamma_{\text{dec}}^2}{L_g}$ iterations by Lemma 3.2. In addition, two queries to $(h, \nabla h)$ are made in each iteration of Algorithm 3, which will stop after at most $\log_{\gamma_{\text{dec}}} \frac{L\gamma_{\text{dec}}^2}{L_g + L_h}$ iterations by Lemma 3.2. Hence, the query number of $(h, \nabla h)$ at iteration k of Algorithm 1 is no more than $(\log_{\gamma_{\text{dec}}} \frac{L\gamma_{\text{dec}}^2}{L_g})T_k + 2\log_{\gamma_{\text{dec}}} \frac{L\gamma_{\text{dec}}^2}{L_g + L_h}$. Applying (4.20) to the right-hand side of (4.2), we can show that the total number of queries to $(h, \nabla h)$ before finding an ε -optimal solution is at most

$$\begin{aligned} T_{\text{crit}}^{\text{sc}} &= K_1 \cdot O \left(\log_{\gamma_{\text{dec}}} \left(\frac{L\gamma_{\text{dec}}^2}{L_g} \right) \sqrt{1 + \frac{L_h}{L} \ln \left(\sqrt{\frac{L_g}{L}} + \frac{L_h}{L} + \frac{L_h^2}{L^2} C_\varepsilon \right)} \right. \\ &\quad \left. + \log_{\gamma_{\text{dec}}} \frac{L\gamma_{\text{dec}}^2}{L_g + L_h} \right). \end{aligned}$$

Using the facts that $K_1 \leq K'_1$ and $\underline{L} = \Theta(L_g)$ and the fact that $\ln(\frac{L_g + L_h}{L_g})\sqrt{\kappa} \leq \sqrt{\frac{L_g + L_h}{L_g}}\sqrt{\kappa} = \sqrt{\frac{L_g + L_h}{\gamma_{\text{dec}}\mu}}$, we obtain the desired result in (4.16). \square

5. Inexact regularized augmented Lagrangian method. In this section, we consider the affine-constrained composite problem

$$(5.1) \quad \min_{\mathbf{x}} \{G(\mathbf{x}) := f(\mathbf{x}) + r(\mathbf{x})\} \text{ s.t. } \mathbf{A}_E \mathbf{x} = \mathbf{b}_E, \mathbf{A}_I \mathbf{x} \leq \mathbf{b}_I,$$

where f is L_f -smooth and μ -strongly convex with $\mu \geq 0$, and r is closed convex and allows easy computation of $\text{prox}_{\eta r}(\mathbf{z})$ and $\text{dist}(\mathbf{0}, \partial r(\mathbf{z}))$ for any $\mathbf{z} \in \mathbb{R}^n$ and $\eta > 0$. We assume that $(f, \nabla f)$ is significantly more expensive than $(\mathbf{A}(\cdot), \mathbf{A}^\top(\cdot))$ to evaluate, where $\mathbf{A} = [\mathbf{A}_E; \mathbf{A}_I]$. We denote the Lagrange multiplier by $\boldsymbol{\lambda} = [\boldsymbol{\lambda}_E; \boldsymbol{\lambda}_I]$ with $\boldsymbol{\lambda}_E$ and $\boldsymbol{\lambda}_I$ associated to the equality and inequality constraints, respectively. We assume (5.1) has an optimal solution \mathbf{x}^* and the multiplier $\boldsymbol{\lambda}^* = [\boldsymbol{\lambda}_E^*; \boldsymbol{\lambda}_I^*]$ satisfying

$$(5.2) \quad \mathbf{0} \in \partial G(\mathbf{x}^*) + \mathbf{A}^\top \boldsymbol{\lambda}^*; \mathbf{A}_E \mathbf{x}^* = \mathbf{b}_E, \mathbf{A}_I \mathbf{x}^* \leq \mathbf{b}_I; \boldsymbol{\lambda}_I^* \geq \mathbf{0}, \langle \boldsymbol{\lambda}_I^*, \mathbf{A}_I \mathbf{x}^* - \mathbf{b}_I \rangle = 0.$$

Our goal is to find an ε -stationary solution of (5.1) defined formally below.

DEFINITION 5.1 (ε -stationary solution). *For a given $\varepsilon \geq 0$, a point $\bar{\mathbf{x}} \in \text{dom}(G)$ is called an ε -stationary solution of (5.1) if there exists $\bar{\boldsymbol{\lambda}} = [\bar{\boldsymbol{\lambda}}_E; \bar{\boldsymbol{\lambda}}_I]$ such that*

$$(5.3) \quad \text{dist}(\mathbf{0}, \partial G(\bar{\mathbf{x}}) + \mathbf{A}^\top \bar{\boldsymbol{\lambda}}) \leq \varepsilon; \sqrt{\|\mathbf{A}_E \bar{\mathbf{x}} - \mathbf{b}_E\|^2 + \|\mathbf{A}_I \bar{\mathbf{x}} - \mathbf{b}_I\|_+^2} \leq \varepsilon;$$

$$(5.4) \quad \bar{\boldsymbol{\lambda}}_I \geq \mathbf{0}, \|\bar{\boldsymbol{\lambda}}_I \odot (\mathbf{A}_I \bar{\mathbf{x}} - \mathbf{b}_I)\| \leq \varepsilon.$$

Algorithm 4 Inexact regularized augmented Lagrangian method (iRALM).**Inputs:** $\mathbf{x}^{(0)} \in \text{dom}(G)$, $\boldsymbol{\lambda}^{(0)}$, $\beta_k > 0$, $\rho_k > 0$, $\bar{\varepsilon}_k > 0$, $\forall, k \geq 0$ and $\varepsilon > 0$ $k \leftarrow 0$ **while** Conditions (5.3) and (5.4) with $(\bar{\mathbf{x}}, \bar{\boldsymbol{\lambda}}) = (\mathbf{x}^{(k)}, \boldsymbol{\lambda}^{(k)})$ do not hold **do**Find $\mathbf{x}^{(k+1)} \approx \arg \min_{\mathbf{x}} \Psi_k(\mathbf{x})$, where Ψ_k is defined in (5.5), such that

(5.6)
$$\text{dist}(\mathbf{0}, \partial \Psi_k(\mathbf{x}^{(k+1)})) \leq \bar{\varepsilon}_k,$$

which can be done, e.g., by the iAPG method. See Settings 1 and 2 below. Let $\boldsymbol{\lambda}_E^{(k+1)} = \boldsymbol{\lambda}_E^{(k)} + \beta_k(\mathbf{A}_E \mathbf{x}^{(k+1)} - \mathbf{b}_E)$, $\boldsymbol{\lambda}_I^{(k+1)} = [\boldsymbol{\lambda}_I^{(k)} + \beta_k(\mathbf{A}_I \mathbf{x}^{(k+1)} - \mathbf{b}_I)]_+$, and set $k \leftarrow k + 1$.**Return:** $(\mathbf{x}^{(k)}, \boldsymbol{\lambda}^{(k)})$

We consider an iRALM presented in Algorithm 4 for finding an ε -stationary solution for (5.1). At iteration k , the iRALM generates the next solution by

(5.5)
$$\mathbf{x}^{(k+1)} \approx \arg \min_{\mathbf{x}} \mathbf{x} \left\{ \Psi_k(\mathbf{x}) := \mathcal{L}_{\beta_k}(\mathbf{x}, \boldsymbol{\lambda}^{(k)}) + \frac{\rho_k}{2} \|\mathbf{x} - \mathbf{x}^{(k)}\|^2 \right\}.$$

Here, \mathcal{L}_{β} is the augmented Lagrangian function of (5.1) with the following form:

$$\begin{aligned} \mathcal{L}_{\beta}(\mathbf{x}, \boldsymbol{\lambda}) &= G(\mathbf{x}) + \langle \boldsymbol{\lambda}_E, \mathbf{A}_E \mathbf{x} - \mathbf{b}_E \rangle + \frac{\beta}{2} \|\mathbf{A}_E \mathbf{x} - \mathbf{b}_E\|^2 \\ &\quad + \frac{1}{2\beta} \left(\|\beta(\mathbf{A}_I \mathbf{x} - \mathbf{b}_I) + \boldsymbol{\lambda}_I\|_+^2 - \|\boldsymbol{\lambda}_I\|^2 \right). \end{aligned}$$

In particular, the iPLAM requires $\mathbf{x}^{(k+1)}$ to be an $\bar{\varepsilon}_k$ -stationary point of Ψ_k . We can guarantee this by applying Algorithm 1 to (5.5). We will show that, compared to existing results, the iRALM finds an ε -stationary solution with a significantly reduced number of queries to $(f, \nabla f)$ but a slightly increased number of queries to $(\mathbf{A}(\cdot), \mathbf{A}^\top(\cdot))$.

Before giving the details, we first present the following lemmas to characterize the relationship between two consecutive iterates of Algorithm 4.

LEMMA 5.2. *Algorithm 4 guarantees that, for any $k \geq 0$,*

(5.7)

$$\begin{aligned} \bar{\varepsilon}_k \|\mathbf{x}^{(k+1)} - \mathbf{x}^*\| &\geq \mu \|\mathbf{x}^{(k+1)} - \mathbf{x}^*\|^2 + \frac{1}{2\beta_k} (\|\boldsymbol{\lambda}^{(k+1)} - \boldsymbol{\lambda}^*\|^2 + \|\boldsymbol{\lambda}^{(k+1)} - \boldsymbol{\lambda}^{(k)}\|^2 - \|\boldsymbol{\lambda}^{(k)} - \boldsymbol{\lambda}^*\|^2) \\ &\quad + \frac{\rho_k}{2} (\|\mathbf{x}^{(k+1)} - \mathbf{x}^{(k)}\|^2 + \|\mathbf{x}^{(k+1)} - \mathbf{x}^*\|^2 - \|\mathbf{x}^{(k)} - \mathbf{x}^*\|^2). \end{aligned}$$

Proof. From (5.6), there exists $\mathbf{v}^{(k)} \in \partial_{\mathbf{x}} \mathcal{L}_{\beta_k}(\mathbf{x}^{(k+1)}, \boldsymbol{\lambda}^{(k)}) + \rho_k(\mathbf{x}^{(k+1)} - \mathbf{x}^{(k)})$ such that $\|\mathbf{v}^{(k)}\| \leq \bar{\varepsilon}_k$, and thus by the μ -strong convexity of G , we have

$$\begin{aligned} (5.8) \quad \langle \mathbf{v}^{(k)}, \mathbf{x}^{(k+1)} - \mathbf{x}^* \rangle &\geq G(\mathbf{x}^{(k+1)}) - G(\mathbf{x}^*) + \frac{\mu}{2} \|\mathbf{x}^{(k+1)} - \mathbf{x}^*\|^2 + \langle \mathbf{A}_E^\top \boldsymbol{\lambda}_E^{(k)}, \mathbf{x}^{(k+1)} - \mathbf{x}^* \rangle \\ &\quad + \langle \beta_k \mathbf{A}_E^\top (\mathbf{A}_E \mathbf{x}^{(k+1)} - \mathbf{b}_E), \mathbf{x}^{(k+1)} - \mathbf{x}^* \rangle \\ &\quad + \langle \mathbf{A}_I^\top [\beta_k(\mathbf{A}_I \mathbf{x}^{(k+1)} - \mathbf{b}_I) + \boldsymbol{\lambda}_I^{(k)}]_+, \mathbf{x}^{(k+1)} - \mathbf{x}^* \rangle \\ &\quad + \langle \rho_k(\mathbf{x}^{(k+1)} - \mathbf{x}^{(k)}), \mathbf{x}^{(k+1)} - \mathbf{x}^* \rangle. \end{aligned}$$

By the Cauchy–Schwarz inequality, it holds that $\langle \mathbf{v}^{(k)}, \mathbf{x}^{(k+1)} - \mathbf{x}^* \rangle \leq \|\mathbf{v}^{(k)}\| \cdot \|\mathbf{x}^{(k+1)} - \mathbf{x}^*\| \leq \bar{\varepsilon}_k \|\mathbf{x}^{(k+1)} - \mathbf{x}^*\|$. Hence, by the update of $\boldsymbol{\lambda}^{(k+1)}$ and the facts $\mathbf{A}_E \mathbf{x}^* = \mathbf{b}_E$ and $\mathbf{A}_I \mathbf{x}^* \leq \mathbf{b}_I$, we obtain from (5.8) that

$$\begin{aligned}
 (5.9) \quad & \bar{\varepsilon}_k \|\mathbf{x}^{(k+1)} - \mathbf{x}^*\| \\
 & \geq G(\mathbf{x}^{(k+1)}) - G(\mathbf{x}^*) + \frac{\mu}{2} \|\mathbf{x}^{(k+1)} - \mathbf{x}^*\|^2 + \langle \boldsymbol{\lambda}_E^{(k+1)}, \mathbf{A}_E \mathbf{x}^{(k+1)} - \mathbf{b}_E \rangle \\
 & \quad + \langle \boldsymbol{\lambda}_I^{(k+1)}, \mathbf{A}_I \mathbf{x}^{(k+1)} - \mathbf{b}_I \rangle + \langle \rho_k(\mathbf{x}^{(k+1)} - \mathbf{x}^{(k)}), \mathbf{x}^{(k+1)} - \mathbf{x}^* \rangle \\
 & = G(\mathbf{x}^{(k+1)}) - G(\mathbf{x}^*) + \frac{\mu}{2} \|\mathbf{x}^{(k+1)} - \mathbf{x}^*\|^2 + \langle \boldsymbol{\lambda}_E^{(k+1)}, \mathbf{A}_E \mathbf{x}^{(k+1)} - \mathbf{b}_E \rangle \\
 & \quad + \langle \boldsymbol{\lambda}_I^{(k+1)}, \mathbf{A}_I \mathbf{x}^{(k+1)} - \mathbf{b}_I \rangle + \frac{\rho_k}{2} (\|\mathbf{x}^{(k+1)} - \mathbf{x}^{(k)}\|^2 \\
 & \quad + \|\mathbf{x}^{(k+1)} - \mathbf{x}^*\|^2 - \|\mathbf{x}^{(k)} - \mathbf{x}^*\|^2).
 \end{aligned}$$

Using the updating equation of $\boldsymbol{\lambda}^{(k+1)}$ again, we have

$$\begin{aligned}
 (5.10) \quad & \langle \boldsymbol{\lambda}_E^{(k+1)} - \boldsymbol{\lambda}_E^*, \mathbf{A}_E \mathbf{x}^{(k+1)} - \mathbf{b}_E \rangle = \left\langle \boldsymbol{\lambda}_E^{(k+1)} - \boldsymbol{\lambda}_E^*, \frac{1}{\beta_k} (\boldsymbol{\lambda}_E^{(k+1)} - \boldsymbol{\lambda}_E^{(k)}) \right\rangle \\
 & = \frac{1}{2\beta_k} \left(\|\boldsymbol{\lambda}_E^{(k+1)} - \boldsymbol{\lambda}_E^*\|^2 + \|\boldsymbol{\lambda}_E^{(k+1)} - \boldsymbol{\lambda}_E^{(k)}\|^2 - \|\boldsymbol{\lambda}_E^{(k)} - \boldsymbol{\lambda}_E^*\|^2 \right).
 \end{aligned}$$

By [75, Lemma 4], it holds that

$$\langle \boldsymbol{\lambda}_I^{(k+1)} - \boldsymbol{\lambda}_I^*, \mathbf{A}_I \mathbf{x}^{(k+1)} - \mathbf{b}_I \rangle \geq \left\langle \boldsymbol{\lambda}_I^{(k+1)} - \boldsymbol{\lambda}_I^*, \max \left\{ -\frac{\boldsymbol{\lambda}_I^{(k)}}{\beta_k}, \mathbf{A}_I \mathbf{x}^{(k+1)} - \mathbf{b}_I \right\} \right\rangle,$$

which together with $\max \left\{ -\frac{\boldsymbol{\lambda}_I^{(k)}}{\beta_k}, \mathbf{A}_I \mathbf{x}^{(k+1)} - \mathbf{b}_I \right\} = \frac{1}{\beta_k} (\boldsymbol{\lambda}_I^{(k+1)} - \boldsymbol{\lambda}_I^{(k)})$ gives

$$\begin{aligned}
 (5.11) \quad & \langle \boldsymbol{\lambda}_I^{(k+1)} - \boldsymbol{\lambda}_I^*, \mathbf{A}_I \mathbf{x}^{(k+1)} - \mathbf{b}_I \rangle \\
 & = \left\langle \boldsymbol{\lambda}_I^{(k+1)} - \boldsymbol{\lambda}_I^*, \frac{1}{\beta_k} (\boldsymbol{\lambda}_I^{(k+1)} - \boldsymbol{\lambda}_I^{(k)}) \right\rangle \\
 & = \frac{1}{2\beta_k} \left(\|\boldsymbol{\lambda}_I^{(k+1)} - \boldsymbol{\lambda}_I^*\|^2 + \|\boldsymbol{\lambda}_I^{(k+1)} - \boldsymbol{\lambda}_I^{(k)}\|^2 - \|\boldsymbol{\lambda}_I^{(k)} - \boldsymbol{\lambda}_I^*\|^2 \right).
 \end{aligned}$$

Adding (5.10) and (5.11) to (5.9) gives

$$\begin{aligned}
 (5.12) \quad & \bar{\varepsilon}_k \|\mathbf{x}^{(k+1)} - \mathbf{x}^*\| \geq G(\mathbf{x}^{(k+1)}) - G(\mathbf{x}^*) + \frac{\mu}{2} \|\mathbf{x}^{(k+1)} - \mathbf{x}^*\|^2 + \langle \boldsymbol{\lambda}^*, \mathbf{A} \mathbf{x}^{(k+1)} - \mathbf{b} \rangle \\
 & \quad + \frac{\rho_k}{2} (\|\mathbf{x}^{(k+1)} - \mathbf{x}^{(k)}\|^2 + \|\mathbf{x}^{(k+1)} - \mathbf{x}^*\|^2 - \|\mathbf{x}^{(k)} - \mathbf{x}^*\|^2) \\
 & \quad + \frac{1}{2\beta_k} (\|\boldsymbol{\lambda}^{(k+1)} - \boldsymbol{\lambda}^*\|^2 + \|\boldsymbol{\lambda}^{(k+1)} - \boldsymbol{\lambda}^{(k)}\|^2 - \|\boldsymbol{\lambda}^{(k)} - \boldsymbol{\lambda}^*\|^2).
 \end{aligned}$$

By the KKT conditions $\mathbf{0} \in \partial G(\mathbf{x}^*) + \mathbf{A}^\top \boldsymbol{\lambda}^*$ and $\langle \boldsymbol{\lambda}_I^*, \mathbf{A}_I \mathbf{x}^* - \mathbf{b}_I \rangle = 0$, it follows that

$$\begin{aligned}
 & G(\mathbf{x}^{(k+1)}) - G(\mathbf{x}^*) + \langle \boldsymbol{\lambda}^*, \mathbf{A} \mathbf{x}^{(k+1)} - \mathbf{b} \rangle \\
 & = G(\mathbf{x}^{(k+1)}) - G(\mathbf{x}^*) + \langle \mathbf{A}^\top \boldsymbol{\lambda}^*, \mathbf{x}^{(k+1)} - \mathbf{x}^* \rangle \geq \frac{\mu}{2} \|\mathbf{x}^{(k+1)} - \mathbf{x}^*\|^2,
 \end{aligned}$$

where the inequality holds from the μ -strong convexity of G . Applying this inequality to (5.12) gives the desired result. \square

5.1. Outer-iteration complexity. In this subsection, we assume that (5.6) can be guaranteed. We specify the choices of $\{\beta_k\}_{k \geq 0}$, $\{\rho_k\}_{k \geq 0}$, and $\{\bar{\varepsilon}_k\}_{k \geq 0}$ and establish the outer-iteration complexity of Algorithm 4. To do so, we first show the uniform boundedness of the primal-dual iterates below.

LEMMA 5.3 (bounded iterates). *Suppose $\beta_k = \beta_0 \sigma^k$ and $\rho_k = \rho_0 \sigma^{-k}$ $\forall k \geq 0$ for some $\beta_0 > 0, \rho_0 > 0$, and $\sigma > 1$ in Algorithm 4. It holds, for any $k \geq 0$, that*

$$(5.13) \quad \begin{aligned} \sqrt{\beta_0 \rho_0 \|\mathbf{x}^{(k+1)} - \mathbf{x}^*\|^2 + \|\boldsymbol{\lambda}^{(k+1)} - \boldsymbol{\lambda}^*\|^2} &\leq \sum_{i=0}^k \frac{2\beta_i \bar{\varepsilon}_i}{\sqrt{\beta_0 \rho_0}} \\ &+ \sqrt{\beta_0 \rho_0 \|\mathbf{x}^{(0)} - \mathbf{x}^*\|^2 + \|\boldsymbol{\lambda}^{(0)} - \boldsymbol{\lambda}^*\|^2}. \end{aligned}$$

Proof. Multiplying $2\beta_k$ to both sides of (5.7) gives

$$(5.14) \quad \begin{aligned} 2\beta_k \bar{\varepsilon}_k \|\mathbf{x}^{(k+1)} - \mathbf{x}^*\| \\ \geq 2\mu\beta_k \|\mathbf{x}^{(k+1)} - \mathbf{x}^*\|^2 + (\|\boldsymbol{\lambda}^{(k+1)} - \boldsymbol{\lambda}^*\|^2 + \|\boldsymbol{\lambda}^{(k+1)} - \boldsymbol{\lambda}^{(k)}\|^2 - \|\boldsymbol{\lambda}^{(k)} - \boldsymbol{\lambda}^*\|^2) \\ + \beta_0 \rho_0 (\|\mathbf{x}^{(k+1)} - \mathbf{x}^{(k)}\|^2 + \|\mathbf{x}^{(k+1)} - \mathbf{x}^*\|^2 - \|\mathbf{x}^{(k)} - \mathbf{x}^*\|^2). \end{aligned}$$

Sum up (5.14) to have

$$\begin{aligned} \beta_0 \rho_0 \|\mathbf{x}^{(k+1)} - \mathbf{x}^*\|^2 + \|\boldsymbol{\lambda}^{(k+1)} - \boldsymbol{\lambda}^*\|^2 \\ \leq \sum_{i=0}^k 2\beta_i \bar{\varepsilon}_i \|\mathbf{x}^{(i+1)} - \mathbf{x}^*\| + \beta_0 \rho_0 \|\mathbf{x}^{(0)} - \mathbf{x}^*\|^2 + \|\boldsymbol{\lambda}^{(0)} - \boldsymbol{\lambda}^*\|^2. \end{aligned}$$

We obtain (5.13) by the inequality above and >Lemma A.1 with $\lambda_i = \frac{2\beta_{i-1} \bar{\varepsilon}_{i-1}}{\sqrt{\beta_0 \rho_0}}$, $u_k = \sqrt{\beta_0 \rho_0 \|\mathbf{x}^{(k)} - \mathbf{x}^*\|^2 + \|\boldsymbol{\lambda}^{(k)} - \boldsymbol{\lambda}^*\|^2}$, and $C = \beta_0 \rho_0 \|\mathbf{x}^{(0)} - \mathbf{x}^*\|^2 + \|\boldsymbol{\lambda}^{(0)} - \boldsymbol{\lambda}^*\|^2$. \square

By Lemmas 5.3 and A.2, we show that Algorithm 4 produces an ε -KKT point.

THEOREM 5.4. *Let β_k and ρ_k be defined as in Lemma 5.3, $\bar{\varepsilon} = \frac{\varepsilon(\sigma-1)}{8(\sigma+1)} \min\{1, \sqrt{\beta_0 \rho_0}\}$, and $\bar{\varepsilon}_k = \min\{\bar{\varepsilon}, \sqrt{\frac{\rho_0}{20\sigma}} \sigma^{-k}\} \forall k \geq 0$ in Algorithm 4. Then Algorithm 4 will stop and return $\mathbf{x}^{(k)}$ as an ε -stationary point of (5.1) with k no more than*

$$(5.15) \quad K := \max \left\{ \left\lceil \log_\sigma \frac{4D_0 \sqrt{\rho_0}}{\sqrt{\beta_0} \bar{\varepsilon}} \right\rceil, \left\lceil \log_\sigma \frac{4D_0}{\beta_0 \varepsilon} \right\rceil, \right. \\ \left. \left\lceil \log_\sigma \frac{5(D_0 + \|\boldsymbol{\lambda}^*\|)^2}{\beta_0 \varepsilon} \right\rceil, \left\lceil 2 \log_\sigma \frac{8}{\varepsilon (\ln \sigma)^2} \right\rceil - 1 \right\} + 1,$$

$$(5.16) \quad \text{where } D_0 = \sqrt{\beta_0 \rho_0 \|\mathbf{x}^{(0)} - \mathbf{x}^*\|^2 + \|\boldsymbol{\lambda}^{(0)} - \boldsymbol{\lambda}^*\|^2}.$$

Proof. Since $\bar{\varepsilon}_i \leq \bar{\varepsilon}$ for $i \geq 0$, we have from (5.13) that

$$(5.17) \quad \|\mathbf{x}^{(k)} - \mathbf{x}^*\| \leq \frac{2\bar{\varepsilon}(\sigma^k - 1)}{\rho_0(\sigma-1)} + \frac{D_0}{\sqrt{\beta_0 \rho_0}}, \quad \|\boldsymbol{\lambda}^{(k)} - \boldsymbol{\lambda}^*\| \leq \frac{2\bar{\varepsilon}\sqrt{\beta_0}(\sigma^k - 1)}{\sqrt{\rho_0}(\sigma-1)} + D_0, \quad \forall k \geq 0,$$

with D_0 defined in (5.16). Hence, by the triangle inequality and (5.17), it holds that

$$\begin{aligned} \|\mathbf{x}^{(k+1)} - \mathbf{x}^{(k)}\| &\leq \frac{2\bar{\varepsilon}(\sigma^{k+1} + \sigma^k - 2)}{\rho_0(\sigma-1)} + \frac{2D_0}{\sqrt{\beta_0 \rho_0}}, \quad \|\boldsymbol{\lambda}^{(k+1)} - \boldsymbol{\lambda}^{(k)}\| \\ &\leq \frac{2\bar{\varepsilon}\sqrt{\beta_0}(\sigma^{k+1} + \sigma^k - 2)}{\sqrt{\rho_0}(\sigma-1)} + 2D_0, \end{aligned}$$

and thus

$$(5.18) \quad \begin{aligned} \rho_{K-1} \|\mathbf{x}^{(K)} - \mathbf{x}^{(K-1)}\| &\leq \frac{2\bar{\varepsilon}(\sigma+1)}{\sigma-1} + \frac{2D_0\sqrt{\rho_0}}{\sqrt{\beta_0}\sigma^{K-1}}, \quad \frac{1}{\beta_{K-1}} \|\boldsymbol{\lambda}^{(K)} - \boldsymbol{\lambda}^{(K-1)}\| \\ &\leq \frac{2\bar{\varepsilon}(\sigma+1)}{\sqrt{\beta_0\rho_0}(\sigma-1)} + \frac{2D_0}{\beta_0\sigma^{K-1}}. \end{aligned}$$

By the choice of $\bar{\varepsilon}$ and the definition of K in (5.15), we have from (5.18) that

$$(5.19) \quad \rho_{K-1} \|\mathbf{x}^{(K)} - \mathbf{x}^{(K-1)}\| \leq \frac{3\varepsilon}{4}, \quad \frac{1}{\beta_{K-1}} \|\boldsymbol{\lambda}^{(K)} - \boldsymbol{\lambda}^{(K-1)}\| \leq \frac{3\varepsilon}{4}.$$

Additionally, since $\bar{\varepsilon}_i \leq \sqrt{\frac{\rho_0}{20\sigma}}\sigma^{-i}$ for $i \geq 0$, it is implied by (5.13) that $\|\boldsymbol{\lambda}^{(k)} - \boldsymbol{\lambda}^*\| \leq k\sqrt{\frac{\beta_0}{5\sigma}} + D_0$ and thus $\|\boldsymbol{\lambda}^{(k)}\| \leq k\sqrt{\frac{\beta_0}{5\sigma}} + D_0 + \|\boldsymbol{\lambda}^*\|$, which further implies $\|\boldsymbol{\lambda}^{(k)}\|^2 \leq \frac{2\beta_0 k^2}{5\sigma} + 2(D_0 + \|\boldsymbol{\lambda}^*\|)^2 \forall k \geq 0$. Hence,

$$(5.20) \quad \frac{1}{\beta_{K-1}} \left(\|\boldsymbol{\lambda}^{(K)}\|^2 + \frac{1}{4} \|\boldsymbol{\lambda}^{(K-1)}\|^2 \right) \leq \frac{1}{\beta_0\sigma^{K-1}} \left(\frac{\beta_0 K^2}{2\sigma} + \frac{5}{2}(D_0 + \|\boldsymbol{\lambda}^*\|)^2 \right).$$

Since $K-1 \geq \log_\sigma \frac{5(D_0 + \|\boldsymbol{\lambda}^*\|)^2}{\beta_0\varepsilon}$, it holds that $\frac{5}{2\beta_0\sigma^{K-1}}(D_0 + \|\boldsymbol{\lambda}^*\|)^2 \leq \frac{\varepsilon}{2}$. Also, $K \geq \lceil 2\log_\sigma \frac{8}{\varepsilon(\ln\sigma)^2} \rceil$ implies $\sigma^K \geq \frac{64}{\varepsilon^2(\ln\sigma)^4}$. Thus $\frac{K^2}{\sigma^K} \leq \varepsilon$ according to Lemma A.2 with $a = \varepsilon$ and $b = \sigma^K$. Hence, the right-hand side of (5.20) is no more than ε , so

$$(5.21) \quad \frac{1}{\beta_{K-1}} \left(\|\boldsymbol{\lambda}^{(K)}\|^2 + \frac{1}{4} \|\boldsymbol{\lambda}^{(K-1)}\|^2 \right) \leq \varepsilon.$$

Now from the updating equations of $\mathbf{x}^{(k+1)}$ and $\boldsymbol{\lambda}^{(k+1)}$, we have for any $k \geq 1$,

$$(5.22a) \quad \text{dist}(\mathbf{0}, \partial G(\mathbf{x}^{(k)}) + \mathbf{A}^\top \boldsymbol{\lambda}^{(k)}) \leq \bar{\varepsilon}_{k-1} + \rho_{k-1} \|\mathbf{x}^{(k)} - \mathbf{x}^{(k-1)}\|,$$

$$(5.22b) \quad \sqrt{\|\mathbf{A}_E \mathbf{x}^{(k)} - \mathbf{b}_E\|^2 + \|[\mathbf{A}_I \mathbf{x}^{(k)} - \mathbf{b}_I]_+\|^2} \leq \frac{1}{\beta_{k-1}} \|\boldsymbol{\lambda}^{(k)} - \boldsymbol{\lambda}^{(k-1)}\|,$$

and, by line 5 of Algorithm 4, we have $\boldsymbol{\lambda}_I^{(k)} \geq \mathbf{0} \forall k \geq 1$ and

$$(5.22c) \quad \begin{aligned} \|\boldsymbol{\lambda}_I^{(k)} \odot (\mathbf{A}_I \mathbf{x}^{(k)} - \mathbf{b}_I)\| &\leq \sum_{i \in I, \boldsymbol{\lambda}_i^{(k)} > 0} |\boldsymbol{\lambda}_i^{(k)} \cdot (\mathbf{A}_i \mathbf{x}^{(k)} - \mathbf{b}_i)| \\ &= \sum_{i \in I, \boldsymbol{\lambda}_i^{(k)} > 0} |\boldsymbol{\lambda}_i^{(k)} \cdot (\boldsymbol{\lambda}_i^{(k)} - \boldsymbol{\lambda}_i^{(k-1)})| / \beta_{k-1} \leq \frac{1}{\beta_{k-1}} \left(\|\boldsymbol{\lambda}_I^{(k)}\|^2 + \frac{1}{4} \|\boldsymbol{\lambda}_I^{(k-1)}\|^2 \right). \end{aligned}$$

Moreover, by (5.19), (5.21), and $\bar{\varepsilon}_k \leq \frac{\varepsilon}{4}$, the three inequalities in (5.22) imply that $(\mathbf{x}^{(K)}, \boldsymbol{\lambda}^{(K)})$ is an ε -stationary solution of (5.1), which completes the proof. \square

5.2. Overall oracle complexity. In this subsection, we discuss the details of how to ensure (5.6) and then characterize the total oracle complexity of Algorithm 4 to produce an ε -stationary point of (5.1). Define

$$(5.23) \quad g_k(\mathbf{x}) = f(\mathbf{x}) + \frac{\rho_k}{2} \|\mathbf{x} - \mathbf{x}^{(k)}\|^2,$$

$$(5.24) \quad h_k(\mathbf{x}) = \langle \boldsymbol{\lambda}_E^{(k)}, \mathbf{A}_E \mathbf{x} - \mathbf{b}_E \rangle + \frac{\beta_k}{2} \|\mathbf{A}_E \mathbf{x} - \mathbf{b}_E\|^2 + \frac{\|[\beta_k(\mathbf{A}_I \mathbf{x} - \mathbf{b}_I) + \boldsymbol{\lambda}_I^{(k)}]_+\|^2}{2\beta_k} - \frac{\|\boldsymbol{\lambda}_I^{(k)}\|^2}{2\beta_k}.$$

Then the iRALM subproblem (5.5) can be written as

$$(5.25) \quad \min_{\mathbf{x}} \{\Psi_k(\mathbf{x}) = g_k(\mathbf{x}) + h_k(\mathbf{x}) + r(\mathbf{x})\},$$

which is an instance of (1.1) with $g = g_k$ and $h = h_k$. This means that (5.6) can be ensured by approximately solving the iRALM subproblem (5.25) using Algorithm 1. This way, we can apply the complexity result in Theorem 4.3 to establish the oracle complexity for each outer iteration of Algorithm 4.

We adopt the following settings on solving each iRALM subproblem.

SETTING 1 (how to solve iRALM subproblems). *In iteration k of Algorithm 4, Algorithm 1 is applied to find $\mathbf{x}^{(k+1)}$ satisfying (5.6). More precisely, we compute $\mathbf{x}^{(k+1)}$ by*

$$(5.26) \quad \mathbf{x}^{(k+1)} = \text{iAPG} \left(g_k, h_k, r, \mathbf{x}^{(k)}, \eta_0, \gamma_0, \mu + \rho_k, \underline{L}, \{\varepsilon_t\}_{t \geq 0}, \bar{\varepsilon}_k \right),$$

where ε_t is defined as in (4.5) for $t \geq 1$, g_k is defined in (5.23), h_k is defined in (5.24), and⁷ $\underline{L} = \Theta(L_f)$.

For simplicity, in the setting above, the values of $\eta_0, \gamma_0, \gamma_{\text{dec}}, \gamma_{\text{inc}}, \underline{L}$, and ε_0 stay the same across the calls of the iAPG by different iterations of the iRALM. Also, we use the previous iRALM iterate $\mathbf{x}^{(k)}$ as the initial point to solve the k th subproblem.

SETTING 2 (choice of parameters). *Given an $\varepsilon \in (0, 1)$, we choose $\{\beta_k\}$, $\{\rho_k\}$, and $\{\bar{\varepsilon}_k\}$ in Algorithm 4 as the same as those in Theorem 5.4.*

Notation and some uniform bounds. To facilitate our analysis, we first give some notation used in this subsection. Given K in (5.15) and D_0 in (5.16), we let

$$(5.27) \quad \begin{aligned} \underline{\rho} &= \rho_{K-1}, \bar{\beta} = \beta_{K-1}, B_{\mathbf{x}} = \frac{2\bar{\varepsilon}(\sigma^K - 1)}{\rho_0(\sigma - 1)} + \frac{D_0}{\sqrt{\beta_0 \rho_0}}, \\ B_{\boldsymbol{\lambda}} &= \frac{2\bar{\varepsilon}\sqrt{\beta_0}(\sigma^K - 1)}{\sqrt{\rho_0}(\sigma - 1)} + D_0, \underline{\varepsilon} = \min \left\{ \bar{\varepsilon}, \sqrt{\frac{\rho_0}{20\sigma}} \sigma^{-K} \right\}. \end{aligned}$$

In order to apply Theorem 4.3 to the iRALM subproblem (5.25), we define

$$(5.28) \quad L_{\Psi_k} = L_f + \rho_k + \beta_k \|\mathbf{A}\|^2, \quad C_L^{(k)} = \frac{L_{\Psi_k}}{\sqrt{\underline{L}}} + \sqrt{\frac{L_{\Psi_k}}{\gamma_{\text{dec}}}}, \quad \forall k < K,$$

$$(5.29) \quad \kappa^{(k)} = \frac{L_f + \rho_k}{\gamma_{\text{dec}}(\mu + \rho_k)}, \quad S^{(k)} = \frac{\sqrt{\kappa^{(k)}}}{2(1-c)^2 \underline{L}} \sum_{t=0}^{\infty} \frac{\varepsilon_0^2}{(t+1)^2} < \infty, \quad \forall k < K, n$$

$$(5.30) \quad \psi_0^{(k)} = \Psi_k(\mathbf{x}^{(k)}) - \Psi_k^* + (1 - (1-c)\alpha_0) \frac{\gamma_0}{2} \|\mathbf{x}_*^{(k+1)} - \mathbf{x}^{(k)}\|^2 \quad \forall k < K.$$

⁷Again we assume $\underline{L} = \Theta(L_f)$ to simplify the results. The analysis holds for any $\underline{L} \leq L_f$.

with $\mathbf{x}_*^{(k+1)} = \arg \min_{\mathbf{x}} \Psi_k(\mathbf{x})$ and $\Psi_k^* = \min_{\mathbf{x}} \Psi_k(\mathbf{x})$. Moreover, we define

$$(5.31) \quad \begin{aligned} C_{\varepsilon_k}^{(k)} &= \max \left\{ \frac{\text{dist}(\mathbf{0}, \partial \Psi_k(\mathbf{x}^{(k)}))}{\varepsilon_0}, \frac{2}{\sqrt{1-c}} \right. \\ &\quad \left. + \frac{6(L_f + \rho_k)}{\varepsilon_0 \gamma_{\text{dec}} \sqrt{c(1-c)}} \sqrt{\frac{2(\psi_0^{(k)} + S^{(k)})}{\mu + \rho_k}} \left[\sqrt{\kappa^{(k)}} \ln \frac{2(\psi_0^{(k)} + S^{(k)})(C_L^{(k)})^2}{\varepsilon_k^2} \right] \right\}, \end{aligned}$$

where $c \in (0, 1)$ is the same as that in (4.5). Because $\underline{\rho} \leq \rho_k, \bar{\beta} \geq \beta_k \forall 0 \leq k < K$, the quantities defined below are respectively upper bounds of $\kappa^{(k)}$, $S^{(k)}$, L_{Ψ_k} , and $C_L^{(k)}$:

$$(5.32) \quad \bar{\kappa} = \frac{L_f + \underline{\rho}}{\gamma_{\text{dec}}(\mu + \underline{\rho})}, \quad \bar{S} = \frac{\sqrt{\bar{\kappa}}}{2(1-c)^2 \bar{L}} \sum_{t=0}^{\infty} \frac{\varepsilon_0^2}{(t+1)^2} < \infty,$$

$$(5.33) \quad \bar{L}_{\Psi} = L_f + \rho_0 + \bar{\beta} \|\mathbf{A}\|^2, \quad \bar{C}_L = \frac{\bar{L}_{\Psi}}{\sqrt{\bar{L}}} + \sqrt{\frac{\bar{L}_{\Psi}}{\gamma_{\text{dec}}}}.$$

By the above notation, we can show the following two lemmas.

LEMMA 5.5. *Suppose Setting 2 is adopted. It holds that $\rho_k \geq \underline{\rho}$ and $\beta_k \leq \bar{\beta} \forall 0 \leq k < K$. In addition, $\|\mathbf{x}^{(k)} - \mathbf{x}^*\| \leq B_{\mathbf{x}}$ and $\|\boldsymbol{\lambda}^{(k)} - \boldsymbol{\lambda}^*\| \leq B_{\boldsymbol{\lambda}}$ hold $\forall 0 \leq k \leq K$. Moreover, $\|\mathbf{x}_*^{(k+1)} - \mathbf{x}^*\| \leq B_{\mathbf{x}} \forall 0 \leq k < K$.*

Proof. It is trivial to show that $\rho_k \geq \underline{\rho}$ and $\beta_k \leq \bar{\beta} \forall 0 \leq k < K$. From (5.17) and the definition of $B_{\mathbf{x}}$ and $B_{\boldsymbol{\lambda}}$ in (5.27), we have $\|\mathbf{x}^{(k)} - \mathbf{x}^*\| \leq B_{\mathbf{x}}$ and $\|\boldsymbol{\lambda}^{(k)} - \boldsymbol{\lambda}^*\| \leq B_{\boldsymbol{\lambda}} \forall 0 \leq k \leq K$. Moreover, notice that the first inequality in (5.17) also applies to $\mathbf{x}_*^{(k+1)}$. Hence, we have

$$(5.34) \quad \|\mathbf{x}_*^{(k+1)} - \mathbf{x}^*\| \leq \frac{2\bar{\varepsilon}(\sigma^{k+1} - 1)}{\rho_0(\sigma - 1)} + \frac{D_0}{\sqrt{\beta_0 \rho_0}} \leq B_{\mathbf{x}} \quad \forall k < K.$$

This completes the proof. \square

LEMMA 5.6. *Let $\psi_0^{(k)}$ be defined in (5.30). Then for any $1 \leq k < K$,*

$$\psi_0^{(k)} \leq 2B_{\mathbf{x}}(1 + 2\rho_0 B_{\mathbf{x}} + \|\mathbf{A}\|(2\sigma B_{\boldsymbol{\lambda}} + B_{\boldsymbol{\lambda}} + \|\boldsymbol{\lambda}^*\|)) + (1 - (1-c)\alpha_0) \frac{\gamma_0 B_{\mathbf{x}}^2}{2}.$$

Proof. From (5.22a) and the definition of Ψ_k , it follows that

(5.35)

$$\begin{aligned} \text{dist}(\mathbf{0}, \partial \Psi_k(\mathbf{x}^{(k)})) &\leq \text{dist}(\mathbf{0}, \partial G(\mathbf{x}^{(k)}) + \mathbf{A}^{\top} \boldsymbol{\lambda}^{(k)}) \\ &\quad + \|\mathbf{A}_E^{\top}(\boldsymbol{\lambda}_E^{(k)} + \beta_k(\mathbf{A}_E \mathbf{x} - \mathbf{b}_E)) + \mathbf{A}_I^{\top}[\boldsymbol{\lambda}_I^{(k)} + \beta_k(\mathbf{A}_I \mathbf{x} - \mathbf{b}_I)]_+ - \mathbf{A}^{\top} \boldsymbol{\lambda}^{(k)}\| \\ &= \text{dist}(\mathbf{0}, \partial G(\mathbf{x}^{(k)})) \\ &\quad + \|\mathbf{A}^{\top} \boldsymbol{\lambda}^{(k)}\| + \|\mathbf{A}_E^{\top}(\beta_k(\mathbf{A}_E \mathbf{x} - \mathbf{b}_E)) + \mathbf{A}_I^{\top}([\boldsymbol{\lambda}_I^{(k)} + \beta_k(\mathbf{A}_I \mathbf{x} - \mathbf{b}_I)]_+ - \boldsymbol{\lambda}_I^{(k)})\| \\ &\leq \bar{\varepsilon}_{k-1} + \rho_{k-1} \|\mathbf{x}^{(k)} - \mathbf{x}^{(k-1)}\| \\ &\quad + \|\mathbf{A}\| \sqrt{\beta_k^2 \|\mathbf{A}_E \mathbf{x}^{(k)} - \mathbf{b}_E\|^2 + \|[\boldsymbol{\lambda}_I^{(k)} + \beta_k(\mathbf{A}_I \mathbf{x} - \mathbf{b}_I)]_+ - \boldsymbol{\lambda}_I^{(k)}\|^2} \\ &\leq \bar{\varepsilon}_{k-1} + \rho_{k-1} \|\mathbf{x}^{(k)} - \mathbf{x}^{(k-1)}\| \\ &\quad + \|\mathbf{A}\| \sqrt{\beta_k^2 \|\mathbf{A}_E \mathbf{x}^{(k)} - \mathbf{b}_E\|^2 + \beta_k^2 \|[\mathbf{A}_I \mathbf{x}^{(k)} - \mathbf{b}_I]_+\|^2 + \|\boldsymbol{\lambda}_I^{(k)}\|^2} \\ &\leq \bar{\varepsilon}_{k-1} + 2\rho_0 B_{\mathbf{x}} + \|\mathbf{A}\|(2\sigma B_{\boldsymbol{\lambda}} + B_{\boldsymbol{\lambda}} + \|\boldsymbol{\lambda}^*\|), \end{aligned}$$

where the third inequality is because of the facts that $\boldsymbol{\lambda}_I^{(k)} \geq \mathbf{0}$ and that $\|[\mathbf{x} + \mathbf{y}]_+ - \mathbf{y}\|^2 \leq \|[\mathbf{x}]_+\|^2 + \|\mathbf{y}\|^2 \forall \mathbf{y} \geq \mathbf{0}$, and the last inequality is because of Lemma 5.5, (5.22b), and the fact that $\sqrt{a+b} \leq \sqrt{a} + \sqrt{b} \forall a, b \geq 0$. The inequality in (5.35), together with the convexity of Ψ_k , Lemma 5.5, and (5.34), gives

$$\Psi_k(\mathbf{x}^{(k)}) - \Psi_k^* \leq 2B_{\mathbf{x}}(\bar{\varepsilon}_{k-1} + 2\rho_0 B_{\mathbf{x}} + \|\mathbf{A}\|(2\sigma B_{\boldsymbol{\lambda}} + B_{\boldsymbol{\lambda}} + \|\boldsymbol{\lambda}^*\|)) \quad \forall 1 \leq k < K.$$

The conclusion follows from the fact that $\bar{\varepsilon}_{k-1} \leq 1$ and $\|\mathbf{x}_*^{(k+1)} - \mathbf{x}^{(k)}\|^2 \leq B_{\mathbf{x}}^2$. \square

By Lemma 5.6, we can bound $\psi_0^{(k)}$ uniformly for $0 \leq k < K$ by the quantity

$$\bar{\psi}_0 := \max \left\{ \psi_0^{(0)}, 2B_{\mathbf{x}}(1 + 2\rho_0 B_{\mathbf{x}} + \|\mathbf{A}\|(2\sigma B_{\boldsymbol{\lambda}} + B_{\boldsymbol{\lambda}} + \|\boldsymbol{\lambda}^*\|)) + (1 - (1 - c)\alpha_0) \frac{\gamma_0 B_{\mathbf{x}}^2}{2} \right\}.$$

Now we are ready to show the overall oracle complexity of Algorithm 4.

THEOREM 5.7 (total oracle complexity to produce an ε -stationary point). *Suppose Settings 1 and 2 are adopted. Let K be given in (5.15). Algorithm 4 will stop and return an ε -stationary point of (5.1) after making Q_f queries to $(f, \nabla f)$ and $Q_{\mathbf{A}}$ queries to $(\mathbf{A}(\cdot), \mathbf{A}^{\top}(\cdot))$ with Q_f and $Q_{\mathbf{A}}$ given as follows. (i) When $\mu = 0$,*

$$(5.36) \quad Q_f = O \left(\left(K + \sqrt{\frac{L_f}{\rho_0}} \frac{\sigma^{K/2} - 1}{\sqrt{\sigma} - 1} \right) \ln \frac{\bar{C}_L^2(\bar{\psi}_0 + \bar{S})}{\varepsilon^2} \right),$$

$$(5.37) \quad Q_{\mathbf{A}} = O \left(\left(K + \sqrt{\frac{L_f}{\rho_0}} \frac{\sigma^{K/2} - 1}{\sqrt{\sigma} - 1} + \frac{\|\mathbf{A}\| \sqrt{\beta_0} \sigma^K - 1}{\sqrt{\rho_0} \sigma - 1} \right) \cdot K \cdot \ln \frac{\bar{C}_L^2(\bar{\psi}_0 + \bar{S})}{\varepsilon^2} \right);$$

and (ii) when $\mu > 0$,

$$(5.38) \quad Q_f = O \left(K \sqrt{\frac{L_f}{\mu}} \ln \frac{\bar{C}_L^2(\bar{\psi}_0 + \bar{S})}{\varepsilon^2} \right),$$

$$Q_{\mathbf{A}} = O \left(\left(K \sqrt{\frac{L_f}{\mu}} + \frac{\|\mathbf{A}\| \sqrt{\beta_0} \sigma^{K/2} - 1}{\sqrt{\mu} \sqrt{\sigma} - 1} \right) \cdot K \cdot \ln \frac{(\bar{\psi}_0 + \bar{S}) \bar{C}_L^2}{\varepsilon^2} \right).$$

Proof. By Theorem 5.4, we only need to bound the overall number of queries that are made to produce $\mathbf{x}^{(K)}$. From Theorem 4.3, we can find an $\bar{\varepsilon}_k$ -stationary point of Ψ_k in (5.25) by Algorithm 1 with $Q_f^{(k)}$ queries to $(f, \nabla f)$ and $Q_{\mathbf{A}}^{(k)}$ queries to $(\mathbf{A}(\cdot), \mathbf{A}^{\top}(\cdot))$, where

$$(5.39) \quad Q_f^{(k)} = O \left(\sqrt{\kappa^{(k)}} \ln \frac{(C_L^{(k)})^2(\psi_0^{(k)} + S^{(k)})}{\varepsilon^2} + \ln \frac{L_{\Psi_k}}{\mu + \rho_k} \right),$$

$$(5.40) \quad Q_{\mathbf{A}}^{(k)} = O \left(\sqrt{\frac{L_{\Psi_k}}{\mu + \rho_k}} \ln \left(\left(\frac{L_{\Psi_k}}{L_f + \rho_k} \right) C_{\bar{\varepsilon}_k}^{(k)} \right) \cdot \ln \frac{(C_L^{(k)})^2(\psi_0^{(k)} + S^{(k)})}{\varepsilon^2} \right).$$

In the two inequalities above, we have used the fact $\bar{\varepsilon}_k \geq \varepsilon$.

When $\mu = 0$, we have from (5.39), $\psi_0^{(k)} + S^{(k)} \leq \bar{\psi}_0 + \bar{S}$, and $C_L^{(k)} \leq \bar{C}_L$ that

$$Q_f = \sum_{k=0}^{K-1} Q_f^{(k)} = \sum_{k=0}^{K-1} O \left(\sqrt{\kappa^{(k)}} \ln \frac{\bar{C}_L^2(\bar{\psi}_0 + \bar{S})}{\varepsilon^2} + \ln \frac{L_{\Psi_k}}{\rho_k} \right)$$

$$\stackrel{(5.29)}{=} \sum_{k=0}^{K-1} O \left(\sqrt{1 + \frac{L_f \sigma^k}{\rho_0}} \ln \frac{\bar{C}_L^2(\bar{\psi}_0 + \bar{S})}{\varepsilon^2} \right),$$

which gives (5.36) by $\sum_{k=0}^{K-1} \sqrt{1 + \frac{L_f \sigma^k}{\rho_0}} \leq K + \sqrt{\frac{L_f}{\rho_0} \frac{\sigma^{K/2} - 1}{\sqrt{\sigma} - 1}}$. Also, it follows from (5.35) that $\text{dist}(\mathbf{0}, \partial\Psi_k(\mathbf{x}^{(k)})) \leq \bar{V}_\Psi$ for any $k \geq 0$ where

$$\bar{V}_\Psi = \max \left\{ \text{dist}(\mathbf{0}, \partial\mathcal{L}_{\beta_0}(\mathbf{x}^{(0)}, \boldsymbol{\lambda}^{(0)})), \bar{\varepsilon} + 2\rho_0 B_\mathbf{x} + \|\mathbf{A}\|(B_\boldsymbol{\lambda}(2\sigma + 1) + \|\boldsymbol{\lambda}^*\|) \right\} = O(1).$$

By the fact that $\text{dist}(\mathbf{0}, \partial\Psi_k(\mathbf{x}^{(k)})) = O(1)$, the definition of $\bar{\varepsilon}_k^2$ in (5.4), and the inequalities $C_L^{(k)} \leq \bar{C}_L$, $\kappa^{(k)} \leq \bar{\kappa}$, and $L_{\Psi_k} = L_f + \rho_k + \beta_k \|\mathbf{A}\|^2 \leq \bar{L}_\Psi$, we can derive from (5.31) that $\ln((\frac{L_{\Psi_k}}{L_f + \rho_k})C_{\bar{\varepsilon}_k}^{(k)}) = O(k)$. Thus, we have from (5.40) that

$$\begin{aligned} Q_{\mathbf{A}} &= \sum_{k=0}^{K-1} Q_{\mathbf{A}}^{(k)} = \sum_{k=0}^{K-1} O\left(\sqrt{\frac{L_{\Psi_k}}{\rho_k}} \ln\left(\left(\frac{L_{\Psi_k}}{L_f + \rho_k}\right)C_{\bar{\varepsilon}_k}^{(k)}\right) \cdot \ln \frac{(C_L^{(k)})^2(\psi_0^{(k)} + S^{(k)})}{\bar{\varepsilon}^2}\right) \\ &\leq \sum_{k=0}^{K-1} O\left(\sqrt{\frac{L_f + \rho_k + \beta_k \|\mathbf{A}\|^2}{\rho_k}} \ln\left(\left(\frac{L_f + \rho_k + \beta_k}{L_f + \rho_k}\right)C_{\bar{\varepsilon}_k}^{(k)}\right) \cdot \ln \frac{(\bar{\psi}_0 + \bar{S})\bar{C}_L^2}{\bar{\varepsilon}^2}\right) \\ &= O\left(\left(K + \sqrt{\frac{L_f}{\rho_0} \frac{\sigma^{K/2} - 1}{\sqrt{\sigma} - 1}} + \frac{\|\mathbf{A}\|\sqrt{\beta_0} \sigma^K - 1}{\sqrt{\rho_0} \sigma - 1}\right) \cdot K \cdot \ln \frac{(\bar{\psi}_0 + \bar{S})\bar{C}_L^2}{\bar{\varepsilon}^2}\right). \end{aligned}$$

When $\mu > 0$, we have

$$\begin{aligned} Q_f &= \sum_{k=0}^{K-1} Q_f^{(k)} = \sum_{k=0}^{K-1} O\left(\sqrt{\kappa^{(k)}} \ln \frac{\bar{C}_L^2(\bar{\psi}_0 + \bar{S})}{\bar{\varepsilon}^2} + \ln \frac{L_{\Psi_k}}{\mu + \rho_k}\right) \\ &= O\left(K \sqrt{\frac{L_f}{\mu}} \ln \frac{\bar{C}_L^2(\bar{\psi}_0 + \bar{S})}{\bar{\varepsilon}^2}\right), \end{aligned}$$

where $\kappa^{(k)} \leq \bar{\kappa} = O(\sqrt{L_f/\mu})$ according to (5.32). This gives (5.38). Also, we have

$$\begin{aligned} Q_{\mathbf{A}} &= \sum_{k=0}^{K-1} Q_{\mathbf{A}}^{(k)} = \sum_{k=0}^{K-1} O\left(\sqrt{\frac{L_{\Psi_k}}{\mu + \rho_k}} \ln\left(\left(\frac{L_{\Psi_k}}{L_f + \rho_k}\right)C_{\bar{\varepsilon}_k}^{(k)}\right) \cdot \ln \frac{(C_L^{(k)})^2(\psi_0^{(k)} + S^{(k)})}{\bar{\varepsilon}^2}\right) \\ &= \sum_{k=0}^{K-1} O\left(\sqrt{\frac{L_f + \rho_k + \beta_k \|\mathbf{A}\|^2}{\mu + \rho_k}} \ln\left(\left(\frac{L_f + \rho_k + \beta_k}{L_f + \rho_k}\right)C_{\bar{\varepsilon}_k}^{(k)}\right) \cdot \ln \frac{(\bar{\psi}_0 + \bar{S})\bar{C}_L^2}{\bar{\varepsilon}^2}\right) \\ &= O\left(\left(K \sqrt{\frac{L_f}{\mu}} + \frac{\|\mathbf{A}\|\sqrt{\beta_0} \sigma^{K/2} - 1}{\sqrt{\mu} \sqrt{\sigma} - 1}\right) \cdot K \cdot \ln \frac{(\bar{\psi}_0 + \bar{S})\bar{C}_L^2}{\bar{\varepsilon}^2}\right), \end{aligned}$$

where, again, we use the fact $\ln((\frac{L_{\Psi_k}}{L_f + \rho_k})C_{\bar{\varepsilon}_k}^{(k)}) = O(k)$ and the inequalities $C_L^{(k)} \leq \bar{C}_L$, $\kappa^{(k)} \leq \bar{\kappa}$, and $L_{\Psi_k} = L_f + \rho_k + \beta_k \|\mathbf{A}\|^2 \leq \bar{L}_\Psi$. This proves the case of $\mu > 0$. \square

Remark 1. Notice $K = O(\ln \frac{1}{\varepsilon})$ by (5.15), $\sigma^K = O(\frac{1}{\varepsilon})$, and $\bar{\varepsilon} = \Theta(\varepsilon)$. Hence, from Theorem 5.7, we have $Q_f = O(\sqrt{\frac{L_f}{\varepsilon} \ln \frac{1}{\varepsilon}})$ and $Q_{\mathbf{A}} = O((\sqrt{\frac{L_f}{\varepsilon}} + \frac{\|\mathbf{A}\|}{\varepsilon})(\ln \frac{1}{\varepsilon})^2)$ for the case of $\mu = 0$, and $Q_f = O(\sqrt{\frac{L_f}{\mu}} (\ln \frac{1}{\varepsilon})^2)$ and $Q_{\mathbf{A}} = O((\ln \frac{1}{\varepsilon} \sqrt{\frac{L_f}{\mu}} + \frac{\|\mathbf{A}\|}{\sqrt{\mu \varepsilon}})(\ln \frac{1}{\varepsilon})^2)$ for the case of $\mu > 0$. If $\rho_0 = O(\varepsilon)$ and $\beta_0 = O(\frac{1}{\varepsilon})$, then $K = O(1)$. For this setting, the factors $(\ln \frac{1}{\varepsilon})^2$ in Q_f and $Q_{\mathbf{A}}$ above will reduce to $\ln \frac{1}{\varepsilon}$. The choice of $\beta_k = \beta_0 \sigma^k$ and $\rho_k = \rho_0 \sigma^{-k}$ enables us to obtain the near-optimal complexity results. This is similar to the setting in [49, Theorem 5]. However, one potential drawback is that if iRALM does need to run to K outer iterations, then $\beta_K \rightarrow \infty$ and $\rho_K \rightarrow 0$ as $\varepsilon \rightarrow 0$ and thus the subproblem becomes ill-conditioned.

6. Smoothed bilinear saddle-point structured optimization. In this section, we consider the bilinear saddle-point structured optimization problem

$$(6.1) \quad p^* = \min_{\mathbf{x} \in \mathbb{R}^n} \left\{ p(\mathbf{x}) := f(\mathbf{x}) + r(\mathbf{x}) + \max_{\mathbf{y} \in \mathbb{R}^m} \{ \langle \mathbf{y}, \mathbf{A}\mathbf{x} \rangle - \phi(\mathbf{y}) \} \right\},$$

where $\mathbf{A} \in \mathbb{R}^{m \times n}$, f is smooth and convex, r and ϕ are closed convex and admit easy proximal mappings, and r allows easy computation of $\text{dist}(\mathbf{z}', \partial r(\mathbf{z}))$ for any $\mathbf{z}, \mathbf{z}' \in \mathbb{R}^n$. We assume that $(f, \nabla f)$ is significantly more expensive than $(\mathbf{A}(\cdot), \mathbf{A}^\top(\cdot))$ to evaluate. We adopt the following notation in this section:

$$(6.2a) \quad G(\mathbf{x}) := f(\mathbf{x}) + r(\mathbf{x}), \quad \bar{h}(\mathbf{x}) := \max_{\mathbf{y} \in \mathbb{R}^m} \{ \langle \mathbf{y}, \mathbf{A}\mathbf{x} \rangle - \phi(\mathbf{y}) \},$$

$$(6.2b) \quad \varphi(\mathbf{y}) := \min_{\mathbf{x} \in \mathbb{R}^n} \{ G(\mathbf{x}) + \langle \mathbf{y}, \mathbf{A}\mathbf{x} \rangle \}, \quad d(\mathbf{y}) := \varphi(\mathbf{y}) - \phi(\mathbf{y}).$$

We call $p(\mathbf{x}) - d(\mathbf{y})$ the *duality gap* at (\mathbf{x}, \mathbf{y}) which is nonnegative by the definition of p and d . A pair $(\mathbf{x}^*, \mathbf{y}^*)$ that satisfies $p(\mathbf{x}^*) = d(\mathbf{y}^*)$ or, equivalently, $\mathbf{0} \in \partial G(\mathbf{x}^*) + \mathbf{A}^\top \mathbf{y}^*$, $\mathbf{0} \in \mathbf{A}\mathbf{x}^* - \partial\phi(\mathbf{y}^*)$ is called a *saddle point* of (6.1). Apparently, $p^* = p(\mathbf{x}^*) = d(\mathbf{y}^*) = G(\mathbf{x}^*) + \langle \mathbf{y}^*, \mathbf{A}\mathbf{x}^* \rangle - \phi(\mathbf{y}^*)$. We make the following assumption on (6.1).

Assumption 1. Function f is L_f -smooth and μ -strongly convex with $\mu > 0$; $D_\phi := \max_{\mathbf{y}_1, \mathbf{y}_2 \in \text{dom}(\phi)} \|\mathbf{y}_1 - \mathbf{y}_2\| < \infty$; (6.1) has a saddle point $(\mathbf{x}^*, \mathbf{y}^*)$.

Our goal is to find an ε -stationary solution of (6.1) defined formally below.

DEFINITION 6.1. For $\varepsilon \geq 0$, a point $(\bar{\mathbf{x}}, \bar{\mathbf{y}})$ is an ε -stationary solution of (6.1) if

$$(6.3) \quad \text{dist}(\mathbf{0}, \partial G(\bar{\mathbf{x}}) + \mathbf{A}^\top \bar{\mathbf{y}}) \leq \varepsilon, \quad \text{dist}(\mathbf{0}, \mathbf{A}\bar{\mathbf{x}} - \partial\phi(\bar{\mathbf{y}})) \leq \varepsilon.$$

The following result shows the duality gap of an ε -stationary solution of (6.1).

THEOREM 6.2. Under Assumption 1, if $(\bar{\mathbf{x}}, \bar{\mathbf{y}})$ is an ε -stationary solution of (6.1), then $p(\bar{\mathbf{x}}) - d(\bar{\mathbf{y}}) \leq 2\varepsilon D_\phi + \frac{3\varepsilon^2}{2\mu}$.

Proof. Since $(\bar{\mathbf{x}}, \bar{\mathbf{y}})$ is an ε -stationary solution, there exist $\bar{\mathbf{u}} \in \partial G(\bar{\mathbf{x}}) + \mathbf{A}^\top \bar{\mathbf{y}}$ and $\bar{\mathbf{v}} \in \mathbf{A}\bar{\mathbf{x}} - \partial\phi(\bar{\mathbf{y}})$ such that $\|\bar{\mathbf{u}}\| \leq \varepsilon$ and $\|\bar{\mathbf{v}}\| \leq \varepsilon$. By the μ -strong convexity of G and the Young's inequality, it follows that

$$(6.4) \quad \begin{aligned} G(\bar{\mathbf{x}}) &\leq G(\mathbf{x}^*) + \langle \bar{\mathbf{u}} - \mathbf{A}^\top \bar{\mathbf{y}}, \bar{\mathbf{x}} - \mathbf{x}^* \rangle - \frac{\mu}{2} \|\bar{\mathbf{x}} - \mathbf{x}^*\|^2 \\ &= G(\mathbf{x}^*) + \langle \bar{\mathbf{u}}, \bar{\mathbf{x}} - \mathbf{x}^* \rangle - \langle \bar{\mathbf{y}}, \mathbf{A}\bar{\mathbf{x}} - \mathbf{A}\mathbf{x}^* \rangle - \frac{\mu}{2} \|\bar{\mathbf{x}} - \mathbf{x}^*\|^2 \\ &\leq G(\mathbf{x}^*) - \langle \bar{\mathbf{y}}, \mathbf{A}\bar{\mathbf{x}} - \mathbf{A}\mathbf{x}^* \rangle + \frac{1}{2\mu} \|\bar{\mathbf{u}}\|^2. \end{aligned}$$

In addition, by the convexity of ϕ and the definition of \bar{h} in (6.2a), we have $\bar{h}(\bar{\mathbf{x}}) + \langle \bar{\mathbf{v}}, \bar{\mathbf{y}} - \hat{\mathbf{y}} \rangle \leq \langle \bar{\mathbf{y}}, \mathbf{A}\bar{\mathbf{x}} \rangle - \phi(\bar{\mathbf{y}})$, where $\hat{\mathbf{y}} \in \arg \max_{\mathbf{y}} \{ \langle \mathbf{y}, \mathbf{A}\bar{\mathbf{x}} \rangle - \phi(\mathbf{y}) \}$. Adding this inequality to (6.4) gives

$$(6.5) \quad \begin{aligned} p(\bar{\mathbf{x}}) + \langle \bar{\mathbf{v}}, \bar{\mathbf{y}} - \hat{\mathbf{y}} \rangle &\leq G(\mathbf{x}^*) + \langle \bar{\mathbf{y}}, \mathbf{A}\mathbf{x}^* \rangle - \phi(\bar{\mathbf{y}}) + \frac{1}{2\mu} \|\bar{\mathbf{u}}\|^2 \\ &= p(\mathbf{x}^*) + \langle \bar{\mathbf{y}} - \mathbf{y}^*, \mathbf{A}\mathbf{x}^* \rangle + \phi(\mathbf{y}^*) - \phi(\bar{\mathbf{y}}) + \frac{1}{2\mu} \|\bar{\mathbf{u}}\|^2, \end{aligned}$$

where the equality holds because $(\mathbf{x}^*, \mathbf{y}^*)$ is a saddle point of (6.1). Now from the convexity of ϕ and the fact $\mathbf{A}\mathbf{x}^* \in \partial\phi(\mathbf{y}^*)$, it follows that $\langle \bar{\mathbf{y}} - \mathbf{y}^*, \mathbf{A}\mathbf{x}^* \rangle + \phi(\mathbf{y}^*) - \phi(\bar{\mathbf{y}}) \leq 0$. Hence, we have from (6.5) and the Cauchy–Schwarz inequality that

$$(6.6) \quad p(\bar{\mathbf{x}}) \leq p(\mathbf{x}^*) - \langle \bar{\mathbf{v}}, \bar{\mathbf{y}} - \hat{\mathbf{y}} \rangle + \frac{1}{2\mu} \|\bar{\mathbf{u}}\|^2 \leq p(\mathbf{x}^*) + \varepsilon D_\phi + \frac{\varepsilon^2}{2\mu}.$$

Similarly, from the convexity of ϕ and $\bar{\mathbf{v}} \in \mathbf{A}\bar{\mathbf{x}} - \partial\phi(\bar{\mathbf{y}})$, it follows that

$$(6.7) \quad -\phi(\bar{\mathbf{y}}) \geq -\phi(\mathbf{y}^*) + \langle \bar{\mathbf{v}} - \mathbf{A}\bar{\mathbf{x}}, \bar{\mathbf{y}} - \mathbf{y}^* \rangle.$$

In addition, by the definition of φ in (6.2) and the fact $\bar{\mathbf{u}} \in \partial G(\bar{\mathbf{x}}) + \mathbf{A}^\top \bar{\mathbf{y}}$, we have $\varphi(\bar{\mathbf{y}}) + \langle \bar{\mathbf{u}}, \bar{\mathbf{x}} - \hat{\mathbf{x}} \rangle \geq G(\bar{\mathbf{x}}) + \langle \bar{\mathbf{y}}, \mathbf{A}\bar{\mathbf{x}} \rangle$, where $\hat{\mathbf{x}} = \arg \min_{\mathbf{x}} \{G(\mathbf{x}) + \langle \bar{\mathbf{y}}, \mathbf{A}\mathbf{x} \rangle\}$. Adding this inequality to (6.7) and using the fact that $p^* = G(\mathbf{x}^*) + \langle \mathbf{y}^*, \mathbf{A}\mathbf{x}^* \rangle - \phi(\mathbf{y}^*)$ yield

$$(6.8) \quad \begin{aligned} d(\bar{\mathbf{y}}) + \langle \bar{\mathbf{u}}, \bar{\mathbf{x}} - \hat{\mathbf{x}} \rangle &\geq G(\bar{\mathbf{x}}) + \langle \mathbf{y}^*, \mathbf{A}\bar{\mathbf{x}} \rangle - \phi(\mathbf{y}^*) + \langle \bar{\mathbf{v}}, \bar{\mathbf{y}} - \mathbf{y}^* \rangle \\ &= p^* - p(\mathbf{x}^*) + G(\bar{\mathbf{x}}) + \langle \mathbf{y}^*, \mathbf{A}\bar{\mathbf{x}} \rangle - \phi(\mathbf{y}^*) + \langle \bar{\mathbf{v}}, \bar{\mathbf{y}} - \mathbf{y}^* \rangle \\ &= p^* - G(\mathbf{x}^*) + G(\bar{\mathbf{x}}) + \langle \mathbf{y}^*, \mathbf{A}\bar{\mathbf{x}} - \mathbf{A}\mathbf{x}^* \rangle + \langle \bar{\mathbf{v}}, \bar{\mathbf{y}} - \mathbf{y}^* \rangle. \end{aligned}$$

Notice $-\mathbf{A}^\top \mathbf{y}^* \in \partial G(\mathbf{x}^*)$. By the convexity of G , we have $-G(\mathbf{x}^*) + G(\bar{\mathbf{x}}) + \langle \mathbf{y}^*, \mathbf{A}\bar{\mathbf{x}} - \mathbf{A}\mathbf{x}^* \rangle \geq 0$. Hence, (6.8) and the Cauchy–Schwarz inequality together imply

$$(6.9) \quad d(\bar{\mathbf{y}}) + \langle \bar{\mathbf{u}}, \bar{\mathbf{x}} - \hat{\mathbf{x}} \rangle \geq p^* + \langle \bar{\mathbf{v}}, \bar{\mathbf{y}} - \mathbf{y}^* \rangle \geq p^* - \varepsilon D_y.$$

Moreover, from $\bar{\mathbf{u}} \in \partial G(\bar{\mathbf{x}}) + \mathbf{A}^\top \bar{\mathbf{y}}$ and $\mathbf{0} \in \partial G(\hat{\mathbf{x}}) + \mathbf{A}^\top \bar{\mathbf{y}}$ together with the μ -strong convexity of G , it holds that $\langle \bar{\mathbf{u}}, \bar{\mathbf{x}} - \hat{\mathbf{x}} \rangle \geq \mu \|\bar{\mathbf{x}} - \hat{\mathbf{x}}\|^2$. Hence, by the Cauchy–Schwarz inequality, we have $\|\bar{\mathbf{x}} - \hat{\mathbf{x}}\| \leq \frac{\|\bar{\mathbf{u}}\|}{\mu}$ and $\langle \bar{\mathbf{u}}, \bar{\mathbf{x}} - \hat{\mathbf{x}} \rangle \leq \frac{\|\bar{\mathbf{u}}\|^2}{\mu} \leq \frac{\varepsilon^2}{\mu}$, which together with (6.9) gives $d(\bar{\mathbf{y}}) \geq p^* - \varepsilon D_\phi - \frac{\varepsilon^2}{\mu}$. Therefore, from (6.6), we conclude that $p(\bar{\mathbf{x}}) - d(\bar{\mathbf{y}}) \leq 2\varepsilon D_\phi + \frac{3\varepsilon^2}{2\mu}$. This completes the proof. \square

Remark 2. By Theorem 6.2, to produce a primal-dual solution of (6.1) with a duality gap at most $\varepsilon > 0$, it suffices to find a $\min \left\{ \frac{\varepsilon}{4D_\phi}, \sqrt{\frac{4\mu\varepsilon}{3}} \right\}$ -stationary solution.

When ϕ is convex but not strongly convex, \bar{h} can be nonsmooth. In this case, [54] introduces a smoothing technique and solves an approximation of (6.1) as follows:

$$(6.10) \quad p_\rho^* = \min_{\mathbf{x} \in \mathbb{R}^n} \{p_\rho(\mathbf{x}) := f(\mathbf{x}) + r(\mathbf{x}) + h_\rho(\mathbf{x})\},$$

where $\rho > 0$ is the smoothing parameter, and h_ρ is defined by

$$(6.11) \quad h_\rho(\mathbf{x}) = \max_{\mathbf{y} \in \mathbb{R}^m} \{ \langle \mathbf{y}, \mathbf{A}\mathbf{x} \rangle - \phi(\mathbf{y}) - \frac{\rho}{2} \|\mathbf{y} - \mathbf{y}^{(0)}\|^2 \}$$

with any $\mathbf{y}^{(0)} \in \text{dom}(\phi)$. The result below is from [54, Theorem 1].

LEMMA 6.3. h_ρ defined in (6.11) is $\frac{\|\mathbf{A}\|^2}{\rho}$ -smooth and $\nabla h_\rho(\mathbf{x}) = \mathbf{A}^\top \mathbf{y}(\mathbf{x})$, where

$$(6.12) \quad \mathbf{y}(\mathbf{x}) = \arg \max_{\mathbf{y} \in \mathbb{R}^m} \left\{ \langle \mathbf{y}, \mathbf{A}\mathbf{x} \rangle - \phi(\mathbf{y}) - \frac{\rho}{2} \|\mathbf{y} - \mathbf{y}^{(0)}\|^2 \right\} = \text{prox}_{\phi/\rho} \left(\mathbf{y}^{(0)} + \frac{1}{\rho} \mathbf{A}\mathbf{x} \right).$$

Lemma 6.3 implies that (6.10) is an instance of (1.1) with $g = f$ and $h = h_\rho$. This means we can compute an ε -stationary point of (6.10) by calling the iAPG method in Algorithm 1. We present this approach in Algorithm 5.

By Lemma 6.3 and Theorem 4.3, we have the following complexity result.

Algorithm 5 Smoothing iAPG method for (6.1).

Inputs: $\mathbf{x}^{(0)} \in \text{dom}(r)$, $\mathbf{y}^{(0)} \in \text{dom}(\phi)$, $\rho > 0$, $\eta_{-1} \leq \frac{1}{L}$, $\gamma_0 \in [\mu, 1/\eta_{-1}]$, $L \in [\mu, L_f]$, and $\varepsilon > 0$

Compute:

$$(6.13) \quad \bar{\mathbf{x}} = \text{iAPG} \left(f, h_\rho, r, \mathbf{x}^{(0)}, \eta_{-1}, \gamma_0, \mu, \underline{L}, \{\varepsilon_k\}_{k \geq 0}, \varepsilon \right),$$

where h_ρ is defined in (6.11) and ε_k is defined as in (4.5) for $k \geq 0$.

Return: $\bar{\mathbf{x}}$ and $\bar{\mathbf{y}} = \mathbf{y}(\bar{\mathbf{x}})$, where $\mathbf{y}(\cdot)$ is defined in (6.12).

THEOREM 6.4 (oracle complexity to produce an ε -stationary solution). *Suppose Assumption 1 holds and $(\bar{\mathbf{x}}, \bar{\mathbf{y}})$ is returned by Algorithm 5 with $\rho = \frac{\varepsilon}{D_\phi}$. Then $(\bar{\mathbf{x}}, \bar{\mathbf{y}})$ is an ε -stationary solution of (6.1). Moreover, if $\underline{L} = \Theta(L_f)$, Algorithm 5 produces $(\bar{\mathbf{x}}, \bar{\mathbf{y}})$ using at most K_{sp} queries to $(f, \nabla f)$ and T_{sp} queries to $(\mathbf{A}(\cdot), \mathbf{A}^\top(\cdot))$, where*

$$(6.14) \quad K_{\text{sp}} = O \left(\sqrt{\kappa_f} \ln \frac{C_L^2(\psi_0 + S_f)}{\varepsilon^2} \right), \quad T_{\text{sp}} = O \left(\left(\sqrt{\kappa_f} + \frac{\|\mathbf{A}\|}{\sqrt{\varepsilon\mu}} \right) \ln \left(\frac{1}{\varepsilon} \right) \ln \frac{C_L^2(\psi_0 + S)}{\varepsilon^2} \right).$$

Here, $\kappa_f = \frac{L_f}{\gamma_{\text{dec}}\mu}$, S_f is the same as S in (4.6) except that κ is replaced by κ_f ,

$$\begin{aligned} \psi_0 &= p_\rho(\mathbf{x}^{(0)}) - p_\rho^* + (1 - (1 - c)\alpha_0) \frac{\gamma_0}{2} \|\mathbf{x}_\rho^* - \mathbf{x}^{(0)}\|^2, \\ C_L &= \frac{L_f + \frac{D_\phi \|\mathbf{A}\|^2}{\varepsilon}}{\sqrt{\underline{L}}} + \sqrt{\frac{L_f + \frac{D_\phi \|\mathbf{A}\|^2}{\varepsilon}}{\gamma_{\text{dec}}}} \end{aligned}$$

with $\mathbf{x}_\rho^* = \arg \min_{\mathbf{x}} p_\rho(\mathbf{x})$, $p_\rho^* = \min_{\mathbf{x}} p_\rho(\mathbf{x}) = p_\rho(\mathbf{x}_\rho^*)$, and $c \in (0, 1)$.

Proof. Suppose that $\bar{\mathbf{x}}$ is an ε -stationary point of p_ρ , i.e., $\text{dist}(\mathbf{0}, \partial p_\rho(\bar{\mathbf{x}})) \leq \varepsilon$. Let $\bar{\mathbf{y}} = \mathbf{y}(\bar{\mathbf{x}})$. Then by Lemma 6.3, we have $\text{dist}(\mathbf{0}, \nabla g(\bar{\mathbf{x}}) + \partial r(\bar{\mathbf{x}}) + \mathbf{A}^\top \bar{\mathbf{y}}) \leq \varepsilon$. Also, notice $\mathbf{0} \in \mathbf{A}\bar{\mathbf{x}} - \partial\phi(\bar{\mathbf{y}}) - \rho(\bar{\mathbf{y}} - \mathbf{y}^{(0)})$. Thus $\text{dist}(\mathbf{0}, \mathbf{A}\bar{\mathbf{x}} - \partial\phi(\bar{\mathbf{y}})) \leq \rho \|\bar{\mathbf{y}} - \mathbf{y}^{(0)}\| \leq \rho D_\phi = \varepsilon$. Therefore, $(\bar{\mathbf{x}}, \bar{\mathbf{y}})$ is an ε -stationary solution of (6.1).

Applying Algorithm 1 to (6.10) by (6.13), the quantity C_ε in Theorem 4.3 becomes

$$\begin{aligned} C_\varepsilon &= \max \left\{ \frac{\text{dist}(\mathbf{0}, \partial p_\rho(\mathbf{x}^{(0)}))}{\varepsilon_0}, \frac{2}{1 - c} \right. \\ &\quad \left. + \frac{3L_f(2 - c)}{\varepsilon_0 \gamma_{\text{dec}} \sqrt{c(1 - c)}} \sqrt{\frac{2(\psi_0 + S_f)}{\mu}} \left\lceil \sqrt{\kappa_f} \ln \frac{2(\psi_0 + S_f) C_L^2}{\varepsilon^2} \right\rceil \right\}. \end{aligned}$$

Now, first notice that by Lemma 6.3, querying ∇h_ρ once needs one query to $(\mathbf{A}(\cdot), \mathbf{A}^\top(\cdot))$. Second, by Lemma 6.3 and the boundedness of $\text{dom}(\phi)$, we have $\text{dist}(\mathbf{0}, \partial p_\rho(\mathbf{x}^{(0)})) = O(1)$. Also, by the μ -strong convexity of p_ρ , it follows that

$$\begin{aligned} \psi_0 &\leq \left[1 + (1 - (1 - c)\alpha_0) \frac{\gamma_0}{\mu} \right] (p_\rho(\mathbf{x}^{(0)}) - p_\rho^*) \\ &\leq \left[1 + (1 - (1 - c)\alpha_0) \frac{\gamma_0}{\mu} \right] (p(\mathbf{x}^{(0)}) - p^* + \frac{\rho}{2} D_\phi^2) = O(1). \end{aligned}$$

Hence, $\ln((1 + \frac{\|\mathbf{A}\|^2}{L_f \rho}) C_\varepsilon) = O(\ln(\frac{1}{\varepsilon}))$. Third, the smoothness constant of h_ρ is $\frac{\|\mathbf{A}\|^2}{\rho} = \frac{D_\phi \|\mathbf{A}\|^2}{\varepsilon}$. Therefore we obtain the bounds on K_{sp} and T_{sp} from Theorem 4.3. \square

Remark 3. Since $\psi_0 = O(1)$, $S_f = O(1)$ and $C_L = O(\frac{\|\mathbf{A}\|^2}{\varepsilon})$, according to Theorem 6.4, we have $Q_f = O(\sqrt{\frac{L_f}{\mu}} \ln(\frac{1}{\varepsilon}))$ and $Q_{\mathbf{A}} = O((\sqrt{\frac{L_f}{\mu}} + \frac{\|\mathbf{A}\|}{\sqrt{\varepsilon\mu}}) \ln^2(\frac{1}{\varepsilon}))$.

7. Experimental results. In this section, we demonstrate the practical performance of the proposed algorithms. All the tests were conducted with MATLAB 2021a on a Windows machine with 10 CPU cores and 128 GB memory.

7.1. Multitask learning. We first test the iAPG on multitask learning [15] and compare it to the exact counterpart. Given m binary-class datasets $\mathcal{D}_l = \{(\mathbf{x}_{l,i}, y_{l,i})\}_{i=1}^{N_l}$, $l = 1, \dots, m$, with $\mathbf{x}_{l,i} \in \mathbb{R}^n$ and the corresponding label $y_{l,i} \in \{+1, -1\}$ for each l and i , we solve the multitask logistic regression [19] and use the regularizer given in [15, equation (23)] together with an ℓ_1 term:

$$\begin{aligned} \min_{\mathbf{W}} & \underbrace{\sum_{l=1}^m \frac{1}{N_l} \sum_{i=1}^{N_l} \ln(1 + \exp(-y_{l,i} \mathbf{w}_l^\top \mathbf{x}_{l,i}))}_{g(\mathbf{W})} + \frac{\mu}{2} \|\mathbf{W}\|_F^2 \\ & + \underbrace{\frac{\lambda_1}{2} \left\| \mathbf{W} - \frac{1}{m} \mathbf{W} \mathbf{1} \mathbf{1}^\top \right\|_F^2}_{h(\mathbf{W})} + \underbrace{\lambda_2 \|\mathbf{W}\|_1}_{r(\mathbf{W})}, \end{aligned}$$

where $\|\mathbf{W}\|_1 = \sum_{i,j} |w_{i,j}|$ and \mathbf{w}_l is the l th of \mathbf{W} and the classifier for task l .

In the experiments, we fixed $\lambda_2 = 10^{-3}$ and chose $\mu \in \{0.01, 0.1\}$ and $\lambda_1 \in \{1, 10, 100\}$. A larger value of λ_1 leads to a stronger correlation between the m classifiers and a larger smoothness constant of h . We randomly generated $m = 4$ binary-class datasets as in [77]. For each $l = 1, \dots, m$, every positive sample follows the Gaussian distribution $\mathcal{N}(\boldsymbol{\mu}_l, \boldsymbol{\Sigma})$ and negative sample following $\mathcal{N}(-\boldsymbol{\mu}_l, \boldsymbol{\Sigma})$ with

$$\boldsymbol{\Sigma} = \begin{bmatrix} \rho \mathbf{1}_{s \times s} + (1 - \rho) \mathbf{I}_{s \times s} & \mathbf{0}_{s \times (n-s)} \\ \mathbf{0}_{(n-s) \times s} & \mathbf{I}_{(n-s) \times (n-s)} \end{bmatrix}, \quad \boldsymbol{\mu}_l = \begin{bmatrix} \mathbf{1}_s \\ \mathbf{0}_{n-s} \end{bmatrix} + \mathbf{d}_l,$$

where the entries of \mathbf{d}_l follow the uniform distribution on $[\frac{1}{2}, 1]$. We set $n = 200$, $N_l = 500$, $\forall l$ or $n = 2000$, $N_l = 5000$, $\forall l$. For each combination of (μ, λ_1, n, N_l) , we conducted 10 independent trials. Since the smoothness constants of g and h can be computed explicitly, we also tested the methods without line search. We terminated the tested method once it produced an ε -stationary point $\bar{\mathbf{W}}$, i.e., $\text{dist}(\mathbf{0}, \partial F(\bar{\mathbf{W}})) \leq \varepsilon$, and $\varepsilon = 10^{-6}$ was set. For both iAPG and APG, we set $\gamma_{\text{inc}} = 2$ and $\gamma_{\text{dec}} = \frac{1}{2}$ as in Algorithm 2 if line search is adopted. In addition, for iAPG, the initial inexactness $\varepsilon_0 = 10^{-3}$ was set. The results are shown in Table 1. Here, $\#g$ represents the number⁸ of calls to g or ∇g , $\#h$ is the number of calls to h or ∇h , stat.viol. denotes $\text{dist}(\mathbf{0}, \partial F(\bar{\mathbf{W}}))$, and the time is in seconds. From the results, we see that the proposed iAPG requires smaller $\#g$ than the exact APG in all cases. Though iAPG has larger $\#h$ than APG, the former takes less time and thus is more efficient. The advantage of iAPG over APG becomes more significant as the problem becomes more difficult, i.e., when μ is smaller and/or λ_1 is bigger. These verify our theoretical results. In addition, even without knowing the smoothness constants, the iAPG by line search has a similar performance to that using the smoothness constants.

⁸We increase $\#g$ by one if g or ∇g or $(g, \nabla g)$ is called. The same rule is adopted for $\#h$.

TABLE 1

Results by the proposed iAPG method (i.e., Algorithm 1) and its exact counterpart APG on solving 10 independent random instances of the regularized multitask logistic regression with different sizes and model parameters. The numbers in the parentheses are the standard deviations.

(μ, λ_1)	iAPG no line search				iAPG with line search				APG no line search				APG with line search					
	# g	# h	stat.	viol.	time	# g	# h	stat.	viol.	time	# (g, h)	stat.	viol.	time	# (g, h)	stat.	viol.	time
Problem size: $n = 200, N_l = 500$ for each $l = 1, \dots, 4$																		
(0.1, 1)	37(0.0)	546(4.1)	7.4e-7(7.9e-8)	0.03	46(4.0)	850(66.6)	7.0e-7(2.1e-7)	0.04	103(0.0)	8.0e-7(3.0e-8)	0.04	158(4.2)	7.5e-7(1.7e-7)	0.05				
(0.1, 10)	37(0.0)	1815(7.4)	7.3e-7(7.7e-8)	0.03	47(2.6)	2209(104.9)	7.0e-7(2.4e-7)	0.04	322(1.0)	9.5e-7(3.1e-8)	0.09	604(4.4)	8.6e-7(9.5e-8)	0.15				
(0.1, 100)	37(0.0)	5946(37.0)	7.7e-7(6.7e-8)	0.06	48(2.1)	5226(298.4)	5.3e-7(2.6e-7)	0.06	1038(4.1)	9.8e-7(9.0e-9)	0.27	1584(6.0)	9.7e-7(1.3e-8)	0.38				
(0.01, 1)	106(1.1)	1806(13.7)	8.9e-7(7.4e-8)	0.05	106(0.9)	2313(24.7)	8.8e-7(8.7e-8)	0.06	288(1.0)	9.6e-7(2.3e-8)	0.08	404(1.2)	9.2e-7(6.0e-8)	0.10				
(0.01, 10)	106(1.0)	6023(76.8)	8.6e-7(6.5e-8)	0.08	106(0.9)	5727(211.3)	8.9e-7(7.6e-8)	0.08	874(4.2)	9.8e-7(1.1e-8)	0.22	1643(10.0)	9.6e-7(3.1e-8)	0.39				
(0.01, 100)	107(0.8)	19666(189.9)	8.6e-7(5.5e-8)	0.16	107(1.4)	13381(430.9)	8.6e-7(1.1e-7)	0.13	2775(13.4)	1.0e-6(3.2e-9)	0.71	4248(22.9)	9.9e-7(8.6e-9)	1.02				
Problem size: $n = 2000, N_l = 5000$ for each $l = 1, \dots, 4$																		
(0.1, 1)	31(0.0)	561(0.6)	5.3e-7(2.1e-8)	4.5	38(4.9)	869(113.3)	3.4e-7(2.8e-7)	4.7	105(0.0)	8.5e-7(1.9e-8)	6.4	165(1.7)	7.5e-7(9.5e-8)	7.9				
(0.1, 10)	31(0.0)	1870(5.6)	5.5e-7(2.0e-8)	4.5	41(4.9)	2149(245.9)	6.8e-7(3.4e-7)	4.8	341(0.6)	9.6e-7(1.6e-8)	12.4	647(0.0)	8.2e-7(2.1e-8)	20.0				
(0.1, 100)	31(0.0)	6102(17.2)	5.6e-7(1.6e-8)	4.9	41(6.1)	5103(854.0)	4.5e-7(3.5e-7)	5.0	1107(2.1)	9.8e-7(1.0e-8)	32.0	1728(8.2)	9.5e-7(3.7e-8)	47.2				
(0.01, 1)	91(0.6)	2131(12.5)	7.9e-7(6.4e-8)	6.1	88(0.0)	2612(7.3)	7.0e-7(1.8e-8)	6.0	319(0.8)	9.7e-7(2.8e-8)	11.8	496(2.2)	9.2e-7(5.7e-8)	16.2				
(0.01, 10)	91(0.0)	7099(17.0)	7.6e-7(2.1e-8)	6.4	88(0.0)	7096(183.2)	7.0e-7(2.2e-8)	6.3	999(0.8)	9.9e-7(1.0e-8)	29.2	1903(4.6)	9.7e-7(1.9e-8)	51.7				
(0.01, 100)	91(0.0)	23013(41.8)	7.6e-7(1.3e-8)	7.3	88(0.0)	18142(281.0)	7.0e-7(1.5e-8)	6.9	3183(4.0)	9.9e-7(1.6e-9)	85.1	4975(14.5)	9.9e-7(1.2e-8)	129.1				

TABLE 2

Results by the iRALM'iAPG method with and without line search and the APD method in [20] with and without line search on solving 10 independent random instances (7.1) with $m = 2000$ and $n = 5000$. The numbers in the parentheses are the standard deviations.

Method	#query_obj	#query_cstr	pres	dres	time
iRALM.iAPG no line search	2521(286.3)	21098(4723.5)	3.0e-7(2.9e-7)	6.2e-8(2.1e-10)	18.2
iRALM.iAPG with line search	2962(347.0)	9760(1200.6)	3.0e-7(2.9e-7)	5.2e-8(9.0e-9)	17.0
APD no line search		7929(606.7)	8.8e-10(1.1e-9)	3.0e-7(2.8e-7)	51.4
APD with line search		4349(334.8)	1.8e-7(2.2e-7)	2.9e-7(2.8e-7)	55.3

7.2. Zero-sum constrained LASSO. In this subsection, we test the iRALM in Algorithm 4 with iAPG as a subroutine, on the zero-sum constrained LASSO [17, 30]:

$$(7.1) \quad \min_{\mathbf{x}} \frac{1}{2} \|\mathbf{Ax} - \mathbf{b}\|^2 + \lambda \|\mathbf{x}\|_1 \text{ s.t. } \frac{1}{\sqrt{n}} \sum_{i=1}^n x_i = 0.$$

Here, $\mathbf{A} \in \mathbb{R}^{m \times n}$ and $\mathbf{b} \in \mathbb{R}^m$, and we divide by \sqrt{n} in the constraint to normalize the coefficient vector. We name the proposed method as iRALM'iAPG and compare it to the accelerated primal-dual method, called APD, in [20]. To apply APD, we solve an equivalent min-max problem by the ordinary Lagrangian function of (7.1). For iRALM'iAPG,⁹ we set in Algorithm 4 $\beta_k = \beta_0 \sigma^k, \rho_k = \rho_0 \sigma^{-k}$ with $\beta_0 = 1, \rho_0 = 10^{-3}, \sigma = 3$, and $\varepsilon_0 = 10^{-5}, \gamma_{\text{inc}} = 3, \gamma_{\text{dec}} = \frac{1}{2}$ in Algorithm 2 if line search is adopted. We set $\tau_0 = 1$ and $\gamma_0 = 10^{-3}$ for APD if line search is adopted; see Algorithm 2.3 in [20].

In the tests, we set $m = 2000, n = 5000$ and fixed $\lambda = 10^{-3}$ in (7.1). Each row of \mathbf{A} took the form of $\frac{\mathbf{a}}{\|\mathbf{a}\|}$, where \mathbf{a} was generated by the standard Gaussian distribution. We generated a zero-sum sparse vector \mathbf{x}^o with 200 nonzero components, whose locations were selected uniformly at random. Then we let $\mathbf{b} = \mathbf{Ax}^o + 10^{-3} \frac{\xi}{\|\mathbf{Ax}^o\|}$ with ξ generated from the standard Gaussian distribution. The stopping tolerance was set to $\varepsilon = 10^{-6}$ to produce an ε -stationary point. We conducted 10 independent runs. The results are reported in Table 2, where the methods without line search used explicitly computed smoothness constants to set a constant stepsize. The quantity #query'obj denotes the number of queries to $(\mathbf{A}, \mathbf{A}^\top)$ and #query'cstr the number

⁹A comparison to iRALM with the exact APG as a subroutine can be found in the longer arXiv version [48] of this paper.

TABLE 3

Results by the proposed iRALM'iAPG, the APD in [20], and the PDS in [41] on solving instances of the portfolio optimization (7.2) with NASDAQ data.

	Method	#query_obj	#query_cstr	pres	dres	cmpl	time
$\mu = 0$	iRALM.iAPG no line search	112704	5530144	0.0e+00	4.2e-07	9.2e-19	350.6
	iRALM.iAPG with line search	37235	715328	0.0e+00	4.2e-07	0.0e+00	97.0
	APD no line search		1118808	0.0e+00	1.3e-06	2.2e-17	3603.8
	PDS	54058	176318909	3.5e-18	1.1e-06	1.5e-25	3604.0
$\mu = 10^{-3}$	iRALM.iAPG no line search	21314	375994	0.0e+00	2.3e-07	7.0e-14	54.1
	iRALM.iAPG with line search	48643	117194	0.0e+00	2.3e-07	7.1e-14	108.4
	APD no line search		1119046	0.0e+00	8.5e-07	4.9e-18	3603.6
	PDS	6278	8927446	0.0e+00	2.2e-07	0.0e+00	195.5
$\mu = 0.1$	iRALM.iAPG no line search	3206	32178	4.4e-09	6.2e-08	4.8e-13	10.8
	iRALM.iAPG with line search	6601	16451	5.2e-09	6.2e-08	5.6e-13	17.8
	APD no line search		1119360	0.0e+00	9.0e-08	2.6e-21	3603.6
	PDS	1404	29512311	0.0e+00	5.6e-08	0.0e+00	591.7

of times the constraint function in (7.1) is evaluated. The quantities `pres` and `dres`, respectively, mean the violations of primal and dual feasibility in the KKT system. From the results, we see that the proposed method needs significantly less time than the APD method to produce comparable solutions. In addition, both methods with line search performed similarly as well as those without line search.

7.3. Portfolio optimization. In this subsection, we test the proposed method iRALM'iAPG on solving the portfolio optimization:

$$(7.2) \quad \min_{\mathbf{x}} \frac{1}{2} \mathbf{x}^\top \mathbf{Q} \mathbf{x} \text{ s.t. } \mathbf{x} \geq 0, \sum_{i=1}^n x_i \leq 1, \boldsymbol{\xi}^\top \mathbf{x} \geq c,$$

where $\boldsymbol{\xi}$ contains expected return rates of n assets, \mathbf{Q} is the covariance matrix of the return rates, and c is the minimum total return.

We solve instances of (7.2) with the real NASDAQ dataset¹⁰ [59], where $\boldsymbol{\xi}$ is the mean of 30-day return rates. The original covariance matrix $\mathbf{Q}_0 \in \mathbb{R}^{2730 \times 2730}$ is rank-deficient, and in (7.2), we set $\mathbf{Q} = \mathbf{Q}_0 + \mu \mathbf{I}$ with $\mu \in \{0, 10^{-3}, 0.1\}$. We set $c = 0.02$, a tolerance to $\varepsilon = 10^{-6}$, and also a maximum running time to one hour. We found that APD with line search did not work well for these instances, possibly because of the rounding error during the line search. Hence, we only reported its results without line search by explicitly computing the smoothness constants and setting constant stepsizes. The results by all methods are shown in Table 3, where `cmpl` represents the amount of violation of complementarity condition in the KKT system, and all other quantities have the same meanings as those in Table 2. From the results, we see that the proposed method iRALM'iAPG was significantly more efficient than APD and PDS in terms of running time. For the hardest case that corresponds to $\mu = 0$, APD and PDS both failed to reach the desired accuracy within one hour. PDS required many more queries to the constraint functions, though its queries to the objective were significantly fewer than the proposed method. This is because the inner loop of PDS needs to run to a theoretically determined maximum number of iterations rather than to a computationally checkable stopping condition.

8. Conclusions. We present an inexact accelerated proximal gradient (iAPG) method for composite convex optimization, which have two smooth components with significantly different computational costs. When the more costly component has

¹⁰More results on synthetic data can be found in the longer arXiv version [48] of this paper.

a significantly smaller smoothness constant than the less costly one, the proposed iAPG can significantly reduce the overall time complexity than its exact counterpart, by querying the more costly component less frequently than the less costly one. Using the iAPG as a subroutine, we proposed gradient-based methods for solving affine-constrained composite convex optimization and for solving bilinear saddle-point structured nonsmooth convex optimization. Our methods can have significantly lower time complexity than existing methods.

Appendix A. Technical lemmas. The following technical lemmas are needed in our convergence analysis. The first lemma below is obtained by applying inequality $\sqrt{a+b} \leq \sqrt{a} + \sqrt{b}$ for $a, b \geq 0$ to the conclusion of Lemma 1 in [63].

LEMMA A.1. *Let $\{u_k\}_{k \geq 1}$ be a sequence of nonnegative numbers. Suppose $u_k^2 \leq C + \sum_{i=1}^k \lambda_i u_i \forall k \geq 1$, where $C \geq 0$ is a constant and $\lambda_i \geq 0 \forall i \geq 1$. Then $u_k \leq \sum_{i=1}^k \lambda_i + \sqrt{C} \forall k \geq 1$.*

LEMMA A.2. *Let $\sigma > 1$ and $a \in (0, 1)$. If $b \geq \frac{64}{a^2(\ln \sigma)^4} \geq 1$, then $(\log_\sigma b)^2 \leq a \cdot b$.*

Proof. Let $\theta(x) = \frac{1}{2}(\ln x)^2 - x$. Then $\theta'(x) = \frac{1}{x} \ln x - 1$. Since $\ln x < x \forall x > 0$, we have $\theta'(x) < 0 \forall x > 0$, so θ is decreasing. Hence, $\theta(x) \leq \theta(1) < 0 \forall x \geq 1$, which implies $(\log_\sigma x^2)^2 \leq \frac{8x}{(\ln \sigma)^2} \forall x \geq 1$. Taking $x = \sqrt{b}$ gives $(\log_\sigma b)^2 \leq \frac{8\sqrt{b}}{(\ln \sigma)^2} \leq a \cdot b$, where the second inequality is by the assumption that $b \geq \frac{64}{a^2(\ln \sigma)^4}$. \square

REFERENCES

- [1] Z. ALLEN-ZHU AND E. HAZAN, *Optimal black-box reductions between optimization objectives*, Adv. Neural Inf. Process. Syst., 29 (2016), pp. 1614–1622.
- [2] A. BECK AND M. TEBBOULLE, *A fast iterative shrinkage-thresholding algorithm for linear inverse problems*, SIAM J. Imaging Sci., 2 (2009), pp. 183–202.
- [3] A. BECK AND M. TEBBOULLE, *Smoothing and first order methods: A unified framework*, SIAM J. Optim., 22 (2012), pp. 557–580.
- [4] Y. BELLO-CRUZ, M. L. GONÇALVES, AND N. KRISLOCK, *On Inexact Accelerated Proximal Gradient Methods with Relative Error Rules*, preprint, arXiv:2005.03766, 2020.
- [5] R. I. BOT, E. R. CSETNEK, AND D.-K. NGUYEN, *Fast Augmented Lagrangian Method in the Convex Regime with Convergence Guarantees for the Iterates*, arXiv:2111.09370, 2021.
- [6] K. BREDIES AND H. SUN, *Accelerated Douglas-Rachford Methods for the Solution of Convex-Concave Saddle-Point Problems*, preprint, arXiv:1604.06282, 2016.
- [7] E. J. CANDÈS, X. LI, Y. MA, AND J. WRIGHT, *Robust principal component analysis?*, J. ACM, 58 (2011), pp. 1–37.
- [8] E. J. CANDÈS AND B. RECHT, *Exact matrix completion via convex optimization*, Found. Comput. Math., 9 (2009), pp. 717–772.
- [9] A. CHAMBOLLE AND T. POCK, *A first-order primal-dual algorithm for convex problems with applications to imaging*, J. Math. Imaging Vision, 40 (2011), pp. 120–145.
- [10] X. CHEN, Q. LIN, S. KIM, J. G. CARBONELL, AND E. P. XING, *Smoothing proximal gradient method for general structured sparse regression*, Ann. Appl. Stat., 6 (2012), pp. 719–752.
- [11] X. CHEN, Q. LIN, AND B. SEN, *On degrees of freedom of projection estimators with applications to multivariate nonparametric regression*, J. Amer. Statist. Assoc., 115 (2020), pp. 173–186.
- [12] Y. CHEN, G. LAN, AND Y. OUYANG, *Optimal primal-dual methods for a class of saddle point problems*, SIAM J. Optim., 24 (2014), pp. 1779–1814.
- [13] Y. CHEN, G. LAN, AND Y. OUYANG, *Accelerated schemes for a class of variational inequalities*, Math. Program., 165 (2017), pp. 113–149.

- [14] D. DVINSKIKH AND A. GASNIKOV, *Decentralized and parallel primal and dual accelerated methods for stochastic convex programming problems*, J. Inverse Ill-Posed Probl., 29 (2021), pp. 385–405.
- [15] T. EVGENIOU, C. A. MICCHELLI, M. PONTIL, AND J. SHAWE-TAYLOR, *Learning multiple tasks with kernel methods*, J. Mach. Learn. Res., 6 (2005).
- [16] J. FRIEDMAN, T. HASTIE, AND R. TIBSHIRANI, *Sparse inverse covariance estimation with the graphical lasso*, Biostatistics, 9 (2008), pp. 432–441.
- [17] B. R. GAINES, J. KIM, AND H. ZHOU, *Algorithms for fitting the constrained lasso*, J. Comput. Graph. Statist., 27 (2018), pp. 861–871.
- [18] E. GORBUNOV, D. DVINSKIKH, AND A. GASNIKOV, *Optimal Decentralized Distributed Algorithms for Stochastic Convex Optimization*, preprint, arXiv:1911.07363, 2019.
- [19] X. GU, F.-L. CHUNG, H. ISHIBUCHI, AND S. WANG, *Multitask coupled logistic regression and its fast implementation for large multitask datasets*, IEEE Trans. Cybern., 45 (2014), pp. 1953–1966.
- [20] E. Y. HAMEDANI AND N. S. AYBAT, *A primal-dual algorithm with line search for general convex-concave saddle point problems*, SIAM J. Optim., 31 (2021), pp. 1299–1329.
- [21] B. HE, S. XU, AND J. YUAN, *Indefinite Linearized Augmented Lagrangian Method for Convex Programming with Linear Inequality Constraints*, preprint, arXiv:2105.02425, 2021.
- [22] B. HE AND X. YUAN, *On the acceleration of augmented lagrangian method for linearly constrained optimization*, Optimization, 3 (2010).
- [23] X. HE, R. HU, AND Y.-P. FANG, *Convergence Rate Analysis of Fast Primal-Dual Methods with Scalings for Linearly Constrained Convex Optimization Problems*, preprint, arXiv:2103.10118, 2021.
- [24] X. HE, R. HU, AND Y.-P. FANG, *Fast Convergence of Primal-Dual Dynamics and Algorithms with Time Scaling for Linear Equality Constrained Convex Optimization Problems*, preprint, arXiv:2103.12931, 2021.
- [25] X. HE, R. HU, AND Y.-P. FANG, *Inertial accelerated primal-dual methods for linear equality constrained convex optimization problems*, Numer. Algorithms, 9 (2022), pp. 1669–1690.
- [26] Y. HE AND R. D. MONTEIRO, *An accelerated hpe-type algorithm for a class of composite convex-concave saddle-point problems*, SIAM J. Optim., 26 (2016), pp. 29–56.
- [27] M. R. HESTENES, *Multiplier and gradient methods*, J. Optim. Theory Appl., 4 (1969), pp. 303–320.
- [28] B. HUANG, S. MA, AND D. GOLDFARB, *Accelerated linearized bregman method*, J. Sci. Comput., 54 (2013), pp. 428–453.
- [29] L. JACOB, G. OBOZINSKI, AND J.-P. VERT, *Group lasso with overlap and graph lasso*, in Proceedings of the 26th Annual International Conference on Machine Learning, 2009, pp. 433–440.
- [30] G. M. JAMES, C. PAULSON, AND P. RUSMEVICHIENTONG, *Penalized and constrained optimization: An application to high-dimensional website advertising*, J. Amer. Statist. Assoc., 115 (2019).
- [31] K. JIANG, D. SUN, AND K.-C. TOH, *An inexact accelerated proximal gradient method for large scale linearly constrained convex SDP*, SIAM J. Optim., 22 (2012), pp. 1042–1064.
- [32] M. KANG, M. KANG, AND M. JUNG, *Inexact accelerated augmented lagrangian methods*, Comput. Optim. Appl., 62 (2015), pp. 373–404.
- [33] M. KANG, S. YUN, H. WOO, AND M. KANG, *Accelerated bregman method for linearly constrained ℓ_1 - ℓ_2 minimization*, J. Sci. Comput., 56 (2013), pp. 515–534.
- [34] W. KONG, J. G. MELO, AND R. D. MONTEIRO, *Complexity of a quadratic penalty accelerated inexact proximal point method for solving linearly constrained nonconvex composite programs*, SIAM J. Optim., 29 (2019), pp. 2566–2593.
- [35] W. KONG AND R. D. MONTEIRO, *Accelerated inexact composite gradient methods for nonconvex spectral optimization problems*, Comput. Optim. Appl., to appear
- [36] G. LAN, *Gradient sliding for composite optimization*, Math. Program., 159 (2016), pp. 201–235.
- [37] G. LAN, S. LEE, AND Y. ZHOU, *Communication-efficient algorithms for decentralized and stochastic optimization*, Math. Program., 180 (2020), pp. 237–284.
- [38] G. LAN AND R. D. MONTEIRO, *Iteration-complexity of first-order penalty methods for convex programming*, Math. Program., 138 (2013), pp. 115–139.
- [39] G. LAN AND R. D. MONTEIRO, *Iteration-complexity of first-order augmented lagrangian methods for convex programming*, Math. Program., 155 (2016), pp. 511–547.
- [40] G. LAN AND Y. OUYANG, *Accelerated gradient sliding for structured convex optimization*, Comput. Optim. Appl., 82 (2022), pp. 361–394.
- [41] G. LAN, Y. OUYANG, AND Y. ZHOU, *Graph Topology Invariant Gradient and Sampling Complexity for Decentralized and Stochastic Optimization*, preprint, arXiv:2101.00143, 2021.

- [42] G. LAN, S. POKUTTA, Y. ZHOU, AND D. ZINK, *Conditional accelerated lazy stochastic gradient descent*, in Proceedings of the International Conference on Machine Learning, PMLR, 2017, pp. 1965–1974.
- [43] G. LAN AND Y. ZHOU, *Conditional gradient sliding for convex optimization*, SIAM J. Optim., 26 (2016), pp. 1379–1409.
- [44] H. LI, C. FANG, AND Z. LIN, *Convergence Rates Analysis of the Quadratic Penalty Method and its Applications to Decentralized Distributed Optimization*, arXiv:1711.10802, 2017.
- [45] H. LI, C. FANG, W. YIN, AND Z. LIN, *Decentralized accelerated gradient methods with increasing penalty parameters*, IEEE Trans. Signal Process., 68 (2020), pp. 4855–4870.
- [46] H. LI, Z. LIN, AND Y. FANG, *Variance Reduced Extra and Diging and their Optimal Acceleration for Strongly Convex Decentralized Optimization*, preprint, arXiv:2009.04373, 2020.
- [47] Q. LIN AND L. XIAO, *An adaptive accelerated proximal gradient method and its homotopy continuation for sparse optimization*, Comput. Optim. Appl., 60 (2015), pp. 633–674.
- [48] Q. LIN AND Y. XU, *Inexact Accelerated Proximal Gradient Method with Line Search and Reduced Complexity for Affine-Constrained and Bilinear Saddle-Point Structured Convex Problems*, preprint, arXiv:2201.01169, 2022.
- [49] Z. LU AND Z. ZHOU, *Iteration-Complexity of First-Order Augmented Lagrangian Methods for Convex Conic Programming*, preprint, arXiv:1803.09941v3, 2018.
- [50] 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.
- [51] A. S. NEMIROVSKI AND D. B. YUDIN, *Problem Complexity and Method Efficiency in Optimization*, Wiley, New York, 1983.
- [52] Y. NESTEROV, *Introductory Lectures on Convex Optimization: A Basic Course*, Appl. Optim. 87, Springer Science & Business Media, New York, 2003.
- [53] Y. NESTEROV, *Excessive gap technique in nonsmooth convex minimization*, SIAM J. Optim., 16 (2005), pp. 235–249.
- [54] Y. NESTEROV, *Smooth minimization of non-smooth functions*, Math. Program., 103 (2005), pp. 127–152.
- [55] Y. NESTEROV, *Gradient methods for minimizing composite functions*, Math. Program., 140 (2013), pp. 125–161.
- [56] Y. OUYANG AND T. SQUIRES, *Universal Conditional Gradient Sliding for Convex Optimization*, preprint, arXiv:2103.11026, 2021.
- [57] Y. OUYANG AND Y. XU, *Lower complexity bounds of first-order methods for convex-concave bilinear saddle-point problems*, Math. Program., 185 (2021), pp. 1–35.
- [58] A. PATRASCU, I. NECOARA, AND Q. TRAN-DINH, *Adaptive inexact fast augmented Lagrangian methods for constrained convex optimization*, Optim. Lett., 11 (2017), pp. 609–626.
- [59] Z. PENG, T. WU, Y. XU, M. YAN, AND W. YIN, *Coordinate-friendly structures, algorithms and applications*, Ann. Math. Sci. Appl., 1 (2016), pp. 57–119.
- [60] M. J. POWELL, *A fast algorithm for nonlinearly constrained optimization calculations*, in *Numerical Analysis*, Springer, New York, 1978, pp. 144–157.
- [61] R. T. ROCKAFELLAR, *Augmented lagrangians and applications of the proximal point algorithm in convex programming*, Math. Oper. Res., 1 (1976), pp. 97–116.
- [62] S. SABACH AND M. TEBOLLE, *Faster lagrangian-based methods in convex optimization*, SIAM J. Optim., 32 (2022), pp. 204–227.
- [63] M. SCHMIDT, N. L. ROUX, AND F. BACH, *Convergence Rates of Inexact Proximal-Gradient Methods for Convex Optimization*, preprint, arXiv:1109.2415, 2011.
- [64] M. TAO AND X. YUAN, *Accelerated uzawa methods for convex optimization*, Math. Comp., 86 (2017), pp. 1821–1845.
- [65] R. TIBSHIRANI, *Regression shrinkage and selection via the lasso*, J. R. Stat. Soc. Ser. B. Methodol., 58 (1996), pp. 267–288.
- [66] R. TIBSHIRANI, M. SAUNDERS, S. ROSSET, J. ZHU, AND K. KNIGHT, *Sparsity and smoothness via the fused lasso*, J. R. Stat. Soc. Ser. B. Methodol., 67 (2005), pp. 91–108.
- [67] Q. TRAN-DINH AND V. CEVHER, *Constrained convex minimization via model-based excessive gap*, Adv. Neural Inf. Process. Syst., 27 (2014), pp. 721–729.
- [68] Q. TRAN-DINH AND V. CEVHER, *A Primal-Dual Algorithmic Framework for Constrained Convex Minimization*, preprint, arXiv:1406.5403, 2014.
- [69] Q. TRAN-DINH, O. FERCOQ, AND V. CEVHER, *A smooth primal-dual optimization framework for nonsmooth composite convex minimization*, SIAM J. Optim., 28 (2018), pp. 96–134.
- [70] P. TSENG, *On Accelerated Proximal Gradient Methods for Convex-Concave Optimization*, manuscript, 2008.
- [71] S. VILLA, S. SALZO, L. BALDASSARRE, AND A. VERRI, *Accelerated and inexact forward-backward algorithms*, SIAM J. Optim., 23 (2013), pp. 1607–1633.

- [72] X. WEI, H. YU, Q. LING, and M. J. NEELY, *Solving non-smooth constrained programs with lower complexity than $O(1/\varepsilon)$ a primal-dual homotopy smoothing approach*, NeurIPS, 2018, pp. 3999–4009.
- [73] Y. XU, *Accelerated first-order primal-dual proximal methods for linearly constrained composite convex programming*, SIAM J. Optim., 27 (2017), pp. 1459–1484.
- [74] Y. XU, *First-order methods for constrained convex programming based on linearized augmented Lagrangian function*, Informs J. Optim., 3 (2021), pp. 89–117.
- [75] Y. XU, *Iteration complexity of inexact augmented lagrangian methods for constrained convex programming*, Math. Program., 185 (2021), pp. 199–244.
- [76] Y. XU, *First-order methods for problems with $O(1)$ functional constraints can have almost the same convergence rate as for unconstrained problems*, SIAM J. Optim., 32 (2022), pp. 1759–1790.
- [77] Y. XU, I. AKROTIRIANAKIS, AND A. CHAKRABORTY, *Proximal gradient method for huberized support vector machine*, Pattern Anal. Appl., 19 (2016), pp. 989–1005.
- [78] Y. XU AND W. YIN, *A block coordinate descent method for regularized multiconvex optimization with applications to nonnegative tensor factorization and completion*, SIAM J. Imaging Sci., 6 (2013), pp. 1758–1789.
- [79] W. YIN, *Analysis and generalizations of the linearized bregman method*, SIAM J. Imaging Sci., 3 (2010), pp. 856–877.
- [80] W. YIN, S. OSHER, D. GOLDFARB, AND J. DARBON, *Bregman iterative algorithms for ℓ_1 -minimization with applications to compressed sensing*, SIAM J. Imaging Sci., 1 (2008), pp. 143–168.
- [81] L. YUAN, J. LIU, AND J. YE, *Efficient methods for overlapping group lasso*, Adv. Neural Inf. Process. Syst., 24 (2011).
- [82] M. YUAN AND Y. LIN, *Model selection and estimation in regression with grouped variables*, J. R. Stat. Soc. Ser. B Methodol., 68 (2006), pp. 49–67.
- [83] A. YURTSEVER, Q. TRAN-DINH, AND V. CEVHER, *A universal primal-dual convex optimization framework*, in Proceedings of the 28th International Conference on Neural Information Processing Systems, Vol. 2, 2015, pp. 3150–3158.
- [84] R. ZHAO, *Accelerated stochastic algorithms for convex-concave saddle-point problems*, Math. Oper. Res., 47 (2022), pp. 1443–1473.
- [85] R. ZHAO, W. B. HASSELL, AND V. Y. TAN, *An optimal algorithm for stochastic three-composite optimization*, in Proceedings of the 22nd International Conference on Artificial Intelligence and Statistics, PMLR, 2019, pp. 428–437.