Optimal Nonergodic Primal-Dual Complexity of Efficient Inexact Parameter-Free Augmented Lagrangian Methods *

Arnesh Sujanani[†] Saeed Ghadimi[‡] Henry Wolkowicz[†]

Revision of August 10, 2026, 4:47am

Contents

*Emails respectively: a3sujana@uwaterloo.ca, sghadimi@uwaterloo.ca, hwolkowicz@uwaterloo.ca

[†]Department of Combinatorics and Optimization, University of Waterloo, ON, Canada

[‡]Department of Management Science and Engineering, University of Waterloo, ON, Canada

Key words and phrases: augmented Lagrangian, first-order methods, parameter-free, optimal complexity, verifiable termination, primal-dual solution, KKT point

AMS subject classifications: 65K05, 49M37, 90C06, 65K10, 90C30

List of Algorithms

List of Tables

Abstract

Augmented Lagrangian (AL) methods are a classical framework for constrained optimization, but for directly verifiable approximate KKT points, known first-order complexity bounds for standard inexact AL methods are suboptimal, while the best known proximal augmented Lagrangian (PAL) bounds retain an additional logarithmic factor. We consider linearly constrained convex composite problems with a smooth convex term and a possibly nonsmooth closed proper convex term with compact domain. We develop three inexact AL schemes that preserve the standard AL subproblem structure and attain the optimal primal-dual complexity $\mathcal{O}(\epsilon^{-1})$ in the convex setting, improving prior AL bounds of $\mathcal{O}(\epsilon^{-4/3})$, $\mathcal{O}(\epsilon^{-7/4})$, and $\mathcal{O}(\epsilon^{-2})$, and removing the logarithmic factor from PAL guarantees. Two variants are parameter-free, and all three admit nonergodic guarantees, including a stronger last-iterate guarantee for one variant.

These results show that proximal regularization, ergodic averaging, and prior knowledge of problem-dependent constants are not intrinsic requirements for attaining optimal verifiable primal-dual complexity within the standard AL framework. A key ingredient is a parameter-free accelerated method that computes verifiable stationarity certificates for the standard, unregularized AL subproblems with optimal complexity. In the strongly convex setting, our methods attain near-optimal complexity $\mathcal{O}(\epsilon^{-1/2}\log(\epsilon^{-1}))$, with two parameter-free variants. Numerical experiments on six problem classes, including elastic-net least-squares regression, group-sparse Huberized support vector machines, and a quantum semidefinite program (SDP), demonstrate substantial computational advantages over a representative PAL method, with speedups frequently ranging from 5 to 50 times.

1 Introduction

Augmented Lagrangian (AL) methods are a fundamental tool for constrained optimization. By replacing difficult projections onto the feasible region with a sequence of penalized subproblems and multiplier updates, they provide a flexible framework that is particularly well suited to large-scale linearly constrained models, including game theory applications, radar communication waveform design, conic and semidefinite optimization problems, and recent low-rank SDP approaches [1, 4, 7, 8, 10, 12, 27, 28, 30, 45, 46].

In this paper, we investigate whether the classical inexact AL architecture is itself sufficient to attain optimal and directly verifiable first-order guarantees for linearly constrained convex composite optimization problems of the form

$$p^*_{LC} := \min\{\psi(z) := \psi_s(z) + \psi_n(z) : \mathcal{A}z = b\}, \tag{1.1}$$

where $\mathcal{A} : \mathbb{E} \to \mathbb{R}^m$ is a nonzero linear transformation, $\mathbb{E}$ is a Euclidean space, and $\psi : \mathbb{E} \to \mathbb{R}$ is convex or $\bar{\mu}$-strongly convex. The nonsmooth term $\psi_n : \mathcal{N} \subset \mathbb{E} \to \mathbb{R}$ is closed, proper, and convex, and its domain is assumed to be a *compact convex set,* $\operatorname{dom}\psi_n$; in particular, its diameter is finite:

$$D := \sup\{\|z - z'\| : z, z' \in \operatorname{dom}\psi_n\} < \infty. \tag{1.2}$$

The smooth term $\psi_s : \mathbb{E} \to \mathbb{R}$ is convex, differentiable, and $\bar{L}$*-smooth*, meaning that

$$\|\nabla\psi_s(x) - \nabla\psi_s(y)\| \le \bar{L}\|x - y\|. \tag{1.3}$$

Given a nondecreasing sequence of positive penalty parameters $\{c_k\}$, an inexact AL method proceeds as follows. At iteration k, given the current multiplier p_{k-1}, it approximately minimizes the *augmented Lagrangian function* $\mathcal{L}_{c_k}(z, p_{k-1})$, where

$$\mathcal{L}_c(z,p) := \mathcal{L}_c^s(z,p) + \psi_n(z), \tag{1.4}$$

and $\mathcal{L}_c^s$ denotes its smooth part, namely,

$$\mathcal{L}_c^s(z,p) := \psi_s(z) + \langle p, \mathcal{A}z - b\rangle + \frac{c}{2}\|\mathcal{A}z - b\|^2. \tag{1.5}$$

It then performs the full multiplier update

$$p_k = p_{k-1} + c_k(\mathcal{A}z_k - b). \tag{1.6}$$

Thus, each outer iteration consists of an inexact primal minimization followed by a dual ascent step.

Inexact proximal augmented Lagrangian (PAL) methods modify this template by adding a proximal regularization term. Given positive scalars $\{\lambda_k\}$, these methods approximately minimize $\mathcal{P}_{c_k}^{\lambda_k}(z, p_{k-1})$, where

$$\mathcal{P}_c^{\lambda}(z,p) := \mathcal{L}_c(z,p) + \frac{1}{2\lambda}\|z - z_{k-1}\|^2. \tag{1.7}$$

The proximal term can simplify the complexity analysis and improve conditioning, but it also changes the subproblem and can be computationally undesirable when the unregularized AL subproblem has exploitable structure. This distinction is structural: PAL methods solve a different sequence of primal subproblems, so their complexity guarantees do not by themselves determine what is achievable within the standard, unregularized AL framework.

Despite substantial progress, existing first-order inexact AL methods do not simultaneously provide all of the following features for (1.1): an optimal primal-dual/KKT complexity bound, a fully verifiable termination criterion, no proximal regularization of the AL subproblems, and no prior knowledge of problem-dependent constants. Some AL complexity guarantees target primal objective and feasibility criteria involving the unknown optimal value [3,30,31,36,49,50]. Although important theoretically, such guarantees do not directly yield implementable stopping rules. For verifiable primal-dual/KKT certificates, the known first-order AL bounds remain suboptimal: Xu obtains an $\mathcal{O}(\epsilon^{-4/3})$ primal-dual complexity bound [49]; Necoara obtains an $\mathcal{O}(\epsilon^{-3/2})$ bound [30]; Liu et al. establish an $\mathcal{O}(\epsilon^{-2})$ nonergodic primal-dual bound [25]; and both Lan and Monteiro's method and the standard first-order I-AL method of Lu and Zhou attain $\mathcal{O}(\epsilon^{-7/4})$ KKT-type bounds [15,27]. Necoara's fast inexact AL method, which incorporates Nesterov-type acceleration into the outer scheme rather than following the traditional inexact AL framework, achieves an $\mathcal{O}(\epsilon^{-4/3})$ primal-dual complexity bound [30]. Lu and Zhou's adaptively regularized PAL method improves the KKT complexity to $\mathcal{O}(\epsilon^{-1}\log(\epsilon^{-1}))$ [27], and related PAL schemes obtain similar logarithmically suboptimal bounds [6,19,24,26]. Thus, the remaining issue is not merely to improve a complexity exponent, but to determine the capabilities of the standard AL framework.

This leads to the central question addressed in this paper: can an unregularized inexact AL method attain the **optimal** $\mathcal{O}(\epsilon^{-1})$ first-order complexity for **directly verifiable approximate KKT points** of (1.1), while avoiding both proximal regularization of the AL subproblems and problem-dependent parameter tuning? We answer this question **affirmatively**. For merely convex objectives, we develop inexact AL methods with the **optimal** $\mathcal{O}(\epsilon^{-1})$ first-order complexity for computing **approximate primal-dual/KKT solutions**; two of the proposed variants are **parameter-free.** For strongly convex objectives, we establish a near-optimal $\mathcal{O}(\epsilon^{-1/2}\log(\epsilon^{-1}))$ complexity bound without requiring knowledge of the strong convexity parameter or any other problem-dependent parameters.

Two ideas are central to the analysis. First, approximate stationarity of the standard AL subproblems, together with the full multiplier update, yields a telescoping dual-distance estimate that directly controls primal feasibility without ergodic averaging. Second, parameter-free accelerated regularization and restart schemes minimize the unregularized AL subproblems and produce verifiable composite-stationarity certificates at optimal or near-optimal inner complexity. For the adaptive last-iterate variant, a uniform multiplier bound further allows the increasing penalty parameter to directly control feasibility of the current iterate.

We also report numerical results for **six** important classes of linearly constrained optimization problems: **quadratic programs**, **logistic regression**, **elastic-net least-squares regression**, **group-sparse Huberized support vector machines (SVMs)**, **Bayesian D-optimal experimental design**, and a **Huberized quantum semidefinite program (SDP)**. These experiments show that the proposed AL methods can substantially outperform a representative PAL scheme [26], with **speedups** frequently exceeding **5–50x**.

1.1 Literature Review

1.1.1 Parameter-Free Optimization Methods for Unconstrained Minimization

When $\mathcal{A} = 0$ in (1.1), the problem reduces to unconstrained composite minimization. Parameter-free first-order methods are attractive in this setting because they spare users from estimating quantities such as global Lipschitz constants, growth constants, and strong convexity moduli [11, 16, 20, 22, 23, 42, 44, 47, 48, 51]. In terms of objective-value accuracy, when ψ is convex, backtracking FISTA and related accelerated composite-gradient methods attain the optimal $\mathcal{O}(\epsilon^{-1/2})$ complexity for computing a point z such that $\psi(z) - \psi_* \le \epsilon$ [5, 33, 40].

For verifiable termination, however, objective residuals are less useful than stationarity measures such as gradient norms or composite gradient mappings. When ψ is convex, standard accelerated schemes generally attain only the suboptimal $\mathcal{O}(\epsilon^{-2/3})$ complexity for reducing such stationarity measures below ϵ [13, 14, 18, 29, 41]. Nesterov's proximal regularization technique yields a near-optimal $\mathcal{O}(\epsilon^{-1/2}\log(\epsilon^{-1}))$ bound [32], while subsequent parameter-dependent methods remove the logarithmic factor under additional algorithmic requirements, such as a prescribed iteration horizon or consecutive method calls [17, 35].

Lan et al. [16] recently developed accumulative-regularization methods that attain the optimal $\mathcal{O}(\epsilon^{-1/2})$ stationarity complexity and include parameter-free variants. Their results substantially clarify the complexity landscape for unconstrained stationarity. The linearly constrained composite setting considered here is more demanding: a verifiable certificate must simultaneously control Lagrangian stationarity and feasibility, while the AL subproblems themselves vary with both the multiplier and the penalty parameter.

1.1.2 Inexact Augmented Lagrangian Methods

The exact AL method dates back to Hestenes and Powell [9, 37], and Rockafellar established convergence of inexact AL schemes through the proximal-point interpretation [39]. Modern first-order analyses quantify both the accuracy with which the AL subproblems must be solved and the total number of inner first-order iterations required to reach a prescribed target accuracy.

One line of work focuses on primal accuracy. For convex objectives, Aybat and Iyengar [3] obtained an $\mathcal{O}(\epsilon^{-1}\log(\epsilon^{-1}))$ complexity bound, and Xu [49] improved this result to the optimal $\mathcal{O}(\epsilon^{-1})$ gradient-evaluation complexity for computing $z \in \operatorname{dom}\psi_n$ satisfying

$$|\psi(z) - p^*_{LC}| \le \epsilon, \quad \|\mathcal{A}z - b\| \le \epsilon, \tag{1.8}$$

where p^*_{LC} is defined in (1.1). Such guarantees are theoretically important, but (1.8) is generally not a practical stopping rule because p^*_{LC} is unknown.

A more implementable target is an approximate KKT point, namely, a triple (z, p, r) satisfying

$$r \in \nabla\psi_s(z) + \partial\psi_n(z) + \mathcal{A}^*p, \quad \|r\| \le \epsilon, \quad \|\mathcal{A}z - b\| \le \epsilon. \tag{1.9}$$

This certificate can be checked directly within an AL method. The main first-order AL complexity results for primal-dual or KKT-type criteria therefore provide the natural benchmarks for this paper. Lan and Monteiro [15] analyze a first-order inexact AL method for computing a primal-dual solution according to the verifiable criterion in (1.9). They establish an $\mathcal{O}(\epsilon^{-7/4})$ complexity bound, although their criterion for terminating the AL subproblem solves is not directly verifiable because it depends on the unknown optimal values of the corresponding augmented Lagrangian subproblems. Lu and Zhou [27] and Necoara [30] establish KKT complexity bounds of $\mathcal{O}(\epsilon^{-7/4})$ and $\mathcal{O}(\epsilon^{-3/2})$, respectively, for standard first-order I-AL methods that also seek points satisfying (1.9). Like the method of Lan and Monteiro, these methods are not parameter-free and use AL-subproblem termination criteria that depend on unknown optimal values of the AL objectives.

Xu [49] and Liu et al. [25] also study the complexity of inexact AL methods for computing primal-dual solutions, although their solution criteria differ slightly from (1.9). Xu establishes an ergodic primal-dual complexity bound of $\mathcal{O}(\epsilon^{-4/3})$, whereas Liu et al. establish a nonergodic primal-dual complexity bound of $\mathcal{O}(\epsilon^{-2})$. These results demonstrate steady progress toward verifiable AL certificates, but their complexities remain worse than $\mathcal{O}(\epsilon^{-1})$, and they do not provide the parameter-free, genuine-AL guarantee pursued here. More recently, easily implementable relative-error AL frameworks have been developed, but they attain a suboptimal $\mathcal{O}(\epsilon^{-2})$ complexity bound for (1.9) [38].

PAL and other proximal or regularized AL frameworks constitute a second line of work. Lu and Mei [26] establish an $\mathcal{O}(\epsilon^{-1}\log(\epsilon^{-1}))$ complexity bound in the convex case and an $\mathcal{O}(\epsilon^{-1/2}\log(\epsilon^{-1}))$ bound in the strongly convex case for KKT-type solutions. Related PAL and inexact proximal-point analyses appear in [6, 19, 24]. Although powerful, these methods introduce the proximal term in (1.7), thereby changing the primal subproblem, and their known convex-case bounds retain a logarithmic dependence on the target accuracy. Moreover, strongly convex variants typically require knowledge of the strong convexity parameter.

Accordingly, our contribution is not merely a sharper complexity exponent. It establishes that, within the problem class studied here, the standard, unregularized AL architecture is sufficient to combine directly verifiable KKT termination with optimal nonergodic primal-dual complexity. In two variants, this is achieved without knowledge of problem-dependent constants.

1.2 Summary of Contributions

We now summarize our main contributions.

1. We establish that standard inexact AL methods can attain the optimal $\mathcal{O}(\epsilon^{-1})$ first-order complexity for computing directly verifiable primal-dual/KKT solutions in the merely convex setting. We realize this result through three new inexact AL schemes. This improves upon the $\mathcal{O}(\epsilon^{-4/3})$ primal-dual bound of Xu [49], the $\mathcal{O}(\epsilon^{-3/2})$ primal-dual bound of Necoara [30], the $\mathcal{O}(\epsilon^{-2})$ nonergodic primal-dual bound of Liu et al. [25], and the $\mathcal{O}(\epsilon^{-7/4})$ primal-dual/KKT-type bounds of Lan–Monteiro [15] and the standard I-AL method of Lu–Zhou [27]. Our complexity bound also improves upon logarithmically suboptimal $\mathcal{O}(\epsilon^{-1}\log(\epsilon^{-1}))$ guarantees available for regularized PAL type methods [6, 19, 24, 26, 27].

2. We provide a detailed nonergodic convergence analysis of all three proposed AL methods, yielding guarantees for individual generated iterates rather than ergodic averages. Two variants yield best-iterate guarantees, whereas the third establishes the stronger last-iterate guarantee under mildly stronger assumptions. The last-iterate variant is also more adaptive and employs a novel mechanism for updating the tolerances used to solve its AL subproblems.

3. In the convex setting, two of the three AL variants are parameter-free. To the best of our knowledge, they are the first parameter-free inexact AL methods that combine verifiable KKT termination with the optimal $\mathcal{O}(\epsilon^{-1})$ primal-dual complexity.

4. A key ingredient in these bounds is a parameter-free regularization method with the optimal $\mathcal{O}(\epsilon^{-1/2})$ complexity for solving the unconstrained composite AL subproblems. The method extends the parameter-free ideas of [16] from smooth unconstrained minimization to the unconstrained composite setting and uses a termination criterion specifically tailored to the AL analysis. Beyond its role in the AL analysis, PF-AR therefore provides an optimal parameter-free method for computing directly verifiable stationarity certificates in convex composite minimization with compact nonsmooth domain.

5. When ψ is $\bar{\mu}$-strongly convex with $\bar{\mu} > 0$, we establish a near-optimal $\mathcal{O}(\epsilon^{-1/2}\log(\epsilon^{-1}))$ complexity bound. Unlike the parameter-dependent methods of [21, 26], two of our methods are fully parameter-free. The AL subproblems are solved using a simplified restarted FISTA scheme developed in this paper and inspired by [44]. Compared with the scheme in [44], our method uses a simpler restart condition and permits a more flexible choice of restart point.

6. Finally, numerical experiments on six important classes of linearly constrained problems demonstrate that the proposed AL methods can substantially outperform a representative PAL method by Lu and Mei [26]. On many instances, the observed speedups often range from 5x to 50x.

The two tables below summarize our six main contributions further.

Table 1.1 Comparison of Methods for Convex Linearly-Constrained Optimization

Name	Primal-Dual Complexity	Parameter-Free	AL or PAL Method
Lan and Monteiro [15]	$\mathcal{O}\left(\epsilon^{-7/4}\right)$	No	AL
Liu et al. [25][1]	$\mathcal{O}\left(\epsilon^{-2}\right)$	No	AL
Lu and Zhou [27]	$\mathcal{O}\left(\epsilon^{-7/4}\right)$	No	AL
Necoara [30]	$\mathcal{O}\left(\epsilon^{-3/2}\right)$	No	AL
Xu [49][1]	$\mathcal{O}\left(\epsilon^{-4/3}\right)$	No	AL
This Work (Algorithm 2.4)	$\mathcal{O}\left(\epsilon^{-1}\right)$	No	AL
This Work (Algorithm 2.5)	$\mathcal{O}\left(\epsilon^{-1}\right)$	Yes	AL
This Work (Algorithm 2.6)	$\mathcal{O}\left(\epsilon^{-1}\right)$ (stronger last-iterate)	Yes	AL
Burns and Liang [6]	$\mathcal{O}\left(\epsilon^{-1}\log(\epsilon^{-1})\right)$	No	PAL
Li and Qu [19]	$\mathcal{O}\left(\epsilon^{-1}\log(\epsilon^{-1})\right)$	No	PAL
Lin and Xu [24]	$\mathcal{O}\left(\epsilon^{-1}\log(\epsilon^{-1})\right)$	No	PAL
Lu and Zhou [27]	$\mathcal{O}\left(\epsilon^{-1}\log(\epsilon^{-1})\right)$	No	PAL
Lu and Mei [26]	$\mathcal{O}\left(\epsilon^{-1}\log\left(\epsilon^{-1}\right)\right)$	Yes	PAL

Table 1.2 Comparison of Methods for Strongly-Convex Linearly-Constrained Optimization

Name	Primal-Dual Complexity	Parameter-Free	AL or PAL Method
This Work (Algorithm 2.4)	$\mathcal{O}\left(\epsilon^{-1/2}\log(\epsilon^{-1})\right)$	No	AL
This Work (Algorithm 2.5)	$\mathcal{O}\left(\epsilon^{-1/2}\log(\epsilon^{-1})\right)$	Yes	AL
This Work (Algorithm 2.6)	$\mathcal{O}\left(\epsilon^{-1/2}\log(\epsilon^{-1})\right)$	Yes	AL
Li and Xu [21]	$\mathcal{O}\left(\epsilon^{-1/2}\log(\epsilon^{-1})\right)$	No	AL
Lu and Mei [26]	$\mathcal{O}\left(\epsilon^{-1/2}\log(\epsilon^{-1})\right)$	No	PAL

[1] It should be noted that the notions of approximate primal–dual solution employed by Liu et al. [25] and Xu [49] are closely related to, but formally distinct from, those employed by the other works listed in the table.

1.3 Basic Definitions and Notation

We introduce the notation used throughout the paper. Let $\mathbb{R}_+$ and $\mathbb{R}_{++}$ denote the sets of nonnegative and strictly positive real numbers, respectively. We work in a finite-dimensional inner product space $\mathbb{E}$ with inner product $\langle \cdot, \cdot \rangle$ and associated norm $\| \cdot \|$. Let $A : \mathbb{E} \to \mathbb{R}^m$ be a nonzero linear transformation, and let σ_A^+ denote its smallest positive singular value.

For a closed convex set $Z \subseteq \mathbb{E}$, we denote its boundary by $\operatorname{bd} Z$, the distance from $z \in \mathbb{E}$ to Z by

$$\operatorname{dist}(z, Z) := \inf_{u \in Z} \|z - u\|,$$

and its indicator function by $\iota_Z : \mathbb{E} \to (-\infty, +\infty]$, where

$$\iota_Z(z) := \begin{cases} 0, & z \in Z, \\ +\infty, & \text{otherwise.} \end{cases}$$

For $h : \mathbb{E} \to (-\infty, +\infty]$, define

$$\operatorname{dom} h := \{z \in \mathbb{E} : h(z) < \infty\}.$$

The function h is proper if $\operatorname{dom} h \neq \emptyset$, and $\overline{\operatorname{Conv}}(\mathbb{E})$ denotes the class of proper, lower semicontinuous, convex functions on $\mathbb{E}$.

For a proper function h, its ε-subdifferential at $z \in \mathbb{E}$ is

$$\partial_\varepsilon h(z) := \left\{u \in \mathbb{E} : h(z') \geq h(z) + \langle u, z' - z \rangle - \varepsilon, \quad \forall z' \in \mathbb{E}\right\},$$

and $\partial h(z) := \partial_0 h(z)$. For a closed convex set C and $z \in C$, its ε-normal cone is

$$N_C^\varepsilon(z) := \{\xi \in \mathbb{E} : \langle \xi, u - z \rangle \leq \varepsilon, \quad \forall u \in C\},$$

and $N_C(z) := N_C^0(z)$.

If $f : \mathbb{E} \to \mathbb{R}$ is differentiable, its affine approximation at $\bar{z}$ is

$$\ell_f(z; \bar{z}) := f(\bar{z}) + \langle \nabla f(\bar{z}), z - \bar{z} \rangle, \quad \forall z \in \mathbb{E}. \tag{1.10}$$

If f is convex, then $\ell_f(\cdot; \bar{z})$ is the unique affine minorant of f at $\bar{z}$.

For $t > 0$ and $b > 1$, define

$$\log_b^+(t) := \max\{\log_b(t), 0\}, \qquad \log_b^{++}(t) := \max\{\log_b(t), 1\}. \tag{1.11}$$

When no ambiguity arises, $\log(t)$ denotes the natural logarithm.

We let the Lagrangian and dual functions of (1.1) be defined respectively as

$$\mathcal{L}(z, p) := \psi(z) + \langle p, \mathcal{A}z - b \rangle, \quad D(p) := \min_z \mathcal{L}(z, p). \tag{1.12}$$

Then the dual problem and weak duality of (1.1) are

$$p_{LC}^* \geq d_{LC}^* := \max_p D(p). \tag{1.13}$$

Let $\mathcal{P}_\star$ denote the set of optimal dual solutions of problem (1.1). For $p \in \mathbb{R}^m$, define the squared distance to dual optimality by

$$\Pi(p) := \operatorname{dist}^2(p, \mathcal{P}_\star) = \inf_{p_\star \in \mathcal{P}_\star} \|p - p_\star\|^2. \tag{1.14}$$

2 Linearly-Constrained Convex Optimization

In this section, we develop inexact augmented Lagrangian (AL) methods for solving the linearly constrained convex optimization problem (1.1) under the assumption that ψ is convex. More specifically, we design inexact AL methods that compute approximate solutions satisfying suitable primal-dual optimality conditions for the pair (1.1) and (1.13), while attaining optimal complexity guarantees.

To establish convergence results, we impose the following standard assumption.

Assumption 2.1. *There exists an optimal primal-dual pair, (z_*, p_*), satisfying the first-order primal-dual optimality conditions*

$$0 \in \nabla\psi_s(z_*) + \partial\psi_n(z_*) + \mathcal{A}^* p_* \ \textit{(d.f.)}, \qquad \mathcal{A}z_* - b = 0 \ \textit{(p.f.)}. \tag{2.1}$$

Here, the multiplier p_* attains the optimal dual value d^*_{LC} in (1.13), and $\mathcal{A}^*$ denotes the adjoint of the linear operator $\mathcal{A}$.

We next define the notion of approximate solution used by our inexact AL methods throughout the paper.

Definition 2.2. *Given problem (1.1) under Assumption 2.1, and a tolerance pair, $(\hat{\epsilon}, \hat{\rho}) \in \mathbb{R}^2_{++}$, we say that a triple (z, p, r) is a $(\hat{\epsilon}, \hat{\rho})$-**approximate primal-dual optimal solution** if it approximately satisfies the primal-dual optimality conditions in (2.1), i.e., the triple satisfies*

$$r \in \nabla\psi_s(z) + \partial\psi_n(z) + \mathcal{A}^* p, \quad \|r\| \le \hat{\rho}, \quad \|\mathcal{A}z - b\| \le \hat{\epsilon}. \tag{2.2}$$

Condition (2.2) encodes approximate dual stationarity and primal feasibility.

To approximately minimize (1.4), our inexact AL methods employ PF-AR, the optimal parameter-free method developed in Subsection 2.1. PF-AR nontrivially extends the parameter-free method of Lan et al. [16] from unconstrained minimization to convex composite optimization by using a new line-search mechanism and a different termination criterion.

The remainder of this section is organized as follows. Subsection 2.1 presents the parameter-free AL subproblem solver, PF-AR, together with its optimal complexity guarantees for convex composite minimization. Subsection 2.2 presents O-IAL, an optimal dual-type inexact AL method that maintains approximate dual stationarity at each iteration, and proves its optimal best-iterate guarantees for (1.1). O-IAL is not parameter-free as it requires knowledge of the diameter D of $\operatorname{dom}\psi_n$. Subsection 2.3 then presents OPF-IAL, an essentially restarted version of O-IAL that is parameter-free and requires no knowledge of D or any other problem parameter while preserving optimal best-iterate guarantees. Finally, Subsection 2.4 presents APF-IAL, a more practical adaptive parameter-free method that chooses the penalty parameter and subproblem tolerances adaptively and achieves optimal last-iterate guarantees under assumptions only mildly stronger than those required for O-IAL and OPF-IAL.

2.1 PF-AR Method: Parameter-Free Convex Subproblem Solver

This section presents a parameter-free accumulative regularization (AR) method, called PF-AR, which will be used by O-IAL, OPF-IAL, and APF-IAL, to approximately solve their AL subproblems. More generally, PF-AR solves the unconstrained variant of the composite optimization problem in (1.1), i.e.,

$$\min\{\psi(x) := \psi_s(x) + \psi_n(x)\}, \tag{2.3}$$

where ψ_s is convex and $\bar{L}$-smooth, and ψ_n is a closed proper convex function with compact domain $\operatorname{dom}\psi_n$ of diameter D. We also assume that the proximal mapping of ψ_n can be computed efficiently. More specifically, for any $\sigma, \eta \geq 0$ and any $x, \bar{x} \in \mathbb{E}$, we assume that the optimization problem

$$x^{++}(\eta, \sigma, \bar{x}; x) := \operatorname{argmin}_u \langle \nabla\psi_s(x), u\rangle + \psi_n(u) + \frac{\sigma}{2}\|u - \bar{x}\|^2 + \frac{\eta}{2}\|u - x\|^2$$

can be solved efficiently, where recall that $\mathbb{E}$ is a finite-dimensional inner product space.

The proposed PF-AR algorithm can be viewed as an extension of the parameter-free accelerated regularization (AR) method of [16], originally developed for unconstrained problems, to the composite setting. This extension, however, is nontrivial. In particular, PF-AR employs a new backtracking line-search mechanism for estimating the Lipschitz constant, which differs fundamentally from the parameter-free strategy in [16]. This modification is essential for obtaining optimal convergence guarantees in terms of the norm of a subgradient of ψ, which is the stationarity criterion required in our augmented Lagrangian analysis.

This stopping criterion also distinguishes our approach from the parameter-dependent composite AR method in [16], which terminates upon finding (x, η) such that $\|G_\eta(x)\|$ is small, where

$$G_\eta(x) := \eta(x - x^+(\eta; x)), \quad x^+(\eta; x) := \operatorname{argmin}_u \langle \nabla\psi_s(x), u\rangle + \psi_n(u) + \frac{\eta}{2}\|u - x\|^2. \tag{2.4}$$

In contrast, PF-AR terminates when $\|v\|$ is small for some $v \in \nabla\psi_s(x) + \partial\psi_n(x)$. To obtain this guarantee, our method enforces two key inequalities through a backtracking line-search procedure at every iteration. Indeed, the PF-AR method is a multi-loop algorithm that is fully parameter-free: it requires no prior knowledge of either the diameter D or the Lipschitz constant $\bar{L}$. To remove the need to know D, PF-AR repeatedly calls a second algorithm, AR-L, with different guesses for the diameter. The AR-L method itself requires no knowledge of $\bar{L}$; instead, each of its iterations uses the new line-search mechanism described above. Since this line-search is central to both PF-AR and AR-L, we present it first.

Algorithm 2.1 Backtracking(ψ_s, ψ_n, σ, x, $\bar{x}$, M_0).

1: **for** $j = 0, 1, \ldots,$ **do**

2: Set $x^{++} := x^{++}(2M_j, \sigma, \bar{x}; x) := \operatorname{argmin}_u \langle \nabla\psi_s(x), u\rangle + \psi_n(u) + \frac{\sigma}{2}\|u - \bar{x}\|^2 + M_j\|u - x\|^2$

3: Set $x^+ := x^+(2M_j; x) := \operatorname{argmin}_u \langle \nabla\psi_s(x), u\rangle + \psi_n(u) + M_j\|u - x\|^2$

4: If

$$\psi_s(x^{++}) - \psi_s(x) - \langle \nabla\psi_s(x), x^{++} - x\rangle \leq \frac{M_j}{2}\|x^{++} - x\|^2, \tag{2.5}$$

$$\|\nabla\psi_s(x^+) - \nabla\psi_s(x)\| \leq M_j\|x^+ - x\|, \tag{2.6}$$

then **terminate** with x^{++}, x^+, and $M = M_j$. Otherwise, set $M_{j+1} = 2M_j$.

5: **end for**

Output: triple (x^{++}, x^+, M).

As discussed above, AR-L uses Algorithm 2.1 as a subroutine and does not require prior knowledge of $\bar{L}$. It is an inexact proximal-point scheme that approximately minimizes a sequence of proximal subproblems. More specifically, as in [16], AR-L computes a sequence of outer iterates x_s satisfying, approximately,

$$x_s \approx \operatorname{argmin}_u \left\{\phi_s(u) := \psi_s(u) + \psi_n(u) + \frac{\sigma_s}{2}\|u - \bar{x}_s\|^2\right\}, \tag{2.7}$$

by using x_{s-1} as an initial point and N_s inner iterations.

Following [16], we assume the availability of an accelerated algorithm $\mathcal{A}$ for approximately solving these proximal subproblems. Many accelerated methods in the literature satisfy the required

property such as the R-FISTA method presented in Subsection 3.1 and the method in [34]. We state it formally in the next assumption.

Assumption 2.3. *Algorithm $\mathcal{A}$ used to compute $x_s = \mathcal{A}(\psi_s, \psi_n, \sigma_s, \bar{x}_s, x_{s-1})$ (see (2.10) in algorithm below) to approximately minimize ϕ_s has the following performance guarantees after the k-th evaluation of $\nabla\psi_s$:*

$$\phi_s(x_s^k) - \phi_s(x_s^*) \leq \frac{L_s^k}{k^2}\|x_{s-1} - x_s^*\|^2. \tag{2.8}$$

Here x_s^k is the computed approximate solution and L_s^k is an estimate of Lipschitz constant $\bar{L}$ such that $L_s^k \leq c_{\mathcal{A}}\bar{L}$ where $c_{\mathcal{A}}$ is a universal constant that depends on algorithm $\mathcal{A}$. We further assume that subroutine $\mathcal{A}$ is terminated whenever

$$k \geq 8\sqrt{\frac{2L_s^k}{\sigma_s}}. \tag{2.9}$$

When this occurs, we output the approximate solution $x_s = x_s^k$.

We now formally present the AR-L method for solving (2.3) which requires no knowledge of $\bar{L}$.

Algorithm 2.2 AR-L$(\psi_s, \psi_n, x_0, \sigma_1, M_0)$.

Input: function pair (ψ_s, ψ_n), initial point $x_0 \in \operatorname{dom}\psi_n$, initial regularization parameter $\sigma_1 > 0$, and initial Lipschitz estimate $M_0 > 0$.

$s \leftarrow 0$
stopcrit $\leftarrow \infty$
while stopcrit > 0 **do**
  set $s \leftarrow s + 1$;
  if $s = 1$ **then**
    $\bar{x}_s = x_0$
  else
    set $\sigma_s = 4\sigma_{s-1}$ and $\bar{x}_s = (1 - \gamma_s)\bar{x}_{s-1} + \gamma_s x_{s-1}$ with $\gamma_s = 1 - \sigma_{s-1}/\sigma_s$;
  end if
  compute an approximate solution $x_s \in \operatorname{dom}\psi_n$ of

$$x_s^* := \operatorname{argmin}_{x \in \operatorname{dom}\psi_n} \left\{ \phi_s(x) := \psi_s(x) + \psi_n(x) + \frac{\sigma_s}{2}\|x - \bar{x}_s\|^2 \right\} \tag{2.10}$$

  by running subroutine $x_s = \mathcal{A}(\psi_s, \psi_n, \sigma_s, \bar{x}_s, x_{s-1})$ with initial point x_{s-1};
  set $(x_s^{++}, x_s^+, M_s) =$ Backtracking $(\psi_s,\ \psi_n,\ \sigma_s,\ x_s,\ \bar{x}_s,\ M_{s-1})$;
  set $v_s := 2M_s\,(x_s - x_s^+) - \nabla\psi_s(x_s) + \nabla\psi_s(x_s^+)$;
  set stopcrit $= M_s - \sigma_s$
end while

Output: triple $(\hat{x}, \hat{v}, \hat{M}) := (x_s^+, v_s, M_s)$.

In the analysis of AR-L, we set $\sigma_0 = 0$ and $\bar{x}_0 = x_0$. The main complexity result for the AR-L method is stated in Theorem 2.4. Due to the length of its proof, the proof of the theorem is given in Section A.

Theorem 2.4. *The AR-L method outputs a triple $(\hat{x}, \hat{v}, \hat{M})$ satisfying*

$$\hat{v} \in \nabla\psi_s(\hat{x}) + \partial\psi_n(\hat{x}), \qquad \|\hat{v}\| \leq 7\sigma_1 D, \tag{2.11}$$

$$M_0 \leq \hat{M} \leq \max\left\{M_0,\, 2\bar{L}\right\}. \tag{2.12}$$

Moreover, if $M_0 \leq 2\bar{L}$, then such a triple is obtained within at most

$$6 + C_1\sqrt{\frac{\bar{L}}{\sigma_1}} + \log_2\left(\frac{2\bar{L}}{M_0}\right) \tag{2.13}$$

gradient evaluations, where

$$C_1 := 2\sqrt{2} + 16\sqrt{2c_{\mathcal{A}}}, \tag{2.14}$$

and $c_{\mathcal{A}}$ is the universal constant in Assumption 2.3.

Finally, if the input is chosen as $\sigma_1 = \epsilon/(7\overline{D})$ for some scalar $\overline{D} > 0$, then AR-L outputs $(\hat{x}, \hat{v}, \hat{M})$ satisfying (2.11) and (2.12), with $\|\hat{v}\| \leq (\epsilon D)/\overline{D}$ within at most $6+\sqrt{7}C_1\sqrt{\frac{\bar{L}\overline{D}}{\epsilon}}+\log_2\left(\frac{2\bar{L}}{M_0}\right)$ gradient evaluations.

Remark 2.5. *The assumption that $M_0 \leq 2\bar{L}$ is not strictly necessary to establish a complexity result similar to (2.13). It is imposed only to simplify the analysis and the resulting complexity bound of the AR-L method. It is easy to construct an initial estimate M_0 satisfying this assumption. Indeed, if ψ_s is not affine and we choose $w_0 \neq x_0$ with $\nabla\psi_s(x_0) \neq \nabla\psi_s(w_0)$, where $x_0 \in \operatorname{dom}\psi_n$ is the initial point, and define $M_0 := \|\nabla\psi_s(x_0) - \nabla\psi_s(w_0)\|/(4\|x_0 - w_0\|)$, then it follows from the definition of $\bar{L}$ that $M_0 \leq 2\bar{L}$. In the affine case, since any positive scalar $\bar{L}$ is a valid smoothness constant for ψ_s, we may choose any $M_0 > 0$ and then take $\bar{L}$ large enough so that $M_0 \leq 2\bar{L}$.*

As follows from Theorem 2.4, the AR-L method requires the knowledge of D in order to compute a point whose subgradient norm is at most ϵ.

We next present the PF-AR method, which requires no prior knowledge of either $\bar{L}$ or D to compute a point whose subgradient norm is at most ϵ. That is, given functions ψ, ψ_s, and ψ_n as in (2.3), and a tolerance parameter $\epsilon > 0$, the goal of PF-AR is to find a pair $(\hat{x}, \hat{v}) \in \operatorname{dom}\psi_n \times \mathbb{E}$ such that

$$\|\hat{v}\| \leq \epsilon, \quad \hat{v} \in \nabla\psi_s(\hat{x}) + \partial\psi_n(\hat{x}). \tag{2.15}$$

The PF-AR method repeatedly invokes AR-L with different guesses for D and is guaranteed to compute a pair $(\hat{x}, \hat{v})$ in at most $\mathcal{O}(1/\epsilon)$ gradient evaluations. The method is now presented in Algorithm 2.3 below.

Algorithm 2.3 PF-AR Method

Input: function pair (ψ_s, ψ_n), initial point $x_0 \in \operatorname{dom}\psi_n$, initial smoothness estimate $M_0 > 0$, initial diameter $D_0 > 0$ estimate, and tolerance $\epsilon > 0$.

1: $t \leftarrow 0$
2: stopcrit $\leftarrow \infty$
3: **while** stopcrit $> \epsilon$ **do**
4: set $t \leftarrow t + 1$;
5: set $D_t = 4D_{t-1}$;
6: set $(v_t, x_t, M_t) =$ AR-L$(\psi_s, \psi_n, x_{t-1}, \epsilon/(7D_t), M_{t-1})$ (see Algorithm 2.2) ;
7: set stopcrit $\leftarrow \|v_t\|$;
8: **end while**

Output: pair $(\hat{x}, \hat{v}) = (x_t, v_t)$ satisfying (2.15).

The main complexity result of the PF-AR method is stated in the following theorem.

Theorem 2.6. *Suppose that inputs M_0 and D_0 satisfy $M_0 \leq 2\bar{L}$ and $D_0 \leq D$. The PF-AR method then computes a pair $(\hat{x}, \hat{v})$ such that*

$$\hat{v} \in \nabla\psi_s(\hat{x}) + \partial\psi_n(\hat{x}), \quad \|\hat{v}\| \leq \epsilon \tag{2.16}$$

within at most

$$\left[6 + \log_2(2\bar{L}/M_0)\right]\left[1 + \left\lceil \log_4\left(\frac{D}{D_0}\right)\right\rceil\right] + 8\sqrt{7}C_1\sqrt{\frac{\bar{L}D}{\epsilon}} \tag{2.17}$$

gradient evaluations where C_1 is a universal constant as in (2.14). Hence, the PF-AR method achieves the optimal gradient evaluation complexity bound of $\mathcal{O}\left(\sqrt{\bar{L}D/\epsilon}\right)$ for finding a pair $(\hat{x},\hat{v})$ satisfying (2.16).

Proof. It follows from step 6 of the PF-AR Method that during the t-th iteration, PF-AR calls the AR-L method with inputs $(\psi_s, \psi_n, x_{t-1}, \epsilon/(7D_t), M_{t-1})$ and outputs a triple (v_t, x_t, M_t). It then follows from (2.11) and (2.12) that the triple (v_t, x_t, M_t) satisfies

$$v_t \in \nabla\psi_s(x_t) + \partial\psi_n(x_t), \qquad \|v_t\| \le \frac{\epsilon D}{D_t}, \tag{2.18}$$

$$M_{t-1} \le M_t \le \max\left\{M_{t-1},\, 2\bar{L}\right\}. \tag{2.19}$$

Let

$$T := 1 + \left\lceil \log_4\left(\frac{D}{D_0}\right)\right\rceil. \tag{2.20}$$

It then follows from the second bound in (2.18), the bound $D_0 \le D$, the fact that step 5 implies that $D_t = 4^t D_0$, the way stopcrit is set in step 7, and the condition of the while loop that PF-AR performs at most T outer iterations where T is as in (2.20).

To show the total gradient evaluation complexity result, we first show that for all $t \ge 0$

$$M_0 \le M_t \le 2\bar{L}. \tag{2.21}$$

We proceed by induction. It follows from the assumption $M_0 \le \bar{2}L$ of Theorem 2.6 that both inequalities in (2.21) must hold for $t = 0$. Let $t \ge 1$ and assume now that $M_0 \le M_{t-1} \le 2\bar{L}$. It then follows from the second inequality in (2.19) and the induction hypothesis that

$$M_t \overset{(2.19)}{\le} \max\left\{M_{t-1}, 2\bar{L}\right\} \le 2\bar{L}.$$

Moreover, it follows from the first inequality in (2.19) and the induction hypothesis that

$$M_0 \le M_{t-1} \le M_t.$$

Hence, it follows by induction that (2.21) holds for all $t \ge 0$.

It then follows from (2.21), the fact that PF-AR calls the AR-L method with inputs as in step 6, and the complexity bound in (2.13) that the call made to the AR-L method in the t-th iteration performs at most

$$6 + C_1\sqrt{\frac{7\bar{L}D_t}{\epsilon}} + \log_2\left(\frac{2\bar{L}}{M_{t-1}}\right) \overset{(2.21)}{\le} 6 + C_1\sqrt{\frac{7\bar{L}D_t}{\epsilon}} + \log_2\left(\frac{2\bar{L}}{M_0}\right) \tag{2.22}$$

gradient evaluations. The total number of gradient evaluations is then bounded by

$$\left[6 + \log_2(2\bar{L}/M_0)\right]T + C_1\sum_{t=1}^{T}\sqrt{\frac{7\bar{L}D_t}{\epsilon}} = \left[6 + \log_2(2\bar{L}/M_0)\right]T + C_1\sqrt{\frac{7\bar{L}}{\epsilon}}\sum_{t=1}^{T}\sqrt{4^t D_0}$$

$$\le \left[6 + \log_2(2\bar{L}/M_0)\right]T + C_1\sqrt{\frac{7\bar{L}}{\epsilon}}\sqrt{D_0}2^{T+1} \overset{(2.20)}{\le} \left[6 + \log_2(2\bar{L}/M_0)\right]T + 8C_1\sqrt{7}\sqrt{\frac{\bar{L}D}{\epsilon}}$$

which implies the complexity result in (2.17) in view of the definition of T in (2.20). The last conclusion of Theorem 2.6 is immediate from (2.17). □

Remark 2.7. *The assumptions in Theorem 2.6 that the inputs M_0 and D_0 satisfy $M_0 \le 2\bar{L}$ and $D_0 \le D$ are not strictly necessary for establishing PF-AR's complexity result. However, they simplify the analysis and the resulting complexity bound. Moreover, both conditions can be easily satisfied. As mentioned in the remarks following the AR-L method, it is easy to construct an M_0 such that $M_0 \le 2\bar{L}$. To construct a D_0 satisfying $D_0 \le D$, select $w_0 \ne x_0$ with $w_0 \in \operatorname{dom}\psi_n$ and define $D_0 := \|x_0 - w_0\|$. It then follows from the definition of the diameter D of $\operatorname{dom}\psi_n$ that $D_0 \le D$.*

2.2 Best Iterate Convergence of an Optimal Augmented Lagrangian Method

In this subsection, we present an **inexact augmented Lagrangian method**, referred to as the **optimal inexact augmented Lagrangian (O-IAL) method**, for solving the linearly constrained convex problem presented in (1.1). The objective of O-IAL is to compute a primal-dual pair that satisfies the approximate primal-dual optimality conditions in (2.2), while achieving optimal first-order complexity bounds.

O-IAL can be viewed as an inexact AL method that maintains approximate dual stationarity at each iteration. Given a primal-dual pair (z_{k-1}, p_{k-1}), an O-IAL iteration uses the PF-AR method described in Subsection 2.1 to compute its next primal iterate, z_k, by approximately minimizing the convex AL subproblem

$$\min_z \mathcal{L}_c(z; p_{k-1}), \tag{2.23}$$

with accuracy less than or equal to $\hat{\rho} > 0$. O-IAL thus maintains approximate dual stationarity, as specified by the dual residual condition $\|r_k\| \leq \hat{\rho}$ in (2.2). After approximately minimizing (2.23), O-IAL then performs the dual update according to (1.6).

Algorithm 2.4 now formally presents the *O-IAL method*.

Algorithm 2.4 O-IAL Method

Input: function pair (ψ_s, ψ_n), initial points $z_0 \in \operatorname{dom}\psi_n$ and $p_0 \in \mathbb{R}^m$, primal-dual tolerance pair $(\hat{\epsilon}, \hat{\rho}) \in \mathbb{R}^2_{++}$, penalty parameter $c > 0$, PF-AR tolerance $\epsilon = \min\left\{\frac{c\hat{\epsilon}^2}{4D}, \hat{\rho}\right\}$, and smoothness and diameter estimates $M_0 := \|\nabla\psi_s(z_0) - \nabla\psi_s(w_0)\|/(4\|z_0 - w_0\|)$ and $D_0 := \|z_0 - w_0\|$, where $w_0 \in \operatorname{dom}\psi_n$ is chosen as in Remark 2.7.

$k \leftarrow 0$
call the PF-AR method (Algorithm 2.3) with function pair $(\mathcal{L}^s_c(\cdot, p_k), \psi_n)$, initial point z_k, estimates M_0 and D_0, and tolerance ϵ; and let (z_{k+1}, r_{k+1}) be its output;
set: $Res_{k+1} \leftarrow \mathcal{A}z_{k+1} - b$;
set: $p_{k+1} = p_k + cRes_{k+1}$;
update: $k \leftarrow k + 1$;
while $\|\mathrm{Res_k}\| > \hat{\epsilon}$ **do**
  call the PF-AR method (Algorithm 2.3) with function pair $(\mathcal{L}^s_c(\cdot, p_k), \psi_n)$, initial point z_k, estimates M_0 and D_0, and tolerance ϵ; and let (z_{k+1}, r_{k+1}) be its output;
  set: $Res_{k+1} \leftarrow \mathcal{A}z_{k+1} - b$;
  set: $p_{k+1} = p_k + cRes_{k+1}$;
  update: $k \leftarrow k + 1$;
end while

Output: triple $(z, p, r) := (z_k, p_k, r_k)$ satisfying (2.2).

It will be shown in Proposition 2.9(b) of the next subsection that every call to the PF-AR method produces a pair (z_k, r_k) satisfying the approximate dual stationarity conditions in (2.2). Consequently, when the while loop terminates, the output triple (z_k, p_k, r_k) satisfies all conditions in (2.2).

Theorem 2.8 provides the main complexity result for O-IAL. The proof of the theorem is given in a collection of propositions and lemmas in Subsection 2.2.1.

Theorem 2.8. *Suppose that Assumption 2.1 holds. Recall that* $\Pi(\cdot)$ *is as in* (1.14), $\bar{L}$ *is the smoothness parameter of* ψ_s, *and* D *is the diameter of* $\operatorname{dom}\psi_n$. *The following statements then hold.*

(a) The O-IAL method requires at most

$$\mathcal{O}\left(\frac{\Pi(p_0)}{\hat{\epsilon}^2 c^2}\sqrt{(\bar{L}+c\|\mathcal{A}\|^2)\max\left\{\frac{D^2}{c\hat{\epsilon}^2},\frac{D}{\hat{\rho}}\right\}}\right) \tag{2.24}$$

resolvent/gradient evaluations to find a $(\hat{\epsilon},\hat{\rho})$-approximate solution of (1.1), i.e., a triple (z,p,r) satisfying (2.2).

(b) By choosing the penalty parameter c as $c=\mathcal{O}(1/\hat{\epsilon})$, the above complexity bound is reduced to the optimal gradient evaluation complexity bound

$$\mathcal{O}\left(\Pi(p_0)\sqrt{\left(\bar{L}+\frac{\|\mathcal{A}\|^2}{\hat{\epsilon}}\right)\max\left\{\frac{D^2}{\hat{\epsilon}},\frac{D}{\hat{\rho}}\right\}}\right). \tag{2.25}$$

Hence, if $\hat{\epsilon}=\hat{\rho}$, it follows that the O-IAL method requires at most $\mathcal{O}(1/\hat{\epsilon})$ total gradient evaluations to find an $(\hat{\epsilon},\hat{\epsilon})$-approximate solution of (1.1).

2.2.1 Proof of Theorem 2.8

This subsection proves Theorem 2.8. We begin with the following proposition, which characterizes the smoothness properties of the augmented Lagrangian subproblem and the behavior of the PF-AR subroutine used in step 7 of O-IAL.

Proposition 2.9. *The following statements about the $(k-1)$-st iteration of O-IAL hold:*

(a) the function $\mathcal{L}_c^s(\cdot,p_{k-1})$ is convex and $(\bar{L}+c\|\mathcal{A}\|^2)$-smooth;

(b) the call made to the PF-AR method in step 7 outputs a pair (z_k,r_k) that satisfies the following relations

$$r_k\in\nabla\psi_s(z_k)+\partial\psi_n(z_k)+\mathcal{A}^*p_k,\quad \|r_k\|\le\min\left\{\frac{c\hat{\epsilon}^2}{4D},\hat{\rho}\right\}, \tag{2.26}$$

within at most

$$\mathcal{O}\left(\sqrt{(\bar{L}+c\|\mathcal{A}\|^2)\max\left\{\frac{D^2}{c\hat{\epsilon}^2},\frac{D}{\hat{\rho}}\right\}}\right) \tag{2.27}$$

gradient evaluations.

Proof. We first show part a). The fact that $\mathcal{L}_c^s(\cdot;p_{k-1})$ is convex is immediate from its definition in (1.5) and the fact that ψ_s is convex. To show the second result, let z and z' be in $\mathbb{E}$. The definition of $\mathcal{L}_c^s(\cdot;p_{k-1})$ in (1.5), the fact that ψ_s is $\bar{L}$-smooth, and the triangle inequality, then imply that

$$\begin{aligned}\|\nabla\mathcal{L}_c^s(z;p_{k-1})-\nabla\mathcal{L}_c^s(z';p_{k-1})\| &\le \|\nabla\psi_s(z)-\nabla\psi_s(z')\|+\|c\mathcal{A}^*[(\mathcal{A}z-b)-(\mathcal{A}z'-b)]\|\\ &\le \bar{L}\|z-z'\|+\|c\mathcal{A}^*\mathcal{A}(z-z')\|\\ &\le (\bar{L}+c\|\mathcal{A}\|^2)\|z-z'\|.\end{aligned}$$

(b) It is easy to see from the definition of $\mathcal{L}_c^s(z_k;p_{k-1})$ in (1.5) and the update formula for p_k in step 9 that

$$\nabla\mathcal{L}_c^s(z_k;p_{k-1})=\nabla\psi_s(z_k)+\mathcal{A}^*(p_{k-1}+c(\mathcal{A}z_k-b))=\nabla\psi_s(z_k)+\mathcal{A}^*p_k. \tag{2.28}$$

It then follows from this observation, the fact that O-IAL calls PF-AR in its step 7 with function pair $(\mathcal{L}_c^s(\cdot, p_k), \psi_n)$ and tolerance $\epsilon = \min\{(c\hat{\epsilon}^2)/4D, \hat{\rho}\}$, and the inclusion in (2.16) that PF-AR outputs a pair (z_k, r_k) satisfying

$$r_k \in \nabla\psi_s(z_k) + \partial\psi_n(z_k) + \mathcal{A}^* p_k, \quad \|r_k\| \leq \min\left\{\frac{c\hat{\epsilon}^2}{4D}, \hat{\rho}\right\},$$

which implies that (2.26) holds.

The gradient evaluation complexity bound of the PF-AR method call in (2.27) is immediate from fact that O-IAL calls PF-AR in its step 7 with function pair $(\mathcal{L}_c^s(\cdot, p_k), \psi_n)$ and tolerance $\epsilon = \min\{(c\hat{\epsilon}^2)/4D, \hat{\rho}\}$, part(a), and the last conclusion of Theorem 2.6. □

The next lemma derives a lower bound on an inner product involving the primal feasibility residual $\mathcal{A}z_k - b$. This estimate will play a key role in the complexity analysis.

Lemma 2.10. *For any iteration index $k \geq 1$ generated by the O-IAL method, it holds that*

$$\langle \mathcal{A}z_k - b, p_* - p_k\rangle \geq \langle r_k, z_* - z_k\rangle, \tag{2.29}$$

where the pair (z_*, p_*) satisfies (2.1).

Proof. It is easy to see that the inclusion in (2.26) and the fact that $\psi(z) := \psi_s(z) + \psi_n(z)$ is a convex function implies that $r_k \in \partial\psi(z_k) + \mathcal{A}^* p_k$. This inclusion and the subdifferential inequality at z_* imply that

$$\psi(z_*) - \psi(z_k) \geq \langle r_k - \mathcal{A}^* p_k, z_* - z_k\rangle. \tag{2.30}$$

Now, suppose that (z_*, p_*) is an optimal primal-dual pair satisfying Assumption 2.1. It then follows that (z_*, p_*) satisfies the inclusion in (2.1). It then follows from this inclusion and the subdifferential inequality at z_k that

$$\psi(z_k) - \psi(z_*) \geq \langle -\mathcal{A}^* p_*, z_k - z_*\rangle. \tag{2.31}$$

It then follows by adding (2.30) and (2.31) that

$$\langle \mathcal{A}^*(p_* - p_k), z_k - z_*\rangle \geq \langle r_k, z_* - z_k\rangle. \tag{2.32}$$

The fact that z_* satisfies the second relation in (2.1) together with (2.32) then imply that

$$\langle \mathcal{A}z_k - b, p_* - p_k\rangle \overset{(2.1)}{=} \langle \mathcal{A}(z_k - z_*), p_* - p_k\rangle = \langle z_k - z_*, \mathcal{A}^*(p_* - p_k)\rangle \overset{(2.32)}{\geq} \langle r_k, z_* - z_k\rangle,$$

which immediately implies the result. □

The following lemma combines the previous inner-product estimate with a telescoping argument to derive a bound on the primal feasibility residuals.

Lemma 2.11. *For any iteration index $k \geq 1$ generated by O-IAL, it holds that*

$$\sum_{\ell=1}^{k} \|\mathcal{A}z_\ell - b\|^2 \leq \frac{\|p_* - p_0\|^2}{c^2} + \frac{k\hat{\epsilon}^2}{2} \tag{2.33}$$

where p_* is as in (2.1) and p_0, c, and $\hat{\epsilon}$ are inputs to O-IAL.

Proof. Suppose that $k \geq 1$ is an index generated by O-IAL and let $\ell \geq 1$ be an index such that $\ell \leq k$. Relation (2.29), Cauchy-Schwarz inequality, and the fact that second relation in (2.26) holds for $k = \ell$ imply that

$$\langle \mathcal{A}z_\ell - b, p_* - p_\ell \rangle \overset{(2.29)}{\geq} \langle r_\ell, z_* - z_\ell \rangle \overset{(1.2)}{\geq} -D\|r_\ell\| \overset{(2.26)}{\geq} -\frac{c\hat{\epsilon}^2}{4} \tag{2.34}$$

holds for all indices $\ell \leq k$. Using the above relation, the update for p_k in step 9, and completing the square, we then have that

$$\begin{aligned} \|p_* - p_{\ell-1}\|^2 - \|p_* - p_\ell\|^2 &= \|p_{\ell-1} - p_\ell\|^2 + 2\langle p_\ell - p_{\ell-1}, p_* - p_\ell \rangle \\ &= c^2\|\mathcal{A}z_\ell - b\|^2 + 2c\langle \mathcal{A}z_\ell - b, p_* - p_\ell \rangle. \end{aligned} \tag{2.35}$$

It then follows by summing relation (2.35) from $\ell = 1$ to k and re-arranging terms that

$$c^2 \sum_{\ell=1}^{k} \|\mathcal{A}z_\ell - b\|^2 \leq \|p_* - p_0\|^2 + \frac{kc^2\hat{\epsilon}^2}{2}.$$

The result then immediately follows from dividing both sides of the above relation by c^2. □

The following proposition characterizes the number of outer iterations that O-IAL performs to find a triple (z, p, r) satisfying the approximate primal-dual optimality conditions in (2.2).

Proposition 2.12. *The O-IAL method performs at most*

$$\left\lceil \frac{2\Pi(p_0)}{\hat{\epsilon}^2 c^2} \right\rceil \tag{2.36}$$

outer iterations to find a triple (z, p, r) satisfying (2.2), where $\Pi(\cdot)$ is as in (1.14), $\hat{\epsilon}$ is an input tolerance, and c is the input penalty parameter.

Proof. It follows from Proposition 2.9(b) that, for each outer index $k \geq 1$ generated by O-IAL, the call made to the PF-AR method in step 7 outputs a pair (z_k, r_k) satisfying

$$r_k \in \nabla\psi_s(z_k) + \partial\psi_n(z_k) + \mathcal{A}^* p_k, \qquad \|r_k\| \leq \hat{\rho}.$$

Hence, by the condition in the while loop of O-IAL, the definition of Res_k in step 8, and the fact that the above inclusion holds for every iteration index $k \geq 1$, if O-IAL terminates, then its output $(z, p, r) = (z_k, p_k, r_k)$ satisfies (2.2).

It remains to show that O-IAL exits the while loop, and therefore terminates, after at most (2.36) outer iterations. To this end, let p_* be an arbitrary optimal solution of the Lagrangian dual of (1.1). Relation (2.33) implies that

$$k \min_{1 \leq \ell \leq k} \|\mathcal{A}z_\ell - b\|^2 \leq \sum_{\ell=1}^{k} \|\mathcal{A}z_\ell - b\|^2 \leq \frac{\|p_* - p_0\|^2}{c^2} + \frac{k\hat{\epsilon}^2}{2}.$$

Dividing both sides by k then gives

$$\min_{1 \leq l \leq k} \|\mathcal{A}z_l - b\|^2 \leq \frac{\|p_* - p_0\|^2}{c^2 k} + \frac{\hat{\epsilon}^2}{2}. \tag{2.37}$$

The outer iteration bound (2.36) then follows from (2.37), the fact that p_* is an arbitrary optimal multiplier, and a simple proof by contradiction. The result of the proposition then follows from the outer iteration complexity bound and the above observations. □

We are now ready to prove Theorem 2.8.

Proof of Theorem 2.8. (a) It follows from Proposition 2.12 that the O-IAL method performs at most

$$\left\lceil \frac{2\Pi(p_0)}{\hat{\epsilon}^2 c^2} \right\rceil \tag{2.38}$$

outer iterations to find a triple (z, p, r) satisfying (2.2). Moreover, it follows from Proposition 2.9(b) that the call to the PF-AR method performs at most

$$\mathcal{O}\left(\sqrt{(\bar{L} + c\|\mathcal{A}\|^2)\max\left\{\frac{D^2}{c\hat{\epsilon}^2}, \frac{D}{\hat{\rho}}\right\}}\right)$$

gradient/resolvent evaluations during each outer iteration of O-IAL. The conclusion of part (a) then follows from multiplying these two complexity bounds.

(b) It follows immediately from part (a) and the assumption that $c = \mathcal{O}(1/\hat{\epsilon})$ that the O-IAL method performs at most

$$\mathcal{O}\left(2\Pi(p_0)\sqrt{\left(\bar{L} + \frac{\|\mathcal{A}\|^2}{\hat{\epsilon}}\right)\max\left\{\frac{D^2}{\hat{\epsilon}}, \frac{D}{\hat{\rho}}\right\}}\right) \tag{2.39}$$

total gradient evaluations to find an $(\hat{\epsilon}, \hat{\rho})$-approximate solution. The first conclusion of part (b) follows immediately from this observation. The second conclusion of part (b) follows immediately from the first conclusion and considering $\hat{\epsilon} = \hat{\rho}$.

□

2.3 Best-Iterate Convergence of a Restarted Parameter-Free AL Method

The O-IAL method in Algorithm 2.4 is not parameter-free, since it requires knowledge of the domain diameter D. In this subsection, we develop OPF-IAL, a restarted parameter-free variant of O-IAL that uses a guess-and-check procedure for D. Each cycle runs O-IAL for a prescribed number of iterations using the current diameter guess in (1.2); if no triple (z_k, p_k, r_k) satisfying (2.2) is found, OPF-IAL restarts with a larger diameter guess and a larger iteration budget.

As in O-IAL, the goal is to compute a $(\hat{\epsilon}, \hat{\rho})$-approximate primal-dual solution satisfying (2.2). When $\hat{\epsilon} = \hat{\rho}$, OPF-IAL retains the optimal $\mathcal{O}(1/\hat{\epsilon})$ complexity bound of O-IAL under Assumption 2.1. We now formally describe the OPF-IAL method.

Algorithm 2.5 OPF-IAL Method

Input: function pair (ψ_s, ψ_n), initial points $z_0 \in \operatorname{dom}\psi_n$ and $p_0 \in \mathbb{R}^m$, primal-dual tolerance pair $(\hat{\epsilon}, \hat{\rho}) \in \mathbb{R}^2_{++}$, penalty parameter $c > 0$, initial guess $I_0 > 0$, and smoothness and diameter estimates $M_0 := \|\nabla\psi_s(z_0) - \nabla\psi_s(w_0)\|/(4\|z_0 - w_0\|)$ and $D_0 := \|z_0 - w_0\|$, where $w_0 \in \operatorname{dom}\psi_n$ is chosen as in Remark 2.7.

1: $\ell \leftarrow 1$

2: stopcrit $\leftarrow \infty$

Initialization: set: $k \leftarrow 0$, $z_0 \leftarrow z_{\ell-1}$, and $p_0 \leftarrow p_0$.

3: **while** stopcrit $> \hat{\epsilon}$ (main outer loop) **do**

4: set $k \leftarrow k + 1$

5: call PF-AR method (Algorithm 2.3) with function pair $(\mathcal{L}_c^s(\cdot, p_{k-1}), \psi_n(\cdot))$, initial point z_{k-1}, estimates M_0 and D_0, and tolerance $\epsilon = \min\left\{\frac{\hat{\epsilon}}{4D_{\ell-1}}, \hat{\rho}\right\}$, and let (z_k, r_k) be its output;

6: $p_k \leftarrow p_{k-1} + c(\mathcal{A}z_k - b)$;

7: **if** $k > \lceil 2I_{\ell-1} \rceil$ **then**

8: **restart**,

9: i.e., set $z_\ell = z_k$, $D_\ell = 2D_{\ell-1}$, $I_\ell = 2I_{\ell-1}$, and $\ell \leftarrow \ell + 1$,

10: and execute **Initialization** above;

11: **else**

12: set stopcrit $\leftarrow \|\mathcal{A}z_k - b\|$.

13: **end if**

14: **end while**

Output: triple $(z, p, r) := (z_k, p_k, r_k)$ satisfying (2.2)

We now make several remarks about the structure of OPF-IAL. The algorithm involves three nested types of iterations. Iterations indexed by ℓ are referred to as *cycles*, iterations indexed by k are referred to as *outer iterations*, and the iterations performed internally by the PF-AR method in step 5 are referred to as *inner iterations*.

During each outer iteration of the ℓ-th cycle, step 5 calls the PF-AR method to approximately minimize the augmented Lagrangian function $\mathcal{L}_c(z, p) := \mathcal{L}_c^s(z, p) + \psi_n(z)$ with tolerance depending on the current estimate $D_{\ell-1}$ of the diameter D. Moreover, step 7 checks whether the current outer iteration count k exceeds the threshold $\lceil 2I_{\ell-1} \rceil$. If this condition is satisfied, then OPF-IAL restarts and a new cycle begins. At the restart, the estimates D_ℓ and I_ℓ are updated according to $D_\ell = 2D_{\ell-1}$ and $I_\ell = 2I_{\ell-1}$, and the last primal iterate generated during the previous cycle is used to initialize the next cycle.

Finally, like O-IAL, the while loop in steps 3–14 of OPF-IAL terminates once $\|\mathcal{A}z_k - b\| \le \hat{\epsilon}$, since OPF-IAL maintains approximate dual stationarity at every outer iteration.

Theorem 2.13 provides the main complexity result for OPF-IAL. The proof of Theorem 2.13 is given in a collection of propositions and lemmas in Subsection 2.3.1.

Theorem 2.13. *If OPF-IAL in Algorithm 2.5 uses penalty parameter input $c = 1/\hat{\epsilon}$, it performs at most*

$$\mathcal{O}\left(\left(1 + \left\lceil \max\left\{ \log_2\left(\frac{D}{D_0}\right), \log_2\left(\frac{\Pi(p_0)}{I_0}\right) \right\} \right\rceil\right) \left[1 + \left\lceil \max\left\{ \Pi(p_0), \frac{DI_0}{D_0} \right\} \right\rceil\right] \times \sqrt{\left(\bar{L} + \frac{\|\mathcal{A}\|^2}{\hat{\epsilon}}\right) \max\left\{ \frac{D^2}{\hat{\epsilon}}, \frac{\Pi(p_0) D_0 D}{I_0 \hat{\epsilon}}, \frac{D}{\hat{\rho}} \right\}} \right)$$

resolvent/gradient evaluations to find an $(\hat{\epsilon}, \hat{\rho})$-approximate solution of (1.1) according to the criterion in (2.2). Hence, if $\hat{\rho} = \hat{\epsilon}$, it follows that OPF-IAL performs at most $\mathcal{O}(1/\hat{\epsilon})$ total gradient evaluations to find an $(\hat{\epsilon}, \hat{\epsilon})$-approximate solution of (1.1).

2.3.1 Proof of Theorem 2.13

This subsection is devoted to the proof of Theorem 2.13. We begin with a key proposition that provides a bound on the number of outer iterations that a cycle of OPF-IAL performs if it is made with a correct guess for the diameter.

Proposition 2.14. *Suppose that the ℓ-th cycle of OPF-IAL is performed with a guess $D_{\ell-1} \geq D$, where D is as in (1.2). It then holds that the ℓ-th cycle of OPF-IAL performs at most*

$$\lceil 2\Pi(p_0) \rceil \tag{2.40}$$

outer iterations, i.e, those indexed by k, to find a triple $(z, p, r) := (z_k, p_k, r_k)$ satisfying (2.2).

Proof. Suppose that the ℓ-th cycle of OPF-IAL is performed with a guess $D_{\ell-1} \geq D$ and let $k \geq 1$ be an iteration index generated during the ℓ-th cycle. It follows from relations (1.2), (2.26) and (2.29) that

$$\langle \mathcal{A}z_k - b, p_* - p_k \rangle \overset{(2.29)}{\geq} \langle r_k, z_* - z_k \rangle \overset{(1.2)}{\geq} -D\|r_k\| \overset{(2.26)}{\geq} -\frac{D\hat{\epsilon}}{4D_{\ell-1}} \geq -\frac{D\hat{\epsilon}}{4D} = -\frac{\hat{\epsilon}}{4}. \tag{2.41}$$

It follows from a similar argument as in the proof of (2.35), the update rule for p_k in step 6 of OPF-IAL, and the fact that $c = 1/\hat{\epsilon}$ that

$$\begin{aligned} \|p_* - p_{k-1}\|^2 - \|p_* - p_k\|^2 &= \frac{1}{\hat{\epsilon}^2}\|\mathcal{A}z_k - b\|^2 + \frac{2}{\hat{\epsilon}}\langle \mathcal{A}z_k - b, p_* - p_k \rangle \\ &\overset{(2.41)}{\geq} \frac{1}{\hat{\epsilon}^2}\|\mathcal{A}z_k - b\|^2 - \frac{1}{2}. \end{aligned} \tag{2.42}$$

It follows from summing the above inequalities from $k = 1$ to m, passing to the minimum, and multiplying both sides by $\hat{\epsilon}^2$ that

$$\min_{1 \leq k \leq m} \|\mathcal{A}z_k - b\|^2 \leq \frac{\hat{\epsilon}^2\|p_* - p_0\|^2}{m} + \frac{\hat{\epsilon}^2}{2}.$$

It then follows from this bound and a simple proof by contradiction that the ℓ-th cycle performs at most $\lceil 2\Pi(p_0) \rceil$ outer iterations. Moreover, for every outer iteration generated during the cycle, a similar argument as in Proposition 2.9 implies that

$$r_k \in \nabla\psi_s(z_k) + \partial\psi_n(z_k) + \mathcal{A}^* p_k, \qquad \|r_k\| \leq \hat{\rho}$$

for all outer indices k generated. Therefore, the returned triple (z, p, r) satisfies (2.2). □

Proposition 2.15. *OPF-IAL performs at most*

$$1 + \left\lceil \max\left\{ \log_2\left(\frac{D}{D_0}\right), \log_2\left(\frac{\Pi(p_0)}{I_0}\right) \right\} \right\rceil$$

cycles to find a triple (z, p, r) satisfying (2.2).

Proof. It follows from Proposition 2.14 that if the ℓ-th cycle of OPF-IAL is performed with $D_{\ell-1} \geq D$ and $I_{\ell-1} \geq \Pi(p_0)$, then OPF-IAL finds a triple (z_k, p_k, r_k) satisfying (2.2). Now, using the definitions of D_ℓ and I_ℓ, it follows that $D_{\ell-1} \geq D$ and $I_{\ell-1} \geq \Pi(p_0)$ if

$$D_{\ell-1} = 2^{\ell-1} D_0 \geq D, \text{ and } I_{\ell-1} = 2^{\ell-1} I_0 \geq \Pi(p_0).$$

Hence, by re-arranging the above inequalities and using the fact that $D_0 \leq D$, it follows that if the number of cycles ℓ satisfies

$$\ell \geq 1 + \max\left\{ \log_2\left(\frac{D}{D_0}\right), \log_2\left(\frac{\Pi(p_0)}{I_0}\right) \right\},$$

then OPF-IAL must find a triple (z, p, r) satisfying (2.2). The result of the lemma then immediately follows from this conclusion. □

The following lemma provides upper and lower bounds on the quantities D_ℓ and I_ℓ.

Lemma 2.16. *The following inequalities hold for any cycle index $\ell \geq 1$ generated by OPF-IAL:*

$$D_0 \leq D_\ell \leq \max\left\{4D, \frac{4\Pi(p_0)D_0}{I_0}\right\} \tag{2.43}$$

$$I_0 \leq I_\ell \leq \max\left\{4\Pi(p_0), \frac{4DI_0}{D_0}\right\}. \tag{2.44}$$

Proof. The facts that $D_0 \leq D_\ell$ and $I_0 \leq I_\ell$ follow immediately from the update rules of D_ℓ and I_ℓ in step 9 of OPF-IAL. To show the second inequality in (2.43), first observe that it follows from Proposition 2.15 that the number of cycles ℓ that OPF-IAL performs satisfies

$$\ell \leq 1 + \left\lceil \max\left\{\log_2\left(\frac{D}{D_0}\right), \log_2\left(\frac{\Pi(p_0)}{I_0}\right)\right\}\right\rceil \leq 2 + \max\left\{\log_2\left(\frac{D}{D_0}\right), \log_2\left(\frac{\Pi(p_0)}{I_0}\right)\right\}.$$

Hence, it follows from the update rule for D_ℓ in step 9 of Algorithm 2.5 that the following inequalities hold for any cycle index $\ell \geq 1$ generated by OPF-IAL:

$$D_\ell = 2^\ell D_0 \leq 2^{2+\max\left\{\log_2\left(\frac{D}{D_0}\right), \log_2\left(\frac{\Pi(p_0)}{I_0}\right)\right\}} D_0 = 2^2 \cdot \max\left\{\frac{D}{D_0}, \frac{\Pi(p_0)}{I_0}\right\} D_0. \tag{2.45}$$

The second inequality in (2.43) then immediately follows from (2.45). Likewise, the second inequality in (2.44) follows from a similar argument as the above argument and the fact that $I_\ell = 2^\ell I_0$. □

The following proposition upper bounds the total number of outer iterations, i.e. those indexed by k, that OPF-IAL performs during each of its cycles. The proposition also characterizes the number of inner iterations that the call to the PF-AR method performs during each of OPF-IAL's outer iterations.

Proposition 2.17. *Let $\ell \geq 1$ be a cycle index generated by OPF-IAL. The following statements hold.*

(a) The ℓ-th cycle of OPF-IAL performs at most

$$1 + \left\lceil \max\left\{8\Pi(p_0), \frac{8DI_0}{D_0}\right\}\right\rceil \tag{2.46}$$

outer iterations, i.e., iterations indexed by k;

(b) During step 5 of each of OPF-IAL's outer iterations, the PF-AR method performs at most

$$\mathcal{O}\left(\sqrt{\left(\bar{L} + \frac{\|\mathcal{A}\|^2}{\hat{\epsilon}}\right)} \max\left\{\frac{D^2}{\hat{\epsilon}}, \frac{\Pi(p_0)D_0 D}{I_0\hat{\epsilon}}, \frac{D}{\hat{\rho}}\right\}\right) \tag{2.47}$$

gradient evaluations.

Proof. (a) Let $\ell \geq 1$ be a cycle index generated by OPF-IAL. It follows from the inequality check that OPF-IAL performs in its step 7 that its ℓ-th cycle performs at most $1+\lceil 2I_{\ell-1}\rceil$ outer iterations, i.e., iterations indexed by k. The result then follows from this bound and the upper bound for $I_{\ell-1}$ in (2.44).

(b) It follows from the last conclusion of Theorem 2.6, a similar argument as in Proposition 2.9(b), and the fact that the OPF-IAL method calls the PF-AR method in its step 5 with tolerance pair $\epsilon = \min\left\{\frac{\hat{\epsilon}}{4D_{\ell-1}}, \hat{\rho}\right\}$ that the PF-AR method performs at most

$$\mathcal{O}\left(\sqrt{(\bar{L} + c\|\mathcal{A}\|^2)} \max\left\{\frac{D_{\ell-1}D}{c\hat{\epsilon}^2}, \frac{D}{\hat{\rho}}\right\}\right) \tag{2.48}$$

gradient evaluations during each of its calls. The conclusion of the proposition then follows from (2.48), the fact that $c = 1/\hat{\epsilon}$, and the upper bound for $D_{\ell-1}$ in (2.43). □

We are now ready to prove Theorem 2.13.

Proof of Theorem 2.13. It follows from Proposition 2.15 that OPF-IAL performs at most

$$1 + \left\lceil \max\left\{ \log_2\left(\frac{D}{D_0}\right), \log_2\left(\frac{\Pi(p_0)}{I_0}\right) \right\} \right\rceil$$

cycles to find a triple (z, p, r) satisfying (2.2). Moreover, it follows from Proposition 2.17(a) that each cycle of OPF-IAL performs at most

$$1 + \left\lceil \max\left\{ 8\Pi(p_0), \frac{8DI_0}{D_0} \right\} \right\rceil$$

outer iterations. Finally, Proposition 2.17(b) implies that the PF-AR method performs at most

$$\mathcal{O}\left(\sqrt{\left(\bar{L} + \frac{\|\mathcal{A}\|^2}{\hat{\epsilon}}\right) \max\left\{ \frac{D^2}{\hat{\epsilon}}, \frac{\Pi(p_0) D_0 D}{I_0 \hat{\epsilon}}, \frac{D}{\hat{\rho}} \right\}} \right)$$

gradient evaluations during each of OPF-IAL's outer iterations. The total complexity result of OPF-IAL presented in Theorem 2.13 then follows from multiplying the above three complexity bounds and absorbing the constants in the big O. □

2.4 Last-Iterate Convergence of an Adaptive Parameter-Free AL Method

This subsection presents an adaptive parameter-free inexact augmented Lagrangian method, namely, APF-IAL, with last-iterate convergence guarantees for approximately solving the primal-dual pair (1.1) and (1.13). Throughout this subsection, we consider problem (1.1) under the following two assumptions.

Assumption 2.18. *The function ψ_n is M_n-Lipschitz continuous on* $\operatorname{dom} \psi_n$*;*

Assumption 2.19. *There exists $\bar{z} \in \operatorname{int}(\operatorname{dom} \psi_n)$ such that $\mathcal{A}\bar{z} = b$.*

Unlike O-IAL and OPF-IAL, APF-IAL achieves stronger last-iterate convergence rate guarantees while the former two methods achieve best-iterate convergence rate guarantees. Like OPF-IAL, APF-IAL is parameter-free and requires no knowledge of any parameters underlying (1.1) and (1.13). However, unlike OPF-IAL and O-IAL, APF-IAL has the advantage that it is provably able to adaptively update the penalty parameter c and the tolerance that it uses to approximately minimize the AL function at every iteration. Finally given a tolerance pair $(\hat{\epsilon}, \hat{\rho}) \in \mathbb{R}_{++}$ and assuming that $\hat{\epsilon} = \hat{\rho}$, APF-IAL, like O-IAL and OPF-IAL, achieves an optimal complexity bound $\mathcal{O}(1/\hat{\epsilon})$ for finding an approximate primal-dual pair of (1.1) and (1.13) according to the criterion in (2.2).

The APF-IAL method is formally presented in Algorithm 2.6 below.

Algorithm 2.6 APF-IAL Method

Input: function pair (ψ_s, ψ_n), initial points $z_0 \in \operatorname{dom}\psi_n$ and $p_0 \in \mathbb{R}^m$, primal-dual tolerance pair $(\hat{\epsilon}, \hat{\rho}) \in \mathbb{R}^2_{++}$, an initial penalty parameter $c_1 > 0$, scalars $\alpha > 1$ and $\omega > 1$, a scalar $\tilde{\epsilon}_1 \geq \hat{\rho}$, and smoothness and diameter estimates $M_0 := \|\nabla\psi_s(z_0) - \nabla\psi_s(w_0)\|/(4\|z_0 - w_0\|)$ and $D_0 := \|z_0 - w_0\|$, where $w_0 \in \operatorname{dom}\psi_n$ is chosen as in Remark 2.7.

1: set $k \leftarrow 0$;
2: set $\mathrm{Res_p} \leftarrow \mathcal{A}z_k - b$ and $\mathrm{Res_d} \leftarrow \infty$;
3: **while** $\|\mathrm{Res_p}\| > \hat{\epsilon}$ or $\|\mathrm{Res_d}\| > \hat{\rho}$ (main outer loop) **do**
4: set $k \leftarrow k + 1$;
5: **if** $\|\mathrm{Res_p}\| > \hat{\epsilon}$ **then**
6: $\epsilon_k = \max\{\tilde{\epsilon}_k, \hat{\rho}\}$
7: **else**
8: $\epsilon_k = \hat{\rho}$
9: **end if**
10: call PF-AR method (Algorithm 2.3) with function pair $(\mathcal{L}^s_c(\cdot, p_{k-1}), \psi_n(\cdot))$, initial point z_{k-1}, estimates M_0 and D_0, and tolerance $\epsilon = \epsilon_k$, and let (z_k, r_k) be its output;
11: set $\mathrm{Res_p} \leftarrow \mathcal{A}\mathrm{z_k} - \mathrm{b}$ and $\mathrm{Res_d} \leftarrow \mathrm{r_k}$;
12: update $p_k = p_{k-1} + c_k\mathrm{Res_p}$;
13: set $c_{k+1} = \alpha c_k$ and $\tilde{\epsilon}_{k+1} = \tilde{\epsilon}_k/\omega$;
14: **end while**

Output: triple $(z, p, r) := (z_k, p_k, r_k)$ satisfying (2.2).

We make a few remarks on APF-IAL. First, the while loop in steps 3–14 terminates only when both primal feasibility and dual stationarity are small, namely, $\|\mathrm{Res_p}\| \leq \hat{\epsilon}$ and $\|\mathrm{Res_d}\| \leq \hat{\rho}$. Unlike O-IAL and OPF-IAL, APF-IAL does not maintain approximate dual stationarity at every outer iteration. Second, ϵ_k is the PF-AR tolerance used in step 10: it is set to $\hat{\rho}$ once primal feasibility is sufficiently small, and otherwise to the decreasing adaptive tolerance $\tilde{\epsilon}_k$. Third, step 12 performs the standard multiplier update, as in O-IAL and OPF-IAL. Finally, step 13 updates the penalty and tolerance according to $c_{k+1} = \alpha c_k$ and $\tilde{\epsilon}_{k+1} = \tilde{\epsilon}_k/\omega$, where $\alpha, \omega > 1$. Gradually increasing the penalty parameter is typically more practical than using a large fixed penalty, while still yielding an optimal convergence rate.

The following result describes the main complexity result of APF-IAL. Its proof is given in Subsection 2.4.1.

Theorem 2.20. *Let*

$$\bar{d} := \operatorname{dist}(\bar{z}, \operatorname{bd}(\operatorname{dom}\psi_n)), \quad \nabla_s := \sup_{z \in \operatorname{dom}\psi_n} \|\nabla\psi_s(z)\| \tag{2.49}$$

where here $\operatorname{bd}(\operatorname{dom}\psi_n)$ *denotes the boundary of* $\operatorname{dom}\psi_n$. *APF-IAL performs at most*

$$\mathcal{O}\left(\frac{\alpha^{3/2}\|\mathcal{A}\|D\sqrt{M_n + \nabla_s + \tilde{\epsilon}_1}}{\sqrt{\bar{d}\sigma^+_{\mathcal{A}}}(\sqrt{\alpha} - 1)\sqrt{\hat{\epsilon}\hat{\rho}}}\right) \tag{2.50}$$

total gradient evaluations to find an $(\hat{\epsilon}, \hat{\rho})$*-approximate solution according to the criterion in* (2.2), *where* $(\hat{\epsilon}, \hat{\rho})$, α, *and* c_1 *are inputs,* D *and* M_n *are as in* (1.2) *and Assumption 2.18, respectively, and* $\sigma^+_{\mathcal{A}}$ *is the smallest positive singular value of the linear transformation* $\mathcal{A}$. *Hence, it follows that APF-IAL performs at most* $\mathcal{O}(1/\hat{\epsilon})$ *total gradient evaluations to find an* $(\hat{\epsilon}, \hat{\epsilon})$*-approximate solution of* (1.1).

2.4.1 Proof of Theorem 2.20

This subsection is dedicated to proving Theorem 2.20. The following key result from [43] is presented with minor changes in notation. It will be instrumental in showing that the sequence of Lagrange multipliers $\{p_k\}$ is bounded.

Lemma 2.21. *Assume that ψ_n satisfies Assumption 2.18, $\mathcal{A} : \mathbb{E} \to \mathbb{R}^m$ is a linear transformation satisfying Assumption 2.19, and the triple $(z, p, s) \in \mathbb{E} \times \mathbb{R}^m \times \mathbb{E}$ satisfies $s \in \partial\psi_n(z) + \mathcal{A}^* p$. Then:*

(a) there holds

$$\bar{d}\sigma_{\mathcal{A}}^{+}\|p\| \leq 2D\,(M_n + \|s\|) - \langle p, \mathcal{A}z - b\rangle; \tag{2.51}$$

(b) if, in addition,

$$p = p^{-} + \chi(\mathcal{A}z - b) \tag{2.52}$$

for some $p^{-} \in \mathbb{R}^m$ and $\chi > 0$, we have

$$\|p\| \leq \max\left\{\|p^{-}\|, \frac{2D(M_n + \|s\|)}{\bar{d}\sigma_{\mathcal{A}}^{+}}\right\}. \tag{2.53}$$

The following proposition characterizes the properties and the output of the call made to the PF-AR method during the k-th iteration of APF-IAL. We omit the proof of the proposition since it is nearly identical to the proof of Proposition 2.9 with the only differences being that the static penalty parameter c is replaced with the adaptive penalty parameter c_k and that the quantity $\min\left\{(c\hat{\epsilon}^2)/4D, \hat{\rho}\right\}$ is now replaced by ϵ_k since the PF-AR method is called in APF-IAL with the latter tolerance.

Proposition 2.22. *The following statements about the k-th iteration of APF-IAL hold:*

(a) the call made to the PF-AR method in step 10 outputs a pair (z_k, r_k) satisfying

$$r_k \in \nabla\psi_s(z_k) + \partial\psi_n(z_k) + \mathcal{A}^* p_k, \quad \|r_k\| \leq \epsilon_k; \tag{2.54}$$

(b) the call made to the PF-AR method performs at most

$$\mathcal{O}\left(\sqrt{\frac{(\bar{L} + c_k\|\mathcal{A}\|^2)\,D}{\epsilon_k}}\right)$$

gradient evaluations to find a pair (z_k, r_k) satisfying (2.54).

The following result provides upper and lower bounds on the sequence of tolerances $\{\epsilon_k\}$ generated in steps 5 to 9 of APF-IAL.

Lemma 2.23. *For any iteration index $k \geq 1$ generated by APF-IAL, we have*

$$\hat{\rho} \leq \epsilon_k \leq \tilde{\epsilon}_1. \tag{2.55}$$

Proof. It follows from the way ϵ_k is set in steps 5 to 9 of APF-IAL that $\epsilon_k = \max\{\tilde{\epsilon}_k, \hat{\rho}\}$ or $\epsilon_k = \hat{\rho}$, which immediately implies the first bound in (2.55). The second bound in (2.55) follows immediately from this observation, the fact that $\tilde{\epsilon}_k$ is a decreasing sequence, and the fact that $\tilde{\epsilon}_1 \geq \hat{\rho}$. □

Proposition 2.24. *The sequence $\{p_k\}$ generated by APF-IAL satisfies*

$$\|p_k\| \leq \max\left\{\frac{2D(M_n + \nabla_s + \tilde{\epsilon}_1)}{\bar{d}\sigma_{\mathcal{A}}^{+}}, \|p_0\|\right\} \quad \forall k \geq 0. \tag{2.56}$$

Proof. It follows from the definition of ∇_s in (2.49), the inequality in (2.54), the second relation in (2.55), and triangle inequality that the following inequality holds

$$\|r_k - \nabla\psi_s(z_k)\| \leq \|r_k\| + \|\nabla\psi_s(z_k)\| \overset{(2.54)}{\leq} \epsilon_k + \|\nabla\psi_s(z_k)\| \overset{(2.55)}{\leq} \tilde{\epsilon}_1 + \|\nabla\psi_s(z_k)\| \overset{(2.49)}{\leq} \tilde{\epsilon}_1 + \nabla_s \quad (2.57)$$

for all $k \geq 1$. Now, it follows from the above inequality, the fact that the inclusion in (2.54) implies that $r_k - \nabla\psi_s(z_k) \in \partial\psi_n(z_k) + \mathcal{A}^* p_k$, and Lemma 2.21(b) with $(z, p, s) = (z_k, p_k, r_k - \nabla\psi_s(z_k))$ that the following holds

$$\|p_k\| \overset{(2.53)}{\leq} \max\left\{\|p_{k-1}\|, \frac{2D(M_n + \|r_k - \nabla\psi_s(z_k)\|)}{\bar{d}\sigma_{\mathcal{A}}^+}\right\} \overset{(2.57)}{\leq} \max\left\{\|p_{k-1}\|, \frac{2D(M_n + \nabla_s + \tilde{\epsilon}_1)}{\bar{d}\sigma_{\mathcal{A}}^+}\right\}$$

for all $k \geq 1$. The conclusion of the proposition then follows immediately from the above relation, the fact that inequality (2.56) is satisfied with $k = 0$, and an induction argument. □

The following proposition bounds the primal feasibility of the iterates generated by APF-IAL.

Proposition 2.25. *The following relation holds for any iteration index $k \geq 1$ generated by APF-IAL:*

$$\|\mathcal{A}z_k - b\| \leq \max\left\{\frac{4D(M_n + \nabla_s + \tilde{\epsilon}_1)}{c_k\bar{d}\sigma_{\mathcal{A}}^+}, \frac{2\|p_0\|}{c_k}\right\}. \quad (2.58)$$

Proof. The update rule for the dual variable p_k in step 12, triangle inequality, and relation (2.56) imply that the following relation holds for any iteration index $k \geq 1$:

$$\|\mathcal{A}z_k - b\| = \frac{1}{c_k}\|p_k - p_{k-1}\| \leq \frac{1}{c_k}(\|p_k\| + \|p_{k-1}\|) \overset{(2.56)}{\leq} \max\left\{\frac{4D(M_n + \nabla_s + \tilde{\epsilon}_1)}{c_k\bar{d}\sigma_{\mathcal{A}}^+}, \frac{2\|p_0\|}{c_k}\right\},$$

from which the result of the proposition immediately follows. □

The following proposition provides an upper bound on the sequence of penalty parameters, c_k, generated by APF-IAL.

Proposition 2.26. *For any iteration index $k \geq 1$ generated by APF-IAL, we have*

$$c_k \leq \max\left\{\alpha c_1, \frac{4\alpha^2 D(M_n + \nabla_s + \tilde{\epsilon}_1)}{\hat{\epsilon}\bar{d}\sigma_{\mathcal{A}}^+}, \frac{2\alpha^2\|p_0\|}{\hat{\epsilon}}\right\} \quad (2.59)$$

where $\hat{\epsilon} > 0$ is the input tolerance to APF-IAL and c_1 is the initial penalty parameter.

Proof. First, observe that relation (2.59) holds for $k = 1$ and $k = 2$ since $\alpha > 1$ and $c_2 = \alpha c_1$. Suppose by contradiction that APF-IAL generates an iteration index $k \geq 3$ such that

$$c_k > \max\left\{\alpha c_1, \frac{4\alpha^2 D(M_n + \nabla_s + \tilde{\epsilon}_1)}{\hat{\epsilon}\bar{d}\sigma_{\mathcal{A}}^+}, \frac{2\alpha^2\|p_0\|}{\hat{\epsilon}}\right\}. \quad (2.60)$$

It then follows from this relation and the way that c_k is updated in step 13 of APF-IAL that

$$c_{k-2} = \frac{c_k}{\alpha^2} > \max\left\{\frac{4D(M_n + \nabla_s + \tilde{\epsilon}_1)}{\hat{\epsilon}\bar{d}\sigma_{\mathcal{A}}^+}, \frac{2\|p_0\|}{\hat{\epsilon}}\right\}. \quad (2.61)$$

This relation together with relation (2.58) then implies that

$$\|\mathcal{A}z_{k-2} - b\| \overset{(2.58)}{\leq} \frac{4D(M_n + \nabla_s + \tilde{\epsilon}_1)}{c_{k-2}\bar{d}\sigma_{\mathcal{A}}^+} < \hat{\epsilon}. \quad (2.62)$$

If $\|r_{k-2}\| \leq \hat{\rho}$, it follows from the above relation that APF-IAL must terminate in its $(k-2)$-th iteration, which contradicts the fact that APF-IAL generated the iteration index $k-1$. For the other situation, suppose now that $\|r_{k-2}\| > \hat{\rho}$. It then follows from relation (2.62) and the way ϵ_{k-1} is set during steps 5 to 9 of APF-IAL that $\epsilon_{k-1} = \hat{\rho}$. From (2.54), we get that $\|r_{k-1}\| \leq \hat{\rho}$. Moreover, it follows from relation (2.61), the update rule for c_k in step 13 of APF-IAL, and the fact that $\alpha > 1$ that $c_{k-1} = \alpha c_{k-2}$ that

$$c_{k-1} > \max\left\{\frac{4D(M_n + \nabla_s + \tilde{\epsilon}_1)}{\hat{\epsilon}\bar{d}\bar{\sigma}^+_{\mathcal{A}}}, \frac{2\|p_0\|}{\hat{\epsilon}}\right\}.$$

Hence, this relation together with relation (2.58) implies that

$$\|\mathcal{A}z_{k-1} - b\| \overset{(2.58)}{\leq} \max\left\{\frac{4D(M_n + \nabla_s + \tilde{\epsilon}_1)}{c_{k-1}\bar{d}\bar{\sigma}^+_{\mathcal{A}}}, \frac{2\|p_0\|}{c_{k-1}}\right\} < \hat{\epsilon}. \tag{2.63}$$

It follows from relation (2.63) and the fact that $\|r_{k-1}\| \leq \hat{\rho}$ that APF-IAL must terminate in its $(k-1)$-th iteration, which contradicts the fact that APF-IAL generated the iteration index k satisfying relation (2.60). Hence, relation (2.59) must hold for all iteration indices k generated by APF-IAL. □

Proposition 2.27. *The number of outer iterations performed by APF-IAL is at most*

$$K := 2 + \left\lceil \log^+_\alpha \left(\max\left\{\frac{4\alpha^2 D(M_n + \nabla_s + \tilde{\epsilon}_1)}{\hat{\epsilon}(c_1\bar{d}\bar{\sigma}^+_{\mathcal{A}})}, \frac{2\alpha^2\|p_0\|}{\hat{\epsilon}c_1}\right\}\right)\right\rceil \tag{2.64}$$

where $\hat{\epsilon} > 0$ is the input tolerance to APF-IAL and c_1 is the initial penalty parameter.

Proof. Suppose by contradiction that APF-IAL generates an iteration index $\hat{K} > K$. It follows from the definition of K above that

$$\begin{aligned}\hat{K} - 1 > K - 1 &= 1 + \left\lceil \log^+_\alpha \left(\max\left\{\frac{4\alpha^2 D(M_n + \nabla_s + \tilde{\epsilon}_1)}{\hat{\epsilon}(c_1\bar{d}\bar{\sigma}^+_{\mathcal{A}})}, \frac{2\alpha^2\|p_0\|}{\hat{\epsilon}c_1}\right\}\right)\right\rceil \\ &> \log^+_\alpha \left(\max\left\{\frac{4\alpha^2 D(M_n + \nabla_s + \tilde{\epsilon}_1)}{\hat{\epsilon}(c_1\bar{d}\bar{\sigma}^+_{\mathcal{A}})}, \frac{2\alpha^2\|p_0\|}{\hat{\epsilon}c_1}\right\}\right).\end{aligned}$$

It is then easy to see that the above relation implies that

$$\alpha^{\hat{K}-1} > \max\left\{\frac{4\alpha^2 D(M_n + \nabla_s + \tilde{\epsilon}_1)}{\hat{\epsilon}(c_1\bar{d}\bar{\sigma}^+_{\mathcal{A}})}, \frac{2\alpha^2\|p_0\|}{\hat{\epsilon}c_1}\right\}.$$

Now, it is easy to see from the way that c_k is updated in step 13 of APF-IAL that $c_{\hat{K}} = c_1\alpha^{\hat{K}-1}$. This fact together with the above bound then imply that

$$c_{\hat{K}} > \max\left\{\alpha c_1, \frac{4\alpha^2 D(M_n + \nabla_s + \tilde{\epsilon}_1)}{\hat{\epsilon}(\bar{d}\bar{\sigma}^+_{\mathcal{A}})}, \frac{2\alpha^2\|p_0\|}{\hat{\epsilon}}\right\}, \tag{2.65}$$

which contradicts the bound on c_k in (2.59). □

We are now ready to prove Theorem 2.20.

Proof of Theorem 2.20. Observe first that it follows from Proposition 2.22(b), Lemma 2.23, and the fact that $c_k = c_1\alpha^{k-1}$ that the PF-AR method performs at most

$$T_k := \sqrt{\frac{\bar{L}D + c_1\alpha^{k-1}D\|\mathcal{A}\|^2}{\hat{\rho}}} \tag{2.66}$$

gradient evaluations during the k-th outer iteration of APF-IAL. It then follows from this observation, the bound (2.64) on the number of outer iterations performed by APF-IAL, Proposition 2.22(a), the fact that $\sqrt{a+b} \leq \sqrt{a} + \sqrt{b}$, and the bound on c_k in Proposition 2.26 that APF-IAL performs at most the following number of gradient evaluations:

$$\sum_{k=1}^{K} T_k \overset{(2.66)}{=} \sum_{k=1}^{K} \sqrt{\frac{\bar{L}D + c_1\alpha^{k-1}D\|\mathcal{A}\|^2}{\hat{\rho}}} \leq K\sqrt{\frac{\bar{L}D}{\hat{\rho}}} + \frac{\sqrt{c_1 D}\|\mathcal{A}\|}{\sqrt{\hat{\rho}}} \sum_{k=1}^{K} \sqrt{\alpha^{k-1}}$$

$$= K\sqrt{\frac{\bar{L}D}{\hat{\rho}}} + \frac{\sqrt{c_1 D}\|\mathcal{A}\|}{\sqrt{\hat{\rho}}} \left[\frac{\sqrt{\alpha^K}-1}{\sqrt{\alpha}-1}\right] \leq K\sqrt{\frac{\bar{L}D}{\hat{\rho}}} + \frac{\|\mathcal{A}\|\sqrt{D}\sqrt{c_1\alpha^K}}{\sqrt{\hat{\rho}}(\sqrt{\alpha}-1)} = K\sqrt{\frac{\bar{L}D}{\hat{\rho}}} + \frac{\sqrt{\alpha}}{\sqrt{\alpha}-1}\frac{\|\mathcal{A}\|\sqrt{D}\sqrt{c_K}}{\sqrt{\hat{\rho}}}$$

$$\overset{(2.59)}{\leq} K\sqrt{\frac{\bar{L}D}{\hat{\rho}}} + \max\left\{\left[\frac{\alpha\|\mathcal{A}\|\sqrt{D}\sqrt{c_1}}{(\sqrt{\alpha}-1)}\right]\frac{1}{\sqrt{\hat{\rho}}}, \frac{2\alpha^{3/2}\|\mathcal{A}\|D\sqrt{M_n+\nabla_s+\tilde{\epsilon}_1}}{\sqrt{\bar{d}\sigma^+_{\mathcal{A}}}(\sqrt{\alpha}-1)}\left[\frac{1}{\sqrt{\hat{\epsilon}}\sqrt{\hat{\rho}}}\right]\right\}$$

$$\overset{(2.64)}{=} \left[2 + \left\lceil \log_\alpha\left(\max\left\{\frac{4\alpha^2 D(M_n+\nabla_s+\tilde{\epsilon}_1)}{\hat{\epsilon}(c_1\bar{d}\sigma^+_{\mathcal{A}})}, \frac{2\alpha^2\|p_0\|}{\hat{\epsilon}c_1}\right\}\right)\right\rceil\right]\sqrt{\frac{\bar{L}D}{\hat{\rho}}}$$

$$+ \max\left\{\left[\frac{\alpha\|\mathcal{A}\|\sqrt{D}\sqrt{c_1}}{(\sqrt{\alpha}-1)}\right]\frac{1}{\sqrt{\hat{\rho}}}, \frac{2\alpha^{3/2}\|\mathcal{A}\|D\sqrt{M_n+\nabla_s+\tilde{\epsilon}_1}}{(\sqrt{\alpha}-1)\sqrt{\bar{d}\sigma^+_{\mathcal{A}}\hat{\epsilon}\hat{\rho}}}\right\}$$

to find an $(\hat{\epsilon}, \hat{\rho})$-approximate solution of (1.1) according to (2.2). Hence, it follows from the above relation that APF-IAL performs at most

$$\mathcal{O}\left(\frac{\alpha^{3/2}\|\mathcal{A}\|D\sqrt{M_n+\nabla_s+\tilde{\epsilon}_1}}{\sqrt{\bar{d}\sigma^+_{\mathcal{A}}}(\sqrt{\alpha}-1)\sqrt{\hat{\epsilon}\hat{\rho}}}\right)$$

total gradient evaluations to find an $(\hat{\epsilon}, \hat{\rho})$-approximate solution, which implies the complexity bound in (2.50).

The last conclusion of the theorem immediately follows from (2.50) and taking $\hat{\epsilon} = \hat{\rho}$. □

3 Linearly-Constrained Strongly Convex Optimization

In this section, we discuss the complexities of the inexact AL methods, O-IAL (Algorithm 2.4), OPF-IAL (Algorithm 2.5) and APF-IAL (Algorithm 2.6), for solving (1.1), where ψ is now assumed to be $\bar{\mu}$-strongly convex with $\bar{\mu} > 0$.

The rest of this section is organized as follows. Subsection 3.1 presents the parameter-free R-FISTA method and its complexity guarantees for unconstrained strongly convex composite minimization. The R-FISTA method will be used as a subroutine inside O-IAL, OPF-IAL, and APF-IAL to approximately minimize their strongly convex AL subproblems. Subsection 3.2 presents the near-optimal complexity bounds of O-IAL, OPF-IAL, and APF-IAL for the linearly-constrained strongly convex problem (1.1)

3.1 R-FISTA: Parameter-Free Strongly Convex Optimization Subproblem Solver

This section describes a restarted FISTA method, namely R-FISTA, which will be used by OPF-IAL and APF-IAL to solve their strongly convex composite AL subproblems. More generally, R-FISTA solves the unconstrained composite optimization problem version of (1.1), i.e.,

$$\min\{\psi(z) := \psi_s(z) + \psi_n(z)\}, \tag{3.1}$$

where ψ is assumed to be $\bar{\mu}$-strongly convex with $\bar{\mu} > 0$, ψ_s is convex and $\bar{L}$-smooth, and ψ_n is a closed proper convex function on its compact domain $\operatorname{dom} \psi_n$, which has diameter D.

Given $\epsilon > 0$, R-FISTA aims to find $(y, v) \in \operatorname{dom} \psi_n \times \mathbb{E}$ such that

$$\|v\| \leq \epsilon, \quad v \in \nabla \psi_s(y) + \partial \psi_n(y). \tag{3.2}$$

This is the unconstrained form of the approximate optimality condition in Definition 2.2, obtained from (2.2) by setting $\hat{\epsilon} = 0$, $\hat{\rho} = \epsilon$, and $\mathcal{A} = 0$. Any pair satisfying (3.2) is therefore called an **ϵ-optimal solution** of (3.1).

The residual criterion (3.2) also yields a function-gap bound. Recall that a proper convex function F satisfies the quadratic growth condition $\mathcal{G}^2_\mu$ if $\frac{\mu}{2} d(x, X^*)^2 \leq F(x) - F^*$ holds for all $x \in \mathbb{E}$. Every μ-strongly convex function satisfies this condition. The following result from [2] relates function gap criterion to the distance from stationarity.

Lemma 3.1. *Let $F : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ be a proper lower semicontinuous convex function with nonempty minimizer set X^*, and let $F^* = \inf F$. If F satisfies $\mathcal{G}^2_\mu$ for some $\mu > 0$, then $2\mu(F(x) - F^*) \leq d\left(0, \partial F(x)\right)^2$ holds for all $x \in \mathbb{R}^n$.*

Applying Lemma 3.1 to $F = \psi$ with $\mu = \bar{\mu}$, and using $v \in \partial \psi(y)$, gives $\psi(y) - \psi(x^*) \leq \frac{\|v\|^2}{2\bar{\mu}}$. Hence, any pair satisfying $\|v\|^2 \leq 2\bar{\mu}\epsilon$ also satisfies $\psi(y) - \psi(x^*) \leq \epsilon$.

R-FISTA is a restarted, parameter-free accelerated gradient method. It begins with an aggressive estimate of the strong convexity parameter and checks a restart condition at each iteration. Whenever this condition fails, the method starts a new cycle with a smaller strong convexity estimate.

R-FISTA can be viewed as a compact-domain specialization of RPF-SFISTA [44]. Compactness simplifies both the analysis and the restart mechanism: the restart condition depends only on the final iterate of the current cycle, so there is no need to track the best function-value iterate or restart from it. Since each cycle's complexity is independent of its initial function value, any point in $\operatorname{dom} \psi_n$ may be used to start the next cycle. This yields a simpler and more flexible method, which we now present.

Algorithm 3.1 R-FISTA Method

Input: scalars $\chi \in (0,1)$, $(\mu_0, \bar{M}_0) \in \mathbb{R}^2_{++}$, tolerance $\epsilon > 0$, function pair (ψ_s, ψ_n), and initial point $\bar{x}_0 \in \operatorname{dom} \psi_n$.

1: set $\ell \leftarrow 1$

2: set stopcrit $\leftarrow \infty$

Initialization: initialize $j = 0$, $\underline{M}_\ell \in [\max\{0.25\bar{M}_{\ell-1}, \bar{M}_0\}, \bar{M}_{\ell-1}]$, $\tilde{\mu} = \mu_{\ell-1}$, $(A_0, \tau_0, L_0) = (0, 1, \underline{M}_\ell)$, and $x_0 \in \operatorname{dom} \psi_n$ with $x_0 = \bar{x}_0$ if $\ell = 1$; set $y_0 = x_0$.

3: **while** stopcrit $> \epsilon$ (main outer loop) **do**

4: set $j \leftarrow j + 1$;

5: set $L_j = L_{j-1}$ and evaluate:

$$a_{j-1} = \frac{\tau_{j-1} + \sqrt{\tau_{j-1}^2 + 4\tau_{j-1}A_{j-1}L_j}}{2L_j}, \quad \tilde{x}_{j-1} = \frac{A_{j-1}y_{j-1} + a_{j-1}x_{j-1}}{A_{j-1} + a_{j-1}}, \tag{3.3}$$

$$y_j := prox_{(1/L_j)\psi_n}\left(\tilde{x}_{j-1} - \frac{1}{L_j}\nabla\psi_s(\tilde{x}_{j-1})\right). \tag{3.4}$$

i. **While**: $\ell_{\psi_s}(y_j; \tilde{x}_{j-1}) + \frac{(1-\chi)L_j}{4}\|y_j - \tilde{x}_{j-1}\|^2 < \psi_s(y_j)$

set $L_j \leftarrow 2L_j$ and update the equations (3.3), (3.4);

EndWhile:

6: evaluate

$$A_j = A_{j-1} + a_{j-1}, \quad \tau_j = \tau_{j-1} + \frac{a_{j-1}\tilde{\mu}}{2}, \tag{3.5}$$

$$s_j = L_j(\tilde{x}_{j-1} - y_j), \tag{3.6}$$

$$x_j = \frac{1}{\tau_j}\left[\frac{\tilde{\mu}a_{j-1}y_j}{2} + \tau_{j-1}x_{j-1} - a_{j-1}s_j\right], \tag{3.7}$$

$$v_j = \nabla\psi_s(y_j) - \nabla\psi_s(\tilde{x}_{j-1}) + s_j; \tag{3.8}$$

7: **if**

$$\|y_j - x_0\|^2 \geq \chi A_j L_j \|y_j - \tilde{x}_{j-1}\|^2 \tag{3.9}$$

does not hold then

8: **restart**, i.e., set $\bar{M}_\ell = L_j$, $\mu_\ell = 0.5\tilde{\mu}$, $\ell \leftarrow \ell + 1$, stopcrit $\leftarrow \infty$,

9: and execute **Initialization** above;

10: **else**

11: set stopcrit $= \|v_j\|$.

12: **end if**

13: **end while**

Output: pair $(y, v) := (y_j, v_j)$ that satisfies (3.2).

Several remarks about R-FISTA are now given. R-FISTA performs two types of iterations, inner ACG iterations indexed by j and cycles indexed by ℓ. At each of its inner ACG iterations, R-FISTA checks in its step 7 a key inequality (3.9) to determine when to restart and reduce its current estimate μ_ℓ of the strong convexity parameter. If the condition fails to hold during R-FISTA's ℓ-th cycle, then R-FISTA sets $\mu_\ell = 0.5\mu_{\ell-1}$ and starts its $(\ell+1)$-st cycle with this strong convexity estimate. Finally, note that each of R-FISTA's inner iterations may perform several gradient/proximal operator evaluations due to its inner while loop in step 5(i). We express our total complexity of R-FISTA in terms of the number of inner ACG iterations that it performs and the total number of gradient evaluations that it performs. These two quantities are on the same order.

Before stating the main results of the R-FISTA method, the following quantities are introduced

$$\theta = 4/(1-\chi), \quad \zeta_\ell := \bar{L} + \max\{\underline{M}_\ell, \theta\bar{L}\}, \quad Q_\ell := 2\sqrt{2}\sqrt{\frac{\max\{\underline{M}_\ell, \theta\bar{L}\}}{\mu_{\ell-1}}}. \tag{3.10}$$

The following proposition and theorem state the main complexity results of the R-FISTA method and key properties of its output. The proofs of Proposition 3.2 and Theorem 3.3 are given in Appendix B.

Proposition 3.2. *The following statements about the ℓ-th cycle of R-FISTA hold:*

(a) its main while loop either stops successfully with stopcrit $\leq \epsilon$ or the cycle stops because the inequality (3.9) in step 7 does not hold (in one of its inner iterations) in at most

$$\left\lceil (1+Q_\ell)\log^{++}\left(\frac{D^2\zeta_\ell^2}{\chi\epsilon^2}\right)+1\right\rceil + \left\lceil \log_2^+\left(\frac{2\bar{L}}{(1-\chi)\underline{M}_\ell}\right)\right\rceil \tag{3.11}$$

ACG iterations/gradient evaluations;

(b) if the main while loop of the cycle terminates successfully with stopcrit $\leq \epsilon$, then the pair (y,v) that R-FISTA outputs satisfies

$$\|v\| \leq \epsilon, \quad v \in \nabla\psi_s(y) + \partial\psi_n(y), \tag{3.12}$$

i.e., (y,v) is an ϵ-optimal solution of (3.1);

(c) if $\mu_{\ell-1} \in (0, \bar{\mu}]$, then a restart is never executed during the cycle and the main while loop of the cycle always stops successfully with stopcrit $\leq \epsilon$ and a pair (y,v) satisfying (3.12) in at most (3.11) ACG iterations/gradient evaluations.

Theorem 3.3. *Suppose that inputs μ_0 and $\bar{M}_0$ satisfy $\mu_0 \geq \bar{\mu}$ and $\bar{M}_0 \leq 2\bar{L}$. Then, R-FISTA terminates with a pair (y,v), that is an ϵ-optimal solution of (3.1), in at most*

$$\mathcal{O}_1\left(\left\lceil \log_2^{++}(\mu_0/\bar{\mu})\right\rceil \left[\sqrt{\frac{\bar{L}}{\bar{\mu}}}\log^{++}\left(\frac{D\bar{L}}{\epsilon}\right) + \log_2^+(\bar{L}/\bar{M}_0)\right]\right) \tag{3.13}$$

ACG iterations/gradient evaluations. Hence R-FISTA performs $\mathcal{O}\left(\sqrt{\bar{L}/\bar{\mu}}\log\left(D\bar{L}/\epsilon\right)\right)$ ACG iterations/gradient evaluations to find an ϵ-optimal solution of (3.1).

Remark 3.4. *The assumptions in Theorem 3.3 that the inputs $(\mu_0, \bar{M}_0)$ satisfy $\mu_0 \geq \bar{\mu}$ and $\bar{M}_0 \leq 2\bar{L}$ are not necessary to establish convergence results, but simplify our analysis and R-FISTA's complexity bound.*

For more details on how to construct such a $\bar{M}_0$, see the discussion after the AR-L method. It is also easy to construct $\mu_0 \geq \bar{\mu}$. For example, select $w_0 \neq x_0$ and compute $w_0^+ = prox_{(1/\bar{M}_0)\psi_n}(w_0 - \frac{1}{\bar{M}_0}\nabla\psi_s(w_0))$ and $x_0^+ = prox_{(1/\bar{M}_0)\psi_n}(x_0 - \frac{1}{\bar{M}_0}\nabla\psi_s(x_0))$. If $x_0^+ \neq w_0^+$, then set $g_{x_0} = \bar{M}_0(x_0 - x_0^+) + \nabla\psi_s(x_0^+) - \nabla\psi_s(x_0)$ and $g_{w_0} = \bar{M}_0(w_0 - w_0^+) + \nabla\psi_s(w_0^+) - \nabla\psi_s(w_0)$. By the optimality conditions of the proximal-gradient steps, $g_{x_0} \in \partial\psi(x_0^+)$ and $g_{w_0} \in \partial\psi(w_0^+)$. Hence, by the $\bar{\mu}$-strong convexity of ψ, we have

$$\frac{\langle g_{x_0} - g_{w_0}, x_0^+ - w_0^+\rangle}{\|x_0^+ - w_0^+\|^2} \geq \bar{\mu}.$$

Thus, if $x_0^+ \neq w_0^+$, then set

$$\mu_0 = \frac{\langle g_{x_0} - g_{w_0}, x_0^+ - w_0^+\rangle}{\|x_0^+ - w_0^+\|^2}.$$

If $x_0^+ = w_0^+$, then recompute both points (possibly with different stepsizes) until the resulting proximal-gradient points are distinct.

3.2 Complexity of O-IAL, OPF-IAL, and APF-IAL for Linearly-Constrained Strongly Convex Optimization

This subsection considers the complexity of the O-IAL, OPF-IAL, and APF-IAL methods presented in Subsection 2.2, Subsection 2.3, and Subsection 2.4, respectively, for solving (1.1) under the assumption that ψ is a $\bar{\mu}$-strongly convex function, and the assumption that R-FISTA is used (instead of the PF-AR method) to approximately minimize the augmented Lagrangian function. All three achieve a near-optimal complexity of $\mathcal{O}(\hat{\epsilon}^{-1/2}\log(1/\hat{\epsilon}))$ for finding an $(\hat{\epsilon},\hat{\epsilon})$ approximate solution of (1.1).

The following theorem states the main complexity result of O-IAL under the strong convexity assumption.

Theorem 3.5. *Suppose that Assumption 2.1 holds and that ψ is $\bar{\mu}$-strongly convex for some $\bar{\mu} > 0$. Assume that O-IAL is initialized with the function pair (ψ_s, ψ_n), a point $z_0 \in \operatorname{dom}\psi_n$, a multiplier $p_0 \in \mathbb{R}^m$, a primal-dual tolerance pair $(\hat{\epsilon},\hat{\rho}) \in \mathbb{R}^2_{++}$, the penalty parameter $c = 1/\hat{\epsilon}$, and the constants M_0 and D_0 defined in Remark 2.7. In addition, let $\chi \in (0,1)$ and let the initial strong convexity estimate $\mu_0 > 0$ satisfy the conditions stated in Remark 3.4.*

Suppose further that, during its k-th iteration, O-IAL calls R-FISTA, rather than PF-AR, in step 7 with inputs $\chi \in (0,1)$, $\mu_0 > 0$, initial Lipschitz estimate $\bar{M}_0 = M_0$, tolerance $\epsilon = \min\left\{\frac{\hat{\epsilon}}{4D}, \hat{\rho}\right\}$, function pair $(\mathcal{L}^s_c(\cdot, p_{k-1}), \psi_n(\cdot))$, and initial point $z_{k-1} \in \operatorname{dom}\psi_n$. Then O-IAL performs at most

$$\mathcal{O}_1\left(\Pi(p_0)\left\lceil \log_2^{++}(\mu_0/\bar{\mu})\right\rceil\left[\sqrt{\frac{\bar{L}+\|\mathcal{A}\|^2/\hat{\epsilon}}{\bar{\mu}}}\log^{++}\left(\frac{D(\bar{L}+\|\mathcal{A}\|^2/\hat{\epsilon})}{\min\left\{\frac{\hat{\epsilon}}{4D},\hat{\rho}\right\}}\right)+\log_2^+\left(\frac{\bar{L}+\|\mathcal{A}\|^2/\hat{\epsilon}}{\bar{M}_0}\right)\right]\right) \tag{3.14}$$

gradient evaluations to obtain an $(\hat{\epsilon},\hat{\rho})$-approximate solution of (1.1). Consequently, O-IAL performs at most $\mathcal{O}(\hat{\epsilon}^{-1/2}\log(1/\hat{\epsilon}))$ total gradient evaluations to obtain an $(\hat{\epsilon},\hat{\epsilon})$-approximate solution of (1.1).

Proof. Suppose that the assumptions of the theorem hold, i.e., O-IAL uses penalty $c = 1/\hat{\epsilon}$ and calls R-FISTA in step 7 of its k-th iteration with inputs $\chi \in (0,1)$, $\mu_0 > 0$ as in Remark 3.4, $\bar{M}_0 = M_0$, tolerance $\epsilon = \min\left\{\frac{\hat{\epsilon}}{4D},\hat{\rho}\right\}$, function pair $(\mathcal{L}^s_c(\cdot,p_{k-1}),\psi_n(\cdot))$, and initial point $z_{k-1} \in \operatorname{dom}\psi_n$. It then follows from these assumptions, the fact that $\mathcal{L}^s_c(\cdot,p_{k-1})$ is $\bar{L}+\|\mathcal{A}\|^2/\hat{\epsilon}$-smooth, and Theorem 3.3 that the call to R-FISTA outputs a pair (z_k,r_k) that satisfies

$$r_k \in \nabla\psi_s(z_k) + \partial\psi_n(z_k) + \mathcal{A}^*p_k, \quad \|r_k\| \le \min\left\{\frac{\hat{\epsilon}}{4D},\hat{\rho}\right\}, \tag{3.15}$$

within at most

$$\mathcal{O}_1\left(\left\lceil \log_2^{++}(\mu_0/\bar{\mu})\right\rceil\left[\sqrt{\frac{\bar{L}+\|\mathcal{A}\|^2/\hat{\epsilon}}{\bar{\mu}}}\log^{++}\left(\frac{D(\bar{L}+\|\mathcal{A}\|^2/\hat{\epsilon})}{\min\left\{\frac{\hat{\epsilon}}{4D},\hat{\rho}\right\}}\right)+\log_2^+\left(\frac{\bar{L}+\|\mathcal{A}\|^2/\hat{\epsilon}}{\bar{M}_0}\right)\right]\right) \tag{3.16}$$

gradient evaluations. It then follows from the fact that O-IAL uses penalty parameter $c = 1/\hat{\epsilon}$, and similar arguments as in Subsection 2.2.1 and Proposition 2.12 that O-IAL performs at most $\lceil 2\Pi(p_0)\rceil$ outer iterations, i.e., those indexed by k. It then follows from this bound, the bound in (3.16), and both relations in (3.15) that the number of gradient evaluations O-IAL performs to find an $(\hat{\epsilon},\hat{\rho})$-approximate solution of (1.1) is on the order of the quantity in (3.14). The final conclusion of Theorem 3.5 is then immediate from the complexity bound in (3.14) and setting $\hat{\rho} = \hat{\epsilon}$. □

Remark 3.6. *It can also be shown under similar assumptions as in Theorem 3.5 that if the parameter-free AL method, OPF-IAL (Algorithm 2.5), calls R-FISTA (instead of PF-AR) during its step 5, it also achieves a near-optimal primal-dual complexity bound of $\mathcal{O}(\hat{\epsilon}^{-1/2}\log(\hat{\epsilon}^{-1}))$ to find an $(\hat{\epsilon},\hat{\epsilon})$-approximate solution of (1.1). We omit the proof since it follows from nearly identical arguments as the proof of Theorem 3.5.*

The theorem below now states the last-iterate complexity of the adaptive APF-IAL method (Algorithm 2.6) under the strong convexity assumption of ψ.

Theorem 3.7. *Suppose that Assumptions 2.18 and 2.19 hold, ψ is $\bar{\mu}$-strongly convex for some $\bar{\mu} > 0$, and K is as in (2.64). Assume that APF-IAL is initialized with the function pair (ψ_s, ψ_n), a point $z_0 \in \operatorname{dom} \psi_n$, a multiplier $p_0 \in \mathbb{R}^m$, a primal-dual tolerance pair $(\hat{\epsilon}, \hat{\rho}) \in \mathbb{R}^2_{++}$, parameters $c_1 > 0$, $\alpha > 1$, $\omega > 1$, and $\tilde{\epsilon}_1 \geq \hat{\rho}$, and constants M_0 and D_0 defined in Remark 2.7. In addition, let $\chi \in (0,1)$ and let the initial strong convexity estimate $\mu_0 > 0$ satisfy the conditions stated in Remark 3.4.*

Suppose further that, during its k-th iteration, APF-IAL calls R-FISTA, rather than PF-AR, in step 10 with inputs $\chi \in (0,1)$, $\mu_0 > 0$, initial Lipschitz estimate $\bar{M}_0 = M_0$, tolerance $\epsilon = \epsilon_k$, function pair $(\mathcal{L}_c^s(\cdot, p_{k-1}), \psi_n(\cdot))$, and initial point $z_{k-1} \in \operatorname{dom} \psi_n$. Then APF-IAL performs at most

$$\mathcal{O}_1 \left(\left\lceil \log_2^{++} (\mu_0/\bar{\mu}) \right\rceil \left[K \sqrt{\frac{\bar{L}}{\bar{\mu}}} + \sqrt{\frac{\|\mathcal{A}\|^2}{\hat{\epsilon}\bar{\mu}}} \right] \log^{++} \left(\frac{D(\bar{L} + \|\mathcal{A}\|^2/\hat{\epsilon})}{\hat{\rho}} \right) \right) \tag{3.17}$$

gradient evaluations to obtain an $(\hat{\epsilon}, \hat{\rho})$-approximate solution of (1.1). Consequently, APF-IAL performs at most $\mathcal{O}(\hat{\epsilon}^{-1/2} \log(1/\hat{\epsilon}))$ total gradient evaluations to obtain an $(\hat{\epsilon}, \hat{\epsilon})$-approximate solution of (1.1).

Proof. It follows from the assumptions of the theorem, Theorem 3.3, and the fact that Lemma 2.23 implies that $\epsilon_k \geq \hat{\rho}$, that the call to R-FISTA during the k-th iteration of APF-IAL outputs a pair (z_k, r_k) that satisfies

$$r_k \in \nabla \psi_s(z_k) + \partial \psi_n(z_k) + \mathcal{A}^* p_k, \quad \|r_k\| \leq \epsilon_k \tag{3.18}$$

within at most

$$\mathcal{O}_1 \left(\left\lceil \log_2^{++} (\mu_0/\bar{\mu}) \right\rceil \left[\sqrt{\frac{\bar{L} + c_k\|\mathcal{A}\|^2}{\bar{\mu}}} \log^{++} \left(\frac{D(\bar{L} + c_k\|\mathcal{A}\|^2)}{\hat{\rho}} \right) \right] \right) \tag{3.19}$$

gradient evaluations. It then follows from the bound in (2.64) that APF-IAL performs at most

$$K := 2 + \left\lceil \log_\alpha^+ \left(\max \left\{ \frac{4\alpha^2 D(M_n + \nabla_s + \tilde{\epsilon}_1)}{\hat{\epsilon}(c_1 \bar{d} \sigma_{\mathcal{A}}^+)}, \frac{2\alpha^2 \|p_0\|}{\hat{\epsilon} c_1} \right\} \right) \right\rceil$$

outer iterations to find an $(\hat{\epsilon}, \hat{\rho})$-approximate solution of (1.1). Moreover, the bound on c_k in (2.59) implies that $c_k = \mathcal{O}(1/\hat{\epsilon})$. It then follows from summing the complexity bound in (3.19) from $k = 1$ to K (using a similar argument as in the proof of Theorem 2.20) that the number of gradient evaluations that APF-IAL performs to find a $(\hat{\epsilon}, \hat{\rho})$ solution is on the order of the quantity in (3.17). The last conclusion of Theorem 3.7 then follows from this bound and setting $\hat{\rho} = \hat{\epsilon}$. □

4 Numerical Experiments

This section compares the numerical performance of our inexact methods O-IAL (Algorithm 2.4) and APF-IAL (Algorithm 2.6) with that of a representative PAL method, which we refer to as ProxALM (Algorithm 5 in [26]), on six classes of linearly constrained convex optimization problems, including both convex and strongly convex instances. To make the comparison as fair as possible, we use exactly the same termination criterion for all three methods. Specifically, each method terminates when it produces a triple (z, p, r) satisfying

$$r \in \nabla \psi_s(z) + \partial \psi_n(z) + \mathcal{A}^* p, \quad \|r\| \leq \hat{\rho}, \quad \|\mathcal{A}z - b\| \leq \hat{\epsilon}. \tag{4.1}$$

In all our experiments, we set $\hat{\rho} = \hat{\epsilon} = 10^{-5}$. We vary the strong convexity modulus $\bar{\mu}$, where $\bar{\mu} = 0$ corresponds to the convex setting.

We next describe the implementation details of the three methods considered in our numerical experiments. ProxALM takes as input $\epsilon = 10^{-5}$, a randomly generated initial primal-dual point $(x_0, \lambda_0) \in \operatorname{dom} \psi_n \times \mathbb{R}^m$, $M = 10$, $\delta = 0.8$, the initial penalty parameter $\rho_0 = \max\left\{10, \bar{\mu} + \sqrt{\bar{\mu}^2 + 4}\right\}$, $\alpha_0 = 1$, the initial tolerance $\eta_0 = 10$ for the first proximal AL subproblem, the penalty-parameter increase factor $\zeta = 1.1$, and the proximal AL subproblem tolerance decrease factor $\sigma = 1/(\zeta + 0.4) = 2/3$. At every iteration $k \geq 0$, the penalty parameter and proximal AL subproblem tolerance are updated according to $\rho_k = \rho_0 \zeta^k$ and $\eta_k = \eta_0 \sigma^k$, respectively.

For many of our test problems $\bar{\mu} < 5$, and hence $\rho_0 = 10$. Some problems have $\bar{\mu} \geq 5$, in which case $\rho_0 = \bar{\mu} + \sqrt{\bar{\mu}^2 + 4}$. Thus, in all cases, the required condition $\rho_0 > \left(\bar{\mu} + \sqrt{\bar{\mu}^2 + 4}\right)/2$ is satisfied. As prescribed in step 2 of ProxALM, we use the strongly convex version of FISTA presented in Algorithm 2 of [26] to approximately minimize each proximal augmented Lagrangian subproblem. Algorithm 2 is implemented exactly as specified in [26]. Finally, we modify the termination criterion in step 4 of ProxALM so that it uses the common criterion in (4.1).

Since ProxALM uses a strongly convex APG/FISTA method to minimize its PAL subproblems, O-IAL and APF-IAL use R-FISTA (Algorithm 3.1) to minimize their AL subproblems. Both methods choose the R-FISTA-specific input parameter $\chi = 0.001$ and use $L_j = \beta L_j$, where $\beta = 1.25$ instead of 2. In particular, $1/\beta = 0.8$, which matches the value $\delta = 0.8$ used by ProxALM. They also use the initial strong convexity estimate $\mu_0 = 2(\psi_s(w_0) - \psi_s(z_0) - \langle \nabla \psi_s(z_0), w_0 - z_0 \rangle)/(\|w_0 - z_0\|^2)$, where $w_0, z_0 \in \operatorname{dom} \psi_n$ and z_0 is the initial point. When ψ_n is the indicator function of a closed convex set, this choice satisfies $\mu_0 \geq \bar{\mu}$.

O-IAL uses the same randomly generated initial primal and multiplier points as ProxALM, together with the fixed penalty parameter $c = 10$, the tolerances $\hat{\epsilon} = 10^{-5}$ and $\hat{\rho} = 10^{-5}$, and a domain-diameter parameter $D > 0$. When the diameter of $\operatorname{dom} \psi_n$ can be estimated from the problem data, we use this estimate for D. Otherwise, if the diameter cannot be readily estimated or $\operatorname{dom} \psi_n$ is not necessarily compact, we set $D = 1$. Hence, some experiments intentionally test the algorithms beyond the theoretical setting considered in this paper, which assumes that $\operatorname{dom} \psi_n$ is compact.

APF-IAL uses the same initial primal-dual point as O-IAL and ProxALM, the initial penalty parameter $c_1 = 10$, the penalty-parameter increase factor $\alpha = 1.1$, $\omega = 1.5$, and the initial tolerance $\tilde{\epsilon}_1 = 10$. Here, $1/\omega = 2/3$ matches the value $\sigma = 2/3$ used by ProxALM, while $\tilde{\epsilon}_1 = 10$ matches its initial tolerance $\eta_0 = 10$.

All experiments are performed in MATLAB R2025b on a 2023 MacBook Pro equipped with an 8-core CPU. The latest codes to reproduce the experiments are available at this clickable-link or with URL `https://github.com/asujanani6/Parameter_Free_ALProj_Codes`. Each of the following six subsections reports numerical results for a different class of linearly constrained problems.

4.1 Quadratic Objective over the Simplex

Given dimensions $(m, n) \in \mathbb{N}^2$, scalars $(\xi, \tau) \in \mathbb{R}^2_{++}$, matrices $A, C \in \mathbb{R}^{m \times n}$ and $B \in \mathbb{R}^{n \times n}$, a positive diagonal matrix $D \in \mathbb{R}^{n \times n}$, and vectors $d, b \in \mathbb{R}^m$, we consider

$$\begin{aligned} \min_x \quad & \psi_s(x) := \frac{\xi}{2}\|DBx\|^2 + \frac{\tau}{2}\|Cx - d\|^2 \\ \text{s.t.} \quad & x \in \Delta^n, \qquad Ax = b, \end{aligned}$$

where

$$\Delta^n := \left\{ x \in \mathbb{R}^n_+ : \mathbf{1}_n^\top x = 1 \right\}$$

is the probability simplex (and hence where $\mathbf{1}_n$ is the vector of all ones). We vary the dimensions over $10 \leq m \leq 700$ and $50 \leq n \leq 1100$, and use target curvatures $\bar{L} \in \{200, 500, 1000\}$ and $\bar{\mu} \in \{5, 10, 100\}$. The entries of A, C, d, and a seed matrix $\widetilde{B}$ are independently sampled from $\mathcal{U}[0, 1]$, while the diagonal entries of D are sampled uniformly from $\{1, \ldots, \alpha\}$. The matrix B and the coefficients (ξ, τ) are then spectrally calibrated so that $\lambda_{\max}(\nabla^2 \psi_s) = \bar{L}$ and $\lambda_{\min}(\nabla^2 \psi_s) = \bar{\mu}$. All tested instances in this subsection are strongly convex. We set $b = A(\mathbf{1}_n/n)$, and generate $x_0 = \widehat{x}/(\mathbf{1}_n^\top \widehat{x})$, where the entries of $\widehat{x}$ are independently sampled from $\mathcal{U}[0, 1]$. The numerical comparisons are reported in Table 4.1.

As seen from Table 4.1, O-IAL and APF-IAL were, on average, 7.24 and 52.37 times faster than ProxALM, respectively. O-IAL was faster than ProxALM on 13 instances, slower on one, and tied on one. It also consistently achieved dual-stationarity residuals on the order of 10^{-10} as it maintains high dual stationarity at each of its iterations. APF-IAL was faster than ProxALM on all 15 instances.

Table 4.1 Quadratic Objective Over Simplex Comparison

Dimensions		Time (s)			Primal Feasibility Residual			Dual Stationarity Residual		
m	n	O-IAL	APF-IAL	ProxALM [26]	O-IAL	APF-IAL	ProxALM [26]	O-IAL	APF-IAL	ProxALM [26]
10	50	0.09	0.07	0.09	9.1959e-06	7.0414e-06	1.1720e-06	1.3176e-10	7.6083e-06	6.9513e-06
50	125	0.18	0.03	0.14	9.9027e-06	5.0281e-06	6.1081e-09	1.7364e-10	9.1839e-06	7.8900e-06
100	200	0.37	0.03	0.52	9.0125e-06	7.6086e-07	2.5729e-09	1.5800e-10	8.5710e-06	5.4161e-06
150	275	0.33	0.04	0.85	8.2824e-06	6.1373e-06	6.0852e-10	1.6413e-10	9.9256e-06	9.7421e-06
200	350	0.39	0.03	1.33	7.6745e-06	2.2687e-06	4.2170e-10	1.6393e-10	9.8571e-06	6.9411e-06
250	425	0.23	0.03	1.99	9.1271e-06	4.6282e-06	1.4520e-10	1.4806e-10	3.3230e-06	9.4211e-06
300	500	0.29	0.02	2.79	8.7465e-06	7.7660e-06	1.4695e-10	1.5001e-10	9.4454e-06	9.3530e-06
350	575	0.38	0.07	4.13	9.0311e-06	1.0688e-06	3.6827e-11	1.3399e-10	6.7502e-06	4.6030e-06
400	650	0.35	0.05	4.66	8.4926e-06	5.5555e-06	1.6717e-10	1.3553e-10	8.8359e-06	9.8516e-06
450	725	0.47	0.09	6.35	9.3200e-06	4.0478e-06	1.0157e-10	1.1782e-10	6.0721e-06	8.5942e-06
500	800	0.96	0.19	7.31	7.4388e-06	3.8783e-06	7.9476e-10	1.7589e-10	9.9275e-06	9.5685e-06
550	875	1.43	0.21	11.42	3.5687e-06	1.3724e-06	4.4192e-09	1.5287e-10	9.6858e-06	9.8851e-06
600	950	1.55	0.24	14.07	1.2666e-06	2.6860e-06	4.4561e-09	1.2568e-10	9.9306e-06	9.8263e-06
650	1025	1.93	0.31	19.59	1.5630e-06	6.8438e-06	2.6687e-09	1.3531e-10	9.7562e-06	9.8604e-06
700	1100	3.08	0.50	26.72	2.3606e-06	6.0140e-06	2.7868e-09	1.6383e-10	9.9026e-06	9.2066e-06

4.2 Logistic Regression over an ℓ_1 Ball

Given a number of observations $N \in \mathbb{N}$, dimensions $(m, n) \in \mathbb{N}^2$, a data matrix $X \in \mathbb{R}^{N \times n}$, labels $y \in \{-1, 1\}^N$, a constraint matrix $A \in \mathbb{R}^{m \times n}$, and $b \in \mathbb{R}^m$, we consider

$$\begin{aligned} \min_x \quad & \psi_s(x) := \frac{1}{N} \sum_{i=1}^{N} \log\left(1 + \exp(-y_i X_i^\top x)\right) + \frac{\bar{\mu}}{2}\|x\|^2 \\ \text{s.t.} \quad & x \in \mathbb{B}_1^n, \qquad Ax = b, \end{aligned}$$

where $X_i^\top$ denotes the i-th row of X and

$$\mathbb{B}_1^n := \{x \in \mathbb{R}^n : \|x\|_1 \leq 1\}$$

is the ℓ_1 ball. In our experiments, we vary parameters $50 \leq N \leq 650$, $50 \leq m \leq 1000$, and $100 \leq n \leq 2500$, and we vary the target curvature pair from $30 \leq \bar{L} \leq 1000$ and $0 \leq \bar{\mu} \leq 5$. Hence, the instances with $\bar{\mu} = 0$ are convex, while those with $\bar{\mu} > 0$ are strongly convex. We generate the matrix X according to

$$X = \frac{\sqrt{4N(\bar{L} - \bar{\mu})}}{\|\widetilde{X}\|_2} \widetilde{X}$$

where $\widetilde{X}$ is a fully dense Gaussian matrix. The entries of y are independently and uniformly sampled from $\{-1, 1\}$.

We set our initial point $x_0 = 0$. The entries of the second to m-th row of A are independently sampled from the standard normal distribution, while its first row is generated as $A_{1,:} = 2\mathbf{1}_n^\top$. Finally, we set $b = A\,(\mathbf{1}_n/2n)$. The numerical comparisons are reported in Table 4.2.

Table 4.2 Logistic Regression Comparison

Dimensions		Time (s)			Primal Feasibility Residual			Dual Stationarity Residual		
m	n	O-IAL	APF-IAL	ProxALM [26]	O-IAL	APF-IAL	ProxALM [26]	O-IAL	APF-IAL	ProxALM [26]
50	100	0.65	0.22	4.71	7.0415e-06	6.8516e-06	3.6004e-09	1.2061e-10	9.9789e-06	9.4590e-06
100	150	1.19	0.19	4.83	6.9268e-06	5.0017e-06	1.2150e-08	1.2494e-10	9.8587e-06	8.9484e-06
125	200	1.51	0.24	7.19	9.9941e-06	6.4578e-06	1.6544e-09	1.2386e-10	9.9646e-06	9.5325e-06
150	300	1.37	0.34	8.96	4.7838e-06	4.7497e-06	1.2518e-09	1.2468e-10	9.9642e-06	9.1886e-06
175	400	2.34	0.42	12.62	5.9537e-06	4.4589e-06	3.4247e-09	1.2468e-10	9.9959e-06	9.9159e-06
200	500	1.25	0.42	15.02	3.9627e-06	6.8020e-06	1.3402e-09	1.2477e-10	9.9725e-06	6.8233e-06
250	750	2.84	0.74	25.77	3.5943e-06	5.5857e-06	3.5027e-09	1.2499e-10	9.9918e-06	9.5666e-06
300	1000	3.34	1.00	28.92	3.8711e-06	5.5445e-06	2.9092e-09	1.2496e-10	9.9868e-06	6.5960e-06
350	1200	3.61	1.31	47.98	9.9795e-06	8.2046e-06	5.2400e-10	1.2308e-10	9.9622e-06	9.8896e-06
400	1400	6.04	1.49	54.06	1.8704e-06	6.7112e-06	2.4277e-09	1.2487e-10	9.9592e-06	9.7753e-06
450	1500	9.61	1.61	47.48	2.8357e-06	6.4066e-06	1.8368e-09	1.2477e-10	9.9766e-06	9.8213e-06
500	1750	14.52	2.29	96.82	1.8923e-06	9.6769e-06	2.5087e-09	1.2433e-10	9.8801e-06	7.7157e-06
600	2000	10.51	3.40	173.07	2.7537e-06	4.0733e-06	3.2842e-09	1.2452e-10	9.9636e-06	7.6926e-06
800	2200	27.92	4.21	230.68	7.6531e-06	4.3184e-06	2.8696e-09	1.2441e-10	9.9961e-06	9.6622e-06
1000	2500	40.77	8.05	336.23	4.0537e-06	6.6977e-06	1.1896e-09	1.2455e-10	9.9896e-06	5.6936e-06

O-IAL and APF-IAL were, on average, 8.30 and 34.99 times faster than ProxALM, respectively, with O-IAL and APF-IAL outperforming ProxALM on all 15 instances. ProxALM often reduced the primal-feasibility residual far below the prescribed tolerance of 10^{-5}, whereas APF-IAL provided a better balance between primal feasibility, dual stationarity, and computational cost.

4.3 Elastic-Net Least-Squares Regression

Given a number of observations $N \in \mathbb{N}$, dimensions $(m, n) \in \mathbb{N}^2$, a data matrix $X \in \mathbb{R}^{N\times n}$, a response vector $y \in \mathbb{R}^N$, a constraint matrix $A \in \mathbb{R}^{m\times n}$, and $b \in \mathbb{R}^m$, we consider

$$\begin{aligned} \min_{x} \quad & \psi_s(x) + \psi_n(x) := \frac{1}{2N}\|Xx - y\|^2 + \frac{\bar{\mu}}{2}\|x\|^2 + \frac{1}{\sqrt{n}}\|x\|_1 \\ \text{s.t.} \quad & Ax = b, \end{aligned}$$

We vary $300 \le N \le 1560$, $50 \le m \le 1000$, $100 \le n \le 1500$, and the target curvature pair over $15 \le \bar{L} \le 480$ and $0 \le \bar{\mu} \le 5$. The matrix X is generated as

$$X = \frac{\sqrt{N(\bar{L} - \bar{\mu})}}{\|\widetilde{X}\|_2}\widetilde{X}$$

where $\widetilde{X} \in \mathbb{R}^{N\times n}$ is a fully dense but rank-deficient Gaussian design matrix.

A sparse reference vector x_{ref} is generated with approximately 10% nonzero entries and normalized to have unit Euclidean norm. We generate $y = Xx_{\text{ref}} + \sigma\varepsilon$, where ε is a standard Gaussian vector and $\sigma = 0.05\max\{\|Xx_{\text{ref}}\|/\sqrt{N}, 1\}$. The initial point is $x_0 = x_{\text{ref}} + (\mathbf{1}_n/\sqrt{n})$. The matrix A is initially Gaussian, after which its first row is shifted along $x_0 - x_{\text{ref}}$ so that $A_{1,:}(x_0 - x_{\text{ref}}) = 1$. Finally, $b = Ax_{\text{ref}}$. The results are reported in Table 4.3.

Table 4.3 Elastic-Net Least Squares Regression Comparison

Dimensions		Time (s)			Primal Feasibility Residual			Dual Stationarity Residual		
m	n	O-IAL	APF-IAL	ProxALM [26]	O-IAL	APF-IAL	ProxALM [26]	O-IAL	APF-IAL	ProxALM [26]
50	100	0.13	0.04	0.29	8.6296e-06	7.1069e-06	7.9688e-08	9.9672e-11	6.9169e-06	3.3123e-06
100	125	0.20	0.02	0.35	7.1036e-06	4.8189e-06	9.9291e-09	2.1931e-10	9.5429e-06	5.1250e-07
140	200	0.30	0.03	0.55	7.0256e-06	6.0711e-06	6.0293e-08	2.4822e-10	9.5494e-06	2.8900e-06
180	250	0.31	0.04	0.76	5.5541e-06	4.5893e-06	8.6723e-09	2.4786e-10	6.3845e-06	6.1334e-07
200	280	0.45	0.05	0.87	6.0040e-06	5.4705e-06	8.7956e-08	2.3822e-10	8.4981e-06	7.5677e-06
250	300	0.63	0.04	1.09	9.2774e-06	3.7229e-06	9.2569e-09	2.2389e-10	6.0941e-06	9.5241e-07
350	400	1.34	0.05	1.91	8.9104e-06	4.4272e-06	8.9300e-09	2.4985e-10	9.9473e-06	8.0528e-06
400	450	2.23	0.07	3.01	9.7796e-06	6.5893e-06	1.4934e-08	1.9934e-10	7.3994e-06	4.1762e-06
450	520	15.62	0.30	122.30	9.9297e-06	5.5994e-06	3.5779e-06	2.4147e-10	9.9162e-06	6.7195e-06
500	600	3.95	0.24	9.85	9.3737e-06	2.6084e-06	1.8283e-08	2.4263e-10	9.8893e-06	9.8151e-06
550	700	2.23	0.34	12.16	7.5875e-06	1.4928e-06	1.4847e-08	2.4637e-10	9.9663e-06	5.6968e-06
650	850	2.36	0.44	16.77	6.3560e-06	6.4186e-06	3.7237e-09	2.4342e-10	9.9593e-06	9.5102e-06
750	1000	3.32	0.64	23.04	7.7759e-06	9.5629e-06	7.3995e-09	2.4916e-10	9.9151e-06	8.7765e-06
850	1200	6.80	1.56	54.32	5.2259e-06	7.5335e-06	2.0233e-09	2.4707e-10	9.9031e-06	7.1109e-06
1000	1500	12.25	3.26	149.21	3.9914e-06	5.8409e-06	4.3336e-09	2.4741e-10	9.9514e-06	8.1034e-06

O-IAL and APF-IAL were, on average, 4.31 and 55.14 times faster than ProxALM, respectively, with O-IAL faster on all 15 instances. This problem class produced the largest average speedup for APF-IAL among the six classes.

4.4 Group-Sparse Huberized-Hinge SVM

Given a number of observations $N \in \mathbb{N}$, dimensions $(m, n) \in \mathbb{N}^2$, a data matrix $X \in \mathbb{R}^{N\times n}$, labels $y \in \{-1, 1\}^N$, a constraint matrix $A \in \mathbb{R}^{m\times n}$, and $b \in \mathbb{R}^m$, we consider

$$\begin{aligned} \min_{x} \quad & \psi_s(x) + \psi_n(x) := \frac{1}{N}\sum_{i=1}^{N} h_\delta\left(1 - y_i X_i^\top x\right) + \frac{\bar{\mu}}{2}\|x\|^2 + \lambda \sum_{G\in\mathcal{G}} \|x_G\|_2 \\ \text{s.t.} \quad & Ax = b, \end{aligned}$$

where $\delta = 2$, $\lambda = 1/\sqrt{n}$, and $\mathcal{G}$ partitions the variables into contiguous groups of at most ten coordinates. The Huberized-hinge loss is defined by

$$h_\delta(t) = \begin{cases} 0, & t \le 0, \\ t^2/(2\delta), & 0 < t < \delta, \\ t - \delta/2, & t \ge \delta. \end{cases}$$

We vary parameters $260 \le N \le 7000$, $100 \le m \le 2000$, $200 \le n \le 4000$, and the target curvature pair over $20 \le \bar{L} \le 800$ and $0 \le \bar{\mu} \le 5$. The matrix X is generated in a similar way as the previous two problem classes.

A group-sparse reference classifier x_{ref} is generated by activating approximately 20% of the groups and normalizing the result to have unit Euclidean norm. Binary labels y are obtained by adding Gaussian noise with standard deviation $0.1 \max\{\|Xx_{\text{ref}}\|/\sqrt{N}, 1\}$ to Xx_{ref} and taking componentwise signs. We set initial point $x_0 = 0$.

Each row of constraint matrix A compares the average linear scores of two synthetic cohorts. The first row compares the highest- and lowest-scoring cohorts under x_{ref} and is scaled so that $A_{1,:}x_{\text{ref}} = 1$. The remaining rows use disjoint, randomly paired cohorts and are normalized to have unit Euclidean norm. Finally, $b = Ax_{\text{ref}}$. The numerical comparisons are reported in Table 4.4.

Table 4.4 Group-Sparse Huberized-Hinge SVM Comparison

Dimensions		Time (s)			Primal Feasibility Residual			Dual Stationarity Residual		
m	n	O-IAL	APF-IAL	ProxALM [26]	O-IAL	APF-IAL	ProxALM [26]	O-IAL	APF-IAL	ProxALM [26]
100	200	1.51	0.14	0.98	9.8647e-06	2.8261e-06	2.0833e-08	2.4535e-10	9.9942e-06	9.9698e-06
125	300	3.06	0.25	2.14	9.4398e-06	4.4665e-07	2.4831e-08	2.4959e-10	9.9785e-06	8.7184e-06
160	350	4.00	0.36	3.66	8.5234e-06	2.8080e-07	2.7076e-08	2.4924e-10	9.9912e-06	9.8796e-06
220	420	6.24	0.23	3.06	9.3216e-06	1.8530e-06	1.2041e-07	2.4800e-10	9.9502e-06	7.0318e-06
300	500	11.38	0.26	2.86	9.8340e-06	7.1945e-06	7.4209e-08	2.4857e-10	9.8958e-06	6.3296e-06
340	650	13.63	0.52	5.69	9.2159e-06	2.1553e-06	2.8755e-08	2.4860e-10	9.9846e-06	6.2586e-06
410	800	17.75	0.76	10.88	9.3374e-06	2.2401e-06	7.6778e-08	2.4880e-10	9.9554e-06	5.0307e-06
500	1000	38.64	1.76	25.09	9.4024e-06	2.3469e-06	1.2538e-07	2.4732e-10	9.9932e-06	7.4665e-06
580	1300	77.52	4.40	50.60	8.3424e-06	2.3128e-06	3.4474e-08	2.4784e-10	9.9934e-06	9.1382e-06
630	1800	103.71	9.61	158.06	7.5187e-06	3.4824e-06	4.8961e-08	2.4937e-10	9.9756e-06	5.3972e-06
720	2200	185.82	16.02	256.62	9.8946e-06	5.4031e-06	1.1028e-08	2.4799e-10	9.9957e-06	7.2998e-06
800	2500	165.66	16.03	310.64	8.0908e-06	3.3600e-06	6.8596e-09	2.4931e-10	9.9595e-06	6.7552e-06
1000	2800	248.27	16.39	360.11	7.8138e-06	3.8748e-06	1.6719e-08	2.4804e-10	9.8386e-06	5.0048e-06
1200	3500	387.89	25.50	639.73	9.7420e-06	9.5725e-06	5.6077e-09	2.4937e-10	9.9801e-06	5.3442e-06
2000	4000	1513.25	42.62	1333.21	9.7640e-06	3.8331e-06	6.5937e-08	2.4522e-10	9.9615e-06	6.6974e-06

APF-IAL was, on average, 15.42 times faster than ProxALM, whereas the average speedup factor of O-IAL relative to ProxALM was 0.94, indicating that ProxALM was marginally faster on average than O-IAL. However, O-IAL consistently achieved much smaller dual-stationarity residuals of approximately 2.5×10^{-10} than ProxALM. APF-IAL was faster than ProxALM on all 15 instances considered.

4.5 Bayesian D-Optimal Experimental Design

Given an information-matrix dimension $N \in \mathbb{N}$, dimensions $(m, n) \in \mathbb{N}^2$, candidate experiment vectors $v_1, \ldots, v_n \in \mathbb{R}^N$, a constraint matrix $A \in \mathbb{R}^{m\times n}$, and $b \in \mathbb{R}^m$, we consider

$$\begin{aligned} \min_{x} \quad & \psi_s(x) := -\gamma \log\det\left(I_N + \sum_{i=1}^{n} x_i v_i v_i^\top\right) + \frac{\bar{\mu}}{2}\|x\|^2 \\ \text{s.t.} \quad & x \in \mathcal{B}^n, \qquad Ax = b, \end{aligned}$$

where

$$\mathcal{B}^n := \left[0, \frac{2}{\sqrt{n}}\right]^n.$$

We vary $3 \le N \le 17$, $5 \le m \le 200$, $25 \le n \le 300$, and the target curvature pair over $5 \le \bar{L} \le 100$ and $0 \le \bar{\mu} \le 5 \times 10^{-2}$. The dimensions satisfy $n > N(N+1)/2$.

The candidate vectors are independently sampled from the standard normal distribution and normalized so that $\|v_i\| = 1/2$. Letting $V = [v_1 \ \cdots \ v_n] \in \mathbb{R}^{N\times n}$ and $Q_0 = V^\top V$, we choose

$$\gamma = \frac{\bar{L} - \bar{\mu}}{\lambda_{\max}(Q_0 \circ Q_0)},$$

where $\circ$ denotes the Hadamard product. We set initial point $x_0 = 0$. The first row of A imposes the total-budget equality $\mathbf{1}_n^\top x = 1$. The remaining rows impose category budgets on a random partition of all but one of the candidate experiments into disjoint experiment families. Finally, we set $b = A(\mathbf{1}_n/n)$. The results are reported in Table 4.5.

Table 4.5 Bayesian Optimal Design Comparison

Dimensions		Time (s)			Primal Feasibility Residual			Dual Stationarity Residual		
m	n	O-IAL	APF-IAL	ProxALM [26]	O-IAL	APF-IAL	ProxALM [26]	O-IAL	APF-IAL	ProxALM [26]
5	25	0.45	0.09	1.25	2.7883e-07	5.9706e-07	4.5457e-09	1.2415e-10	9.9057e-06	9.4089e-06
10	30	0.61	0.27	2.55	4.5228e-07	8.2275e-07	3.9348e-08	1.1904e-10	9.9933e-06	8.7112e-06
20	50	1.12	0.60	5.23	7.9678e-06	5.8942e-06	8.5390e-08	1.2430e-10	9.9840e-06	9.3609e-06
30	70	5.88	2.10	24.69	9.8960e-06	1.0200e-06	2.2628e-07	1.2487e-10	9.9899e-06	3.0808e-06
50	80	7.15	0.51	4.75	8.2063e-06	1.5803e-06	8.4101e-08	1.2217e-10	9.9959e-06	9.8783e-06
60	90	9.33	0.59	5.57	8.9226e-06	6.4077e-06	8.0434e-08	1.2488e-10	9.9937e-06	4.0106e-08
75	115	21.45	0.85	11.64	9.1407e-06	4.1737e-06	1.3451e-07	1.2238e-10	9.9774e-06	8.2729e-06
85	125	23.16	1.03	12.00	9.7713e-06	1.8210e-06	1.9272e-07	1.2352e-10	9.9849e-06	9.3506e-06
100	175	44.07	2.41	34.99	9.1592e-06	4.1352e-06	4.2160e-07	1.2367e-10	9.9955e-06	5.8512e-08
120	200	71.97	3.58	73.93	9.6673e-06	2.3166e-06	9.1791e-06	1.2487e-10	9.9984e-06	6.9781e-07
130	220	72.12	4.48	78.33	9.7896e-06	4.1079e-06	3.5511e-06	1.2437e-10	9.9813e-06	2.0627e-07
140	230	97.34	5.40	117.74	9.7533e-06	4.9281e-06	8.5930e-06	1.2353e-10	9.9925e-06	3.5834e-06
160	250	135.96	7.10	194.70	9.7997e-06	6.4149e-06	5.2830e-06	1.2497e-10	9.9970e-06	1.8520e-07
180	275	192.57	8.58	317.97	9.9753e-06	5.9361e-06	5.2906e-06	1.2344e-10	9.9924e-06	1.4934e-07
200	300	204.77	11.30	282.49	9.9368e-06	9.2390e-06	7.9677e-06	1.2329e-10	9.9961e-06	3.3229e-07

O-IAL and APF-IAL were, on average, 1.78 and 16.79 times faster than ProxALM, respectively. O-IAL was faster than ProxALM on 10 of the 15 instances, while APF-IAL was faster than ProxALM on all 15 instances.

4.6 Huberized Quantum State Tomography Semidefinite Program (SDP)

Let $\mathbb{H}^d$ denote the space of $d \times d$ complex Hermitian matrices, equipped with the inner product $\langle U, V\rangle_F := \operatorname{Re}\operatorname{tr}(U^*V)$, and set $n = d^2$. Given measured observables $O_1, \ldots, O_N \in \mathbb{H}^d$, measurements $y \in \mathbb{R}^N$, a constraint matrix $A \in \mathbb{R}^{m\times d^2}$, and $b \in \mathbb{R}^m$, we consider

$$\begin{aligned} \min_{\rho\in\mathbb{H}^d} \quad & \psi_s(\rho) + \psi_n(\rho) := \frac{\gamma}{N}\sum_{i=1}^{N} h_\delta\left(\langle O_i, \rho\rangle_F - y_i\right) + \frac{\bar\mu}{2}\|\rho\|_F^2 + \underbrace{\iota_{\mathcal{D}_d}(\rho)}_{\psi_n(\rho)} \\ \text{s.t.} \quad & A\operatorname{svec}_{\mathbb{H}}(\rho) = b, \end{aligned}$$

where $\iota_{\mathcal{D}_d}$ is the indicator function of the density-matrix spectrahedron and $h_\delta(t)$ is defined as follows:

$$\mathcal{D}_d := \left\{\rho \in \mathbb{H}^d : \rho \succeq 0,\ \operatorname{tr}(\rho) = 1\right\}, \quad h_\delta(t) := \begin{cases} t^2/(2\delta), & |t| \le \delta, \\ |t| - \delta/2, & |t| > \delta. \end{cases}$$

The operator $\operatorname{svec}_{\mathbb{H}}(H)$ is a linear isometry that takes in a matrix $H = (H_{ij}) \in \mathbb{H}^d$ as input and outputs a vector $\mathbb{R}^{d^2}$:

$$\operatorname{svec}_{\mathbb{H}}(H) := \left[H_{11}, \ldots, H_{dd},\ \left(\sqrt{2}\operatorname{Re}(H_{ij})\right)_{i<j},\ \left(\sqrt{2}\operatorname{Im}(H_{ij})\right)_{i<j}\right]^\top.$$

We set $\delta = 0.5$ and vary $60 \le N \le 970$, $8 \le m \le 86$, and $12 \le d \le 64$. Hence, n satisfies $144 \le n \le 4096$. The target curvatures satisfy $14 \le \bar L \le 630$ and $0 \le \bar\mu \le 5$.

Each O_i is generated as a dense, traceless Hermitian matrix and normalized so that $\|O_i\|_F = 1$. We set $\gamma = [N\delta(\bar L - \bar\mu)]/\|B\|_2^2$, where $B \in \mathbb{R}^{N\times d^2}$ has rows $B_i = \operatorname{svec}_{\mathbb{H}}(O_i)^\top$. A normalized complex Gaussian vector $g \in \mathbb{C}^d$ is generated, and the reference density matrix and measurements are defined by $\rho_{\text{ref}} = 0.35\, gg^* + 0.65\,\frac{I_d}{d}$ and $y_i = \langle O_i, \rho_{\text{ref}}\rangle_F + \varepsilon_i$, where $\varepsilon_i \sim \mathcal{U}[-0.02, 0.02]$. The initial point is the maximally mixed state $\rho_0 = I_d/d$. The matrix A is initially Gaussian, after which its first row is shifted along $\operatorname{svec}_{\mathbb{H}}(\rho_0 - \rho_{\text{ref}})$, and we set $b = A\operatorname{svec}_{\mathbb{H}}(\rho_{\text{ref}})$. The results are reported in Table 4.6.

Table 4.6 Huberized Quantum State Tomography Comparison

Dimensions		Time (s)			Primal Feasibility Residual			Dual Stationarity Residual		
m	n	O-IAL	APF-IAL	ProxALM [26]	O-IAL	APF-IAL	ProxALM [26]	O-IAL	APF-IAL	ProxALM [26]
8	144	0.68	0.44	10.18	1.5906e-12	6.9649e-06	9.8046e-10	1.7489e-10	9.9437e-06	6.4462e-06
10	196	0.45	0.28	11.67	5.2495e-13	4.2385e-06	2.2599e-09	1.7676e-10	9.9640e-06	6.3797e-06
12	256	0.49	0.41	14.21	9.9633e-13	3.2955e-06	2.3523e-09	1.7645e-10	9.9992e-06	5.4347e-06
15	324	6.78	5.90	358.13	8.2771e-06	9.0839e-07	3.5424e-09	1.7670e-10	9.9997e-06	8.1455e-06
18	400	4.24	2.17	181.61	2.3007e-12	1.5707e-06	2.4364e-09	1.7641e-10	9.9970e-06	7.7798e-06
21	484	3.27	1.88	55.63	9.7588e-08	1.6571e-06	5.1399e-10	1.7654e-10	9.9968e-06	6.5909e-06
24	576	2.55	1.45	35.36	6.6984e-07	2.7532e-06	1.5051e-09	1.7600e-10	9.9852e-06	6.7207e-06
27	676	3.92	1.27	35.01	1.7832e-06	8.3042e-07	1.9601e-09	1.7504e-10	9.9963e-06	9.1124e-06
30	784	3.70	1.33	37.50	3.2894e-06	1.6693e-06	1.3051e-10	1.7633e-10	9.9162e-06	7.9736e-06
35	1024	5.80	1.38	34.42	8.1250e-06	1.8668e-06	3.7588e-10	1.7563e-10	9.9952e-06	6.8599e-06
42	1296	4.57	1.65	38.22	1.6795e-07	1.6715e-06	4.2726e-10	1.7626e-10	9.9749e-06	9.3537e-06
50	1600	4.16	1.84	45.34	2.6703e-07	1.7037e-06	3.6872e-10	1.7611e-10	9.9895e-06	9.2976e-06
61	2304	7.75	2.51	82.04	3.4542e-07	1.0970e-06	5.2736e-10	1.7339e-10	9.9901e-06	9.1086e-06
73	3136	11.14	3.66	115.43	4.8447e-07	1.4990e-06	6.9661e-10	1.7567e-10	9.9307e-06	9.9191e-06
86	4096	18.02	6.10	232.20	7.0560e-07	1.8597e-06	1.1957e-09	1.7302e-10	9.9470e-06	9.3482e-06

O-IAL and APF-IAL were, on average, 18.30 and 35.24 times faster than ProxALM, respectively. O-IAL consistently attained dual-stationarity residuals near 1.7×10^{-10}, while APF-IAL achieved the smallest runtimes and kept both residuals within the requested tolerances.

4.7 Summary of Numerical Results

In conclusion, APF-IAL was the fastest method in these experiments and struck the best balance between achieving small primal-feasibility and dual-stationarity residuals. Overall, O-IAL was the second-fastest method and frequently achieved exceptionally small dual-stationarity residuals. This behavior is consistent with its design: O-IAL explicitly maintains approximate stationarity throughout its iterations and consequently produces dual residuals near 10^{-10}, despite the requested tolerance being only 10^{-5}.

ProxALM was the least computationally efficient of the three methods. Although it often reduced the primal-feasibility residual well below the prescribed threshold, its main bottleneck was efficiently reducing the dual-stationarity residual. This behavior can be explained by the distinction between AL and PAL methods. After the multiplier update in an AL method, approximate stationarity of the AL subproblem directly controls the dual stationarity residual of the original linearly constrained problem at the updated multiplier. By contrast, the stationarity condition for a PAL subproblem contains an additional proximal-displacement term. Consequently, accurate minimization of the PAL subproblem does not translate as directly into a small stationarity residual for the original problem unless the proximal displacement is also sufficiently small. This helps explain why ProxALM had greater difficulty efficiently reducing the dual-stationarity residual.

5 Conclusion

In this paper, we developed three inexact augmented Lagrangian methods, O-IAL, OPF-IAL, and APF-IAL, that attain an optimal complexity bound of $\mathcal{O}(\epsilon^{-1})$ in the convex setting and a near-optimal bound of $\mathcal{O}(\epsilon^{-1/2}\log(\epsilon^{-1}))$ in the strongly convex setting. Among these methods, OPF-IAL and APF-IAL are parameter-free. To solve the inexact AL subproblems, our methods employ either PF-AR or R-FISTA, both of which are accelerated parameter-free methods developed in this paper. Finally, numerical experiments on six important classes of problems demonstrate that O-IAL and APF-IAL are often 5–50 times faster than a representative proximal augmented Lagrangian method.

A Proof of Theorem 2.4

The following lemma characterizes properties of the output (x_s^{++}, x_s^+, M_s) of the Backtracking function.

Lemma A.1. *For every $s \geq 1$, the following relations hold:*

$$x_s^{++} := x^{++}(2M_s, \sigma_s, \bar{x}_s; x_s) \tag{A.1}$$

$$x_s^{+} := x^{+}(2M_s; x_s) := \operatorname{argmin}_u \langle \nabla \psi_s(x_s), u\rangle + \psi_n(u) + M_s\|u - x_s\|^2 \tag{A.2}$$

$$\psi_s(x_s^{++}) - \psi_s(x_s) - \langle \nabla \psi_s(x_s), x_s^{++} - x_s\rangle \leq \frac{M_s}{2}\|x_s^{++} - x_s\|^2 \tag{A.3}$$

$$\|\nabla \psi_s(x_s^{+}) - \nabla \psi_s(x_s)\| \leq 2M_s\|x_s^{+} - x_s\| \tag{A.4}$$

$$M_0 \leq M_s \leq \max\left\{M_0, 2\bar{L}\right\}. \tag{A.5}$$

Proof. Relations (A.1) and (A.2) follow immediately from the way x^{++} and x^+ are computed in steps 2 and 3 of Backtracking and the fact that the s-th iteration of Algorithm 2.2 calls Backtracking with inputs $(\psi_s, \psi_n, \sigma_s, x_s, M_{s-1})$ and outputs $(x_s^{++}, x_s^+, M_s) := (x^{++}, x^+, M)$. Likewise, relations (A.3) and (A.4) follow immediately from the checks that Backtracking performs in (2.5) and (2.6).

To see (A.5), observe that (A.3) and (A.4) in Backtracking always hold if $M_j \geq \bar{L}$ for some j generated during Backtracking because ψ_s is assumed to be $\bar{L}$-smooth. It follows from this fact, the fact that M_{j+1} is set inside of Backtracking as $M_{j+1} = 2M_j$ if (A.3) and (A.4) do not hold, and the fact that Algorithm 2.2 calls Backtracking with $M_0 = M_{s-1}$ that $M_{s-1} \leq M_s \leq \max\left\{M_{s-1}, 2\bar{L}\right\}$. Relation (A.5) then follows from this fact and a simple induction argument. □

The following lemma provides key bounds which involve the gradient mapping. We omit the proof of the lemma since the proof is identical to the proofs of (3.5) and (3.6) in [16] with $(\eta, \sigma, \bar{x}, x)$ replaced by $(2M_s, \sigma_s, \bar{x}_s, x_s)$.

Lemma A.2. *For every $s \geq 1$, we have that*

$$\|x^+(2M_s + \sigma_s; x_s) - x^{++}(2M_s, \sigma_s, \bar{x}_s; x_s)\| \leq \frac{\sigma_s}{2M_s + \sigma_s}\|x_s - \bar{x}_s\|; \tag{A.6}$$

$$\|G_{2M_s}(x_s)\| \leq \|G_{\sigma_s + 2M_s}(x_s)\|. \tag{A.7}$$

The next lemma characterizes key descent properties of x_s^{++}.

Lemma A.3. *For every $s \geq 1$, we have that the following relation holds:*

$$\begin{aligned}&\left[\psi_s(x_s^{++}) + \psi_n(x_s^{++}) + \frac{\sigma_s}{2}\|x_s^{++} - \bar{x}_s\|^2\right] - \left[\psi_s(u) + \psi_n(u) + \frac{\sigma_s}{2}\|u - \bar{x}_s\|^2\right] \\ &\leq M_s\|u - x_s\|^2 - \frac{\sigma_s + 2M_s}{2}\|u - x_s^{++}\|^2 - \frac{M_s}{2}\|x_s^{++} - x_s\|^2 \quad \forall u \in \operatorname{dom}\psi_n.\end{aligned} \tag{A.8}$$

Proof. It follows from (A.1) that

$$\psi_s(x_s^{++}) - \psi_s(x_s) - \langle \nabla \psi_s(x_s), x_s^{++} - x_s\rangle \leq \frac{M_s}{2}\|x_s^{++} - x_s\|^2$$

where $x_s^{++} := x^{++}(2M_s, \sigma_s, \bar{x}_s; x_s)$. The result is then immediate from (3.8) in [16] with $(x^{++}, \eta, \bar{x}, x, M)$ in the statement replaced with $(x_s^{++}, 2M_s, \sigma_s, \bar{x}_s, x_s, M_s)$. □

The following proposition gives a key bound on the norm of the gradient mapping in terms of the distances between x_s^* and x_s and x_s^* and $\bar{x}_s$.

Proposition A.4. *For any $s \geq 1$, we have that the following relation holds:*

$$\|G_{2M_s}(x_s)\| \leq (3\sigma_s + 4M_s)\|x_s^* - x_s\| + \sigma_s\|x_s^* - \bar{x}_s\|. \tag{A.9}$$

where x_s^* is as in (2.10) and $G_{2M_s}(x_s) := 2M_s(x_s - x_s^+)$.

Proof. It follows from the definition of x_s^* in (2.10) and Lemma A.3 with $u = x_s^*$ that the following holds:

$$\begin{aligned} 0 &\overset{(2.10)}{\leq} \left[\psi_s(x_s^{++}) + \psi_n(x_s^{++}) + \frac{\sigma_s}{2}\|x_s^{++} - \bar{x}_s\|^2\right] - \left[\psi_s(x_s^*) + \psi_n(x_s^*) + \frac{\sigma_s}{2}\|x_s^* - \bar{x}_s\|^2\right] \\ &\overset{(A.8)}{\leq} M_s\|x_s^* - x_s\|^2 - \frac{\sigma_s + 2M_s}{2}\|x_s^* - x_s^{++}\|^2 - \frac{M_s}{2}\|x_s^{++} - x_s\|^2. \end{aligned}$$

Thus, rearranging the above inequality, we have that $0.5M_s\|x_s^{++} - x_s\|^2 \leq M_s\|x_s^* - x_s\|^2$ and hence $\|x_s^{++} - x_s\| \leq \sqrt{2}\|x_s^* - x_s\|$. It then follows from this relation, (A.6), (A.7) that

$$\begin{aligned} \|G_{2M_s}(x_s)\| &\leq \|G_{\sigma_s+2M_s}(x_s)\| = (\sigma_s + 2M_s)\|x_s - x^+(2M_s + \sigma_s; x_s)\| \\ &\leq (\sigma_s + 2M_s)\|x_s - x_s^{++}\| + (\sigma_s + 2M_s)\|x_s^{++} - x^+(2M_s + \sigma_s; x_s)\| \\ &\overset{(A.6)}{\leq} (\sigma_s + 2M_s)\|x_s - x_s^{++}\| + \sigma_s\|x_s - \bar{x}_s\| \\ &\leq (2\sigma_s + 4M_s)\|x_s^* - x_s\| + \sigma_s\|x_s - x_s^*\| + \sigma_s\|x_s^* - \bar{x}_s\| \\ &= (3\sigma_s + 4M_s)\|x_s^* - x_s\| + \sigma_s\|x_s^* - \bar{x}_s\| \end{aligned}$$

which immediately implies the result holds. □

The following proposition provides key bounds on the distances of the iterates generated by AR-L. We omit the proof since the proof follows similarly to the proof of Propositions 2.1 and 3.2 in [16].

Proposition A.5. *For all $s \geq 1$, we have that the following relations must hold:*

$$\|x_{s-1} - x_s^*\| \leq \|x_{s-1} - x_{s-1}^*\| \tag{A.10}$$

$$\sigma_s\|\bar{x}_s - x_s^*\| \leq \sum_{i=1}^{s}(\sigma_{i-1} + \sigma_i)\|x_{i-1}^* - x_{i-1}\|. \tag{A.11}$$

The following proposition shows that the iterate v_s computed in step 12 lies in the subdifferential of ϕ and also shows a key bound on its norm.

Proposition A.6. *For every $s \geq 1$, the following relations must hold:*

$$v_s \in \nabla\psi_s(x_s^+) + \partial\psi_n(x_s^+) \tag{A.12}$$

$$\|v_s\| \leq (6\sigma_s + 8M_s)\|x_s^* - x_s\| + 2\sigma_s\|x_s^* - \bar{x}_s\|. \tag{A.13}$$

where recall v_s is computed as in step 12 of Algorithm 2.2.

Proof. It follows from the definition of x_s^+ in (A.2) and the optimality of the subproblem in the definition that $0 \in \nabla\psi_s(x_s) + \partial\psi_n(x_s^+) + 2M_s(x_s^+ - x_s)$. Hence, it follows that $2M_s(x_s - x_s^+) + \nabla\psi_s(x_s^+) - \nabla\psi_s(x_s) \in \nabla\psi_s(x_s^+) + \partial\psi_n(x_s^+)$ which immediately implies the inclusion in (A.12) in view of the definition of v_s in step 12 of Algorithm 2.2.

Now observe that it follows from the definition of x_s^+ in (A.2) and the definition of $G_\eta(x)$ in (2.4) that $G_{2M_s}(x_s) = 2M_s(x_s - x_s^+)$. It follows from this observation, the definition of v_s in step 12 of Algorithm 2.2, (A.4), and (A.9) that

$$\begin{aligned}\|v_s\| &= \|2M_s(x_s - x_s^+) + \nabla\psi_s(x_s^+) - \nabla\psi_s(x_s)\| \le \|2M_s(x_s - x_s^+)\| + \|\nabla\psi_s(x_s^+) - \nabla\psi_s(x_s)\| \\ &\overset{(A.4)}{\le} 2M_s\|(x_s - x_s^+)\| + 2M_s\|x_s^+ - x_s\| = 2\|G_{2M_s}(x_s)\| \overset{(A.9)}{\le} (6\sigma_s + 8M_s)\|x_s^* - x_s\| + 2\sigma_s\|x_s^* - \bar{x}_s\|\end{aligned}$$

which shows (A.13). □

We are now ready to prove the main complexity theorem of AR-L.

Proof of Theorem 2.4. Observe that it follows from (A.11) and (A.13) that for every $s \ge 1$, it holds that

$$\begin{aligned}\|v_s\| &\le (6\sigma_s + 8M_s)\|x_s^* - x_s\| + 2\sigma_s\|x_s^* - \bar{x}_s\| \\ &\overset{(A.11)}{\le} (6\sigma_s + 8M_s)\|x_s^* - x_s\| + 2\sum_{i=1}^{s}(\sigma_{i-1} + \sigma_i)\|x_{i-1}^* - x_{i-1}\| \qquad \text{(A.14)} \\ &= (6\sigma_s + 8M_s)\|x_s^* - x_s\| + 2\sigma_1\|x_0 - x^*\| + 2\sum_{i=2}^{s}(\sigma_{i-1} + \sigma_i)\|x_{i-1}^* - x_{i-1}\|. \qquad \text{(A.15)}\end{aligned}$$

Now observe that since ϕ_s in (2.10) is σ_s-strongly convex, it follows that

$$\|x - x_s^*\| \le \sqrt{\frac{2}{\sigma_s}}\sqrt{\phi_s(x) - \phi_s(x_s^*)} \quad \forall x \in \operatorname{dom}\psi_n. \qquad \text{(A.16)}$$

It then follows from (A.16), (2.8), and the bound $L_s^k \le c_{\mathcal{A}}\bar{L}$ that, after the k-th evaluation of $\nabla\psi_s$, subroutine $\mathcal{A}$ computes an approximate solution x_s^k such that

$$\begin{aligned}\|x_s^k - x_s^*\| &\overset{(A.16)}{\le} \sqrt{\frac{2}{\sigma_s}}\sqrt{\phi_s(x_s^k) - \phi_s(x_s^*)} \\ &\overset{(2.8)}{\le} \sqrt{\frac{2}{\sigma_s}}\sqrt{\frac{L_s^k\|x_{s-1} - x_s^*\|^2}{k^2}} \le \frac{1}{k}\sqrt{\frac{2c_{\mathcal{A}}\bar{L}}{\sigma_s}}\|x_{s-1} - x_s^*\|. \qquad \text{(A.17)}\end{aligned}$$

Moreover, it follows from the assumption on $\mathcal{A}$ in (2.9) and the bound $L_s^k \le c_{\mathcal{A}}\bar{L}$ that the number of gradient evaluations that $\mathcal{A}$ performs is bounded by

$$N_s := \left\lceil 8\sqrt{\frac{2L_s^k}{\sigma_s}} \right\rceil \le \left\lceil 8\sqrt{\frac{2c_{\mathcal{A}}\bar{L}}{\sigma_s}} \right\rceil. \qquad \text{(A.18)}$$

It then follows from (A.10), (A.17), and (A.18) that

$$\begin{aligned}\|x_s - x_s^*\| &\overset{(A.17)}{\le} \frac{1}{N_s}\sqrt{\frac{2c_{\mathcal{A}}\bar{L}}{\sigma_s}}\|x_{s-1} - x_s^*\| \\ &\overset{(A.18)}{\le} \frac{1}{8}\|x_{s-1} - x_s^*\| \overset{(A.10)}{\le} \frac{1}{8}\|x_{s-1} - x_{s-1}^*\|. \qquad \text{(A.19)}\end{aligned}$$

It then follows from the bound on v_s in (A.15), (A.19), the fact that $8^{-s} \leq (0.5)4^{-s}$ for $s \geq 2$, and the fact that $\sigma_s = 4\sigma_{s-1}$ that the following relations hold for every $s \geq 2$:

$$\begin{aligned}
\|v_s\| &\overset{(A.15)}{\leq} (6\sigma_s + 8M_s)\|x_s^* - x_s\| + 2\sigma_1\|x_0 - x^*\| + 2\sum_{i=2}^{s}(\sigma_{i-1} + \sigma_i)\|x_{i-1}^* - x_{i-1}\| \\
&\overset{(A.19)}{\leq} (6\sigma_s + 8M_s)8^{-s}\|x^* - x_0\| + 2\sigma_1\|x_0 - x^*\| + 10\sigma_1\sum_{i=2}^{s} 4^{i-2}8^{-i+1}\|x^* - x_0\| \\
&\leq (3\sigma_s + 4M_s)4^{-s}\|x^* - x_0\| + 2\sigma_1\|x_0 - x^*\| + \frac{5}{2}\sigma_1\|x_0 - x^*\| \\
&= (3\sigma_s + 4M_s)4^{-s}\|x^* - x_0\| + \frac{9}{2}\sigma_1\|x_0 - x^*\|. \qquad \text{(A.20)}
\end{aligned}$$

We now estimate $\|\hat{v}\|$ when the termination condition stopcrit ≤ 0 is satisfied, where stopcrit $= M_s - \sigma_s$. If the criterion is satisfied with $s = 1$, then it follows from step 14 of AR-L that it terminates with $(\hat{x}, \hat{v}, M) = (x_1^+, v_1, M_1)$. Moreover, it follows from the fact that $\sigma_0 = 0$, $\sigma_1 \geq M_1$, (A.14) and (A.19) with $s = 1$, and the fact that $8^{-1} \leq (0.5)4^{-1}$ that

$$\begin{aligned}
\|\hat{v}\| = \|v_1\| &\leq (6\sigma_1 + 8M_1)\|x_1^* - x_1\| + 2\sigma_1\|x_0 - x^*\| \\
&\overset{(A.19)}{\leq} (3\sigma_1 + 4\sigma_1) * \frac{1}{4}\|x_0 - x^*\| + 2\sigma_1\|x_0 - x^*\| \\
&\leq 7\sigma_1\|x_0 - x^*\| \leq 7\sigma_1 D. \qquad \text{(A.21)}
\end{aligned}$$

Otherwise, let $S > 1$ be the smallest index s for which the termination criterion $M_s - \sigma_s \leq 0$ is satisfied. It follows from (A.20), the fact that the diameter of $\operatorname{dom}\psi_n$ is D, the identity $\sigma_s = 4^{s-1}\sigma_1$, and the termination condition $\sigma_S \geq M_S$ that the output $\hat{v} = v_S$ satisfies

$$\begin{aligned}
\|\hat{v}\| = \|v_S\| &\leq (3\sigma_S + 4M_S)4^{-S}\|x^* - x_0\| + \frac{9}{2}\sigma_1\|x_0 - x^*\| \\
&\leq (7\sigma_S)4^{-S}\|x^* - x_0\| + \frac{9}{2}\sigma_1\|x_0 - x^*\| \\
&= \frac{7}{4}\sigma_1\|x^* - x_0\| + \frac{9}{2}\sigma_1\|x^* - x_0\| \leq 7\sigma_1\|x^* - x_0\| \leq 7\sigma_1 D. \qquad \text{(A.22)}
\end{aligned}$$

Hence, it follows from (A.21) and (A.22) that the output $\hat{v}$ always satisfies the inequality in (2.11). Moreover, since the inclusion (A.12) holds for every $s \geq 1$, it follows from the way the output $(\hat{x}, \hat{v}, \hat{M})$ is defined in step 14 that the pair $(\hat{x}, \hat{v})$ satisfies the inclusion in (2.11). Finally, because (A.5) holds at every iteration $s \geq 1$, and $\hat{M} = M_s$ for some iteration index s, we conclude that $\hat{M}$ satisfies (2.12).

To establish the complexity bound in (2.13), assume that $M_0 \leq 2\bar{L}$ and let S be the smallest index for which termination criterion $\sigma_s \geq M_s$ is satisfied. First, observe that (A.5) implies that, for every $s = 1, \ldots, S$, the backtracking procedure outputs M_s satisfying $M_s \leq \max\{M_0, 2\bar{L}\}$. We next derive an upper bound on S. Using the bound above, the update $\sigma_s = 4\sigma_{s-1}$ for $s > 1$, the assumption $M_0 \leq 2\bar{L}$, and the fact that $S > 1$ is the first index such that $\sigma_S \geq M_S$, we obtain

$$M_S \leq 4^{S-1}\sigma_1 = \sigma_S = 4\sigma_{S-1} < 4M_{S-1} \leq \max\{4M_0, 8\bar{L}\} \leq 8\bar{L}.$$

It is then immediate from the above relations that $S \leq 2 + \log_4^+(2\bar{L}/\sigma_1)$.

We now bound the total number of gradient evaluations performed during AR-L to find such a $\hat{v}$. Recall that during each outer iteration of AR-L, algorithm $\mathcal{A}$ performs N_s gradient evaluations where N_s is bounded as in (A.18). Furthermore, as shown above AR-L performs at most S outer iterations, where $S \leq 2 + \log_4(2\bar{L}/\sigma_1)$. Hence, the total number of gradient evaluations that $\mathcal{A}$

performs throughout the AR-L method can be bounded as

$$\sum_{s=1}^{S} N_s \leq \sum_{s=1}^{S} 1 + 8\sqrt{\frac{2c_{\mathcal{A}}\bar{L}}{\sigma_s}}$$
$$\leq S + 8\sqrt{\frac{2c_{\mathcal{A}}\bar{L}}{\sigma_1}} \sum_{s=1}^{\infty} 2^{-(s-1)} \leq S + 16\sqrt{\frac{2c_{\mathcal{A}}\bar{L}}{\sigma_1}}.$$

Now observe that the call made to the Backtracking function in step 11 of the s-th iteration of AR-L performs at most $2 + \log_2(M_s/M_{s-1})$ gradient evaluations. Thus, it follows from summing and using (A.5) with $s = S$ that the total number of extra gradient evaluations from the Backtracking function is bounded by $2S + \log_2(2\bar{L}/M_0)$. Thus, summing everything together and using the fact that $3\log_4 t < 2\sqrt{t}$ for all $t > 0$, the total number of gradient evaluations that the AR-L method performs is bounded by

$$3S + 16\sqrt{\frac{2c_{\mathcal{A}}\bar{L}}{\sigma_1}} + \log_2\left(\frac{2\bar{L}}{M_0}\right) \leq 3\left[2 + \log_4^+\left(\frac{2\bar{L}}{\sigma_1}\right)\right] + 16\sqrt{\frac{2c_{\mathcal{A}}\bar{L}}{\sigma_1}} + \log_2\left(\frac{2\bar{L}}{M_0}\right)$$
$$= 6 + 3\log_4^+\left(\frac{2\bar{L}}{\sigma_1}\right) + 16\sqrt{2c_{\mathcal{A}}}\sqrt{\frac{\bar{L}}{\sigma_1}} + \log_2\left(\frac{2\bar{L}}{M_0}\right)$$
$$< 6 + 2\sqrt{\frac{2\bar{L}}{\sigma_1}} + 16\sqrt{2c_{\mathcal{A}}}\sqrt{\frac{\bar{L}}{\sigma_1}} + \log_2\left(\frac{2\bar{L}}{M_0}\right) = 6 + C_1\sqrt{\frac{\bar{L}}{\sigma_1}} + \log_2\left(\frac{2\bar{L}}{M_0}\right)$$

where $C_1 := 2\sqrt{2} + 16\sqrt{2c_{\mathcal{A}}}$. This immediately proves the gradient evaluation complexity bound (2.13) of the AR-L method.

The last conclusion of Theorem 2.4 follows immediately from the first conclusion of Theorem 2.4 and the assumption that $\sigma_1 = \epsilon/(7\overline{D})$ for some scalar $\overline{D} > 0$. □

B Proof of Proposition 3.2

This appendix section is dedicated to proving Proposition 3.2. It is divided into two subsections. The first subsection is dedicated to proving Proposition 3.2(a)-(b) while the second subsection is dedicated to proving Proposition 3.2(c).

B.1 Proof of Proposition 3.2(a)-(b)

The following lemma presents key properties of the iterates generated during the ℓ-th cycle of R-FISTA. Its proof is not given as it closely resembles the proofs of Lemmas A.3 and A.4 in [43].

Lemma B.1. *Let θ, ζ_ℓ, and Q_ℓ be as in (3.10). For every inner iteration index $j \geq 1$ generated during the ℓ-th cycle of R-FISTA, the following statements hold:*

(a) $\{L_j\}$ is nondecreasing;

(b) the following relations hold

$$\tau_{j-1} = 1 + \frac{\tilde{\mu}A_{j-1}}{2}, \quad \frac{\tau_{j-1}A_j}{a_{j-1}^2} = L_j, \tag{B.1}$$

$$\underline{M}_\ell \leq L_{j-1} \leq \max\{\underline{M}_\ell, \theta\bar{L}\}, \tag{B.2}$$

$$v_j \in \nabla\psi_s(y_j) + \partial\psi_n(y_j), \quad \|v_j\| \leq \zeta_\ell\|y_j - \tilde{x}_{j-1}\|; \tag{B.3}$$

(c) it holds that

$$A_j L_j \geq \max\left\{\frac{j^2}{4}, \left(1+Q_\ell^{-1}\right)^{2(j-1)}\right\}. \tag{B.4}$$

The proof of Proposition 3.2(a)-(b) is now presented. The proof of Proposition 3.2(a)-(b) has a similar flavor to the proof of Proposition 2.1(a)-(b) in [44], but we include it here for completeness.

Proof of Proposition 3.2(a)-(b). (a) Consider the ℓ-th cycle of R-FISTA and let m denote the first term in the sum in (3.11). Using this definition and the inequality $\log(1+\alpha) \geq \alpha/(1+\alpha)$ for any $\alpha > -1$, it is easy to verify that

$$\left(1+Q_\ell^{-1}\right)^{2(m-1)} \geq \frac{D^2\zeta_\ell^2}{\chi\epsilon^2}. \tag{B.5}$$

We claim that the ℓ-th cycle of R-FISTA either terminates successfully with stopcrit $\leq \epsilon$ or stops in its step 7 because the inequality (3.9) did not hold (in one of its inner iterations) in at most m inner ACG iterations. Indeed, it suffices to show that if the ℓ-th cycle of R-FISTA has not stopped in step 7 up to (and including) the m-th inner ACG iteration, then the main while loop of R-FISTA must successfully stop with stopcrit $\leq \epsilon$ at the m-th iteration. So, assume that the ℓ-th cycle of R-FISTA has not stopped in its step 7 up to (and including) the m-th inner ACG iteration. Hence, it follows that (3.9) holds with $j = m$.

This observation together with the inequality (B.3) with $j = m$, relation (B.5), and inequality (B.4) with $j = m$, then imply that

$$D^2 \geq \|y_m - x_0\|^2 \overset{(3.9)}{\geq} \chi A_m L_m \|y_m - \tilde{x}_{m-1}\|^2 \overset{(B.3)}{\geq} \frac{\chi}{\zeta_\ell^2} A_m L_m \|v_m\|^2 \tag{B.6}$$

$$\overset{(B.4)}{\geq} \frac{\chi}{\zeta_\ell^2}\left(1+Q_\ell^{-1}\right)^{2(m-1)}\|v_m\|^2 \overset{(B.5)}{\geq} \frac{D^2}{\epsilon^2}\|v_m\|^2 \tag{B.7}$$

and hence that $\|v_m\| \leq \epsilon$. In view of the criterion of the main while loop of R-FISTA, the ℓ-th cycle of R-FISTA must thus successfully stop at the end of its m-th inner ACG iteration. Hence, the above claim holds. Moreover, in view of (B.2), it follows that the second term in (3.11) is a bound on the total number of times L_j is multiplied by 2 and step 5 is repeated. Since exactly one gradient evaluation occurs every time step 5 is executed, the result of part (a) must hold.

(b) Suppose that the main while loop of the ℓ-th cycle of R-FISTA terminates successfully with stopcrit $\leq \epsilon$. Since stopcrit$=\|v_j\|$ whenever no restart occurs and the inclusion in (B.3) holds for every generated iterate, the output pair (y, v) satisfies (3.12). □

B.2 Proof of Proposition 3.2(c)

For the remainder of this subsection, consider the ℓ-th cycle of R-FISTA. We assume that $j \geq 1$ is an inner ACG iteration index generated during the cycle and that $\tilde{\mu} = \mu_{\ell-1}$ is in the interval $(0, \bar{\mu}]$.

A useful fact that is used in the proof of following result is that if a function $\Psi : \mathbb{E} \to \mathbb{R} \cup \{+\infty\}$ is ν-convex with modulus $\nu > 0$, then it has a unique global minimum x^* and

$$\Psi(x^*) + \frac{\nu}{2}\|\cdot - x^*\|^2 \leq \Psi(\cdot). \tag{B.8}$$

The first lemma below characterizes a key property of the iterate y_j.

Lemma B.2. *For every iteration index $j \geq 1$ generated during the ℓ-th cycle of R-FISTA, it holds that*

$$y_j := \operatorname*{argmin}_{u \in \operatorname{dom}\psi_n}\left\{q_{j-1}(u; \tilde{x}_{j-1}, L_j) := \ell_{\psi_s}(u; \tilde{x}_{j-1}) + \psi_n(u) + \frac{L_j}{2}\|u - \tilde{x}_{j-1}\|^2\right\}. \tag{B.9}$$

Proof. It follows from the update rule for y_j in (3.4), the definition of the proximal operator, the definition of the linearization of ψ_s in (1.10), and algebraic manipulation that

$$y_j \overset{(3.4)}{=} \operatorname{argmin}_{u\in\operatorname{dom}\psi_n}\left\{\frac{1}{L_j}\psi_n(u)+\frac{1}{2}\left\|u-\left(\tilde{x}_{j-1}-\frac{1}{L_j}\nabla\psi_s(\tilde{x}_{j-1})\right)\right\|^2\right\}$$
$$\overset{(1.10)}{=} \operatorname{argmin}_{u\in\operatorname{dom}\psi_n}\left\{\ell_{\psi_s}(u;\tilde{x}_{j-1})+\psi_n(u)+\frac{L_j}{2}\|u-\tilde{x}_{j-1}\|^2\right\}$$

which immediately implies that (B.9) holds. □

It can easily be seen that Lemma B.2 and the fact that $\tilde{\mu}=\mu_{\ell-1}$ is in the interval $(0,\bar{\mu}]$ imply that Lemmas A.2-A.6 in [44] hold. The next lemma thus restates Lemma A.6 of [44] which lower bounds the difference in our potential function $\sigma_j(x)$.

Lemma B.3. *For every $j\geq 1$ and $x\in\operatorname{dom}\psi_n$, we have*

$$\sigma_{j-1}(x)-\sigma_j(x)\geq\frac{\chi A_j L_j}{2}\|y_j-\tilde{x}_{j-1}\|^2 \tag{B.10}$$

where

$$\sigma_j(x):=A_j[\psi(y_j)-\psi(x)]+\frac{\tau_j}{2}\|x-x_j\|^2.$$

Next, we state a basic result that is useful in deriving complexity bounds for R-FISTA.

Lemma B.4. *For every $j\geq 2$ and $x\in\operatorname{dom}\psi_n$, it holds that*

$$A_{j-1}[\psi(y_{j-1})-\psi(x)]+\frac{\tau_{j-1}}{2}\|x-x_{j-1}\|^2\leq\frac{1}{2}\|x-x_0\|^2-\frac{\chi}{2}\sum_{i=1}^{j-1}A_iL_i\|y_i-\tilde{x}_{i-1}\|^2. \tag{B.11}$$

Proof. The proof follows immediately by summing relation (B.10) from 1 to $j-1$ and noting that $A_0=0$ and $\tau_0=1$. □

We are now ready to prove Proposition 3.2(c).

Proof of Proposition 3.2(c). Consider the ℓ-th cycle of R-FISTA and assume that it is performed with $\tilde{\mu}=\mu_{\ell-1}$ in the interval $(0,\bar{\mu}]$. Using relation (B.11) with $x=y_{j-1}$, it follows that

$$\|y_{j-1}-x_0\|^2\overset{(B.11)}{\geq}\chi\sum_{i=1}^{j-1}A_iL_i\|y_i-\tilde{x}_{i-1}\|^2\geq\chi A_{j-1}L_{j-1}\|y_{j-1}-\tilde{x}_{j-2}\|^2.$$

It then follows from the above relation that for any inner iteration index $j\geq 1$ generated during the ℓ-th cycle, it holds that

$$\|y_j-x_0\|^2\geq\chi A_jL_j\|y_j-\tilde{x}_{j-1}\|^2. \tag{B.12}$$

Hence, relation (B.12) implies that the inequality (3.9) checked in step 7 of R-FISTA always holds for every inner iteration index $j\geq 1$ generated during the ℓ-th cycle. Hence, the ℓ-th cycle of R-FISTA never terminates in its step 7 and a restart is not executed. This observation together with Proposition 3.2(a)-(b) then immediately imply that the main while loop of the ℓ-th cycle must terminate successfully with stopcrit$\leq\epsilon$ and a pair (y,v) satisfying (3.12) in at most (3.11) ACG iterations/gradient evaluations. □

C Proof of Theorem 3.3

This subsection is dedicated to proving Theorem 3.3. As mentioned in the statement of Theorem 3.3 we assume that $(\mu_0, \bar{M}_0)$ satisfy $\mu_0 \geq \bar{\mu}$ and $\bar{M}_0 \leq 2\bar{L}$.

The proposition below establishes an upper bound on the number of cycles that R-FISTA performs and an upper bound on the number of inner ACG iterations that each cycle of R-FISTA performs. Its proof is similar to that of Proposition 2.2 in [44], but we include it here for completeness.

Proposition C.1. *The following statements about R-FISTA hold:*

(a) R-FISTA performs at most $\lceil \log_2^{++}(2\mu_0/\bar{\mu}) \rceil$ *cycles before terminating with a pair* (y, v) *that is an* ϵ*-optimal solution of* (3.1)*;*

(b) each cycle of R-FISTA performs at most

$$\mathcal{O}_1\left(\sqrt{\frac{\bar{L}}{\bar{\mu}}}\log^{++}\left(\frac{D\bar{L}}{\epsilon}\right) + \log_2^+(\bar{L}/\bar{M}_0)\right) \tag{C.1}$$

ACG iterations/gradient evaluations.

Proof. (a) It follows from Proposition 3.2(b) that if the main while loop of a cycle of R-FISTA terminates successfully with stopcrit $\leq \epsilon$, then the pair (y, v) that R-FISTA outputs satisfies (3.12), i.e., the pair is an ϵ-optimal solution of (3.1). Moreover, Proposition 3.2(c) implies that if the ℓ-th cycle of R-FISTA is performed with $\mu_{\ell-1} \in (0, \bar{\mu}]$, then the main while loop of the cycle always stops successfully with stopcrit$\leq \epsilon$ and hence R-FISTA terminates. The conclusion of part(a) then follows from these two observations and the fact that μ_ℓ is updated according to the rule $\mu_\ell = \mu_{\ell-1}/2$ if a restart occurs and a new cycle begins.

(b) First observe that part(a) and the fact that $\mu_0 \geq \bar{\mu}$ imply that $\mu_\ell \geq \min\{\mu_0, \bar{\mu}/2\} = \bar{\mu}/2$ for all $\ell \geq 1$. Moreover, it follows from the way $\underline{\mathrm{M}}_\ell$ is set in the Initialization phase of R-FISTA that $\bar{M}_0 \leq \underline{\mathrm{M}}_\ell$. This conclusion, the fact that $\bar{M}_0 \leq 2\bar{L} \leq \theta\bar{L}$, and a similar argument as the proof of Lemma 2.5 in [44] then imply that

$$\bar{M}_0 \leq \underline{\mathrm{M}}_\ell \leq \max\left\{\bar{M}_0, \theta\bar{L}\right\} \leq \theta\bar{L}. \tag{C.2}$$

Also observe that it follows from Proposition 3.2(a) and the definitions of θ, ζ_ℓ, and Q_ℓ in (3.10) that each cycle of R-FISTA performs at most

$$\mathcal{O}_1\left(\sqrt{\frac{\max\left\{\underline{\mathrm{M}}_\ell, \bar{L}\right\}}{\mu_{\ell-1}}}\log^{++}\left(\frac{D^2(\bar{L}^2 + \underline{\mathrm{M}}_\ell^2)}{\epsilon^2}\right) + \log_2^+(\bar{L}/\underline{\mathrm{M}}_\ell)\right) \tag{C.3}$$

ACG iterations/gradient evaluations. The result of part (b) then follows from the lower bound on μ_ℓ, the fact that $\mathcal{O}(\log(A)) = \mathcal{O}(\log(A^2))$ for a scalar $A > 0$, and bounds (C.2) and (C.3). □

We are now ready to prove Theorem 3.3.

Proof of Theorem 3.3. Proposition C.1(a) implies that R-FISTA performs at most $\lceil \log_2^{++}(2\mu_0/\bar{\mu}) \rceil$ cycles before finding a pair (y, v) that is an ϵ-optimal solution of (3.1). The result then follows from this observation and the fact that the total number of ACG iterations/gradient evaluations that R-FISTA performs can be upper bounded by the product of the number of cycles $\lceil \log_2^{++}(2\mu_0/\bar{\mu}) \rceil$ and the bound in (C.1). □

Index

References

[1] J. M. Aguirre, D. Cifuentes, V. Guigues, R. D. Monteiro, V. H. Nascimento, and A. Sujanani, *cuhallar: A gpu accelerated low-rank augmented lagrangian method for large-scale semidefinite programming*, arXiv preprint arXiv:2505.13719, (2025). 3

[2] J.-F. Aujol, C. Dossal, H. Labarrière, and A. Rondepierre, *Fista restart using an automatic estimation of the growth parameter*, Journal of Optimization Theory and Applications, 206 (2025), p. 51. 28

[3] N. S. Aybat and G. Iyengar, *An augmented lagrangian method for conic convex programming*, arXiv preprint arXiv:1302.6322, (2013). 4, 5

[4] A. Bazzi and M. Chafii, *On integrated sensing and communication waveforms with tunable papr*, IEEE Transactions on Wireless Communications, 22 (2023), pp. 7345–7360. 3

[5] A. Beck and M. Teboulle, *A fast iterative shrinkage-thresholding algorithm for linear inverse problems*, SIAM journal on imaging sciences, 2 (2009), pp. 183–202. 5

[6] M. X. Burns and J. Liang, *Improved analysis of restarted accelerated gradient and augmented lagrangian methods via inexact proximal point frameworks*, arXiv preprint arXiv:2602.17878, (2026). 4, 6, 7

[7] K. Deng, R. Wang, Z. Zhu, J. Zhang, and Z. Wen, *The augmented lagrangian methods: Overview and recent advances*, SIAM Review, 68 (2026), pp. 471–576. 3

[8] L. Ding, H. Lu, and J. Yang, *New understandings and computation on augmented lagrangian methods for low-rank semidefinite programming*, arXiv preprint arXiv:2505.15775, (2025). 3

[9] M. R. Hestenes, *Multiplier and gradient methods*, Journal of optimization theory and applications, 4 (1969), pp. 303–320. 5

[10] D. Hou, T. Tang, and K.-C. Toh, *A low-rank augmented lagrangian method for doubly nonnegative relaxations of mixed-binary quadratic programs*, Operations Research, 74 (2026), pp. 1626–1647. 3

[11] L. Jiang, K. Tang, and Z. Zhang, *Optimal parameter-free first-order methods for convex optimization with unknown growth and smoothness*, arXiv preprint arXiv:2607.11878, (2026). 5

[12] P. Kananian, A. Sujanani, and S. M. Zahedi, *Asymptotically fair and truthful allocation of public goods*, Journal of Artificial Intelligence Research, 85 (2026). 3

[13] D. Kim and J. A. Fessler, *Another look at the fast iterative shrinkage/thresholding algorithm (fista)*, SIAM Journal on Optimization, 28 (2018), pp. 223–250. 5

[14] ———, *Generalizing the optimized gradient method for smooth convex minimization*, SIAM Journal on Optimization, 28 (2018), pp. 1920–1950. 5

[15] G. Lan and R. D. Monteiro, *Iteration-complexity of first-order augmented lagrangian methods for convex programming*, Mathematical Programming, 155 (2016), pp. 511–547. 4, 6, 7

[16] G. Lan, Y. Ouyang, and Z. Zhang, *Optimal and parameter-free gradient minimization methods for convex and nonconvex optimization*, Mathematical Programming, (2026), pp. 1–40. 5, 7, 9, 10, 40, 41

[17] J. Lee, C. Park, and E. Ryu, *A geometric structure of acceleration and its role in making gradients small fast*, Advances in Neural Information Processing Systems, 34 (2021), pp. 11999–12012. 5

[18] B. Li, B. Shi, and Y.-x. Yuan, *Proximal subgradient norm minimization of ista and fista*, Applied and Computational Harmonic Analysis, (2025), p. 101848. 5

[19] F. Li and Z. Qu, *An inexact proximal augmented lagrangian framework with arbitrary linearly convergent inner solver for composite convex optimization*, Mathematical Programming Computation, 13 (2021), pp. 583–644. 4, 6, 7

[20] T. Li and G. Lan, *A simple uniformly optimal method without line search for convex optimization*, Mathematical Programming, (2025), pp. 1–38. 5

[21] Z. Li and Y. Xu, *Augmented lagrangian–based first-order methods for convex-constrained programs with weakly convex objective*, INFORMS Journal on Optimization, 3 (2021), pp. 373–397. 7

[22] J. Liang and R. D. Monteiro, *An average curvature accelerated composite gradient method for nonconvex smooth composite optimization problems*, SIAM Journal on Optimization, 31 (2021), pp. 217–243. 5

[23] ———, *Average curvature fista for nonconvex smooth composite optimization problems*, Computational Optimization and Applications, 86 (2023), pp. 275–302. 5

[24] Q. Lin and Y. Xu, *Reducing the complexity of two classes of optimization problems by inexact accelerated proximal gradient method*, SIAM journal on optimization, 33 (2023), pp. 1–35. 4, 6, 7

[25] Y.-F. Liu, X. Liu, and S. Ma, *On the nonergodic convergence rate of an inexact augmented lagrangian framework for composite convex programming*, Mathematics of Operations Research, 44 (2019), pp. 632–650. 4, 6, 7

[26] Z. Lu and S. Mei, *Accelerated first-order methods for convex optimization with locally lipschitz continuous gradient*, SIAM Journal on Optimization, 33 (2023), pp. 2275–2310. 4, 5, 6, 7, 32, 33, 34, 35, 36, 37, 38, 39

[27] Z. Lu and Z. Zhou, *Iteration-complexity of first-order augmented lagrangian methods for convex conic programming*, SIAM journal on optimization, 33 (2023), pp. 1159–1190. 3, 4, 6, 7

[28] R. D. Monteiro, A. Sujanani, and D. Cifuentes, *A low-rank augmented lagrangian method for large-scale semidefinite programming based on a hybrid convex-nonconvex approach.*, Mathematical Programming, (2026), pp. 1–57. 3

[29] R. D. Monteiro and B. F. Svaiter, *An accelerated hybrid proximal extragradient method for convex optimization and its implications to second-order methods*, SIAM Journal on Optimization, 23 (2013), pp. 1092–1125. 5

[30] I. Necoara, A. Patrascu, and F. Glineur, *Complexity of first-order inexact lagrangian and penalty methods for conic convex programming*, Optimization Methods and Software, 34 (2019), pp. 305–335. 3, 4, 6, 7

[31] V. Nedelcu, I. Necoara, and Q. Tran-Dinh, *Computational complexity of inexact gradient augmented lagrangian methods: application to constrained mpc*, SIAM Journal on Control and Optimization, 52 (2014), pp. 3109–3134. 4

[32] Y. Nesterov, *How to make the gradients small*, Optima. Mathematical Optimization Society Newsletter, (2012), pp. 10–11. 5

[33] Y. Nesterov, *Gradient methods for minimizing composite functions*, Mathematical programming, 140 (2013), pp. 125–161. 5

[34] ———, *Universal gradient methods for convex optimization problems*, Mathematical Programming, 152 (2015), pp. 381–404. 11

[35] Y. Nesterov, A. Gasnikov, S. Guminov, and P. Dvurechensky, *Primal–dual accelerated gradient methods with small-dimensional relaxation oracle*, Optimization Methods and Software, 36 (2021), pp. 773–810. 5

[36] A. Patrascu, I. Necoara, and Q. Tran-Dinh, *Adaptive inexact fast augmented lagrangian methods for constrained convex optimization*, Optimization Letters, 11 (2017), pp. 609–626. 4

[37] M. J. Powell, *A method for nonlinear constraints in minimization problems*, Optimization, (1969), pp. 283–298. 5

[38] Y. Qu, X. Cai, H. Liu, and D. Han, *Convergence rate of inexact augmented lagrangian method with practical relative error criterion for composite convex programming*, Computational Optimization and Applications, 91 (2025), pp. 1227–1261. 6

[39] R. Rockafellar, *Augmented lagrangians and applications of the proximal point algorithm in convex programming*, Mathematics of Operations Research, 1 (1976), pp. 97–116. 5

[40] K. Scheinberg, D. Goldfarb, and X. Bai, *Fast first-order methods for composite convex optimization with backtracking*, Foundations of computational mathematics, 14 (2014), pp. 389–417. 5

[41] B. Shi, S. S. Du, M. I. Jordan, and W. J. Su, *Understanding the acceleration phenomenon via high-resolution differential equations*, Mathematical Programming, 195 (2022), pp. 79–148. 5

[42] J. J. Suh and S. Ma, *An adaptive and parameter-free nesterov's accelerated gradient method for convex optimization*, arXiv preprint arXiv:2505.11670, (2025). 5

[43] A. Sujanani and R. Monteiro, *An adaptive superfast inexact proximal augmented Lagrangian method for smooth nonconvex composite optimization problems*, J. Scientific Computing, 97 (2023). 24, 44

[44] A. Sujanani and R. D. Monteiro, *Efficient parameter-free restarted accelerated gradient methods for convex and strongly convex optimization*, Journal of Optimization Theory and Applications, 206 (2025), p. 52. 5, 7, 28, 45, 46, 47

[45] T. Tang and K.-C. Toh, *A preconditioned augmented lagrangian method for solving semidefinite programming problems*, arXiv preprint arXiv:2605.17089, (2026). 3

[46] J. Wang and L. Hu, *Solving low-rank semidefinite programs via manifold optimization*, Journal of Scientific Computing, 104 (2025), p. 33. 3

[47] Y. Wu and B. Grimmer, *A parameter-free restart scheme with only a parallelizable* $\log\log(1/\epsilon)$ *overhead*, arXiv preprint arXiv:2605.30502, (2026). 5

[48] Y. Wu, Y. Ouyang, Z. Zhang, and Q. Luo, *Universal and parameter-free gradient sliding for composite optimization*, arXiv preprint arXiv:2603.23492, (2026). 5

[49] Y. Xu, *Iteration complexity of inexact augmented lagrangian methods for constrained convex programming*, Mathematical Programming, 185 (2021), pp. 199–244. 4, 5, 6, 7

[50] A. Yurtsever, O. Fercoq, and V. Cevher, *A conditional-gradient-based augmented lagrangian framework*, in International Conference on Machine Learning, PMLR, 2019, pp. 7272–7281. 4

[51] D. Zhou, S. Ma, and J. Yang, *Adabb: Adaptive barzilai-borwein method for convex optimization*, Mathematics of Operations Research, 51 (2026), pp. 715–745. 5