MATHEMATICS OF OPERATIONS RESEARCH
Vol. 00, No. 0, Xxxxx 0000, pp. 000–000
ISSN 0364-765X | EISSN 1526-5471 | 00 | 0000 | 0001

INFORMS
DOI 10.1287/xxxx.0000.0000

Iteration complexity of a proximal augmented Lagrangian method for solving nonconvex composite optimization problems with nonlinear convex constraints

Weiwei Kong
Computer Science and Mathematics Division, Oak Ridge National Laboratory, Oak Ridge, TN, 37830. wwkong92@gmail.com

Jefferson G. Melo
Instituto de Matemática e Estatística, Universidade Federal de Goiás, Campus II- Caixa Postal 131, CEP 74001-970, Goiânia-GO, Brazil. jefferson@ufg.br

Renato D.C. Monteiro
School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, GA, 30332-0205. monteiro@isye.gatech.edu

This paper proposes and analyzes a proximal augmented Lagrangian (NL-IAPIAL) method for solving smooth nonconvex composite optimization problems with nonlinear $\mathcal{K}$-convex constraints, i.e., the constraints are convex with respect to the order given by a closed convex cone $\mathcal{K}$. Each NL-IAPIAL iteration consists of inexactly solving a proximal augmented Lagrangian subproblem by an accelerated composite gradient (ACG) method followed by a Lagrange multiplier update. Under some mild assumptions, it is shown that NL-IAPIAL generates an approximate stationary solution of the constrained problem in $\mathcal{O}(\log(1/\rho)/\rho^3)$ inner iterations, where $\rho > 0$ is a given tolerance. Numerical experiments are also given to illustrate the computational efficiency of the proposed method.

Key words: inexact proximal augmented Lagrangian method, $\mathcal{K}$-convexity, nonlinear constrained smooth nonconvex composite programming, accelerated first-order methods, iteration complexity.
MSC2000 subject classification: 49M05, 49M37, 90C26, 90C30, 90C60, 65K05, 65K10, 68Q25, 65Y20.
OR/MS subject classification: Primary: programming: nonlinear: theory, algorithms; Secondary: programming: nonlinear: nondifferentiable; mathematics: convexity

1. Introduction

This paper presents a nonlinear inner-accelerated proximal inexact augmented Lagrangian (NL-IAPIAL) method for solving the cone convex constrained nonconvex composite optimization (CCC-NCO) problem

$$\phi^* := \inf_{z\in\Re^n} \{\phi(z) := f(z) + h(z) : g(z) \preceq_{\mathcal{K}} 0\}, \tag{1}$$

where $\mathcal{K}$ is a closed convex cone such that $\emptyset \neq \mathcal{K} \neq \Re^\ell$, $g : \Re^n \mapsto \Re^\ell$ is a differentiable $\mathcal{K}$-convex function with a Lipschitz continuous gradient, h is a proper closed convex function with compact domain, f is a nonconvex differentiable function on the domain of h with a Lipschitz continuous gradient, and the relation $g(z) \preceq_{\mathcal{K}} 0$ means $g(z) \in -\mathcal{K}$.

More specifically, the NL-IAPIAL method is based on the augmented Lagrangian (AL) (see [29] and [38, Section 11.K])

$$\mathcal{L}_\beta(z,p) := (f+h)(z) + \frac{1}{2\beta}\left[\mathrm{dist}^2(p+\beta g(z), -\mathcal{K}) - \|p\|^2\right] \quad \forall \beta > 0, \tag{2}$$

where $\mathrm{dist}(y,S)$ denotes the Euclidean distance between a point $y \in \Re^\ell$ and a set $S \subseteq \Re^\ell$. It performs the following proximal point-type update to generate its k-th iterate: given (z_{k-1}, p_{k-1}) and (λ_k, β_k), compute

$$z_k \approx \underset{u}{\mathrm{argmin}} \left\{ \lambda_k \mathcal{L}_{\beta_k}(u; p_{k-1}) + \frac{1}{2}\|u - z_{k-1}\|^2 \right\}, \tag{3}$$

$$p_k = \Pi_{\mathcal{K}^*}(p_{k-1} + \beta_k g(z_k)), \tag{4}$$

where $\mathcal{K}^*$ denotes the dual cone of $\mathcal{K}$, the function $\Pi_{\mathcal{K}^*}$ denotes the projection onto $\mathcal{K}^*$, and z_k is a suitable approximate solution of the composite problem underlying (3). Even though there are different approaches for obtaining z_k as in (3), NL-IAPIAL employs an accelerated composite gradient (ACG) algorithm to obtain it, and hence the "inner-accelerated" qualifier in its name. Moreover, at the end of the k-th iteration above, it performs a key test to decide whether β_k is left unchanged or doubled.

Under a Slater-like assumption[1] and a suitable choice of the inputs (λ, β), it is shown that for any $(\hat\rho, \hat\eta) \in \Re^2_{++}$, the NL-IAPIAL method obtains a near stationary solution, i.e., a quadruple $(\hat z, \hat p, \hat w, \hat q)$ satisfying

$$\hat w \in \nabla f(\hat z) + \partial h(\hat z) + \nabla g(\hat z)\hat p, \quad \langle g(\hat z) + \hat q, \hat p\rangle = 0, \quad g(\hat z) + \hat q \preceq_{\mathcal{K}} 0, \quad \hat p \succeq_{\mathcal{K}^*} 0 \tag{5}$$

$$\|\hat w\| \le \hat\rho, \quad \|\hat q\| \le \hat\eta, \tag{6}$$

in $\mathcal{O}((\hat\eta^{-1/2}\hat\rho^{-2} + \hat\rho^{-3})\log(\hat\rho^{-1} + \hat\eta^{-1}))$ ACG iterations. If (1) satisfies a certain regularity condition, then it is well-known that a necessary condition for a point $\hat z$ to be a local minimum of (1) is that there exists a multiplier $\hat p \in \mathcal{K}^*$ such that $(\hat z, \hat p, \hat q, \hat w) = (\hat z, \hat p, 0, 0)$ satisfies (5). Moreover, the aforementioned complexity bound is derived without assuming that the initial point $z_0 \in \mathrm{dom}\, h$ is feasible, i.e., it also satisfies $g(z_0) \preceq_{\mathcal{K}} 0$. A key fact derived in this work is that the sequence of Lagrange multipliers generated by NL-IAPIAL is bounded, and its proof strongly uses the fact that its constraint function g is $\mathcal{K}$-convex (although (1) is nonconvex due to the nonconvexity assumption on f).

Overview of AL methods. The discussion below separates the AL methods into two classes:

(i) *Proximal AL (PAL) methods* whose k-th iteration is: given a pair (z_{k-1}, p_{k-1}) and a penalty parameter β_k, choose a prox parameter λ_k such that the objective function of (3) is strongly convex, compute an approximate solution z_k of (3), set

$$p_k = (1-\theta)\Pi_{\mathcal{K}^*}(p_{k-1} + \chi_k \beta_k g(z_k)) \tag{7}$$

for some $\chi_k \in (0,1]$ and fixed $\theta \in [0,1)$, and choose the next penalty parameter β_{k+1} from $[\beta_k, \infty)$. A classical PAL method for the case where f is convex has been studied by Rockafellar [36] under the assumption that $\theta = 0$, $\chi_k = 1$, and $\lambda_k = \beta_k$ for every k. It is worth noting that when f is convex, his method, as well as the aforementioned PAL method, can be viewed as a primal-dual, variable stepsize, inexact proximal point method, i.e, one which inexactly solves

$$\partial_z \mathcal{L}_0(z;p) + \frac{1}{\lambda_k}(z - z_{k-1}) \ni 0, \quad -\partial_p \mathcal{L}_0(z;p) + \frac{1}{\chi_k \beta_k}(p - p_{k-1}) \ni 0, \tag{8}$$

[1] See Proposition 2.1 in view of assumption (A4) in Subsection 2.1.

for $(z,p)=(z_k,p_k)$ where $\mathcal{L}_0(z;p) := (f+h)(z) + \langle p, g(z)\rangle - \delta_{\mathcal{K}^*}(p)$, for every $(z,p)\in\Re^n\times\Re^\ell$ with the convention that $+\infty-\infty=+\infty$, and $\delta_{\mathcal{K}^*}(p)$ takes value 0 if $p\succeq_{\mathcal{K}^*}0$ and $+\infty$ otherwise. Note that system (8) is equivalent to

$$\nabla f(z)+\partial h(z)+\nabla g(z)p+\frac{1}{\lambda_k}(z-z_{k-1})\ni 0, \quad -g(z)+\partial\delta_{\mathcal{K}^*}(p)+\frac{1}{\chi_k\beta_k}(p-p_{k-1})\ni 0.$$

(ii) *Non-proximal AL (n-PAL) methods* whose k-th iteration is: given a pair (z_{k-1},p_{k-1}) and a penalty parameter β_k, compute an approximate stationary point z_k of $\mathcal{L}_{\beta_k}(\cdot;p_{k-1})$, set

$$p_k=\Pi_{\mathcal{K}^*}(p_{k-1}+\chi_k\beta_k g(z_k)) \tag{9}$$

for some $\chi_k\in(0,1]$, and choose the next penalty parameter β_{k+1} from $[\beta_k,\infty)$. Detailed discussion of dual-only methods can be found, for example, in [5] where the conditions $\beta_k>\beta_{k-1}>0$ for all $k\geq 1$ and $\beta_k\uparrow\infty$ are assumed, and in [9, 33] where $\beta_k=\beta_{k-1}$ is allowed at iterations for which the feasibility gap decreases sufficiently. It is worth noting that when f is convex, these methods can be viewed as a dual-only, variable stepsize, inexact proximal point method for the same operator above, i.e., one which inexactly solves

$$\partial_z\mathcal{L}_0(z;p)\ni 0, \quad -\partial_p\mathcal{L}_0(z;p)+\frac{1}{\chi_k\beta_k}(p-p_{k-1})\ni 0, \tag{10}$$

for $(z,p)=(z_k,p_k)$ and $\mathcal{L}_0(\cdot;\cdot)$ is as in (i).

Notice how both kinds of AL methods include a prox term in the p block, which leads to the multiplier update (9). However, while the first one adds a proximal term to the z-block (hence the qualifier PAL), the other ones do not (hence the qualifier n-PAL). For a more detailed comparison of the above classes, see the first paragraph in Section 5.

Related works. The literature of AL-based methods is quite vast, so we focus our attention on those dealing with iteration complexities. Since AL-based methods for the convex case have been extensively studied in the literature (see, for example, [1, 2, 22, 23, 28, 29, 31, 35, 41]), we focus on papers that deal with nonconvex problems with nontrivial composite functions Methods for the nonconvex problems where the composite h is the zero function have already been studied in [14, 40].

Papers [12, 19, 30] as well as this one propose and study the complexity of PAL methods for solving the CCC-NCO problem or its linearly constrained version in which $\mathcal{K}=\{0\}$. More specifically, both papers [12, 30] consider PAL methods applied to the linearly constrained CCC-NCO problem where $\theta\in(0,1]$ and $\chi_k=1$ for every k. However, as θ approaches zero, the prox stepsizes λ_k of both methods converge to zero which causes the following issues: 1) their derived complexity bounds diverge to infinity (see the second column in Table 2 below), which makes their analyses invalid for the case where $\theta=0$; and 2) deteriorating computational performance. Using a different approach, i.e., one that does not rely on a merit function, paper [19] establishes the iteration complexity of a PAL method, with $\theta=0$ and $\chi_k=1$ for every k, for solving the linearly constrained CCC-NCO problem under the condition that p_k is reset to zero whenever β_k is increased.

Papers [24, 39] propose and study the iteration complexity of n-PAL methods for solving nonlinearly constrained NCO problems. More specifically, [39] uses the AG method of [11] to obtain the approximate stationary point z_k of $\mathcal{L}_{\beta_k}(\cdot;p_{k-1})$. On the other hand, [24] obtains such z_k by applying an inner accelerated prox method as in [7, 17] whose generated subproblems are convex and similar to the ones generated by the PAL methods. It is worth mentioning that both of these papers make a strong assumption about how the feasibility of an iterate is related to its stationarity (see condition $\mathcal{F}$ in Table 1).

We now describe other papers that have motivated this work or are tangentially related to it. Papers [17, 18, 21, 26] establish the complexity of quadratic penalty-based methods for solving (1).

Paper [6] considers a primal-dual proximal point scheme and analyzes its complexity under strong conditions on the initial point. Papers [42, 43] present a primal-dual first-order algorithm for solving (1) when h is the indicator function of a box (in [43]) or more generally a polyhedron (in [42]). Paper [15] considers a penalty-ADMM method that solves an equivalent reformulation of (1). Paper [25] presents an inexact proximal point method applied to the function defined as $\phi(z)$ if z is feasible and $+\infty$ otherwise. It can be viewed as an extension to the nonconvex setting of the proximal point method (PPM) applied to (1) (see, for example, [36] for the analysis of inexact versions of PPMs for solving (1) in the convex setting).

Before closing this literature review, we list the assumptions of the above PAL and n-PAL methods in Table 1 and give a summary of these methods in Table 2, which compares some of the more recent methods in terms of iteration complexity, type of constraints, necessary conditions, and ranges of θ and χ_k.

$\mathcal{B}$	Either (i) the quantity $\sup_{x\in\operatorname{dom} h}\lvert\phi(x)\rvert$ is finite, (ii) $\operatorname{dom} h$ is bounded, and/or (iii) the feasible set is bounded.
$\mathcal{A}$	If the constraints have an affine component of the form $Ax=b$ then A has full row rank.
$\mathcal{F}$	There exists some $\nu>0$ such that $\nu\|g(x_k)\|\le \operatorname{dist}(0,\nabla g(x_k)g(x_k)+\beta_k^{-1}\partial h(x_k))$ for algorithmically generated sequences $\{x_k\}_{k\ge1}$ and $\{\beta_k\}_{k\ge1}$.
$\mathcal{N}$	The function h restricted to its domain is r-Lipschitz continuous.
$\mathcal{SP}$	If $g(x)\preceq_{\mathcal{K}} 0$ can be divided into $g_e(x)=0$ and $g_\iota(x)\preceq_{\mathcal{J}} 0$ for some closed convex cone $\mathcal{J}$, then there exists $\bar{x}\in\operatorname{int}(\operatorname{dom} h)$ such that $g_e(x)=0$ and $g_\iota(x)\prec_{\mathcal{J}} 0$.

Table 1. Abbreviations for common boundedness and regularity conditions. A discussion of the relationship between $\mathcal{SP}$ and $\mathcal{SP}^\circ$ is given in Subsection 2.1. It is known (see, for example, [19]) that $\mathcal{N}$ is equivalent to requiring that, for every $x\in\operatorname{dom} h$, there exists $r>0$ such that $\partial h(x)\subseteq \mathcal{N}_{\operatorname{dom} h}(x)+\mathcal{B}_r(0)$ where $\mathcal{B}_r(0)=\{x:\|x\|\le r\}$.

Name	**Complexity**	**Constraints**	θ	χ_k	**Key Conditions**	**AL group**
PProx-PDA[2] [12]	$\mathcal{O}(\theta^{-2}\varepsilon^{-4})$	Linear	$(0,1)$	1	$\mathcal{B}$, $\mathcal{A}$	PAL
θ-IPAAL[3] [30]	$\tilde{\mathcal{O}}(\theta^{-15/4}\varepsilon^{-2.5})$	Linear	$(0,1)$	1	$\mathcal{N},\mathcal{SP}$	PAL
IAIPAL [19]	$\tilde{\mathcal{O}}(\varepsilon^{-3})$	Linear	0	1	$\mathcal{B},\mathcal{N},\mathcal{SP}$	PAL
iALM (2019) [39]	$\tilde{\mathcal{O}}(\varepsilon^{-3})$	Nonlinear	-	$\mathcal{O}(\beta_k^{-1})$	$\mathcal{B},\mathcal{F}$	n-PAL
iALM (2020)[4] [24]	$\tilde{\mathcal{O}}(\varepsilon^{-3})$	Nonlinear	-	$\mathcal{O}(\beta_k^{-1})$	$\mathcal{B},\mathcal{F}$	n-PAL
NL-IAPIAL	$\tilde{\mathcal{O}}(\varepsilon^{-3})$	$\mathcal{K}$**-Convex**	0	1	$\mathcal{B},\mathcal{N},\mathcal{SP}$	PAL

Table 2. Comparison of relevant PAL and n-PAL methods with NL-IAPIAL where the first three methods assume that g is an affine function of the form $g(x)=Ax-b$. For simplicity, we let $\varepsilon=\min\{\hat{\rho},\hat{\eta}\}$, and let $\tilde{\mathcal{O}}(\cdot)$ be the same as $\mathcal{O}(\cdot)$ with all logarithmic dependencies on ε removed.

Contributions. We start by highlighting the differences and novelties of the NL-IAPIAL compared to the ones in [12, 19, 30]. In contrast to the PAL methods of [12, 30], whose iteration-complexities in

[2] This method generates prox subproblems of the form $\operatorname{argmin}_{x\in X}\{\lambda h(x)+c\|Ax-b\|^2/2+\|x-x_0\|^2/2\}$ and the analysis of [12] makes the strong assumption that they can be solved exactly for any x_0, c, and λ.

[3] It is also shown that conditions $\mathcal{N}$ and $\mathcal{SP}$ can be removed to yield an iteration complexity of $\tilde{\mathcal{O}}(\theta^{-4}\varepsilon^{-3})$.

[4] An $\tilde{\mathcal{O}}(\varepsilon^{-2.5})$ iteration complexity bound is established for the case where the constraints are linear. Moreover, the method considered in this table is Algorithm 3 of [24] where it is shown that the associated sequence of multipliers is bounded under assumption $\mathcal{F}$.

terms of θ only (see the second column in Table 2) are $\mathcal{O}(\theta^{-2})$ and $\mathcal{O}(\theta^{-15/4})$, respectively, this work presents a PAL method and its corresponding iteration-complexity, both of which do not depend on θ. Moreover, its analysis only assumes the existence of a Slater point and its multiplier update uses $\theta = 0$ and $\chi_k = 1$ for every k, as prescribed in the classical versions of both PAL and n-PAL methods. In contrast to [19] (see the end of the second paragraph of *Related Works*), our proposed PAL method has the following extra features: 1) it always updates p_k as in (4), regardless of whether β_k increases or not; and 2) it solves the more general nonlinear CCC-NCO problem.

Even though NL-IAPIAL is not an n-PAL method, it is still worth discussing some of its features relative to the n-PAL methods of [24, 39]. First, in contrast to [24, 39], this work does not assume the strong condition $\mathcal{F}$ of Table 1 on the iterates generated by their methods (see the fifth column of Table 2). Second, in contrast to the methods in [24, 39] whose choices of χ_k in (7) converge to zero as β_k tends to infinity[5], NL-IAPIAL chooses $\chi_k = 1$ for every k (see the sixth columns of Table 2).

Additional discussion of how NL-IAPIAL compares with other related first-order methods that are neither PAL nor n-PAL methods (i.e., [25, 42, 43]) is given in Section 5.

Organization of the Paper. Subsection 1.1 provides some basic definitions and notation. Section 2 contains three subsections. The first one describes the main problem of interest and the assumptions made on it. The second one motivates and states the NL-IAPIAL method, whereas the third one presents its main complexity results. Section 3 is divided into two subsections. The first one proves Proposition 2.6(b)–(c) which presents iteration-complexity bounds for NL-IAPIAL. The second one proves Proposition 2.5 which gives a bound on the multipliers sequence generated by NL-IAPIAL. Section 4 is devoted to numerical experiments that illustrate the numerical efficiency of NL-IAPIAL. Section 5 gives several concluding remarks. The Appendix section contains three subsections. Appendix A reviews an ACG variant, Appendix B describes some basic convex analysis results, and Appendix C is devoted to the proof of a basic result considered in the main part of the paper.

1.1. Basic Definitions and Notations

This subsection presents notation and basic definitions used in this paper.

Let $\Re_+$ and $\Re_{++}$ denote the set of nonnegative and positive real numbers, respectively, and let $\Re^2_{++} := \Re_{++} \times \Re_{++}$. We denote by R^n an n-dimensional inner product space with inner product and associated norm denoted by $\langle \cdot, \cdot \rangle$ and $\|\cdot\|$, respectively. For a given closed convex set $Z \subset \Re^n$, its boundary is denoted by ∂Z and the distance of a point $z \in \Re^n$ to Z is denoted by $\text{dist}(z, Z)$. The indicator function of Z, denoted by δ_Z, is defined by $\delta_Z(z) = 0$ if $z \in Z$, and $\delta_Z(z) = +\infty$ otherwise. For any $t > 0$, we let $\log_1^+(t) := \max\{\log t, 1\}$, and we define $\mathcal{O}_1(\cdot) = \mathcal{O}(1 + \cdot)$.

The domain of a function $h : \Re^n \to (-\infty, \infty]$ is the set $\text{dom}\, h := \{x \in \Re^n : h(x) < +\infty\}$. Moreover, h is said to be proper if $\text{dom}\, h \neq \emptyset$. The set of all lower semi-continuous proper convex functions defined in $\Re^n$ is denoted by $\overline{\text{Conv}}\ \Re^n$. The ε-subdifferential of a proper function $h : \Re^n \to (-\infty, \infty]$ is defined by

$$\partial_\varepsilon h(z) := \{u \in \Re^n : h(z') \geq h(z) + \langle u, z' - z\rangle - \varepsilon, \quad \forall z' \in \Re^n\} \tag{11}$$

for every $z \in \Re^n$. The classical subdifferential, denoted by $\partial h(\cdot)$, corresponds to $\partial_0 h(\cdot)$. Recall that, for a given $\varepsilon \geq 0$, the ε-normal cone of a closed convex set C at $z \in C$, denoted by $N_C^\varepsilon(z)$, is

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

The normal cone of a closed convex set C at $z \in C$ is denoted by $N_C(z) = N_C^0(z)$. If ψ is a real-valued function which is differentiable at $\bar{z} \in \Re^n$, then its affine approximation $\ell_\psi(\cdot, \bar{z})$ at $\bar{z}$ is given by

$$\ell_\psi(z; \bar{z}) := \psi(\bar{z}) + \langle \nabla \psi(\bar{z}), z - \bar{z}\rangle \quad \forall z \in \Re^n. \tag{12}$$

[5] Methods with this feature tend to become more like penalty-type methods as more iterations are performed.

For a closed convex cone $\mathcal{K}\subset\Re^l$, the dual cone $\mathcal{K}^*$ is defined as

$$\mathcal{K}^* := \{y\in\Re^l : \langle y, x\rangle \geq 0,\, x\in\mathcal{K}\}\,. \tag{13}$$

For given $u,v\in\Re^l$, the notation $u\preceq_{\mathcal{K}} v$ (or $v\succeq_{\mathcal{K}} u$) means that $v-u\in\mathcal{K}$. Moreover, the notation $u\prec_{\mathcal{K}} v$ means that $v-u\in\operatorname{int}\mathcal{K}$. A function $g:\Re^n\to\Re^\ell$ is said to be $\mathcal{K}$-convex if

$$g(tz'+[1-t]z)-tg(z')-[1-t]g(z)\preceq_{\mathcal{K}} 0 \quad \forall z,z'\in\Re^n,\, \forall t\in[0,1]. \tag{14}$$

Under the assumption that g is differentiable, it is well-known that g is $\mathcal{K}$-convex if and only if

$$\langle p, g'(z)(z'-z)\rangle \leq \langle p, g(z')-g(z)\rangle \quad \forall z,z'\in\Re^n,\, \forall p\in\mathcal{K}^*. \tag{15}$$

2. The NL-IAPIAL Method

This section consists of three subsections. The first one precisely describes the problem of interest and its assumptions. The second one motivates and states the NL-IAPIAL method. The third one presents the main complexity results for NL-IAPIAL.

2.1. Problem of Interest

This subsection presents the main problem of interest and discusses the assumptions underlying it.

Consider problem (1) where $\mathcal{K}$ is a closed convex cone such that $\emptyset\neq\mathcal{K}\neq\Re^l$, and functions f,g and h satisfy the following assumptions:

(A1) $h\in\overline{\text{Conv}}\,\Re^n$ and its domain $\mathcal{H}:=\operatorname{dom} h$ is a compact set; moreover, for some scalar $K_h\geq 0$, function h is K_h-Lipschitz continuous on $\mathcal{H}$, i.e., it satisfies

$$|h(z')-h(z)|\leq K_h\|z'-z\| \qquad \forall z,z'\in\mathcal{H};$$

(A2) f is a nonconvex function which is differentiable on $\mathcal{H}$, and there exist $0<m_f\leq L_f$ such that f is m_f-weakly convex on $\mathcal{H}$ (i.e., $f+m_f\|\cdot\|^2/2$ is convex on $\mathcal{H}$) and

$$\|\nabla f(z')-\nabla f(z)\|\leq L_f\|z'-z\| \qquad \forall z',z\in\mathcal{H}; \tag{16}$$

(A3) $g:\Re^n\mapsto\Re^\ell$ is $\mathcal{K}$-convex and differentiable, and there exists $L_g>0$ such that

$$\|\nabla g(z')-\nabla g(z)\|\leq L_g\|z'-z\| \qquad \forall z',z\in\Re^n;$$

(A4) there exist $\bar z\in\operatorname{int}\mathcal{H}$ and $\tau_g>0$ such that $g(\bar z)\preceq_{\mathcal{K}} 0$ and

$$\max\{\|\nabla g(z)p\|, |\langle p, g(\bar z)\rangle|\}\geq \tau_g\|p\| \qquad \forall z\in\mathcal{H},\;\; \forall p\succeq_{\mathcal{K}^*} 0. \tag{17}$$

We now make some comments about the above assumptions. First, any function h of the form $h=\tilde h+\delta_Z$ where $\tilde h$ is a finite everywhere Lipschitz continuous convex function and Z is a compact convex set clearly satisfies condition (A1). Second, it is easy to see that (A2) implies that

$$-\frac{m_f}{2}\|z'-z\|^2\leq f(z')-\ell_f(z';z) \quad \forall z',z\in\mathcal{H}, \tag{18}$$

where $\ell_f(\cdot;\cdot)$ is as in (12). Moreover, it is well-known that (16) implies that $|f(z')-\ell_f(z';z)|\leq L_f\|z'-z\|^2/2$ for every $z,z'\in\mathcal{H}$, and hence that (18) holds with $m_f=L_f$. However, we will show that better iteration-complexity bounds for our method can be derived when a scalar $m_f<L_f$ satisfying (18) is available. Third, since f is nonconvex on $\mathcal{H}$, (A2) implies the smallest m_f satisfying (18) is positive. Fourth, the assumption that $\mathcal{K}\neq\Re^l$ implies that $\mathcal{K}^*\neq\{0\}$. Finally, the cone $\mathcal{K}$ is not assumed to have a nonempty interior.

The result below, whose proof is given in Appendix D, shows that if $\mathcal{K}=\mathcal{J}\times\{0\}$ where $\mathcal{J}$ is a closed convex cone such that $\operatorname{int}\mathcal{J}\neq\emptyset$, then (A4) is equivalent to a Slater-like assumption with respect to g. Hence, (A4) is a mild assumption on (1).

Proposition 2.1. *Suppose $\mathcal{J} \subseteq \Re^s$ is a closed convex cone with nonempty interior, $g_\iota : \Re^n \mapsto \Re^s$ is a (possibly nonconvex) continuously differentiable function, and $g_e : \Re^n \mapsto \Re^t$ is an onto affine map. Moreover, suppose $\nabla g_\iota(\cdot)$ is L_{g_ι}-Lipschitz continuous on the set $\mathcal{H}$ defined in (A1), and let $g := (g_\iota, g_e)$ and $\mathcal{K} := \mathcal{J} \times \{0\}$. Then, the following statements are equivalent:*
(a) there exists $\tau_g > 0$ and $\bar z \in \operatorname{int}\mathcal{H}$ such that $g(\bar z) \preceq_{\mathcal{K}} 0$ and (17) holds;
(b) there exists $\tilde\tau_g > 0$ and $\bar z \in \operatorname{int}\mathcal{H}$ such that $g(\bar z) \preceq_{\mathcal{K}} 0$ and

$$\max\left\{\|\nabla g(\bar z)p\|, |\langle p, g(\bar z)\rangle|\right\} \ge \tilde\tau_g \|p\| \quad \forall p \succeq_{\mathcal{K}^*} 0; \tag{19}$$

(c) there exists $\bar z \in \operatorname{int}\mathcal{H}$ such that $g_\iota(\bar z) \prec_{\mathcal{J}} 0$ and $g_e(\bar z) = 0$;

Some comments about Proposition 2.1 are in order. First, if g_ι is $\mathcal{J}$-convex and g_e is affine, then g is $\mathcal{K}$-convex. Second, the Slater condition is in regard to a single point $\bar z \in \mathcal{H}$, as opposed to condition (17) which involves inequality (17) at all pairs $(z,p) \in \mathcal{H} \times \mathcal{K}^*$. Third, (A4) can be replaced by the Slater-like assumption of Proposition 2.1 when $\mathcal{K} = \mathcal{J} \times \{0\}$ since the former is equivalent to the latter in this case. Actually, a slightly more involved analysis can be done to show that the assumption that g_e is onto (which is part of the assumption of Proposition 2.1) can be removed at the expense of obtaining a weaker version of (A4), namely: inequality (17) holds for every pair $(z,p) \in \mathcal{H} \times (\mathcal{J}^* \times \operatorname{Im}\nabla g_e)$, instead of $(z,p) \in \mathcal{H} \times (\mathcal{J}^* \times \Re^t) = \mathcal{H} \times \mathcal{K}^*$. Finally, since the analysis of this paper can be easily adapted to this slightly weaker version of (A4), the Slater-like condition of Proposition 2.1 without g_e assumed to be onto (or equivalently, ∇g_e to have full column rank) can be used in place of (A4) in order to guarantee that all of the results derived in this paper for NL-IAPIAL hold.

Under assumptions (A1)–(A4), it can be shown that: (i) a necessary condition for a point z^* to be a local minimum of (1) is that there exists a multiplier $p^* \in \mathcal{K}^*$ satisfying

$$0 \in \nabla f(z^*) + \partial h(z^*) + \nabla g(z^*)p^*, \quad \langle g(z^*), p^*\rangle = 0, \quad g(z^*) \preceq_{\mathcal{K}} 0, \quad p^* \succeq_{\mathcal{K}^*} 0; \tag{20}$$

and (ii) the last three conditions in (20) are equivalent[6] to the inclusion $g(z^*) \in N_{\mathcal{K}^*}(p^*)$. The following definition describes the type of approximate solution of (1) that is sought after by the NL-IAPIAL method.

Definition 2.2. *Given a tolerance pair $(\hat\rho, \hat\eta) \in \Re_{++} \times \Re_{++}$, a quadruple $(\hat z, \hat p, \hat w, \hat q) \in \mathcal{H} \times \Re^l \times \Re^n \times \Re^l$ is said to be a $(\hat\rho, \hat\eta)$-approximate stationary quadruple of (1) if it satisfies (5) and (6).*

We now make some observations about Definition 2.2. Another notion of approximate stationarity for (1) is as follows: a pair $(\hat z, \hat p) \in \mathcal{H} \times \Re^l$ is a $(\hat\rho, \hat\eta)$-approximate stationary solution of (1) if it satisfies the inequalities

$$\operatorname{dist}(0, \nabla f(\hat z) + \partial h(\hat z) + \nabla g(\hat z)\hat p) \le \hat\rho, \qquad \operatorname{dist}(g(\hat z), N_{\mathcal{K}^*}(\hat p)) \le \hat\eta. \tag{21}$$

It turns out that $(\hat z, \hat p)$ is a $(\hat\rho, \hat\eta)$-*approximate stationary solution* in the above sense if and only if there exists a residual pair $(\hat w, \hat q) \in \Re^n \times \Re^l$ such that $(\hat z, \hat p, \hat w, \hat q)$ is a $(\hat\rho, \hat\eta)$-approximate stationary quadruple of (1). In this regard, the residual pair $(\hat w, \hat q)$ in Definition 2.2 can be viewed as a certificate that the pair $(\hat z, \hat p)$ in the same definition is a $(\hat\rho, \hat\eta)$-approximate stationary solution of (1). Finally, our analysis is entirely based on the notion of Definition 2.2 even though it could also have been carried out using the notion of a $(\hat\rho, \hat\eta)$-approximate stationary solution instead. The main reason for this choice is that the NL-IAPIAL method presented in Subsection 2.2 naturally generates residual pairs which always satisfy (5), and eventually (6) after a sufficient number of iterations. Moreover, as opposed to the residual pairs which "realize" the two distances in (21), the computation of these residual pairs do not require projections onto $\partial h(\hat z)$ or $N_{\mathcal{K}^*}(\hat p)$.

We end this subsection by stating a technical result which describes some properties about the smooth part of the Lagrangian in (2).

[6] See Lemma B.1(c).

Lemma 2.3. *Assume that conditions (A2) and (A3) hold, and define the function*

$$\widetilde{\mathcal{L}}_\beta(z,p) := f(z) + \frac{1}{2\beta}\left[\mathrm{dist}^2(p+\beta g(z), -\mathcal{K}) - \|p\|^2\right] \quad \forall (z,p,\beta) \in \Re^n \times \Re^\ell \times \Re_{++} \tag{22}$$

and the quantities

$$B_g^{(0)} := \sup_{z\in\mathcal{H}} \|g(z)\|, \quad B_g^{(1)} := \sup_{z\in\mathcal{H}} \|\nabla g(z)\|. \tag{23}$$

Then, for every $\beta > 0$ and $p \in \Re^\ell$, the following properties hold:

(a) $\widetilde{\mathcal{L}}_\beta(\cdot,p)$ is m_f-weakly convex on $\mathcal{H}$, where m_f is as in (A2);

(b) $\widetilde{\mathcal{L}}_\beta(\cdot,p)$ is a differentiable function whose gradient is given by

$$\nabla_z \widetilde{\mathcal{L}}_\beta(z,p) = \nabla f(z) + \nabla g(z)\Pi_{\mathcal{K}^*}(p+\beta g(z)) \quad \forall z \in \Re^n;$$

(c) $\nabla_z \widetilde{\mathcal{L}}_\beta(\cdot,p)$ is $\widetilde{\mathcal{M}}$-Lipschitz continuous where

$$\widetilde{\mathcal{M}} = \widetilde{\mathcal{M}}(\beta,p) := L_f + L_g\|p\| + \beta M_g, \quad M_g := B_g^{(0)} L_g + [B_g^{(1)}]^2, \tag{24}$$

and the quantities L_f and L_g are as in (A2) and (A3), respectively.

Proof. The statements of the lemma with $f \equiv 0$ (and hence $m_f = L_f = 0$) immediately follow from [29, Proposition 5]. Hence, the general case of the lemma easily follows from assumption (A2) and the definition of $\widetilde{\mathcal{L}}_\beta$ in (22). □

2.2. The NL-IAPIAL Method This subsection motivates and states the NL-IAPIAL method.

Before presenting the method, we give a short but precise outline of its key steps, as well as a description of how its iterates are generated. Recall from the introduction that the NL-IAPIAL method, whose goal is to find a $(\hat\rho,\hat\eta)$-approximate stationary quadruple as in (5) and (6), is an iterative method which, at its k-th step, computes its next iterate (z_k,p_k) according to (3) and (4).

We now describe the conditions which are required on the approximate solution z_k of (3). For a given scalar $\sigma \in (0, 1/\sqrt{2}]$, NL-IAPIAL requires that z_k, together with a residual pair $(v_k,\varepsilon_k) \in \Re^n \times \Re_{++}$, satisfy

$$v_k \in \partial_{\varepsilon_k}\left(\lambda \mathcal{L}_{\beta_k}(\cdot,p_{k-1}) + \frac{1}{2}\|\cdot - z_{k-1}\|^2\right)(z_k), \quad \|v_k\|^2 + 2\varepsilon_k \le \sigma_k^2 \|v_k + z_{k-1} - z_k\|^2. \tag{25}$$

where

$$\sigma_k := \frac{\sigma}{\sqrt{\widetilde{\mathcal{M}}_k}}, \qquad \widetilde{\mathcal{M}}_k := \lambda\widetilde{\mathcal{M}}(\beta_k,p_{k-1}) + 1, \tag{26}$$

and $\widetilde{\mathcal{M}}(\cdot,\cdot)$ is as in (24). Note that if $\sigma = 0$ then the inequality in (25) implies that $(v_k,\varepsilon_k) = (0,0)$, and hence that z_k is a global solution of (3) in view of the inclusion in (25) and the definition of ε-subdifferential given in (11). By relaxing σ to be positive, we are then allowing z_k to be an inexact (global) solution of (3).

The following result now describes a way of computing the approximate triple (z_k,v_k,ε_k) as in the above paragraph. Its proof strongly relies on the fact that z_{k-1} is chosen to be the initial point for the ACG variant (see the fifth identity in (27)) and Proposition A.1 of Appendix A.

Lemma 2.4. *Let $\lambda = 1/(2m_f)$ where m_f is as in (A2), and define*

$$\begin{gathered}\psi_s = \lambda\widetilde{\mathcal{L}}_{\beta_k}(\cdot,p_{k-1}) + \frac{1}{2}\|\cdot - z_{k-1}\|^2, \quad \psi_n = \lambda h, \\ \widetilde{M} = \widetilde{\mathcal{M}}_k, \quad \widetilde{\mu} = \frac{1}{2}, \quad x_0 = z_{k-1}, \quad \tilde\sigma = \sigma_k,\end{gathered} \tag{27}$$

where $\widetilde{\mathcal{M}}_k$ *is as in* (26)*. Then, the ACG algorithm of Appendix* A*, with inputs given by* (27)*, computes a triple* $(z_k, v_k, \varepsilon_k) := (y, u, \eta)$ *satisfying* (25) *in a number of ACG iterations bounded by*

$$\left\lceil 5\sqrt{\widetilde{\mathcal{M}}_k}\log_1^+\left(\frac{4\widetilde{\mathcal{M}}_k}{\sigma}\right)\right\rceil. \tag{28}$$

Proof. We first show that the inputs in (27) satisfy conditions (B1)–(B2) in Appendix A. Indeed, using assumption (A1) and Lemma 2.3(a), it is easy to see that both $\psi_s + (\lambda m_f - 1)\|\cdot\|^2/2$ and ψ_n are convex. Since $\lambda = 1/(2m_f)$, it then follows that ψ_s is a 1/2-strongly convex and hence that $\tilde{\mu}$ satisfies the first inequality in (69). Now, in view of Lemma 2.3(c) and the definition of ψ_s in (27), it follows that $\widetilde{M}$ satisfies the second inequality in (69). Hence, we conclude that the inputs in (27) satisfy the conditions (B1)–(B2) in Appendix A.

We now derive the desired complexity bound. It follows from Proposition A.1 and the above result that the ACG algorithm of Appendix A with inputs given by (27) generates a triple $(z_k, v_k, \varepsilon_k) := (y, u, \eta)$ satisfying (25) in at most

$$\left\lceil 1 + \left(\frac{1}{2} + \sqrt{2\widetilde{\mathcal{M}}_k - 1}\right)\log_1^+\widetilde{\mathcal{A}}\right\rceil \tag{29}$$

iterations, where $\widetilde{\mathcal{A}} = 4(1+\widetilde{\sigma})^2(\widetilde{\mathcal{M}}_k - 1/2)\widetilde{\sigma}^{-2}$. Now, note that the definitions of σ_k and $\widetilde{\sigma}$ in (26) and (27), respectively, yield $\widetilde{\mathcal{A}} \le 16(\widetilde{\mathcal{M}}_k)^2\sigma^{-2}$. Hence, (28) follows from (29), the latter inequality, and the fact that $\log_1^+(\cdot) \ge 1$ and $\widetilde{\mathcal{M}}_k \ge 1$. □

It is worth mentioning that the main effort of an ACG iteration consists of: (i) the computation of $\nabla\psi_s(\tilde{x}_j)$ where $\tilde{x}_j$ is one of the iterates obtained in the j-th iteration of ACG (see (71)); and, (ii) the solution of the prox subproblem in (71). Its description given in Appendix A assumes that both (i) and (ii) can be carried out exactly with the aid of given oracles. Moreover, for the case where the functions ψ_s and ψ_n are chosen as in (27), it follows from Lemma 2.3(b) that

$$\nabla\psi_s(z) = \lambda\left[\nabla f(z) + \nabla g(z)\Pi_{\mathcal{K}^*}(p_{k-1} + \beta_k g(z))\right] + z - z_{k-1}.$$

Finally, since we make the blanket assumption that an oracle for exactly evaluating $\Pi_{\mathcal{K}^*}(\cdot)$ at any given point is available, it follows that $\nabla\psi_s(x)$ can be obtained exactly by means of the above formula.

We are now ready to provide a complete description of the NL-IAPIAL method.

NL-IAPIAL Method

Input: a function triple (f, g, h) and a quadruple of parameters (K_h, m_f, L_f, L_g) satisfying assumptions (A1)–(A4), a scalar $\sigma \in (0, 1/\sqrt{2}]$, a penalty parameter $\beta_1 > 0$, an initial pair $(z_0, p_0) \in \mathcal{H} \times \Re^l$, and a tolerance pair $(\hat{\rho}, \hat{\eta}) \in \Re^2_{++}$;
Output: a triple $(\hat{z}, \hat{p}, \hat{w}, \hat{q})$ satisfying (5)-(6);

0. set $\hat{k} = 0$, $k = 1$ and

$$\lambda = \frac{1}{2m_f}, \quad \beta = \beta_1, \quad C_\sigma = \frac{2(1+2\sigma)^2}{1-\sigma^2}; \tag{30}$$

1. use the ACG described in Appendix A with inputs $(\widetilde{M}, \widetilde{\mu}, \psi_s, \psi_n)$, x_0 and $\tilde{\sigma}$ given by (27) to obtain a triple $(z_k, v_k, \varepsilon_k) := (y, u, \eta)$ satisfying (25), and compute

$$p_k := \Pi_{\mathcal{K}^*}(p_{k-1} + \beta_k g(z_k)), \qquad r_k := v_k + z_{k-1} - z_k; \tag{31}$$

2. compute the point $\hat{z}_k$ as

$$\hat{z}_k := \underset{u}{\operatorname{argmin}}\left\{\lambda\left[\left\langle\nabla_z\widetilde{\mathcal{L}}_{\beta_k}(z_k, p_{k-1}), u - z_k\right\rangle + h(u)\right] - \langle r_k, u - z_k\rangle + \frac{\widetilde{\mathcal{M}}_k}{2}\|u - z_k\|^2\right\}, \tag{32}$$

and the triple $(\hat{p}_k, \hat{w}_k, \hat{q}_k)$ as

$$\begin{aligned} \hat{p}_k &:= \Pi_{\mathcal{K}^*}\left(p_{k-1} + \beta_k g(\hat{z}_k)\right), \\ \hat{w}_k &:= w_k + \nabla_z \widetilde{\mathcal{L}}_{\beta_k}(\hat{z}_k, p_{k-1}) - \nabla_z \widetilde{\mathcal{L}}_{\beta_k}(z_k, p_{k-1}), \\ \hat{q}_k &:= \frac{1}{\beta_k}(p_{k-1} - \hat{p}_k), \end{aligned} \tag{33}$$

where $\widetilde{\mathcal{M}}_k$ and $\widetilde{\mathcal{L}}_{\beta_k}$ are as in (22) and (26), respectively, and

$$w_k := \frac{1}{\lambda}\left[r_k + \widetilde{\mathcal{M}}_k(z_k - \hat{z}_k)\right]; \tag{34}$$

if $(\hat{w}, \hat{q}) := (\hat{w}_k, \hat{q}_k)$ satisfies (6) then stop and output $(\hat{z}, \hat{p}, \hat{w}, \hat{q}) = (\hat{z}_k, \hat{p}_k, \hat{w}_k, \hat{q}_k)$;

3. if $k > \hat{k} + 1$ and

$$\Delta_k := \frac{1}{k - \hat{k} - 1}\left[\mathcal{L}_{\beta_k}(z_{\hat{k}+1}, p_{\hat{k}}) - \mathcal{L}_{\beta_k}(z_k, p_k) - \frac{\|p_k\|^2}{2\beta_k}\right] \le \frac{\lambda\hat{\rho}^2}{2C_\sigma}, \tag{35}$$

then set $\beta_{k+1} = 2\beta_k$ and $\hat{k} = k$; otherwise, set $\beta_{k+1} = \beta_k$;

4. update $k \leftarrow k + 1$, and go to step 1.

Some remarks about NL-IAPIAL are in order. First, it performs two kinds of iterations, namely, the ones indexed by k and the ones performed by the ACG algorithm every time it is called in step 1. We refer to the former as "outer" iterations and the latter as "inner" (or ACG) iterations. Second, its input z_0 can be any element in the domain of h and does not necessarily need to be a point satisfying the constraint $g(z_0) \preceq_{\mathcal{K}} 0$. Third, the ACG described in Appendix A is invoked in step 1 to compute a triple $(z_k, v_k, \varepsilon_k)$ satisfying (25), which can be seen as an approximate stationary solution for the prox-subproblem (3). Fourth, it will be shown in Lemma 3.4 that the refined quadruple $(\hat{z}, \hat{p}, \hat{w}, \hat{q}) := (\hat{z}_k, \hat{p}_k, \hat{w}_k, \hat{q}_k)$ computed in step 2 satisfies all the relations in (5) at any outer iteration. As a consequence, the NL-IAPIAL output $(\hat{z}, \hat{p}, \hat{w}, \hat{q})$ is a $(\hat{\rho}, \hat{\eta})$-approximate stationary quadruple of (1) in the sense of Definition 2.2. Finally, it follows from Lemma 2.3(b), and the first identities in (31) and (33), that the gradients of the function $\widetilde{\mathcal{L}}_{\beta_k}(\cdot, p_{k-1})$ which appear in (33) can be computed as $\nabla_z \widetilde{\mathcal{L}}_{\beta_k}(z_k, p_{k-1}) = \nabla f(z_k) + \nabla g(z_k) p_k$ and $\nabla_z \widetilde{\mathcal{L}}_{\beta_k}(\hat{z}_k, p_{k-1}) = \nabla f(\hat{z}_k) + \nabla g(\hat{z}_k)\hat{p}_k$.

In the remaining part of this subsection, we give some intuition about step 3 of NL-IAPIAL. Define the l-th cycle $\mathcal{C}_l$ as the l-th set of consecutive indices k for which β_k remains constant, i.e.,

$$\mathcal{C}_l := \{k : \beta_k = \tilde{\beta}_l := 2^{l-1}\beta_1\}. \tag{36}$$

For every $l \ge 1$, we let k_l denote the largest index in $\mathcal{C}_l$. Hence,

$$\mathcal{C}_l = \{k_{l-1} + 1, \ldots, k_l\} \quad \forall l \ge 1$$

where $k_0 := 0$. Clearly, the different values of $\hat{k}$ that arise in step 3 are exactly the indices in the index set $\{k_l : l \ge 0\}$. Moreover, in view of the test performed in step 3, we have that $k_l - k_{l-1} \ge 2$ for every $l \ge 1$, or equivalently, every cycle contains at least two indices. While generating the indices in the l-th cycle, if an index $k \ge k_{l-1} + 2$ satisfying (35) is found, k becomes the last index k_l in the l-th cycle and the $(l+1)$-th cycle is started at iteration $k_l + 1$ with the penalty parameter set to $\tilde{\beta}_{l+1} = 2\tilde{\beta}_l$, where $\tilde{\beta}_l$ is as in (36).

Finally, the role played by criterion (35) is as follows. It is shown in Lemma 3.5 that for every $k \in \mathcal{C}_\ell$, there exists $j \in \mathcal{C}_\ell$, $j \le k$ such that

$$\|\hat{w}_j\|^2 = \frac{C_\sigma \Delta_k}{\lambda} + \mathcal{O}\left(\frac{1}{\tilde{\beta}_l}\right), \quad \|\hat{q}_j\| = \mathcal{O}\left(\frac{1}{\tilde{\beta}_l}\right). \tag{37}$$

Hence, if criterion (35) holds, then (37) implies that $\|\hat{w}_j\|^2 = \hat{\rho}^2/2 + \mathcal{O}(1/\tilde{\beta}_l)$ and $\|\hat{q}_j\| = \mathcal{O}(1/\tilde{\beta}_l)$. On the other hand, since $\tilde{\beta}_l$ is doubled from one cycle to another, these residual estimates imply that the stopping criterion in step 2 will eventually be satisfied.

2.3. Complexity results for NL-IAPIAL This subsection contains the main complexity results for NL-IAPIAL.

We start by considering a proposition, whose proof is presented in Section 3.2, that shows that the sequence of Lagrange multipliers $\{p_k\}$ is bounded. Before presenting the result, we first introduce the following quantities:

$$\bar{d} := \operatorname{dist}(\bar{z}, \partial\mathcal{H}), \qquad D_h := \sup_{z',z\in\mathcal{H}} \|z'-z\|, \qquad \theta_h := \frac{D_h}{\min\{1,\bar{d}\}} \qquad B_f^{(1)} := \sup_{z\in\mathcal{H}} \|\nabla f(z)\|, \tag{38}$$

$$\kappa_0 := 2\left[K_h + B_f^{(1)}\right] + \left[\frac{\sigma^2}{(1-\sigma)^2} + 4\left(\frac{1+\sigma}{1-\sigma}\right)\right] m_f D_h, \tag{39}$$

where $\sigma \in (0, 1/\sqrt{2}]$ is an input of NL-IAPIAL, K_h and m_f are as in (A1) and (A2), respectively, and $\partial\mathcal{H}$ denotes the boundary of $\mathcal{H}$. Observe that $\bar{d} > 0$ in view of the fact that, by (A4), $\bar{z} \in \operatorname{int}\mathcal{H}$. Moreover, using the fact that $\mathcal{H}$ is compact and ∇f is continuous on $\mathcal{H}$ due to (A1) and (A2), respectively, it follows that D_h and $B_f^{(1)}$ are finite. These two observations then imply that θ_h and κ_0 are also finite.

Proposition 2.5. *The sequence $\{p_k\}$ generated by NL-IAPIAL satisfies*

$$\|p_k\| \le \kappa_p := \max\left\{\|p_0\|, \frac{\theta_h \kappa_0}{\tau_g}\right\}, \qquad \forall k \ge 0, \tag{40}$$

where θ_h, κ_0, and τ_g, are as in (38), (39), and (A4), respectively.

The following quantities will be used in the subsequent results:

$$\Delta\phi^* := \phi^* - \phi_*, \quad \phi_* := \inf_{z\in\Re^n} \phi(z), \tag{41}$$

$$\kappa_1 := \left(\frac{3L_f + L_g\kappa_p}{2m_f}\right)^{1/2}, \quad \kappa_2 := 6\kappa_p\sqrt{M_g C_\sigma}, \quad \kappa_3 := \left[\left(\tau_g + 4\sqrt{M_g}\right)\frac{\kappa_p\sqrt{M_g}}{2m_f}\right]^{1/2}, \tag{42}$$

$$\bar{\beta} = \bar{\beta}(\hat{\rho}, \hat{\eta}) := \frac{m_f}{M_g}\left(\frac{\kappa_2^2}{\hat{\rho}^2} + \frac{\kappa_3^2}{\hat{\eta}}\right), \tag{43}$$

where the quantities (m_f, L_f), L_g, ϕ^*, M_g, C_σ, D_h, and κ_p are as in (A2), (A3), (1), (24), (30), (38), and (40), respectively.

The following result, whose proof is given in Subsection 3.1, establishes bounds on the number of ACG and outer iterations performed during an NL-IAPIAL cycle, and shows that NL-IAPIAL outputs a $(\hat{\rho}, \hat{\eta})$-approximate stationary quadruple of (1) within a logarithmic number of cycles.

Proposition 2.6. *The following statements about NL-IAPIAL hold:*

(a) every outer iteration within the l-th cycle performs at most

$$\left\lceil 5\left(\kappa_1 + \sqrt{\frac{\tilde{\beta}_l M_g}{2m_f}}\right)\log_1^+\left(\frac{4\kappa_1^2}{\sigma} + \frac{2\tilde{\beta}_l M_g}{\sigma m_f}\right)\right\rceil$$

ACG iterations, where m_f, M_g, $\tilde{\beta}_l$, and κ_1 are as in (A2), (24), (36), and (42), respectively;

(b) every cycle performs at most

$$\left\lceil \frac{4m_f C_\sigma\left(\Delta\phi^* + 2m_f D_h\right)}{\hat{\rho}^2}\right\rceil$$

outer iterations, where C_σ, D_h, and $\Delta\phi^*$ are as in (30), (38), and (41), respectively;

(c) the last cycle $\bar{l}$ outputs a $(\hat\rho,\hat\eta)$-approximate stationary quadruple of (1) and satisfies

$$\bar{l} \le \log_1^+\left(\frac{4\bar\beta}{\beta_1}\right), \quad \tilde\beta_{\bar{l}} \le \max\{\beta_1, 2\bar\beta\}$$

where $\bar\beta$ is as in (43).

Notice that if $\beta_1 > 4\bar\beta$, then Proposition 2.6(c) implies the number of ACG iterations of NL-IAPIAL is bounded above by the product of the quantities in Proposition 2.6(a)–(b). The next result bounds the number of ACG iterations of NL-IAPIAL when $\beta_1 \le 4\bar\beta$.

Theorem 2.7. *Suppose $\beta_1 \le 4\bar\beta$. Then NL-IAPIAL outputs a $(\hat\rho,\hat\eta)$-approximate stationary quadruple of (1) in*

$$O\left(\left[1+\frac{m_f C_\sigma\left(\Delta\phi^* + m_f D_h\right)}{\hat\rho^2}\right]\left[\kappa_1 + \frac{\kappa_2}{\hat\rho} + \frac{\kappa_3}{\sqrt{\hat\eta}}\right](\log_1^+)^2\left[\frac{\bar\beta}{\beta_1} + \frac{\kappa_1^2}{\sigma} + \frac{\bar\beta M_g}{\sigma m_f}\right]\right) \tag{44}$$

ACG iterations, where m_f, C_σ, D_h, $\Delta\phi^*$, $(\kappa_1,\kappa_2,\kappa_3)$, and $\bar\beta$ are as in (A2), (30), (38), (41), (42), and (43), respectively.

Proof. First recall that in the l-th cycle of NL-IAPIAL, we have $\beta_k = \tilde\beta_l = 2^{l-1}\beta_1$, for every $l \ge 1$ (see (36)). Also, Proposition 2.6(c) implies that NL-IAPIAL outputs a $(\hat\rho,\hat\eta)$-approximate stationary quadruple of (1) in at most $\bar{l} := \lfloor \log_1^+(4\bar\beta/\beta_1)\rfloor$ cycles. Hence, since $\beta_1 \le 4\bar\beta$, we have

$$\tilde\beta_l = 2^{l-1}\beta_1 \le 4\bar\beta, \qquad \forall l = 1,\ldots,\bar{l}.$$

It now follows from the above inequality and the definition of $\bar\beta$ in (43) that the number of ACG iterations performed by NL-IAPIAL at every outer iteration (see Proposition 2.6 (a)) is

$$O\left(\left[\kappa_1 + \frac{\kappa_2}{\hat\rho} + \frac{\kappa_3}{\sqrt{\hat\eta}}\right]\log_1^+\left[\frac{\kappa_1^2}{\sigma} + \frac{\bar\beta M_g}{\sigma m_f}\right]\right).$$

The conclusion now follows from the above fact and Proposition 2.6 (b)–(c). □

It is worth mentioning that the iteration complexity bound in Theorem 2.7, in terms of the tolerance pair $(\hat\rho,\hat\eta)$, is

$$\mathcal{O}_1\left(\left[\frac{1}{\sqrt{\hat\eta}\cdot\hat\rho^2} + \frac{1}{\hat\rho^3}\right](\log_1^+)^2\left(\frac{1}{\hat\eta} + \frac{1}{\hat\rho^2}\right)\right),$$

as previously claimed in Section 1.

3. Proofs of Proposition 2.5 and Proposition 2.6

This section contains two subsections, the first of which proves Proposition 2.6 and the second one proves Proposition 2.5. It is worth noting that the proof of Proposition 2.6 uses Proposition 2.5, but the proof of Proposition 2.5 is self-contained. Moreover, we opted to postpone the proof of Proposition 2.5 due to its technicalities.

3.1. Proof of Proposition 2.6

The first result below presents some relations about the iterates generated by NL-IAPIAL.

Lemma 3.1. *Let $\{(z_k,p_k,\beta_k)\}$ be generated by NL-IAPIAL and define, for every $k \ge 1$,*

$$s_k := \Pi_{-\mathcal{K}}(p_{k-1} + \beta_k g(z_k)). \tag{45}$$

Then, the following relations hold for every $k \ge 1$:

$$p_{k-1} + \beta_k g(z_k) = p_k + s_k, \quad \langle p_k, s_k\rangle = 0, \quad (p_k,s_k) \in \mathcal{K}^* \times (-\mathcal{K}), \tag{46}$$

$$\mathcal{L}_{\beta_k}(z_k,p_{k-1}) = \phi(z_k) + \frac{1}{2\beta_k}\left(\|p_k\|^2 - \|p_{k-1}\|^2\right). \tag{47}$$

Proof. The relations in (46) follow from the definitions of p_k and s_k in (31) and (45), respectively, and Theorem III.3.2.5 of [13]. Now, in view of the definitions of $\mathcal{L}_\beta$ in (2) and s_k in (45), respectively, we have

$$\mathcal{L}_{\beta_k}(z_k,p_{k-1}) = \phi(z_k) + \frac{1}{2\beta_k}\left[\|p_{k-1}+\beta_k g(z_k)-s_k\|^2 - \|p_{k-1}\|^2\right]$$

which, in view of the first identity in (46), immediately implies (47). □

The next technical result characterizes the change in the augmented Lagrangian between consecutive iterations of the NL-IAPIAL method.

Lemma 3.2. *The sequence $\{(z_k,p_k)\}$ generated by NL-IAPIAL satisfies, for every $k\ge 1$, the relations*

$$\mathcal{L}_{\beta_k}(z_k,p_k) \le \mathcal{L}_{\beta_k}(z_k,p_{k-1}) + \frac{1}{\beta_k}\|p_k-p_{k-1}\|^2, \tag{48}$$

$$\mathcal{L}_{\beta_k}(z_k,p_k) \le \mathcal{L}_{\beta_k}(z_{k-1},p_{k-1}) - \left(\frac{1-\sigma^2}{2\lambda}\right)\|r_k\|^2 + \frac{1}{\beta_k}\|p_k-p_{k-1}\|^2, \tag{49}$$

where (σ,λ) is given by the input of NL-IAPIAL and $\{r_k\}$ is as in (31).

Proof. Let s_k be as in (45). Using (47), the definition of $\mathcal{L}_\beta$ in (2), the fact that $s_k\in-\mathcal{K}$ and $p_{k-1}+\beta_k g(z_k) = p_k + s_k$ in view of (46), we have that

$$\begin{aligned}
\mathcal{L}_{\beta_k}(z_k,p_k) - \mathcal{L}_{\beta_k}(z_k,p_{k-1}) &= \mathcal{L}_{\beta_k}(z_k,p_k) - \phi(z_k) - \frac{1}{2\beta_k}\left(\|p_k\|^2 - \|p_{k-1}\|^2\right)\\
&= \frac{1}{2\beta_k}\left(\operatorname{dist}^2(p_k+\beta_k g(z_k),-\mathcal{K}) - \|p_k\|^2\right) - \frac{1}{2\beta_k}\left(\|p_k\|^2-\|p_{k-1}\|^2\right)\\
&\le \frac{1}{2\beta_k}\left(\|p_k+\beta_k g(z_k)-s_k\|^2 - \|p_k\|^2\right) - \frac{1}{2\beta_k}\left(\|p_k\|^2-\|p_{k-1}\|^2\right)\\
&= \frac{1}{2\beta_k}\left(\|2p_k-p_{k-1}\|^2 - 2\|p_k\|^2 + \|p_{k-1}\|^2\right),
\end{aligned}$$

which immediately implies (48). Now, in view of the definition of the ε-subdifferential given in (11) and the fact that (z_k,v_k,ε_k) satisfies both the inclusion and the inequality in (25), we conclude that

$$\begin{aligned}
&\lambda\mathcal{L}_{\beta_k}(z_k,p_{k-1}) - \lambda\mathcal{L}_{\beta_k}(z_{k-1},p_{k-1}) \le -\frac{1}{2}\|z_k-z_{k-1}\|^2 + \langle v_k, z_k-z_{k-1}\rangle + \varepsilon_k\\
&= -\frac{1}{2}\|v_k+z_k-z_{k-1}\|^2 + \frac{1}{2}\|v_k\|^2 + \varepsilon_k \le -\left(\frac{1-\sigma_k^2}{2}\right)\|r_k\|^2 \le -\left(\frac{1-\sigma^2}{2}\right)\|r_k\|^2,
\end{aligned} \tag{50}$$

where the last inequality follows from the fact that $\sigma_k\le\sigma$ in view of (26). Inequality (49) now follows by combining (48) with (50). □

Recall that the l-th cycle $\mathcal{C}_l$ of NL-IAPIAL is defined in (36). The next results present some properties of the iterates generated during an NL-IAPIAL cycle. The first one shows that the sequence $\{\|r_k\|\}_{k\in\mathcal{C}_l}$ is bounded and can be controlled by $\{\Delta_k\}_{k\in\mathcal{C}_l}$ plus a term which is of $\mathcal{O}(1/\tilde{\beta}_l)$.

Lemma 3.3. *Consider the sequences $\{(z_k,v_k,\varepsilon_k)\}$ and $\{\Delta_k\}$ generated by NL-IAPIAL and the sequence $\{r_k\}$ as in (31). Then, the following statements hold:*

(a) for every $k\ge 1$, we have

$$\|r_k\| \le \frac{D_h}{1-\sigma}; \tag{51}$$

(b) $k\in\mathcal{C}_l$ and $k\ge k_{l-1}+2$, there exists an index $j\in\{k_{l-1}+2,\ldots,k\}$ such that

$$\|r_j\|^2 \le \frac{2\lambda}{1-\sigma^2}\left(\Delta_k + \frac{9\kappa_p^2}{\tilde{\beta}_l}\right), \tag{52}$$

where σ, κ_p, and D_h are as in (26), (40), and (38), respectively.

Proof. (a)The definition of σ_k in (26), the inequality in (25), the triangle inequality for norms, and the fact that $z_k, z_{k-1} \in \mathcal{H}$ imply that

$$\|r_k\| = \|v_k + z_{k-1} - z_k\| \le \|v_k\| + D_h \le \sigma_k \|r_k\| + D_h \le \sigma \|r_k\| + D_h,$$

which, after a simple re-arrangement, proves (51).
(b) Now, to simplify notation, let $\bar{k} = k_{l-1} + 1$. Now, using (40) and the fact that $\|p_j - p_{j-1}\|^2 \le 2\|p_j\|^2 + 2\|p_{j-1}\|^2$, it follows that for any $k \ge \bar{k} + 1$,

$$\frac{\|p_k\|^2}{2} + \sum_{j=\bar{k}}^{k} \|p_j - p_{j-1}\|^2 \le \frac{\kappa_p^2}{2} + 4(k - \bar{k} + 1)\kappa_p^2 = \frac{(1 + 8(k - \bar{k} + 1))\kappa_p^2}{2} \le 9(k - \bar{k})\kappa_p^2. \tag{53}$$

Hence, (48) with $k = \bar{k}$, (49), (53), and the fact that $\beta_k = \tilde{\beta}_l$ for every $k \in \mathcal{C}_l$, imply that for any $k \in \mathcal{C}_l$ such that $k \ge \bar{k} + 1$,

$$\begin{aligned}
\frac{(1-\sigma^2)}{2\lambda} \sum_{j=\bar{k}+1}^{k} \|r_j\|^2 &\overset{(49)}{\le} \sum_{j=\bar{k}+1}^{k} \left[\mathcal{L}_{\beta_j}(z_{j-1}, p_{j-1}) - \mathcal{L}_{\beta_j}(z_j, p_j) + \frac{1}{\beta_j} \|p_j - p_{j-1}\|^2 \right] \\
&\overset{j \in \mathcal{C}_l}{=} \sum_{j=\bar{k}+1}^{k} \left[\mathcal{L}_{\tilde{\beta}_l}(z_{j-1}, p_{j-1}) - \mathcal{L}_{\tilde{\beta}_l}(z_j, p_j) + \frac{1}{\tilde{\beta}_l} \|p_j - p_{j-1}\|^2 \right] \\
&\le \mathcal{L}_{\tilde{\beta}_l}(z_{\bar{k}}, p_{\bar{k}}) - \mathcal{L}_{\tilde{\beta}_l}(z_k, p_k) + \frac{1}{\tilde{\beta}_l} \sum_{j=\bar{k}+1}^{k} \|p_j - p_{j-1}\|^2 \\
&\overset{(48)}{\le} \mathcal{L}_{\tilde{\beta}_l}(z_{\bar{k}}, p_{\bar{k}-1}) - \mathcal{L}_{\tilde{\beta}_l}(z_k, p_k) + \frac{1}{\tilde{\beta}_l} \sum_{j=\bar{k}}^{k} \|p_j - p_{j-1}\|^2 \\
&\overset{(53)}{\le} \mathcal{L}_{\tilde{\beta}_l}(z_{\bar{k}}, p_{\bar{k}-1}) - \mathcal{L}_{\tilde{\beta}_l}(z_k, p_k) - \frac{\|p_k\|^2}{2\tilde{\beta}_l} + \frac{9(k-\bar{k})\kappa_p^2}{\tilde{\beta}_l} \\
&= (k - \bar{k}) \left[\Delta_k + \frac{9\kappa_p^2}{\tilde{\beta}_l} \right],
\end{aligned}$$

where the last equality follows from the definition of Δ_k in (35) and the fact that $\hat{k} = \bar{k} - 1$. The proof of (52) now follows by dividing the above inequality by $(k - \bar{k})(1 - \sigma^2)/(2\lambda)$ and by taking j such that $\|r_j\| = \min_{\bar{k}+1 \le j \le k} \|r_j\|$. ☐

The next result, whose proof can be found in Appendix C, contains some useful relations about the sequence $\{(\hat{z}_k, \hat{p}_k, \hat{w}_k, \hat{q}_k)\}$ generated by NL-IAPIAL.

Lemma 3.4. *Consider the sequences $\{(\hat{z}_k, \hat{p}_k, \hat{w}_k, \hat{q}_k)\}$, $\{p_k\}$, and $\{r_k\}$ generated by NL-IAPIAL. Then, for every $k \ge 1$, we have:*

$$\hat{w}_k \in \nabla f(\hat{z}_k) + \partial h(\hat{z}_k) + \nabla g(\hat{z}_k)\hat{p}_k, \quad \langle g(\hat{z}_k) + \hat{q}_k, \hat{p}_k \rangle = 0, \quad g(\hat{z}_k) + \hat{q}_k \preceq_{\mathcal{K}} 0, \quad \hat{p}_k \succeq_{\mathcal{K}^*} 0, \tag{54}$$

$$\|\hat{w}_k\| \le \frac{1}{\lambda}(1 + 2\sigma)\|r_k\|, \quad \|\hat{q}_k\| \le \frac{B_g^{(1)}\sigma}{\widetilde{\mathcal{M}}_k}\|r_k\| + \frac{1}{\beta_k}\|p_k - p_{k-1}\|, \tag{55}$$

where $B_g^{(1)}$ is as in (23) and $(\widetilde{\mathcal{M}}_k, \sigma)$ is given in (26).

Some comments about Lemma 3.4 are in order. First, in view of the fact that (54) implies that the quadruple $(\hat{z}, \hat{p}, \hat{w}, \hat{q}) = (\hat{z}_k, \hat{p}_k, \hat{w}_k, \hat{q}_k)$ satisfies all the relations in (5), it follows that such a quadruple becomes a $(\hat{\rho}, \hat{\eta})$-approximate stationary quadruple of (1) whenever $\|\hat{w}_k\| \le \hat{\rho}$ and $\|\hat{q}_k\| \le \hat{\eta}$. The inequalities in (55) provide useful bounds for these residual pair in terms of $\|r_k\|$ and $\|p_k - p_{k-1}\|/\beta_k$ which are used to prove that $\{(\hat{w}_k, \hat{q}_k)\}$ eventually approaches zero. Hence, the latter two inequalities will eventually be satisfied, which implies that NL-IAPIAL computes a $(\hat\rho,\hat\eta)$-approximate stationary quadruple of (1) after a finite number of iterations.

The next result shows that during an l-th cycle of NL-IAPIAL, the residual sequence $\{(\hat w_k,\hat q_k)\}$ can be controlled by $\tilde\beta_l$ and $\{\Delta_k\}$ defined in (35).

Lemma 3.5. *Consider the sequence $\{(\hat w_k,\hat q_k)\}_{k\in\mathcal{C}_l}$ generated during the l-th cycle of NL-IAPIAL. Then, for every $k\in\mathcal{C}_l$ and $k\ge k_{l-1}+2$, there exists an index $j\in\{k_{l-1}+2,\ldots,k\}$ such that*

$$\|\hat w_j\|^2\le 2m_f C_\sigma\Delta_k+\frac{m_f\kappa_2^2}{2M_g\tilde\beta_l},\qquad \|\hat q_j\|\le\frac{m_f\kappa_3^2}{M_g\tilde\beta_l},\tag{56}$$

where C_σ, Δ_k, and (κ_2,κ_3) are as in (30), (35), and (42), respectively.

Proof. First, recall that for any $k\in\mathcal{C}_l$, we have $\beta_k=\tilde\beta_l$ in view of (36). Hence, the proof of the first inequality in (56) for some $j\in\{k_{l-1}+2,\ldots,k\}$ follows immediately from Lemma 3.3(b), the first inequality in (55), and the definitions of (C_σ,λ) and κ_2 in (30) and (42), respectively. Now, from the second inequality in (55), the definition of λ in (30), the triangle inequality for norms, Proposition 2.5, (51), and the fact that $\widetilde{\mathcal{M}}_k\ge\lambda\tilde\beta_l M_g$ (see (24) and (26)), we have

$$\begin{aligned}\|\hat q_j\|&\le\frac{B_g^{(1)}\sigma}{\widetilde{M}_k}\|r_j\|+\frac{1}{\tilde\beta_l}\left(\|p_j\|+\|p_{j-1}\|\right)\le\frac{\sigma B_g^{(1)}D_h}{\lambda(1-\sigma)M_g\tilde\beta_l}+\frac{2\kappa_p}{\tilde\beta_l}\\&=\left(\frac{\sigma B_g^{(1)}D_h}{1-\sigma}+\frac{M_g\kappa_p}{m_f}\right)\frac{2m_f}{M_g\tilde\beta_l}.\end{aligned}$$

On the other hand, it follows from the fact that $B_g^{(1)}\le\sqrt{M_g}$ (see (24)) and the definitions of θ_h, κ_0, and κ_p in (38), (39), and (40), respectively, that

$$\frac{\sigma B_g^{(1)}D_h}{1-\sigma}\le\frac{\sigma\min\{1,\bar d\}\theta_h\sqrt{M_g}}{1-\sigma}\le\frac{\sigma D_h\theta_h\sqrt{M_g}}{1-\sigma}\le\frac{\kappa_0\theta_h\sqrt{M_g}}{4m_f}\le\frac{\tau_g\kappa_p\sqrt{M_g}}{4m_f}.$$

Hence, we conclude that

$$\|\hat q_j\|\le\left(\tau_g\sqrt{M_g}+4M_g\right)\frac{\kappa_p}{2M_g\tilde\beta_l}\quad\forall j\in\{k_{l-1}+2,\ldots,k\},$$

which, together with the previous conclusion about $\|\hat w_j\|$ and the definition of κ_3 in (42), implies the existence of an index $j\in\{k_{l-1}+2,\ldots,k\}$ satisfying (56). □

The next result establishes the rate in which the sequence $\{\Delta_k\}$ defined in (35) converges to zero

Lemma 3.6. *Consider the sequence $\{(z_k,p_k)\}_{k\in\mathcal{C}_l}$ generated during the l-th cycle of NL-IAPIAL and let Δ_k be as in (35). Then, for every $k\in\mathcal{C}_l$ and $k\ge k_{l-1}+2$, we have*

$$\Delta_k\le\frac{\Delta\phi^*+2m_fD_h}{k-k_{l-1}-1},$$

where D_h, $\Delta\phi^*$, and m_f are as in (38), (41), and (A2), respectively.

Proof. From step 1 of NL-IAPIAL we have that $(\lambda,z_k,v_k,\varepsilon_k,\sigma_k)$ satisfies (25). Moreover, we also have $1-2\sigma_k^2\ge0$ due to $\sigma_k\le\sigma\in(0,1/\sqrt{2}]$ (see NL-IAPIAL input and (26)). Hence, it follows from Lemma B.3 with $\tilde\phi=\lambda\mathcal{L}_{\beta_k}(\cdot,p_{k-1})$, $(\tilde\sigma,s)=(\sigma_k,1)$, and $(x_0,x)=(z_{k-1},z_k)$ that

$$\lambda\mathcal{L}_{\beta_k}(z_k,p_{k-1})\le\lambda\mathcal{L}_{\beta_k}(z,p_{k-1})+\|z-z_{k-1}\|^2,\qquad\forall z\in\mathcal{H}.\tag{57}$$

Since the definition of $\mathcal{L}_\beta$ in (2) implies that $\mathcal{L}_{\beta_k}(z,p_{k-1})\le\phi(z)$ for every $z\in\mathcal{F}:=\{z\in\mathcal{H}:g(z)\preceq_{\mathcal{K}}0\}$, it follows from (57) and the definitions of ϕ^* and D_h in (1) and (38), respectively, that

$$\mathcal{L}_{\beta_k}(z_k,p_{k-1})\le\phi^*+\frac{D_h^2}{\lambda}.\tag{58}$$

Now, in view of the definitions of $\mathcal{L}_\beta$ and ϕ_* given in (2) and (43), respectively, we have

$$\mathcal{L}_{\beta_k}(z_k,p_k)+\frac{\|p_k\|^2}{2\tilde{\beta}_l}=\phi(z_k)+\frac{1}{2\tilde{\beta}_l}\text{dist}^2(p_k+\tilde{\beta}_l g(z_k),-\mathcal{K})\ge\phi_*.$$

Since the l-th cycle $\mathcal{C}_l$ starts at iteration $k_{l-1}+1$ and $\beta_k=\tilde{\beta}_l$ for any $k\in\mathcal{C}_l$, it follows from the definition of Δ_k given in (35), (58) with $k=k_{l-1}+1$, and the above inequality that

$$\Delta_k=\frac{1}{k-k_{l-1}-1}\left(\mathcal{L}_{\tilde{\beta}_l}(z_{k_{l-1}+1},p_{k_{l-1}})-\mathcal{L}_{\tilde{\beta}_l}(z_k,p_k)-\frac{\|p_k\|^2}{2\tilde{\beta}_l}\right)\le\frac{1}{k-k_{l-1}-1}\left(\phi^*+\frac{D_h^2}{\lambda}-\phi_*\right),$$

which proves the lemma in view of the definitions of λ and $\Delta\phi^*$ in (30) and (43), respectively. □

Now we are ready to present the proof of Proposition 2.6.

Proof of Proposition 2.6.
(a) First note that NL-IAPIAL calls in its step 1 the ACG algorithm of Appendix A with inputs given by (27). Note also that within the l-th cycle, we have $\beta_k=\tilde{\beta}_l$ in view of (36). Hence, since $m_f\le L_f$ (see (A2)), we conclude that (a) follows from Lemma 2.4 and the fact that (40) and the definitions of $\widetilde{\mathcal{M}}_k$, λ, and κ_1 given in (26), (30), and (42), respectively, imply that

$$\begin{aligned}\widetilde{\mathcal{M}}_k&=\lambda\left(L_f+L_g\|p_{k-1}\|+\beta_k M_g\right)+1\\&\le\lambda\left(L_f+L_g\kappa_p+\tilde{\beta}_l M_g\right)+1\le\kappa_1^2+\frac{\tilde{\beta}_l M_g}{2m_f}.\end{aligned}$$

(b) Fix a cycle l and note that $\hat{k}$ in step 3 corresponds to $\hat{k}=k_{l-1}$. It follows from Lemma 3.6 that, for every $k\in\mathcal{C}_l$ and $k\ge\hat{k}+2$,

$$\Delta_k\le\frac{\Delta\phi^*+2m_f D_h}{k-\hat{k}-1}.$$

Hence, we have that if some $k\in\mathcal{C}_l$ is such that

$$k>\hat{k}+1+\frac{2C_\sigma\left(\Delta\phi^*+2m_f D_h\right)}{\lambda\hat{\rho}^2} \tag{59}$$

then Δ_k satisfies inequality (35), ending the l-th cycle. Hence, (b) follows immediately from this conclusion, the definition of λ in (30), and the fact that the l-th cycle starts at $\hat{k}+1$.

(c) First, recall that in the l-th cycle of NL-IAPIAL, we have $\beta_k=\tilde{\beta}_l=2^{l-1}\beta_1$, for every $l\ge1$ (see (36)). If NL-IAPIAL performs just one cycle then $\bar{l}=1$ and then the result immediately follows from (54), the stopping criterion in step 2 and Definition 2.2. Assume then that NL-IAPIAL performs more than one cycle. We argue that NL-IAPIAL stops before or at the first cycle $\bar{l}$ where $\tilde{\beta}_{\bar{l}}\ge\bar{\beta}(\hat{\rho},\hat{\eta})$ and $\bar{\beta}(\hat{\rho},\hat{\eta})$ is as in (43). Suppose that the algorithm has not stopped before a cycle $\bar{l}$, and note that the definition of $\bar{\beta}(\hat{\rho},\hat{\eta})$ in (43) implies

$$\tilde{\beta}_{\bar{l}}\ge\frac{m_f}{M_g}\left(\frac{\kappa_2^2}{\hat{\rho}^2}+\frac{\kappa_3^2}{\hat{\eta}}\right), \tag{60}$$

where κ_2 and κ_3 are as in (42). Now, if at the $\bar{l}$-th cycle, NL-IAPIAL performs at least $\bar{k}\ge k_{\bar{l}-1}+2$ outer iterations, where $\bar{k}$ is the smallest index such that

$$\frac{2m_f C_\sigma\left(\Delta\phi^*+2m_f D_h\right)}{\bar{k}-k_{\bar{l}-1}-1}\le\frac{\hat{\rho}^2}{2}, \tag{61}$$

then, in view of (56), Lemma 3.6, (60), and (61), there exists an index $j \in \{k_{\bar{l}-1}+2,\ldots,\bar{k}\}$ such that

$$\|\hat{w}_j\|^2 \le 2m_f C_\sigma \Delta_{\bar{k}} + \frac{m_f \kappa_2^2}{2M_g \tilde{\beta}_{\bar{l}}} \le \frac{2m_f C_\sigma\left(\Delta\phi^* + 2m_f D_h\right)}{\bar{k} - k_{\bar{l}-1} - 1} + \frac{\kappa_2^2}{2}\left(\frac{\kappa_2^2}{\hat{\rho}^2} + \frac{\kappa_3^2}{\hat{\eta}}\right)^{-1} \le \frac{\hat{\rho}^2}{2} + \frac{\hat{\rho}^2}{2} = \hat{\rho}^2,$$

and also

$$\|\hat{q}_j\| \le \frac{m_f \kappa_3^2}{M_g \tilde{\beta}_{\bar{l}}} \le \kappa_3^2\left(\frac{\kappa_2^2}{\hat{\rho}^2} + \frac{\kappa_3^2}{\hat{\eta}}\right)^{-1} \le \hat{\eta}.$$

More specifically, since we assumed that at least $\bar{k}$ iterations are performed, we have $j = \bar{k}$. Hence, NL-IAPIAL must stop before or on iteration $\bar{k}$ within the $\bar{l}$ cycle, in view of the stopping criterion in step 2. In view of step 3 of NL-IAPIAL, we then have that

$$\beta_k = \tilde{\beta}_l = 2^{l-1}\beta_1 \le 2\bar{\beta}, \quad \forall l \le \bar{l}.$$

The conclusion now follows from the above bound, step 2 of NL-IAPIAL, (54), and Definition 2.2. □

3.2. Proof of Proposition 2.5 The first lemma describes some basic facts about the sequence $\{(z_k, p_k, w_k, r_k, \varepsilon_k)\}$ generated by NL-IAPIAL.

Lemma 3.7. *Consider the sequence $\{(z_k, p_k, w_k, r_k, \varepsilon_k)\}$ generated by NL-IAPIAL. Then, the following statements hold for every $k \ge 1$:*

(a) the quintuple $(z_k, p_k, w_k, r_k, \varepsilon_k)$ satisfies

$$\begin{aligned} &w_k \in \nabla f(z_k) + \partial_{(\lambda^{-1}\varepsilon_k)} h(z_k) + \nabla g(z_k) p_k, \\ &\|w_k\| \le \frac{1}{\lambda}(1+\sigma)\|r_k\|, \quad \varepsilon_k \le \frac{\sigma^2}{2}\|r_k\|^2; \end{aligned} \tag{62}$$

(b) the residual pair (w_k, ε_k) satisfies

$$\varepsilon_k \le \frac{\sigma^2 D_h^2}{2(1-\sigma)^2}, \quad \|w_k\| \le \left(\frac{1+\sigma}{1-\sigma}\right)\frac{D_h}{\lambda}, \tag{63}$$

where σ and D_h are as in (26) and (38), respectively.

Proof. (a) The proof of this statement is presented in Appendix C.

(b) The first inequality in (63) follows by combining (51), the inequality in (25), and the definition of σ_k in (26). The last inequality in (63) follows from (51) and the first inequality in (62). □

The following technical result, whose proof can be found in Lemma 3.10 of [19], plays an important role in the proof of Lemma 3.10 below.

Lemma 3.8. *Let h be a function as in (A1). Then, for every $u, z \in \mathcal{H}$, $\delta \ge 0$, and $\xi \in \partial_\delta h(z)$, we have*

$$\|\xi\| \mathrm{dist}(u, \partial\mathcal{H}) \le \left[\mathrm{dist}(u, \partial\mathcal{H}) + \|z-u\|\right] K_h + \langle \xi, z-u\rangle + \delta,$$

where $\partial\mathcal{H}$ denotes the boundary of $\mathcal{H}$.

The idea behind the proof of Lemma 3.10 of [19] is based on the following two observations: i) any h as in (A1) satisfies the condition that $\partial_\varepsilon h(z) \subset \mathcal{N}_{\mathcal{H}}^{\varepsilon}(z) + \bar{B}(0, K_h)$ (see Lemma A.2(ii) of [19]); and, ii) any closed convex function satisfying the latter condition satisfies the conclusion of Lemma 3.8. It is worth mentioning that the proof of the second observation uses a technical inequality that appears in the proof of Lemma 3 of [26].

The following technical result, whose proof is based on the two previous lemmas, is used in Lemma 3.10 to derive a recursive formula below relating p_{k-1} and p_k.

Lemma 3.9. *Consider the sequence $\{(z_k, p_k)\}$ generated by NL-IAPIAL and let $\bar z$, κ_0, and $\bar d$ be as in (A4), (39), and (38), respectively. Then, the following inequality holds*

$$\langle \nabla g(z_k)p_k, z_k - \bar z\rangle \le D_h\kappa_0 - \bar d\|\nabla g(z_k)p_k\|, \qquad \forall k \ge 1. \tag{64}$$

Proof. Let $\{(z_k, p_k, w_k)\}$ be generated by NL-IAPIAL and note that, in view of the inclusion in (62), we have $w_k - \nabla f(z_k) - \nabla g(z_k)p_k \in \partial_{(\lambda^{-1}\varepsilon_k)} h(z_k)$ for every $k \ge 1$. Hence, it follows from the definition of $\bar d$, and Lemma 3.8 with $\xi = w_k - \nabla f(z_k) - \nabla g(z_k)p_k$, $z = z_k$, $u = \bar z$ and $\delta = \lambda^{-1}\varepsilon_k$, that

$$\begin{aligned}\bar d\|w_k - \nabla f(z_k) - \nabla g(z_k)p_k\| &\le \left(\bar d + \|z_k - \bar z\|\right)K_h + \langle w_k - \nabla f(z_k) - \nabla g(z_k)p_k, z_k - \bar z\rangle + \frac{\varepsilon_k}{\lambda}\\ &\le (\bar d + D_h)K_h - \langle \nabla g(z_k)p_k, z_k - \bar z\rangle + \|w_k - \nabla f(z_k)\|D_h + \frac{\varepsilon_k}{\lambda},\end{aligned}$$

where the last inequality is due to Cauchy-Schwarz inequality and the fact that $\|z_k - \bar z\| \le D_h$ (in view of $\bar z$, $z_k \in \mathcal{H}$ and the definition of D_h in (38)). Now, using the reverse triangle inequality for norms and rearranging the resulting inequality, we have

$$\begin{aligned}\langle \nabla g(z_k)p_k, z_k - \bar z\rangle + \bar d\|\nabla g(z_k)p_k\| &\le (\bar d + D_h)K_h + \|w_k - \nabla f(z_k)\|\,(\bar d + D_h) + \frac{\varepsilon_k}{\lambda}\\ &\le 2D_hK_h + 2\left(\frac{(1+\sigma)D_h}{\lambda(1-\sigma)} + B_f^{(1)}\right)D_h + \frac{\sigma^2 D_h^2}{2\lambda(1-\sigma)^2}\end{aligned}$$

where the last inequality is due to the definition of $B_f^{(1)}$ in (38), the inequalities in (63), and the fact $\bar d \le D_h$. Hence, (64) follows in view of the definition of κ_0 in (39). □

We are now ready to show that the sequence $\{p_k\}$ is bounded.

Lemma 3.10. *Consider the sequence $\{(p_k, \beta_k)\}$ generated by NL-IAPIAL and let κ_0, τ_g, and $\bar d$ be as in (39), (A4) and (38), respectively. Then, for every $k \ge 1$, we have*

$$\min\{1, \bar d\}\tau_g\|p_k\| + \frac{\|p_k\|^2}{\beta_k} \le D_h\kappa_0 + \frac{1}{\beta_k}\langle p_k, p_{k-1}\rangle. \tag{65}$$

Proof. First note that the first two identities in (46) imply that

$$\langle p_k, g(z_k)\rangle = \frac{1}{\beta_k}\langle p_k, s_k + p_k - p_{k-1}\rangle = \frac{\|p_k\|^2}{\beta_k} - \frac{1}{\beta_k}\langle p_k, p_{k-1}\rangle.$$

Using this identity, (64), the fact that $p_k \in \mathcal{K}^*$, and relation (15) with $(z, z', p) = (z_k, \bar z, p_k)$, we conclude that

$$\begin{aligned}D_h\kappa_0 - \bar d\|\nabla g(z_k)p_k\| &\overset{(64)}{\ge} \langle \nabla g(z_k)p_k, z_k - \bar z\rangle = \langle p_k, g'(z_k)(z_k - \bar z)\rangle\\ &\overset{(15)}{\ge} \langle p_k, g(z_k)\rangle - \langle p_k, g(\bar z)\rangle = \frac{\|p_k\|^2}{\beta_k} - \frac{1}{\beta_k}\langle p_k, p_{k-1}\rangle + |\langle p_k, g(\bar z)\rangle|,\end{aligned}$$

or equivalently,

$$\bar d\|\nabla g(z_k)p_k\| + |\langle p_k, g(\bar z)\rangle| + \frac{\|p_k\|^2}{\beta_k} \le D_h\kappa_0 + \frac{1}{\beta_k}\langle p_k, p_{k-1}\rangle.$$

Inequality (65) now follows from (17) and the latter inequality. □

Based on the recursive formula (65), we are now ready to give the proof of Proposition 2.5.

Proof of Proposition 2.5. The proof is done by induction. Inequality (40) trivially holds for $k=0$. Assume that (40) holds with $k=i-1$ for some $i\geq 1$. This assumption together with (65), the Cauchy-Schwarz inequality, and the definitions of θ_h and κ_p in (38) and (40), respectively, imply that

$$\left(\min\{1,\bar{d}\}\tau_g + \frac{\|p_i\|}{\beta_i}\right)\|p_i\| \leq D_h\kappa_0 + \frac{\|p_i\|\cdot\|p_{i-1}\|}{\beta_i} \leq D_h\kappa_0 + \frac{\|p_i\|\kappa_p}{\beta_i}$$
$$= \min\{1,\bar{d}\}\tau_g\frac{\theta_h\kappa_0}{\tau_g} + \frac{\|p_i\|\kappa_p}{\beta_i} \leq \left(\min\{1,\bar{d}\}\tau_g + \frac{\|p_i\|}{\beta_i}\right)\kappa_p,$$

which implies that $\|p_i\|\leq\kappa_p$. Then, (40) also holds with $k=i$ and hence, by induction, we conclude that (40) holds for the whole sequence $\{p_k\}$. □

4. Numerical Experiments

This section presents numerical experiments that highlight the performance of two variants of NL-IAPIAL, named IPL and IPL(A), against six other benchmark methods for solving NCO problems with linear or nonlinear convex constraints. It contains five subsections. The first four present the numerical results on different classes of constrained NCO problems, while the last one contains a summary and some comments. For replication purposes, the MATLAB code for generating the results of this section is available online[7].

Before proceeding, we first precisely describe the implementations of NL-IAPIAL. The IPL and IPL(A) variants considered differ from the description in Section 2 in two important ways. First, they both modify the parameter $\tilde{\sigma}$ that is given to the ACG algorithm in its step 1. More specifically, instead of choosing $\tilde{\sigma}=\sigma_k$ at the k-th iteration, the implementation chooses $\tilde{\sigma}=\min\{\nu/(\widetilde{\mathcal{M}}_k)^{1/2},\sigma\}$ for $\nu\gg 0$. Second, in view of the first modification, they both replace condition (35) with the modified condition

$$\Delta_k \leq \frac{\lambda(1-\sigma^2)\hat{\rho}^2}{4(1+2\nu)^2},$$

where ν is as previously described. In addition to these modifications, IPL(A) replaces the ACG algorithm with an ACG variant that adapts the ACG stepsize for every ACG prox subproblem. In particular, it uses the line search subroutine outlined in Appendix A, and it applies a warm-start strategy[8] for choosing the parameter $\widetilde{M}$ given to ACG for each prox-subproblem. Regarding (σ,ν) and the other hyperparameters, both variants choose

$$\beta_1 = \max\left\{1, \frac{L_f}{\left[B_g^{(1)}\right]^2}\right\},\quad \lambda = \frac{1}{2m_f},\quad \sigma=\sqrt{0.3},\quad \nu=\sqrt{\sigma\left(\lambda L_f+1\right)},\quad p_0=0.$$

While we do not show how the above changes affect the convergence of IPL and IPL(A), we do note that their convergence can be analyzed using the techniques of this paper and those in [19].

We also describe the six benchmark algorithms of this section namely, two variants of the QP-AIPP method of [17] (nicknamed QP and QP(A)), the iALM of [24], two variants of the S-prox-ALM (nicknamed SPA1 and SPA2) of [42, 43], and the HiAPeM of [25] (nicknamed HPM). QP is the method in [16, Algorithm 4.1.1] while QP(A) is a modification of QP that uses the same adaptive ACG variant and parameter warm-start strategy used by IPL(A). iALM was implemented by the authors to be exactly as stated in [24, Algorithm 3] with the parameters σ, β_0, w_0, $\mathbf{y}^0$, and γ_k chosen as

$$\sigma=2,\quad \beta_0=\max\left\{1,\frac{L_f}{\|\mathcal{A}\|^2}\right\},\quad w_0=1,\quad \boldsymbol{y}^0=0,\quad \gamma_k=\frac{(\log 2)\,\|c(x^1)\|}{(k+1)\left[\log(k+2)\right]^2}\quad \forall k\geq 1,$$

[7] See the examples in `./tests/papers/nl-IAPIAL` from the GitHub repository https://github.com/wwkong/nc_opt/.

[8] For the first prox subproblem, $\widetilde{M}$ is initialized to $\lambda\widetilde{\mathcal{M}}_k/2+1$. For $k\geq 1$, if $L_{\hat{j}}$ is the last (estimated) curvature constant generated by the adaptive ACG for the k^{th} prox-subproblem, then $\widetilde{M}$ for the $(k+1)^{\text{th}}$ subproblem is initialized to $\lambda J_{k+1}/2+1$, where $J_{k+1}:=(L_{\hat{j}}-1)/\lambda$.

as suggested in [24, Theorem 2]. Moreover, the starting point for each APG[9] call is the prox center for the current prox subproblem. SPA1–SPA2 were also implemented by the authors to be exactly as stated in [42, Algorithm 2] with the parameters α_1, p, c, β, y_0, and z_0 chosen as

$$\alpha_1 = \frac{\Gamma}{4}, \quad p = 2(L_f + \Gamma\|A\|^2), \quad c = \frac{1}{2(L_f + \Gamma\|A\|^2)}, \quad \beta = 0.5, \quad y_0 = 0, \quad z_0 = x_0,$$

where $\Gamma = 1$ in SPA1 and $\Gamma = 10$ in SPA2. Finally, the code for HiAPeM was provided by the authors of [25] with the parameters σ, β_0, γ, γ_1, γ_2, N_0, and N_1 chosen as

$$\sigma = 3, \quad \beta_0 = 10^{-2}, \quad \gamma = 1.1, \quad \gamma_1 = 1.5, \quad \gamma_2 = 1, \quad N_0 = 100, \quad N_1 = 2.$$

We next describe numerical and mathematical details that are common to all the experiments. First, throughout this section, we denote I to be the identity matrix, $\mathbb{S}^n$ to be the set of symmetric n-by-n matrices, and $\mathbb{S}^n_+$ to be the set of positive semidefinite matrices in $\mathbb{S}^n$. Second, given a tolerance pair $(\hat{\rho}, \hat{\eta}) \in \Re^2_{++}$, a pointed convex cone $\mathcal{K}$, and $z_0 \in \operatorname{dom} h$, all the methods attempt to find a pair $(\hat{z}, \hat{p})$ satisfying

$$\frac{\operatorname{dist}(0, \nabla f(\hat{z}) + \partial h(\hat{z}) + \nabla g(\hat{z})\hat{p})}{1 + \|\nabla f(z_0)\|} \le \hat{\rho}, \quad \frac{\operatorname{dist}(g(\hat{z}), N_{\mathcal{K}^*}(\hat{p}))}{1 + \operatorname{dist}(g(z_0), -\mathcal{K})} \le \hat{\eta}. \tag{66}$$

Third, as all the methods tested utilize an ACG variant to solve a sequence of convex proximal subproblems, the number of iterations reported in the experiments are the total number of ACG iterations needed to obtain a quadruple satisfying (66) (including those which fail to satisfy parameter line searches within the adaptive ACG variants used in IPL(A), QP(A), and HiAPeM). Fourth, the bold numbers in each of the tables of this section indicate the method that performed the most efficiently for a given metric, e.g., runtime or iteration count. Finally, all algorithms described at the beginning of this section are implemented in MATLAB 2021a and are run on Linux 64-bit machines, each containing Xeon E5520 processors and at least 8 GB of memory.

We now end with some comments about the choice of algorithms in the experiments presented in the subsections below. First, QP and QP(A) methods are not included in the experiments of Subsections 4.2 and 4.3 because their current implementations are only available for linearly-constrained problems (even though they can be extended to nonlinearly-constrained problems). Second, HiAPeM is only included in the experiments of Subsection 4.3 because the code provided to the authors is specifically designed to solve the problem class considered in that subsection. Third, S-prox-ALM is only included in the experiments of Subsection 4.4 because its convergence is only guaranteed when the composite function h is the indicator function of a polyhedron. Finally, we do not include QP and IPL in Subsection 4.4 because the results of Subsections 4.1, 4.2, and 4.3 show that their adaptive variants are substantially more efficient.

4.1. Nonconvex QSDP Given a pair of dimensions $(\ell, n) \in \mathbb{N}^2$, a scalar pair $(\alpha_1, \alpha_2) \in \Re^2_{++}$, linear operators $\mathcal{A} : \mathbb{S}^n_+ \mapsto \Re^\ell$, $\mathcal{B} : \mathbb{S}^n_+ \mapsto \Re^n$, and $\mathcal{C} : \mathbb{S}^n_+ \mapsto \Re^\ell$ defined pointwise by

$$[\mathcal{A}(Z)]_i = \langle A_i, Z \rangle, \quad [\mathcal{B}(Z)]_j = \langle B_j, Z \rangle, \quad [\mathcal{C}(Z)]_i = \langle Q_i, Z \rangle,$$

for matrices $\{A_i\}_{i=1}^\ell, \{B_j\}_{j=1}^n, \{Q_i\}_{i=1}^\ell \subseteq \Re^{n\times n}$, positive diagonal matrix $D \in \Re^{n\times n}$, and a vector pair $(b, d) \in \Re^\ell \times \Re^\ell$, we consider the following nonconvex quadratic semidefinite programming (QSDP) problem:

$$\begin{aligned} \min_{z \in \mathbb{S}^n_+} \ & -\frac{\alpha_1}{2}\|D\mathcal{B}(z)\|^2 + \frac{\alpha_2}{2}\|\mathcal{C}(z) - d\|^2 \\ \text{s.t. } & \mathcal{A}(z) = b, \quad 0 \preceq z \preceq rI. \end{aligned}$$

Parameters				**Iteration Count**					**Runtime**				
n	r	m	L_f	iALM	QP	QP(A)	IPL	IPL(A)	iALM	QP	QP(A)	IPL	IPL(A)
50	1.0	1	10	-	23296	1633	18618	**1257**	-	201.7	17.2	172.9	**15.1**
50	1.0	1	20	-	15402	1210	10610	**782**	-	132.8	12.5	98.6	**9.3**
50	1.0	1	40	-	12611	1076	7614	**884**	-	108.7	11.0	70.8	**10.5**
50	1.0	5	40	-	16499	1239	10578	**753**	-	144.8	13.5	100.6	**9.8**
50	1.0	10	40	-	17868	1582	15238	**1207**	-	157.5	17.4	147.4	**16.1**
50	1.0	20	40	-	74732	4425	53599	**1633**	-	665.1	51.6	506.2	**22.6**
50	5.0	1	20	-	40716	2648	35138	**2335**	-	353.3	28.3	326.9	**28.2**
50	10.0	1	20	-	110657	6130	99621	**5998**	-	964.1	**66.9**	928.7	72.8
50	20.0	1	20	-	129175	7112	116263	**6936**	-	1125.8	**77.7**	1088.4	86.5
75	1.0	1	10	-	41201	1948	35565	**1626**	-	363.6	21.0	336.2	**19.5**
75	1.0	1	20	-	32647	1576	27857	**1289**	-	289.1	16.8	264.0	**15.4**
75	1.0	1	40	-	24932	1289	19939	**984**	-	220.7	**13.7**	202.4	18.3
75	1.0	5	40	-	31641	1462	23537	**1025**	-	375.5	**17.5**	317.1	17.9
75	1.0	10	40	-	31874	1557	25519	**1011**	-	367.1	27.8	344.3	**18.4**
75	1.0	20	40	-	38605	1945	23725	**1077**	-	481.9	27.3	312.9	**21.8**
75	5.0	1	20	-	92271	3830	87426	**3648**	-	1137.5	57.2	1088.7	**42.0**
75	10.0	1	20	-	104348	4245	98207	**4060**	-	886.5	**44.3**	926.3	48.2
75	20.0	1	20	-	152856	5961	143057	**5807**	-	1312.4	**66.2**	1380.6	71.3
100	1.0	1	10	-	103570	3251	95110	**2928**	-	1641.3	62.2	1590.0	**61.6**
100	1.0	1	20	-	74587	2466	66010	**2262**	-	1180.4	**46.9**	1102.5	47.2
100	1.0	1	40	-	59253	2040	50282	**1689**	-	934.5	38.6	837.6	**35.1**
100	1.0	5	40	-	55305	1646	46890	**1499**	-	880.3	**32.4**	790.3	32.9
100	1.0	10	40	-	82005	3133	61144	**2698**	-	1311.5	63.9	1034.8	**62.2**
100	1.0	20	40	-	70045	2266	50591	**1499**	-	1127.7	46.7	866.5	**36.3**
100	5.0	1	20	-	129478	3998	119623	**3649**	-	2059.9	77.6	2008.2	**76.8**
100	10.0	1	20	-	174666	5178	163769	**4844**	-	2774.6	**99.5**	2750.9	101.7
100	20.0	1	20	-	238866	6887	225963	**6563**	-	3798.7	**133.3**	3789.0	139.3

Table 3. Iteration counts and runtimes (in seconds) for the Nonconvex QSDP Problem in Subsection 4.1. Cells marked with "–" are those that did not obtain a solution within the given time limit.

In particular, the problem instances tested are given in Table 3 for algorithms QP, QP(A), IPL, IPL(A), and iALM. For additional clarity, we describe below how the instances were generated.

First, we chose $\ell = 10$, varied n across different problem instances, set $\hat{\rho} = 10^{-2}$ and $\hat{\eta} = 10^{-4}$, and ensured that only 5% of the entries of A_i, B_j, and Q_i were set to be nonzero. Second, the entries of A_i, B_j, Q_i, and d (resp. D) were generated by sampling from the uniform distribution $\mathcal{U}[0,1]$ (resp. $\mathcal{U}\{1,...,1000\}$). Third, the vector b was set to $b = \mathcal{A}(\mathrm{diag}(u))$ where u is a random vector in $\mathcal{U}[0,r]^{n\times n}$. Fourth, the initial starting point z_0 was set to be the zero matrix. Finally, each problem instance considered was based on a specific triple (r, m_f, L_f), for which the scalar pair (α_1, α_2) is selected so that $L_f = \lambda_{\max}(\nabla^2 f)$ and $-m_f = \lambda_{\min}(\nabla^2 f)$, and we set a time limit of 6000 seconds.

4.2. Nonconvex QC-QSDP Given a dimension pair $(\ell, n) \in \mathbb{N}^2$, scalar $r > 0$, matrices $P, Q, R \in \Re^{n\times n}$, and the quantities (α_1, α_2), $\mathcal{B}$, $\mathcal{C}$, D, and d as in Subsection 4.1, we consider the

[9] APG is the name of the ACG subroutine used by iALM.

nonconvex quadratically constrained QSDP (QC-QSDP) problem:

$$\begin{aligned} \min_{Z} \ & -\frac{\alpha_1}{2}\|D\mathcal{B}(Z)\|^2 + \frac{\alpha_2}{2}\|\mathcal{C}(Z) - d\|^2 \\ \text{s.t.} \ & \frac{1}{2}(PZ)^* PZ + \frac{1}{2}Q^*QZ + \frac{1}{2}ZQ^*Q \preceq R^*R, \\ & 0 \preceq Z \preceq rI. \end{aligned}$$

In particular, the problem instances tested are given in Table 4 for algorithms iALM, IPL, and IPL(A). For additional clarity, we describe below how the instances were generated.

Parameters					**Iteration Count**			**Runtime**		
n	r	m	L_f	L_g	iALM	IPL	IPL(A)	iALM	IPL	IPL(A)
50	1.0	10^0	10^3	6.2	-	11058	**6760**	-	108.5	**80.1**
50	1.0	10^0	10^4	10.9	-	244	**213**	-	2.4	**2.4**
50	1.0	10^0	10^5	17.1	1862	778	**580**	18.2	7.5	**6.7**
50	1.0	10^1	10^5	10.9	-	244	**213**	-	2.3	**2.4**
50	1.0	10^2	10^5	6.2	-	11058	**6760**	-	107.5	**79.7**
50	1.0	10^3	10^5	2.7	-	13062	**7381**	-	134.4	**89.5**
50	5.0	10^0	10^5	3.4	724	778	**580**	7.2	7.5	**6.7**
50	10.0	10^0	10^5	1.7	726	778	**580**	7.1	7.4	**6.7**
50	20.0	10^0	10^5	0.9	720	778	**580**	7.1	7.5	**6.7**
75	1.0	10^0	10^3	8.9	-	22766	**12386**	-	418.4	**280.3**
75	1.0	10^0	10^4	15.8	-	244	**212**	-	4.4	**4.5**
75	1.0	10^0	10^5	24.7	3409	777	**579**	61.5	14.1	**12.8**
75	1.0	10^1	10^5	15.8	-	244	**212**	-	4.4	**4.6**
75	1.0	10^2	10^5	8.9	-	20257	**12317**	-	377.3	**281.3**
75	1.0	10^3	10^5	4.0	-	135657	**19950**	-	2515.9	**571.6**
75	5.0	10^0	10^5	4.9	5879	777	**579**	140.4	14.2	**13.0**
75	10.0	10^0	10^5	2.5	1115	777	**579**	20.2	14.2	**13.0**
75	20.0	10^0	10^5	1.2	10832	777	**579**	194.9	14.2	**13.0**
100	1.0	10^0	10^3	11.9	-	40755	**16292**	-	1230.0	**612.6**
100	1.0	10^0	10^4	21.2	-	252	**213**	-	7.5	**7.7**
100	1.0	10^0	10^5	33.2	4710	778	**580**	128.2	23.1	**21.5**
100	1.0	10^1	10^5	21.2	-	244	**213**	-	7.3	**7.7**
100	1.0	10^2	10^5	11.9	-	158085	**22101**	-	4714.2	**831.4**
100	1.0	10^3	10^5	5.3	-	-	**61179**	-	-	**2306.2**
100	5.0	10^0	10^5	6.6	3575	778	**580**	97.7	23.1	**21.5**
100	10.0	10^0	10^5	3.3	2406	778	**580**	65.8	23.3	**21.5**
100	20.0	10^0	10^5	1.7	1706	778	**580**	46.5	23.1	**21.4**

TABLE 4. Iteration counts and runtimes (in seconds) for Nonconvex QC-QSDP Problems in Subsection 4.2. Cells marked with "–" are those that did not obtain a solution within the given time limit.

First, we chose $\ell = 10$, varied n across different problem instances, and chose $\hat{\rho} = \hat{\eta} = 10^{-3}$. Second, the quantities $\mathcal{B}$, $\mathcal{C}$, D, and d were generated in the same way as in Subsection 4.1, the matrix R was set to I, and the entries of matrices P and Q were sampled from the uniform distributions $\log(L_f/m_f) \cdot \mathcal{U}[0, 1/\sqrt{100nr}]$ and $\mathcal{U}[0, 1/n]$, respectively. Third, the initial starting point z_0 was set to be the zero matrix. Finally, like in Subsection 4.1, each problem instance considered was based on a specific triple (r, m_f, L_f), for which the scalar pair (α_1, α_2) is selected so that $L_f = \lambda_{\max}(\nabla^2 f)$ and $-m_f = \lambda_{\min}(\nabla^2 f)$, and a time limit of 6000 seconds.

4.3. Nonconvex QC-QP Given a dimension pair $(\ell, n) \in \mathbb{N}^2$, matrices $\{Q_j\}_{j=0}^{\ell}$, vectors $\{c_j\}_{j=0}^{\ell}$, scalars $\{d_j\}_{j=0}^{\ell}$, and scalar $r > 0$, we consider the nonconvex quadratically constrained quadratic programming (QC-QP) problem:

$$\begin{aligned} \min_z \ & \frac{1}{2} z^T Q_0 z + c_0^T z + d_0 \\ \text{s.t.} \ & \frac{1}{2} z^T Q_j z + c_j^T z + d_j \le 0, \quad && j \in \{1, ..., \ell\}, \\ & -r \le z_i \le r, && i \in \{1, ..., n\}, \end{aligned}$$

where $Q_j \succeq 0$ for $j = 1, ..., \ell$, Q_0 is indefinite, and the constraint set has nonempty interior. In particular, the problem tested are given in Table 5 for algorithms iALM, IPL, IPL(A), and HPM. For additional clarity, we describe below how the instances were generated and the organization of the tables.

Parameters					**Iteration Count**			
n	r	m	L_f	L_g	iALM	IPL	IPL(A)	HPM
250	1.0	10^0	10^3	7.3	-	2690	**273**	2679
250	1.0	10^0	10^4	9.7	-	2973	**644**	27934
250	1.0	10^0	10^5	12.1	-	3521	**1788**	59381
250	1.0	10^1	10^5	9.7	-	2690	**1717**	60335
250	1.0	10^2	10^5	7.3	-	947	**676**	8206
250	1.0	10^3	10^5	4.8	-	487	**390**	8262
250	5.0	10^0	10^5	12.1	-	13766	**863**	14963
250	10.0	10^0	10^5	12.1	-	27590	**1632**	11390
250	20.0	10^0	10^5	12.1	-	28430	**2694**	10545
500	1.0	10^0	10^3	7.3	-	3834	**332**	2383
500	1.0	10^0	10^4	9.7	-	3287	**659**	26618
500	1.0	10^0	10^5	12.1	-	4316	**2554**	49287
500	1.0	10^1	10^5	9.7	-	3605	**1912**	61336
500	1.0	10^2	10^5	7.3	-	1498	**908**	9221
500	1.0	10^3	10^5	4.8	-	1000	**750**	8659
500	5.0	10^0	10^5	12.1	-	14452	**1075**	13387
500	10.0	10^0	10^5	12.1	-	29301	**1877**	10549
500	20.0	10^0	10^5	12.1	-	91119	**4720**	7311
1000	1.0	10^0	10^3	7.3	-	8862	**679**	16812
1000	1.0	10^0	10^4	9.7	-	4678	**726**	22044
1000	1.0	10^0	10^5	12.1	-	5969	**1825**	42739
1000	1.0	10^1	10^5	9.7	-	5108	**2026**	58180
1000	1.0	10^2	10^5	7.3	-	1018	**594**	142579
1000	1.0	10^3	10^5	4.8	-	1187	**847**	36673
1000	5.0	10^0	10^5	12.1	-	13553	**1491**	17706
1000	10.0	10^0	10^5	12.1	-	26983	**2621**	11514
1000	20.0	10^0	10^5	12.1	-	53820	**5658**	13451

TABLE 5. Iteration counts for the Nonconvex QC-QP Problem in Subsection 4.3. Cells marked with "–" are those that did not obtain a solution within the given time.

First, we chose $\ell = 10$, varied n across different problem instances, and set $\hat{\rho} = \hat{\eta} = 10^{-5}$. Second, the entries of d_0 and c_j for $j = 0, .., \ell$ were generated from the $\mathcal{U}[0,1]$ distribution. On the other hand, the entries of d_j were generated from the $-20 - 10 \cdot \mathcal{U}[0,10]$ distribution, the eigenvectors of

Q_j were taken from the QR decomposition of a random matrix from the $\mathcal{U}[0,1]^{n\times n}$ distribution, the eigenvalues of Q_0 are taken from the $\mathcal{U}[-m_f, L_f]$ distribution for a given $(m_f, L_f) \in \Re^2$, and the eigenvalues of Q_j for $j = 1,..,n$ are taken from the $\log(L_f/m_f)\cdot\mathcal{U}[0,1/3]$ distribution. Third, the initial starting point z_0 was taken from the $\mathcal{U}[-r,r]^{n\times n}$ distribution. Finally, each problem instance considered was based on a specific triple (r, m_f, L_f), that specifies the eigenvalues for Q_0 and the domain of h, a time limit of 3000 seconds, and an iteration limit of 1000000.

Also, for the sake of fairness, we compare HPM against iALM, IPL, and IPL(A) in terms of ACG iteration counts only. This is because: (i) all the tested methods perform ACG iterations that essentially require the same amount of effort; and (ii) there is substantially more computational overhead found in the more general implementations of iALM, IPL, and IPL(A) compared to the more specialized implementation of HPM [10].

4.4. Nonconvex QP Given a pair of dimensions $(\ell, n) \in \mathbb{N}^2$, a scalar pair $(\omega_1, \omega_2) \in \Re^2_{++}$, matrices $Q, C \in \Re^{\ell\times n}$ and $B \in \Re^{n\times n}$, positive diagonal matrix $D \in \Re^{n\times n}$, and a vector pair $(b, d) \in \Re^\ell \times \Re^\ell$, we consider the problem

$$\begin{aligned} \min_z \ & f(z) - \frac{\omega_1}{2}\|DBz\|^2 + \frac{\omega_2}{2}\|\mathcal{C}z - d\|^2 \\ \text{s.t.} \ & Qz = b, \\ & -r \le z_i \le r, \qquad\qquad i \in \{1,...,n\}. \end{aligned}$$

In particular, the problem instances tested are given in Table 6 for algorithms IPL(A), QP(A), SPA1, and SPA2. For additional clarity, we describe below some differences between NL-IAPIAL and S-prox-ALM, as well as how the instances were generated.

We now describe the experiment parameters for the problem instances considered. First, we chose $\ell = 25$, varied n across different problem instances, set $\hat\rho = \hat\eta = 10^{-5}$, and ensured all generated matrices were fully dense. Second, the entries of Q, B, C, and d (resp. D) were generated by sampling from the uniform distribution $\mathcal{U}[0,1]$ (resp. $\mathcal{U}\{1,...,1000\}$), and the vector b was set to $b = Q(u)$ where u is a random vector in $\mathcal{U}[-r,r]^n$. Third, the initial starting point z_0 was a set to be a random vector in $\mathcal{U}[-r,r]^n$. Finally, all experiments were run with a time limit of 3000 seconds, and the tables of this subsection also report the minimum of the aggregate residuals

$$\hat r := \max\left\{ \frac{\mathrm{dist}(0, \nabla f(\hat z) + \partial h(\hat z) + \nabla g(\hat z)\hat p)}{1 + \|\nabla f(z_0)\|}, \frac{\mathrm{dist}(g(\hat z), N_{\mathcal{K}^*}(\hat p))}{1 + \mathrm{dist}(g(z_0), -\mathcal{K})} \right\}. \tag{67}$$

It is worth mentioning that we only report the above residuals in our numerical experiments because it is (computationally) difficult to choose the right parameters in the S-prox-ALM that guarantee convergence (see Section 5 for more details).

[10] More specifically, the implementation of HPM given by authors of [25] takes the problem data $\{Q_j\}_{j=0}^\ell$, $\{c_j\}_{j=0}^\ell$, $\{d_j\}_{j=0}^\ell$, and r as input and directly applies the HiAPeM algorithm instance for QC-QP problems. In contrast, the implementations of iALM, IPL, and IPL(A) take function oracles for f, ∇f, h, g, ∇g, and

$$\mathrm{prox}_{\lambda h}(\cdot) = \underset{u\in\mathrm{dom}\,h}{\mathrm{argmin}}\{\lambda h(u) + \frac{1}{2}\|u - z\|^2\}, \quad \Pi_{\mathcal{K}}(\cdot), \quad \Pi_{\mathcal{K}^*}(\cdot),$$

as input and manipulate these oracles to run their algorithm instances. As executing floating-point operations is substantially less costly than manipulating (symbolic) function oracles, the HPM implementation is drastically more efficient on an iteration-to-iteration basis (roughly 8-10x more) compared to the iALM, IPL, and IPL(A) implementations, at the cost of a less general-purpose API.

Parameters				Iteration Count					Residual $\hat{r}$/Runtime				
n	r	m	L_f	iALM	QP(A)	IPL(A)	SPA1	SPA2	iALM	QP(A)	IPL(A)	SPA1	SPA2
250	1.0	10^0	10^3	111250	53625	**23000**	-	-	-/894	-/403	**-/177**	3E-04/-	2E-03/-
250	1.0	10^0	10^4	103710	60997	**50195**	-	-	-/1009	-/541	**-/452**	3E-04/-	3E-04/-
250	1.0	10^0	10^5	58049	38963	**30024**	-	-	-/406	-/255	**-/199**	2E-05/-	2E-05/-
250	1.0	10^1	10^5	103800	60851	**50195**	-	-	-/550	-/344	**-/284**	2E-04/-	2E-04/-
250	1.0	10^2	10^5	130970	49208	**20775**	-	-	-/695	-/277	**-/119**	4E-04/-	3E-04/-
250	1.0	10^3	10^5	427430	279680	**16146**	269460	256820	-/2257	-/1609	**-/96**	-/1860	-/1771
250	5.0	10^0	10^5	52603	40483	**33431**	-	-	-/277	-/228	**-/187**	2E-05/-	2E-05/-
250	10.0	10^0	10^5	67225	41561	**33706**	-	-	-/355	-/233	**-/190**	2E-05/-	2E-05/-
250	20.0	10^0	10^5	57393	41786	**34756**	-	-	-/302	-/234	**-/195**	2E-05/-	2E-05/-
500	1.0	10^0	10^3	-	-	**35529**	-	-	8E-04/-	6E-02/-	**-/677**	5E-03/-	5E-03/-
500	1.0	10^0	10^4	-	67928	**48991**	-	-	5E-03/-	-/1103	**-/807**	6E-04/-	5E-04/-
500	1.0	10^0	10^5	69861	49650	**35549**	-	-	-/1491	-/789	**-/568**	4E-04/-	4E-05/-
500	1.0	10^1	10^5	-	67875	**48991**	-	-	7E-03/-	-/1089	**-/801**	2E-03/-	6E-04/-
500	1.0	10^2	10^5	-	123980	**24988**	-	-	7E-02/-	-/2009	**-/425**	1E-03/-	1E-03/-
500	1.0	10^3	10^5	-	-	**67534**	-	-	1E+00/-	6E-01/-	**-/1185**	1E-03/-	5E-04/-
500	5.0	10^0	10^5	68644	50567	**35274**	-	-	-/1441	-/791	**-/556**	5E-04/-	3E-05/-
500	10.0	10^0	10^5	73137	50497	**35396**	-	-	-/1566	-/794	**-/559**	3E-04/-	3E-05/-
500	20.0	10^0	10^5	79126	50586	**35242**	-	-	-/1599	-/760	**-/534**	2E-04/-	3E-05/-
1000	1.0	10^0	10^3	-	-	**30340**	-	-	6E-03/-	3E-02/-	**-/2868**	2E-03/-	6E-03/-
1000	1.0	10^0	10^4	-	27184	**16540**	-	-	4E-03/-	-/2250	**-/1380**	1E-04/-	1E-04/-
1000	1.0	10^0	10^5	-	35192	**27672**	-	-	4E-04/-	-/2952	**-/2515**	3E-02/-	2E-05/-
1000	1.0	10^1	10^5	-	27217	**16540**	-	-	4E-03/-	-/2298	**-/1411**	3E-02/-	1E-04/-
1000	1.0	10^2	10^5	-	-	**16129**	-	-	4E-02/-	3E-02/-	**-/1461**	2E-02/-	3E-03/-
1000	1.0	10^3	10^5	-	-	**11325**	-	-	3E-01/-	2E-01/-	**-/1155**	7E-03/-	3E-03/-
1000	5.0	10^0	10^5	-	35564	**27810**	-	-	4E-04/-	-/2986	**-/2340**	3E-02/-	2E-05/-
1000	10.0	10^0	10^5	-	35515	**27973**	-	-	4E-04/-	-/2983	**-/2354**	3E-02/-	2E-05/-
1000	20.0	10^0	10^5	-	-	**28033**	-	-	4E-04/-	7E-06/-	**-/2358**	3E-02/-	2E-05/-

TABLE 6. Iteration counts, runtimes, and residuals (see (67)) for the Nonconvex QP Problem in Subsection 4.4. Entries marked with "–" are those that either: (i) obtained a solution with a residual below the prescribed tolerance; or (ii) did not obtain a solution within the given time limit.

4.5. Comments about the numerical results Overall, the most efficient methods for the above experiments were the NL-IAPIAL variants (IPL and IPL(A)). IPL(A) performed particularly well on the linearly-constrained instances where the ratio L_f/m was relatively small. Between the two NL-IAPIAL variants, IPL(A) is substantially more efficient. In the QC-QP experiments, we also noticed that the results of IPL variants did not fluctuate as much as the ones of HiAPeM across different problem instances.

We conjecture that IPL and IPL(A) perform significantly better than HiAPeM and iALM on some instances because they apply their multiplier updates more often.

5. Concluding Remarks We first discuss how the n-PAL methods and PAL methods described in the *Overview of AL methods* part Section 1 above compare to one another. First, the subproblems generated by the n-PAL methods can be nonconvex whereas the ones generated by the PAL methods are always strongly convex. Second, some n-PAL algorithms compute the approximate stationary point z_k of $\mathcal{L}_{\beta_k}(\cdot; p_{k-1})$ by using prox-type methods that generate a sequence of convex subproblems similar to those of the PAL methods. Hence, the subproblems generated by the n-PAL methods are generally much harder to solve than those generated by the PAL methods.

We now give a detailed comparison of NL-IAPIAL with the HiAPeM of [25]. Both methods employ an ACG-type subroutine to inexactly solve a generated sequence of strongly convex proximal subproblems. Using nearly the same assumptions as in this paper and denoting $\varepsilon = \min\{\hat{\rho}, \hat{\eta}\}$, [25] establishes an improved $\mathcal{O}(\varepsilon^{-2.5}\log\varepsilon^{-1})$ ACG iteration complexity of HiAPeM starting from any point in $\mathrm{dom}\, h$ for problems where $\mathcal{K} = \{0\} \times \Re^n_+$. However, as noted in the *Related works* part of Section 1, HiAPeM is neither a PAL method (like NL-IAPIAL), nor an n-PAL method (like the iALM of [24]), but rather an inexact PPM applied to nonconvex problem (1) (see, for example, [36] for the analysis of inexact PPMs for solving (1) in the convex setting). Loosely speaking, for some suitable prox stepsize $\lambda > 0$, its k-th prox iteration computes an approximate stationary point z_k of the strongly convex subproblem $\min_z\{\lambda\phi(z) + \|z - z_{k-1}\|^2/2 : g(z) \preceq_{\mathcal{K}} 0\}$ by using either an accelerated penalty method or an accelerated AL method. It is worth mentioning that in the case where f is convex, solving the k-th subproblems corresponds to inexactly solving

$$\partial_z\mathcal{L}_0(z;p) + \frac{1}{\lambda_k}(z - z_{k-1}) \ni 0, \quad -\partial_p\mathcal{L}_0(z;p) \ni 0,$$

for $(z,p) = (z_k, p_k)$ (cf. (8) and (10)).

We next compare NL-IAPIAL with the S-prox-ALM of [43], which is neither a PAL nor n-PAL method, but is based on the augmented Lagrangian function and performs multiplier updates similar to the ones in PAL or n-PAL methods. First, it is shown in [43] that S-prox-ALM has an $\mathcal{O}(\varepsilon^{-2})$ iteration complexity under the assumption that g is affine and the strong assumption that the function h in (1) is the indicator function of a polyhedron. Second, S-prox-ALM generates a sequence of proximal subproblems as in (3), but applies a single composite gradient step to inexactly solve a variant[11] of (3) instead of an ACG-type subroutine. Finally, while the NL-IAPIAL method only requires choosing its parameters based on the scalars m_f, L_f, L_g, and M_g to guarantee convergence, the S-prox-ALM requires choosing its parameters based on the supremum of a set of Hoffman constants (see the proof of [43, Lemma 3.10] and [43, Lemma 4.8]) that is generally difficult to compute and compare with the other constants of NL-IAPIAL.

Finally, it is worth mentioning that NL-IAPIAL is a slightly modified version of the proximal method of multipliers (PMM) studied by Rockafellar in [37]. More specifically, the k-th iteration of the PMM consists of (3)–(4) with $\mathcal{K} = \Re^\ell_+$ and $\lambda_k = \beta_k$ for every k and, hence, can be viewed as inexactly solving (8) with $\lambda_k = \beta_k$ and $\chi_k = 1$ so that both inclusions on it have the same prox stepsize. Under the assumption that (1) is a convex optimization problem, Rockafellar then uses classical results for inexact proximal point methods to analyze the convergence of the PMM. However, the approach outlined above does not generalize to the nonconvex setting in several aspects, namely: (i) while the PMM converges when β_k is constant, convergence of NL-IAPIAL requires β_k to grow significantly; (ii) in contrast to the PMM, NL-IAPIAL chooses λ_k to be a sufficiently small constant to convexify the subproblem in (3); and (iii) the analysis of NL-IAPIAL does not rely on proximal point theory for maximal monotone operators since the operator $(z,p) \mapsto [\partial_z\mathcal{L}_0(z;p), -\partial_p\mathcal{L}_0(z;p)]$ is not monotone in the setting of NL-IAPIAL.

Appendix A Review of an ACG Algorithm This section reviews an ACG algorithm invoked by NL-IAPIAL for solving the sequence of subproblems (3) which arise during its implementation. It also describes a bound on the number of ACG iterations performed in order to obtain a certain type of inexact solution of each subproblem.

Consider the composite optimization problem

[11] Instead of inexactly minimizing the function $\lambda\mathcal{L}(\cdot; p_{k-1}) + \|\cdot - z_{k-1}\|^2/2$, the S-prox-ALM exactly minimizes the linear approximation of the function $\lambda\mathcal{L}(\cdot; p_{k-1}) + \|z - \tilde{z}_{k-1}\|/2$ for a point $\tilde{z}_{k-1}$ different from z_{k-1}. Hence, S-prox-ALM is neither a PAL method nor an n-PAL method.

$$\min \{\psi(x) := \psi_s(x) + \psi_n(x) : x \in \Re^n\}, \tag{68}$$

where the following conditions are assumed to hold: where the following conditions are assumed to hold:

(B1) $\psi_n : \Re^n \to (-\infty, +\infty]$ is a proper closed convex function;

(B2) ψ_s is a convex differentiable function on $\operatorname{dom}\psi_n$ and there exists $(\widetilde{\mu}, \widetilde{M}) \in \Re^2_+$ satisfying $\widetilde{M} > \widetilde{\mu}$ and

$$\widetilde{\mu}\|u - x\|^2/2 \le \psi_s(u) - \ell_{\psi_s}(u; x) \le \widetilde{M}\|u - x\|^2/2 \tag{69}$$

for every $x, u \in \operatorname{dom}\psi_n$, where $\ell_{\psi_s}(\cdot;\cdot)$ is defined in (12).

The ACG algorithm, given $(y_0, \tilde{\sigma}) \in \operatorname{dom}\psi_n \times \Re_{++}$, inexactly solves (68) by computing a triple $(y, u, \eta) \in \operatorname{dom}\psi_n \times \Re^n \times \Re_+$ satisfying

$$u \in \partial_\eta(\psi_s + \psi_n)(y) \quad \|u\|^2 + 2\eta \le \tilde{\sigma}^2\|y_0 - y + u\|^2. \tag{70}$$

With this in mind, we now state the ACG variant considered in this paper.

ACG

---

(0) Let a pair of functions (ψ_s, ψ_n) satisfying **(B1)** and **(B2)** for some $(\widetilde{\mu}, \widetilde{M}) \in \Re^2_+$, a scalar $\tilde{\sigma} > 0$, and an initial point $y_0 \in \operatorname{dom}\psi_n$ be given; set $x_0 = y_0$, $A_0 = 0$, $\tau_0 = 1$, and $j = 0$;

(1) $\zeta = 1/(\widetilde{M} - \widetilde{\mu})$ and compute the quantities

$$\begin{aligned} a_{j+1} &= \frac{\zeta\tau_j + \sqrt{(\zeta\tau_j)^2 + 4\tau_j A_j}}{2}, \quad A_{j+1} = A_j + a_{j+1}, \quad \tilde{x}_{j+1} = \frac{A_j y_j + a_{j+1} x_j}{A_{j+1}} \\ \tau_{j+1} &= \tau_j + \widetilde{\mu} a_{j+1}, \quad y_{j+1} = \operatorname*{argmin}_{y \in \Re^n} \left\{ \ell_{\psi_s}(y; \tilde{x}_{j+1}) + \psi_n(y) + \frac{\widetilde{M}}{2}\|y - \tilde{x}_{j+1}\|^2 \right\}, \\ x_{j+1} &= \frac{1}{\tau_{j+1}}\left[\frac{a_{j+1}}{\zeta}(y_{j+1} - \tilde{x}_{j+1}) + \widetilde{\mu} a_{j+1} y_{j+1} + \tau_j x_j\right]; \end{aligned} \tag{71}$$

(2) compute the quantities

$$\begin{aligned} u_{j+1} &= \widetilde{\mu}(y_{j+1} - x_{j+1}) + \frac{x_0 - x_{j+1}}{A_{j+1}}, \\ \eta_{j+1} &= \frac{1}{2A_{j+1}}\left(\|x_0 - y_{j+1}\|^2 - \tau_{j+1}\|x_{j+1} - y_{j+1}\|^2\right); \end{aligned}$$

(3) if the inequality

$$\|u_{j+1}\|^2 + 2\eta_{j+1} \le \tilde{\sigma}^2\|y_0 - y_{j+1} + u_{j+1}\|^2$$

holds, then stop and output $(y, u, \eta) := (y_{j+1}, u_{j+1}, \eta_{j+1})$; otherwise, set $j = j + 1$ and go to (1).

Some remarks about ACG follow. First, the most common way of describing an iteration of ACG is as in step 1. Second, the auxiliary iterates pair $\{(u_j, \eta_j)\}$ computed in step 2 is used to develop a stopping criterion for ACG when it is called as a subroutine for solving the subproblems generated in step 1 of NL-IAPIAL in Subsection 2.2. Third, it can be shown (see for example [10, 20]) that ACG (without steps 2 and 3) with $\widetilde{\mu} = 0$ corresponds to the well-known FISTA algorithm. Fourth, the sequence $\{A_j\}$ has the following increasing property:

$$A_j \ge \frac{1}{\widetilde{M} - \widetilde{\mu}} \max\left\{ \frac{j^2}{4}, \left(1 + \sqrt{\frac{\widetilde{\mu}}{4(\widetilde{M} - \widetilde{\mu})}}\right)^{2(j-1)} \right\}, \qquad \forall j \ge 1.$$

Finally, notice that each iteration of an ACG-type method consists of an $\mathcal{O}(1)$ number of ψ_s function, ψ_s gradient, and ψ_n prox evaluations.

It is worth mentioning that adaptive variants[12] of ACG have been studied, for example, in [4, 16, 27, 32, 34]. One kind of adaptiveness used in these variants, which is also used inside some methods benchmarked in Section 4, involves replacing $\widetilde{M}$ in the computation of y_{j+1} in step 1 by an estimate M_{j+1} computed as follows: M_{j+1} is initially set to be M_j and, if necessary, is increased (either additively, multiplicatively, or both) and step 1 is repeated a few times (if needed) until the inequality $\psi_s(y_{j+1}) - \ell_{\psi_s}(y_{j+1}; \tilde{x}_{j+1}) \le M_{j+1}\|y_{j+1} - \tilde{x}_{j+1}\|^2/2$ is satisfied. Observe that every time step 1 is repeated within the j-th iteration of ACG, ζ changes (and hence so do a_{j+1}, A_{j+1}, $\tilde{x}_{j+1}$, τ_{j+1}, and y_{j+1}) since $M_{j+1} = \tilde{M}$ changes adaptively.

The next result, whose proof can be found in [20, Lemma 2.13], summarizes the main properties of the above ACG.

Proposition A.1. *Let $\{(y_j, u_j, \eta_j)\}_{j\ge 1}$ be the sequence generated by ACG applied to (68), where (ψ_s, ψ_n) is a given pair of data functions satisfying* **(B1)** *and* **(B2)***. Then, the following statements hold:*

(a) for every $j \ge 1$, we have $\eta_j \ge 0$ and $u_j \in \partial_{\eta_j}(\psi_s + \psi_n)(y_j)$;

(b) for any $\tilde{\sigma} > 0$, the ACG method outputs a triple $(y, u, \eta) \in \operatorname{dom}\psi_n \times \Re^n \times \Re_+$ satisfying

$$u \in \partial_\eta(\psi_s + \psi_n)(y) \quad \|u\|^2 + 2\eta \le \tilde{\sigma}^2\|y_0 - y + u\|^2 \tag{72}$$

in at most

$$\left\lceil 1 + \left(\frac{1}{2} + \sqrt{\frac{\widetilde{M} - \widetilde{\mu}}{\widetilde{\mu}}}\right) \log_1^+ \widetilde{\mathcal{A}} \right\rceil \tag{73}$$

iterations, where

$$\widetilde{\mathcal{A}} := (2\widetilde{\mu} + 3)(1 + \widetilde{\sigma})^2(\widetilde{M} - \widetilde{\mu})\widetilde{\sigma}^{-2}.$$

Appendix B Convex Analysis The first result presents some well-known properties about the projection and distance functions over a closed convex set.

Lemma B.1. *Let $\mathcal{K} \subseteq \Re^n$ be a nonempty closed convex cone and S be a nonempty closed convex set. Then the following properties hold:*

(a) for every $u, z \in \Re^n$, we have $\|\Pi_S(u) - \Pi_S(u)\| \le \|u - z\|$;

(b) the function $d(\cdot) := \operatorname{dist}^2(\cdot, S)/2$ is differentiable, and its gradient is given by

$$\nabla d(u) = u - \Pi_S(u) \in N_S(\Pi_S(u)) \quad \forall u \in \Re^n; \tag{74}$$

(c) it holds that $u \in N_{\mathcal{K}^*}(p)$ if and only if $\langle u, p\rangle = 0$, $u \in -\mathcal{K}$, and $p \in \mathcal{K}^*$.

Proof. See [3, Theorem 5.4] for (a), [3, Example 6.61] and [3, Theorem 6.39(ii)] for (b), and [38, Example 11.4] for (c). □

The next result presents a well-known fact (see, for example, [8, Sub-subsection 2.13.2]) about closed convex cones.

Lemma B.2. *For any closed convex cone $\mathcal{K}$, we have that $x \in \operatorname{int}\mathcal{K}$ if and only if*

$$\langle x, p\rangle > 0 \quad \forall p \in \mathcal{K}^* \quad \text{such that} \quad \|p\| = 1. \tag{75}$$

The below technical result presents a fact about approximate subdifferentials, and its proof can be found, for example, in [30, Lemma A.3].

[12] The closest variant to ACG in this paper can be found in [16, Section 5.2].

Lemma B.3. *Let a proper function $\tilde{\phi}:\Re^n \to (-\infty,\infty]$, scalar $\tilde{\sigma}\in(0,1)$ and $(x_0,x)\in\Re^n\times \mathrm{dom}\,\tilde{\phi}$ be given, and assume that there exists (v,ε) such that*

$$v\in\partial_\varepsilon\left(\tilde{\phi}+\frac{1}{2}\|\cdot-x_0\|^2\right)(x),\quad \|v\|^2+2\varepsilon\le\tilde{\sigma}^2\|v+x_0-x\|^2. \tag{76}$$

Then, for every $x\in\Re^n$ and $s>0$, we have

$$\tilde{\phi}(x)+\frac{1}{2}\left[1-\tilde{\sigma}^2(1+s^{-1})\right]\|v+x_0-x\|^2\le\tilde{\phi}(z)+\frac{s+1}{2}\|z-x_0\|^2.$$

Appendix C Proof of Lemma 3.4 and Lemma 3.7(a) The first result, whose proof is given in [17, Appendix A], describes some properties of a composite gradient step.

Lemma C.1. *Assume that $\tilde{h}\in\overline{\mathrm{Conv}}\,\Re^n$, $\tilde{g}$ is a differentiable function on $\mathrm{dom}\,\tilde{h}$, and $(z,\varepsilon)\in\mathrm{dom}\,\tilde{h}\times\Re_+$ is such that*

$$0\in\partial_\varepsilon(\tilde{g}+\tilde{h})(z). \tag{77}$$

Assume also that there exists $\tilde{L}>0$ such that

$$\tilde{g}(u)-\ell_{\tilde{g}}(u;z)\le\frac{\tilde{L}}{2}\|u-z\|^2\qquad\forall u\in\mathrm{dom}\tilde{h}, \tag{78}$$

and define

$$\tilde{z}:=\operatorname*{argmin}_u\left\{\ell_{\tilde{g}}(u;z)+\tilde{h}(u)+\frac{\tilde{L}}{2}\|u-z\|^2\right\},\qquad \widetilde{w}:=\tilde{L}(z-\tilde{z}). \tag{79}$$

Then, the quadruple $(z,\tilde{z},\widetilde{w},\varepsilon)$ satisfies

$$\widetilde{w}\in\nabla\tilde{g}(z)+\partial\tilde{h}(\tilde{z}),\qquad \widetilde{w}\in\nabla\tilde{g}(z)+\partial_\varepsilon\tilde{h}(z),\qquad \|\widetilde{w}\|\le\sqrt{2\tilde{L}\varepsilon}. \tag{80}$$

The next result specializes the above results to our setting and gives two technical identities.

Lemma C.2. *Let $\widetilde{\mathcal{L}}_\beta$ be as in (22), let β_k, (z_k,v_k,ε_k), $\hat{z}_k$, and (z_{k-1},p_{k-1}) be as in the k-th iteration of NL-IAPIAL, and define*

$$\tilde{g}:=\lambda\widetilde{\mathcal{L}}_{\beta_k}(\cdot;p_{k-1})-\langle v_k,\cdot\rangle+\frac{1}{2}\|\cdot-z_{k-1}\|^2,\quad \tilde{h}:=\lambda h,\quad \tilde{w}_k:=\widetilde{\mathcal{M}}_k(z_k-\hat{z}_k) \tag{81}$$

Then, it holds that

$$\tilde{w}_k\in\nabla\tilde{g}(z_k)+\partial\tilde{h}(\hat{z}_k),\quad \tilde{w}_k\in\nabla\tilde{g}(z_k)+\partial_{\varepsilon_k}\tilde{h}(z_k),\quad \|\tilde{w}_k\|\le\sqrt{2\varepsilon_k\widetilde{\mathcal{M}}_k}. \tag{82}$$

where $\widetilde{\mathcal{M}}_k$ is as in (26). Moreover, it holds that

$$\frac{1}{\lambda}\left(r_k+\nabla\tilde{g}(z_k)\right)=\nabla_z\widetilde{\mathcal{L}}_{\beta_k}(z_k;p_{k-1})=\nabla f(z_k)+\nabla g(z_k)\Pi_{\mathcal{K}^*}(p_{k-1}+\beta_k g(z_k))\quad\forall u\in\Re^n. \tag{83}$$

Proof. It follows from the definition of ε-subdifferential in (11) and the fact that the triple (z_k,v_k,ε_k) satisfies the inclusion in (25) that (77) holds with $(\tilde{g},\tilde{h})$ and $(z,\varepsilon)=(z_k,\varepsilon_k)$. In view of assumptions (A1)–(A3), Lemma 2.3, and the definition of $\widetilde{\mathcal{M}}_k$ in (26), the functions pair $(\tilde{g},\tilde{h})$ defined above satisfies the assumptions of Lemma C.1 with $\tilde{L}=\widetilde{\mathcal{M}}_k$. Note also that the element $\tilde{z}$ computed according to (79) corresponds to $\hat{z}_k$ computed in (32), in view of the definition of r_k given in (31). Hence, it follows from Lemma C.1 that (82) holds. The last statement of the lemma follows from the definition of r_k in (31) and Lemma 2.3(b). □

We are now ready to prove Lemma 3.7(a).

Proof of Lemma 3.7(a). Let $\tilde{h}$ be as in (31). In view of (11), the definitions of p_k and w_k in (31) and (34), respectively, and Lemma C.2, we have

$$\begin{aligned} w_k = \frac{1}{\lambda}\left(r_k + \widetilde{\mathcal{M}}_k(z_k - \hat{z}_k)\right) &\in \frac{1}{\lambda}\left(r_k + \nabla \tilde{g}(z_k) + \partial_{\varepsilon_k}\tilde{h}(z_k)\right) \\ &= \nabla f(z_k) + \nabla g(z_k)\Pi_{\mathcal{K}^*}(p_{k-1} + \beta_k g(z_k)) + \partial_{(\lambda^{-1}\varepsilon_k)} h(z_k) \\ &= \nabla f(z_k) + \nabla g(z_k)p_k + \partial_{(\lambda^{-1}\varepsilon_k)} h(z_k), \end{aligned}$$

which proves the inclusion in (62). We now show that the inequalities in (62) hold. The bound on ε_k in (62) follows immediately from the inequality in (25) and the definition of r_k given in (34). Now, it follows from the inequality in (25), the definition of r_k and w_k in (31) and (34), respectively, the triangle inequality for norms, and Lemma C.2 that

$$\begin{aligned} \lambda\|w_k\| = \|r_k + \widetilde{\mathcal{M}}_k(z_k - \hat{z}_k)\| &\le \|r_k\| + \widetilde{\mathcal{M}}_k\|z_k - \hat{z}_k\| \\ &\le \|r_k\| + \sqrt{2\varepsilon_k\widetilde{\mathcal{M}}_k} \le \left(1 + \sigma_k\sqrt{\widetilde{\mathcal{M}}_k}\right)\|r_k\|, \end{aligned} \tag{84}$$

which immediately implies the desired bound on $\|w_k\|$ in view of the definition of σ_k in (26). □

We now close with the proof of Lemma 3.4.

Proof of Lemma 3.4.

We first show that the inclusion in (54) holds. Using the first identity in (83), Lemma C.2, Lemma 2.3(b), and the definitions of w_k and $(\hat{w}_k, \hat{p}_k)$ in (34) and (33), respectively, we have

$$\begin{aligned} \hat{w}_k &= \frac{1}{\lambda}\left[r_k + \widetilde{\mathcal{M}}_k(z_k - \hat{z}_k)\right] + \left[\nabla_z\widetilde{\mathcal{L}}_{\beta_k}(\hat{z}_k; p_{k-1}) - \nabla_z\widetilde{\mathcal{L}}_{\beta_k}(z_k; p_{k-1})\right] \\ &\in \frac{1}{\lambda}\left[r_k + \nabla\tilde{g}(z_k) + \partial\tilde{h}(\hat{z}_k)\right] + \left[\nabla_z\widetilde{\mathcal{L}}_{\beta_k}(\hat{z}_k; p_{k-1}) - \nabla_z\widetilde{\mathcal{L}}_{\beta_k}(z_k; p_{k-1})\right] \\ &= \nabla_z\widetilde{\mathcal{L}}_{\beta_k}(\hat{z}_k; p_{k-1}) + \partial h(\hat{z}_k) = \nabla f(\hat{z}_k) + \nabla g(\hat{z}_k)\Pi_{\mathcal{K}^*}(p_{k-1} + \beta_k g(\hat{z}_k)) + \partial h(\hat{z}_k) \\ &= \nabla f(\hat{z}_k) + \nabla g(\hat{z}_k)\hat{p}_k + \partial h(\hat{z}_k), \end{aligned}$$

which is the desired inclusion in (54). We now show that the bound on $\|\hat{w}_k\|$ in (55) holds. Using its definition in (33), Lemma 2.3(c) and the definition of $\widetilde{\mathcal{M}}_k$ in (26), the inequality in (25), the definition of r_k given in (34), Lemma C.2, the triangle inequality for norms, and (84), we have

$$\begin{aligned} \lambda\|\hat{w}_k\| &\le \lambda\|w_k\| + \lambda\|\nabla_z\widetilde{\mathcal{L}}_{\beta_k}(\hat{z}_k; p_{k-1}) - \nabla_z\widetilde{\mathcal{L}}_{\beta_k}(z_k; p_{k-1})\| \\ &\le \left(1 + \sigma_k\sqrt{\widetilde{\mathcal{M}}_k}\right)\|r_k\| + \widetilde{\mathcal{M}}_k\|\hat{z}_k - z_k\| \le \left(1 + 2\sigma_k\sqrt{\widetilde{\mathcal{M}}_k}\right)\|r_k\|, \end{aligned}$$

which immediately implies the desired bound on $\|\hat{w}_k\|$ in view of the definition of σ_k in (26).

To show the bound on $\hat{q}_k$, we first use the definitions of $B_g^{(1)}$, p_k, and $\hat{p}_k$ given in (23), (31), and (33), respectively, the last two inequalities in (84), the Mean Value Inequality, and Lemma B.1(a) to obtain

$$\begin{aligned} \frac{1}{\beta_k}\|\hat{p}_k - p_k\| &= \frac{1}{\beta_k}\|\Pi_{\mathcal{K}^*}(p_{k-1} + \beta_k g(\hat{z}_k)) - \Pi_{\mathcal{K}^*}(p_{k-1} + \beta_k g(z_k))\| \le \frac{1}{\beta_k}\|\beta_k g(\hat{z}_k) - \beta_k g(z_k)\| \\ &\le \sup_{t\in[0,1]}\|\nabla g(t\hat{z}_k + [1-t]z_k)\| \cdot \|\hat{z}_k - z_k\| \le B_g^{(1)}\|\hat{z}_k - z_k\| \le \frac{B_g^{(1)}\sigma_k}{\sqrt{\widetilde{\mathcal{M}}_k}}\|r_k\|. \end{aligned}$$

Hence, using the triangle inequality for norms and the definition of $\hat{q}_k$ given in (33), we have

$$\|\hat{q}_k\| = \frac{1}{\beta_k}\|\hat{p}_k - p_{k-1}\| \le \frac{1}{\beta_k}\|\hat{p}_k - p_k\| + \frac{1}{\beta_k}\|p_k - p_{k-1}\| \le \frac{B_g^{(1)}\sigma_k}{\sqrt{\widetilde{\mathcal{M}}_k}}\|r_k\| + \frac{1}{\beta_k}\|p_k - p_{k-1}\|,$$

which proves the bound on $\hat{q}_k$ in view of the definition of σ_k in (26).

To finish the proof of Lemma 3.4, it remains to show that the last three relations in (54) hold. The last relation in (54) follows immediately from the definition of $\hat{p}_k$ in (33). Now, using Lemma B.1(b) with $S = \mathcal{K}^*$ and $u = p_{k-1} + \beta_k g(\hat{z}_k)$ as well as the definitions of $\hat{q}_k$ and $\hat{p}_k$ in (33), we have that

$$g(\hat{z}_k) + \hat{q}_k = \frac{1}{\beta_k}\left[p_{k-1} + \beta_k g(\hat{z}_k) - \hat{p}_k\right] \in N_{\mathcal{K}^*}(\hat{p}_k). \tag{85}$$

Hence, the remaining relations in (54) follow from the above relation and Lemma B.1(c) with $u = g(\hat{z}_k) + \hat{q}_k$ and $p = \hat{p}_k$. □

Appendix D Proof of Proposition 2.1

(a) $\Longrightarrow$ (b). This is immediate.

[(b) $\Longrightarrow$ (c)] Suppose (b) holds. If $\bar{z}$ satisfies (c) then we are done, so suppose that $g_\iota(\bar{z}) \not\prec_{\mathcal{J}} 0$ and $g_e(\bar{z}) = 0$. Our goal is to find $d \in \Re^n$ such that (c) holds with $\bar{z} = \bar{z} + d$, which in view of Lemma B.2 with $x = -g_\iota(\bar{z} + d)$ and the fact that g_e is affine, is equivalent to

$$g_e'(\bar{z})d = 0, \qquad \inf_{\|p_\iota\|=1, p_\iota \in \mathcal{J}^*} \langle -g_\iota(\bar{z} + d), p_\iota \rangle > 0. \tag{86}$$

We now bound the left-hand-side of the inequality in (86). Using the assumption that $\nabla g_\iota(\cdot)$ is L_{g_ι}-Lipschitz, we have

$$\begin{aligned}
\inf_{\|p_\iota\|=1, p_\iota \in \mathcal{J}^*} - \langle g_\iota(\bar{z} + d), p_\iota \rangle &= \inf_{\|p_\iota\|=1, p_\iota \in \mathcal{J}^*} - \left\langle g_\iota(\bar{z}) + g_\iota'(\bar{z})d + \left[g_\iota(\bar{z} + d) - g_\iota(\bar{z}) - g_\iota'(\bar{z})d\right], p_\iota \right\rangle \\
&\geq \inf_{\|p_\iota\|=1, p_\iota \in \mathcal{J}^*} \langle -g_\iota(\bar{z}) - g_\iota'(\bar{z})d, p_\iota \rangle - \|g_\iota(\bar{z} + d) - g_\iota(\bar{z}) - g_\iota'(\bar{z})d\| \\
&\geq \inf_{\|p_\iota\|=1, p_\iota \in \mathcal{J}^*} \langle -g_\iota(\bar{z}) - g_\iota'(\bar{z})d, p_\iota \rangle - \frac{L_{g_\iota}\|d\|^2}{2},
\end{aligned} \tag{87}$$

for any $d \in \Re^n$, so it suffices to find $d \in \Re^n$ so that the last expression in (87) is positive. To find an appropriate direction, we let $0 \neq q_\iota \in \operatorname{int}\mathcal{J}$ and consider the primal-dual conic optimization problems

$$\left(\underbrace{\begin{array}{ll} \min\limits_{p} & -\langle p_\iota, g_\iota(\bar{z}) \rangle \\ \text{s.t.} & \nabla g_\iota(\bar{z})p_\iota + \nabla g_e(\bar{z})p_e = 0 \\ & \langle q_\iota, p_\iota \rangle = 1 \\ & p_\iota \in \mathcal{K}^*, p_e \in \Re^t \end{array}}_{(P)}\right) \equiv \left(\underbrace{\begin{array}{ll} \max\limits_{d,\mu} & \mu \\ \text{s.t.} & -g_\iota(\bar{z}) - g_\iota'(\bar{z})d \succeq_{\mathcal{J}} \mu q_\iota \\ & g_e'(\bar{z})d = 0 \\ & d \in \Re^n, \mu \in \Re \end{array}}_{(D)}\right). \tag{88}$$

Denoting p_ι^* and (d^*, μ^*) to be optimal solutions of (P) and (D), respectively, we show that μ^* is positive and then argue that d^* is an appropriate direction. Using the fact that (D) has a Slater point (and hence strong duality holds for (88)), our assumption that $-g(\bar{z}) \in \mathcal{K}$ (and hence $-\langle p^*, g(\bar{z}) \rangle \geq 0$), and (19), it follows that

$$\mu^* = -\langle p_\iota^*, g_\iota(\bar{z}) \rangle = \max\left\{ \left| \left\langle \begin{bmatrix} p_\iota^* \\ 0 \end{bmatrix}, g(\bar{z}) \right\rangle \right|, \left\| \nabla g(\bar{z}) \begin{bmatrix} p_\iota^* \\ 0 \end{bmatrix} \right\| \right\} \geq \tilde{\tau}_g \|p_\iota^*\| > 0, \tag{89}$$

where the last inequality follows from the second constraint in (P), the fact that $q_\iota \in \operatorname{int}\mathcal{J}$, and Lemma B.2 with $(p, x) = (p_\iota^*, q_\iota)$. Since $g_e'(\bar{z})d^* = 0$ from the second constraint of (D), it only remains to show that the last expression in (87) is positive for some positive multiple of d^*, i.e., $d = \lambda d^*$ for some $\lambda > 0$. Using the fact that d^* is feasible to (D) and our assumption that $g_\iota(\bar{z}) \preceq_{\mathcal{J}} 0$ (and hence $-\langle p_\iota, g(\bar{z})\rangle \geq 0$ for every $p_\iota \in \mathcal{J}^*$), we first have that for $\lambda < 1$ and $d = \lambda d^*$,

$$\begin{aligned}
&\inf_{\|p_\iota\|=1, p_\iota\in\mathcal{J}^*} \langle -g_\iota(\bar{z}) - g_\iota'(\bar{z})d, p_\iota\rangle - \frac{L_{g_\iota}\|d\|^2}{2} \\
&= \lambda\left[\inf_{\|p_\iota\|=1, p_\iota\in\mathcal{J}^*} \left\langle -\frac{1}{\lambda} g_\iota(\bar{z}) - g_\iota'(\bar{z})d^*, p_\iota\right\rangle - \frac{\lambda L_{g_\iota}\|d^*\|^2}{2}\right] \\
&\geq \lambda\left[\inf_{\|p_\iota\|=1, p_\iota\in\mathcal{J}^*} \langle -g_\iota(\bar{z}) - g_\iota'(\bar{z})d^*, p_\iota\rangle - \frac{\lambda L_{g_\iota}\|d^*\|^2}{2}\right] \\
&\geq \lambda\left[\mu^* \inf_{\|p_\iota\|=1, p_\iota\in\mathcal{J}^*} \langle q_\iota, p_\iota\rangle - \frac{\lambda L_{g_\iota}\|d^*\|^2}{2}\right] \\
&= \lambda\left[\mu^*\nu - \frac{\lambda L_{g_\iota}\|d^*\|^2}{2}\right],
\end{aligned} \tag{90}$$

where $\nu := \inf_{\|p_\iota\|=1, p_\iota\in\mathcal{J}^*}\langle q_\iota, p_\iota\rangle$. Using (89) and Lemma B.2 with $(p, x) = (p_\iota, q_\iota)$, it holds that $\mu^*\nu > 0$ and, hence, there exists $\lambda > 0$ sufficiently small so that the last expression in (90) is positive. As a consequence, it follows from (87) that (86) holds, or equivalently, (c) holds with $\bar{z} = \bar{z} + \lambda d^*$.

[(c) $\implies$ (a)] Suppose (c) holds. Since g_e is affine and onto, its gradient matrix $G_e := \nabla g_e$ is independent of z and has full column rank. Hence, there exists $\tau_{g_e} > 0$ such that

$$\|G_e p_e\| \geq \tau_{g_e}\|p_e\|_1 \quad \forall p_e \in \Re^t. \tag{91}$$

On the other hand, the assumption that $g_\iota(\bar{z}) \prec_{\mathcal{J}} 0$, and Lemma B.2 with $\mathcal{K} = \mathcal{J}$ and $x = -g_\iota(\bar{z})$, imply that there exists $\tau_{g_\iota} > 0$ such that

$$-\langle p_\iota, g_\iota(\bar{z})\rangle \geq \tau_{g_\iota}\|p_\iota\| \quad \forall p_\iota \in \mathcal{J}^*.$$

Using the previous inequality and the fact that $\|\nabla g_\iota(z)\|$ is bounded on $\mathcal{H}$, we conclude that there exists $\gamma > 0$ such that

$$-\|\nabla g_\iota(z)p_\iota\| - 2\gamma\langle p_\iota, g_\iota(\bar{z})\rangle \geq [2\gamma\tau_{g_\iota} - \|\nabla g_\iota(z)\|]\cdot\|p_\iota\| \geq \tau_{g_\iota}\|p_\iota\|_1 \quad \forall z \in \mathcal{H}. \tag{92}$$

Relations (91), (92), and the reverse triangle inequality, then imply that for every $z \in \mathcal{H}$,

$$\begin{aligned}
&\|\nabla g(z)p\| - 2\gamma\langle p, g(\bar{z})\rangle = \|\nabla g_\iota(z)p_\iota + G_e p_e\| - 2\gamma\langle p_\iota, g_\iota(\bar{z})\rangle \\
&\geq \|G_e p_e\| - \|\nabla g_\iota(z)p_\iota\| - 2\gamma\langle p_\iota, g_\iota(\bar{z})\rangle \geq \tau_{g_e}\|p_e\|_1 + \tau_{g_\iota}\|p_\iota\|_1 \geq \hat{\tau}\|p\|_1 \geq \hat{\tau}\|p\|,
\end{aligned}$$

where $\hat{\tau} := \min\{\tau_{g_e}, \tau_{g_\iota}\}$. It is now straightforward to see that the above inequality yields inequality (17) with $\tau_g = \hat{\tau}/(1+2\gamma)$. Statement (a) now follows from (17) and the previous conclusion. □

Acknowledgments. The first author has been supported by the US Department of Energy (DOE) and UT-Battelle, LLC, under contract DE-AC05-00OR22725 and also supported by the Exascale Computing Project (17-SC-20-SC), a collaborative effort of the U.S. Department of Energy Office of Science and the National Nuclear Security Administration. The second author was partially supported by ONR Grant N00014-18-1-2077 and AFOSR Grant FA9550-22-1-0088.

References

[1] Aybat N, Iyengar G (2011) A first-order smoothed penalty method for compressed sensing. *SIAM J. Optim.* 21(1):287–313.

[2] Aybat N, Iyengar G (2012) A first-order augmented Lagrangian method for compressed sensing. *SIAM J. Optim.* 22(2):429–459, URL http://dx.doi.org/10.1137/100786721.

[3] Beck A (2017) *First-order methods in optimization* (SIAM).

[4] Beck A, Teboulle M (2009) A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM J. Imaging Sci.* 2(1):183–202.

[5] Bertsekas D (2016) *Nonlinear programming* (Athena Scientific), "3" edition.

[6] Boob D, Deng Q, Lan G (2019) Stochastic first-order methods for convex and nonconvex functional constrained optimization. *Available on arXiv:1908.02734* .

[7] Carmon Y, Duchi JC, Hinder O, Sidford A (2018) Accelerated methods for nonconvex optimization. *SIAM J. Optim.* 28(2):1751–1772, URL http://dx.doi.org/10.1137/17M1114296.

[8] Dattorro M, Dattorro J (2005) *Convex Optimization & Euclidean Distance Geometry* (Meeboo Publishing).

[9] Fletcher R (2013) *Practical methods of optimization* (John Wiley & Sons).

[10] Florea MI, Vorobyov SA (2018) An accelerated composite gradient method for large-scale composite objective problems. *IEEE Transactions on Signal Processing* 67(2):444–459.

[11] Ghadimi S, Lan G (2016) Accelerated gradient methods for nonconvex nonlinear and stochastic programming. *Math. Program.* 156:59–99, ISSN 1436-4646.

[12] Hajinezhad D, Hong M (2019) Perturbed proximal primal–dual algorithm for nonconvex nonsmooth optimization. *Math. Program.* 176:207–245.

[13] Hiriart-Urruty J, Lemarechal C (1993) *Convex Analysis and Minimization Algorithms I* (Berlin: Springer).

[14] Hong M (2016) Decomposing linearly constrained nonconvex problems by a proximal primal dual approach: algorithms, convergence, and applications. *Available on arXiv:1604.00543* .

[15] Jiang B, Lin T, Ma S, Zhang S (2019) Structured nonconvex and nonsmooth optimization algorithms and iteration complexity analysis. *Comput. Optim. Appl.* 72(3):115–157.

[16] Kong W (2021) Accelerated inexact first-order methods for solving nonconvex composite optimization problems. *arXiv:2104.09685* .

[17] Kong W, Melo JG, Monteiro RDC (2019) Complexity of a quadratic penalty accelerated inexact proximal point method for solving linearly constrained nonconvex composite programs. *SIAM J. Optim.* 29(4):2566–2593.

[18] Kong W, Melo JG, Monteiro RDC (2019) An efficient adaptive accelerated inexact proximal point method for solving linearly constrained nonconvex composite problems. *Comput. Optim. Appl.* 76(2):305–346.

[19] Kong W, Melo JG, Monteiro RDC (2020) Iteration-complexity of an inner accelerated inexact proximal augmented Lagrangian method based on the classical Lagrangian function. *arXiv preprint arXiv:2008.00562* .

[20] Kong W, Melo JG, Monteiro RDC (2021) FISTA and Extensions - Review and New Insights. *Optimization Online* .

[21] Kong W, Monteiro RDC (2021) An accelerated inexact proximal point method for solving nonconvex-concave min-max problems. *SIAM Journal on Optimization* 31(4):2558–2585.

[22] Lan G, Monteiro RDC (2013) Iteration-complexity of first-order penalty methods for convex programming. *Math. Program.* 138(1):115–139.

[23] Lan G, Monteiro RDC (2016) Iteration-complexity of first-order augmented Lagrangian methods for convex programming. *Math. Program.* 155(1):511–547.

[24] Li Z, Chen PY, Liu S, Lu S, Xu Y (2020) Rate-improved inexact augmented Lagrangian method for constrained nonconvex optimization. *Available on arXiv:2007.01284* .

[25] Li Z, Xu Y (2020) Augmented Lagrangian based first-order methods for convex and nonconvex programs: nonergodic convergence and iteration complexity. *arXiv e-prints* arXiv–2003.

[26] Lin Q, Ma R, Xu Y (2019) Inexact proximal-point penalty methods for non-convex optimization with non-convex constraints. *Available on Arxiv:1908.11518* .

[27] Lin Q, Xiao L (2014) An adaptive accelerated proximal gradient method and its homotopy continuation for sparse optimization. *Proc. 31st Int. Conf. Mach. Learn.* 32:73–81.

[28] Liu Y, Liu X, Ma S (2019) On the nonergodic convergence rate of an inexact augmented Lagrangian framework for composite convex programming. *Math. Oper. Res.* 44(2):632–650.

[29] Lu Z, Zhou Z (2018) Iteration-complexity of first-order augmented Lagrangian methods for convex conic programming. *Available on arXiv:1803.09941* .

[30] Melo JG, Monteiro RDC, Wang H (2020) Iteration-complexity of an inexact proximal accelerated augmented Lagrangian method for solving linearly constrained smooth nonconvex composite optimization problems. *Available on arXiv:2006.08048* .

[31] Necoara I, Patrascu A, Glineur F (2017) Complexity of first-order inexact Lagrangian and penalty methods for conic convex programming. *Optim. Methods Softw.* 1–31.

[32] Nesterov Y (2012) Gradient methods for minimizing composite functions. *Math. Program.* 1–37.

[33] Nocedal J, Wright S (2006) *Numerical optimization* (Springer Science & Business Media).

[34] Parikh N, Boyd S (2014) Proximal algorithms. *Foundations and Trends in optimization* 1(3):127–239.

[35] Patrascu A, Necoara I, Tran-Dinh Q (2017) Adaptive inexact fast augmented Lagrangian methods for constrained convex optimization. *Optim. Lett.* 11(3):609–626.

[36] Rockafellar R (1976) Augmented Lagrangians and applications of the proximal point algorithm in convex programming. *Mathematics of operations research* 1(2):97–116.

[37] Rockafellar RT (1976) Augmented Lagrangians and applications of the proximal point algorithm in convex programming. *Math. Oper. Res.* 1(2):97–116, ISSN 0364-765X.

[38] Rockafellar RT, Wets RJB (1998) *Variational analysis* (Berlin: Springer), ISBN 3-540-62772-3, URL http://opac.inria.fr/record=b1093869.

[39] Sahin M, Eftekhari A, Alacaoglu A, Latorre F, Cevher V (2019) An inexact augmented Lagrangian framework for nonconvex optimization with nonlinear constraints. *Available on arXiv:1906.11357* .

[40] Xie Y, Wright S (2019) Complexity of proximal augmented Lagrangian for nonconvex optimization with nonlinear equality constraints. *arXiv preprint arXiv:1908.00131* .

[41] Xu Y (2019) Iteration complexity of inexact augmented Lagrangian methods for constrained convex programming. *Math. Program.* ISSN 1436-4646, URL http://dx.doi.org/10.1007/s10107-019-01425-9.

[42] Zhang J, Luo ZQ (2020) A global dual error bound and its application to the analysis of linearly constrained nonconvex optimization. *Available on arXiv:2006.16440* .

[43] Zhang J, Luo ZQ (2020) A proximal alternating direction method of multiplier for linearly constrained nonconvex optimization. *Available on arXiv:2006.16440* .