Journal of Machine Learning Research ? (2020) ?-? Submitted 7/22; Published ??/??

# Accelerated Inexact Composite Gradient Methods for Nonconvex Spectral Optimization Problems

**Weiwei Kong** WWKONG92@GMAIL.COM
**Renato D.C. Monteiro** RENATO.MONTEIRO@ISYE.GATECH.EDU
*School of Industrial and Systems Engineering,*
*Georgia Institute of Technology,*
*Atlanta, GA 30332-0205*

**Editor:** TBD.

## Abstract

This paper presents two inexact composite gradient methods, one inner accelerated and another doubly accelerated, for solving a class of nonconvex spectral composite optimization problems. More specifically, the objective function for these problems is of the form $f_1+f_2+h$ where $f_1$ and $f_2$ are differentiable nonconvex matrix functions with Lipschitz continuous gradients, $h$ is a proper closed convex matrix function, and both $f_2$ and $h$ can be expressed as functions that operate on the singular values of their inputs. The methods essentially use an accelerated composite gradient method to solve a sequence of proximal subproblems involving the linear approximation of $f_1$ and the singular value functions underlying $f_2$ and $h$. Unlike other composite gradient-based methods, the proposed methods take advantage of both the composite and spectral structure underlying the objective function in order to efficiently generate their solutions. Numerical experiments are presented to demonstrate the practicality of these methods on a set of real-world and randomly generated spectral optimization problems.

**Keywords:** nonconvex optimization, spectral functions, inexact proximal point methods, composite gradient methods, accelerated methods, spectral methods

## 1. Introduction

There are numerous applications in electrical engineering, machine learning, and medical imaging that can be formulated as nonconvex spectral optimization problems of the form

$$\min_{U\in\mathbb{R}^{m\times n}} \left\{\phi(U) := f_1(U) + (f_2^{\mathcal{V}}\circ\sigma)(U) + (h^{\mathcal{V}}\circ\sigma)(U)\right\}, \tag{1}$$

where $\sigma$ is the function that maps a matrix to its singular value vector (in nonincreasing order of magnitude), $f_1$ and $f_2^{\mathcal{V}}$ are continuously differentiable functions with Lipschitz continuous gradients, and $h^{\mathcal{V}}$ is a proper, lower semicontinuous, convex function. Moreover, such problems are typically formulated so that: (i) the resolvents of $\lambda\partial h$ and $\lambda\partial h^{\mathcal{V}}$ are easy compute for any $\lambda > 0$; and (ii) both $f_2^{\mathcal{V}}$ and $h^{\mathcal{V}}$ are absolutely symmetric in their arguments, i.e., they do not depend on the ordering or the sign of their arguments.

A typical approach for solving (1) is to employ a composite gradient (CG) method (or an accelerated version of it) that solves composite optimization problems of the form $\min_U [g(U) + h(U)]$, where $g$ is a continuously differentiable function with Lipschitz continuous gradient and $h$ is a proper, lower semicontinuous, convex function. More specifically,

the method is applied to (1) with $g = f_1 + f_2^{\mathcal{V}} \circ \sigma$ and $h = h^{\mathcal{V}} \circ \sigma$ and typically does not use any of the spectral structure underlying $f_2^{\mathcal{V}} \circ \sigma$ and $h^{\mathcal{V}} \circ \sigma$.

Our goal in this paper is to develop two efficient inexact composite gradient (ICG) methods that solve (1) by exploiting the spectral structure underlying the objective function. More specifically, one of the methods, called the inner accelerated ICG (IA-ICG) method inexactly solves a sequence of matrix prox subproblems of the form

$$\min_{U \in \mathbb{R}^{m \times n}} \left\{ \lambda \left[ \langle \nabla f_1(Y_{k-1}), U \rangle + (f_2^{\mathcal{V}} \circ \sigma)(U) + (h^{\mathcal{V}} \circ \sigma)(U) \right] + \frac{1}{2} \|U - Y_{k-1}\|^2 \right\} \tag{2}$$

where $\lambda > 0$ and the point $Y_{k-1}$ is the previous iterate. It is shown (see Subsection 4.1) that the effort of finding the required inexact solution $Y_k$ of (2) consists of computing one singular value decomposition (SVD) and applying an accelerated gradient (ACG) algorithm to the related vector prox subproblem

$$\min_{u \in \mathbb{R}^r} \left\{ \lambda \left[ f_2^{\mathcal{V}}(u) - \langle c_{k-1}, u \rangle + h^{\mathcal{V}}(u) \right] + \frac{1}{2} \|u\|^2 \right\} \tag{3}$$

where $r = \min\{m, n\}$ and $c_{k-1} = \sigma(Y_{k-1} - \lambda \nabla f_1(Y_{k-1}))$. Note that (3) is a problem over the vector space $\mathbb{R}^r$, and hence, has significantly fewer dimensions than (2) which is a problem over the matrix space $\mathbb{R}^{m \times n}$. The other ICG method, called the doubly accelerated ICG (DA-ICG) method, solves a similar prox subproblem as in (2) but with $Y_{k-1}$ selected in an accelerated manner (and hence its qualifier of "doubly accelerated"). Given $\hat{\rho} > 0$, it is shown that both methods obtain a pair $(\hat{Y}, \hat{V})$ satisfying

$$\hat{V} \in \nabla f_1(\hat{Y}) + \nabla \left( f_2^{\mathcal{V}} \circ \sigma \right)(\hat{Y}) + \partial \left( h^{\mathcal{V}} \circ \sigma \right)(\hat{Y}), \quad \|\hat{V}\| \leq \hat{\rho}$$

by solving at most $\mathcal{O}(\hat{\rho}^{-2})$ matrix prox subproblems as in (2).

It is worth mentioning that the IA-ICG method can be viewed an inexact version of the exact composite gradient (ECG) method applied to (1), which solves a sequence of subproblems

$$\min_{U \in \mathbb{R}^{m \times n}} \left\{ \lambda \left[ \left\langle \nabla \left[ f_1 + f_2^{\mathcal{V}} \circ \sigma \right](Y_{k-1}), U \right\rangle + (h^{\mathcal{V}} \circ \sigma)(U) \right] + \frac{1}{2} \|U - Y_{k-1}\|^2 \right\}, \tag{4}$$

where $\lambda > 0$ and the point $Y_{k-1}$ is the previous iterate. Similarly, the DA-ICG method can be viewed as an inexact version of an exact (monotone) accelerated composite gradient (EACG) method, which also solves a sequence of subproblems (4) but with $Y_{k-1}$ chosen in an accelerated manner.

For high-dimensional instances of (1) where $\min\{m, n\}$ is large, and hence, SVDs are expensive to compute, it will be shown that the larger the Lipschitz constant of $\nabla f_2^{\mathcal{V}}$ is, the better the performance of the ICG methods is compared to that of their exact counterparts. This is due to the following facts: (i) solving (4) or (2) involves a single SVD computation; (ii) even though (4) requires fewer resolvent evaluations to solve than (2), the cost of solving these subproblems is comparable due to the fact that the aforementioned SVD is the bottleneck step; and (iii) the larger the Lipschitz constant of $\nabla f_2^{\mathcal{V}}$, is the smaller the stepsize $\lambda$ in (4) must be, and hence, the more subproblems of form (4) need to be solved during the execution of the exact counterparts.

*Related works.* The earliest complexity analysis of an ACG method for solving nonconvex composite problems like the one in (1) is given in (Ghadimi and Lan, 2016). Building on the results in (Ghadimi and Lan, 2016), many other papers (Drusvyatskiy and Paquette, 2019; Ghadimi et al., 2015; Liang et al., 2019) have proposed similar ACG-based methods.

Another common approach for solving problems like (1) is to employ an inexact proximal point method where each prox subproblem is constructed to be convex, and hence, solvable by an ACG variant. For example, papers (Carmon et al., 2018; Paquette et al., 2017; Kong et al., 2019, 2020) present inner accelerated inexact proximal point methods whereas (Liang and Monteiro, 2018) presents a doubly accelerated inexact proximal point method.

To the best our knowledge, this paper is the first one to present ICG methods that exploit both the spectral and composite structure in (1).

*Organization of the paper.* Subsection 1.1 gives some notation and basic definitions. Subsection 1.2 presents several real-world problems that are of the form in (1). Section 2 presents some necessary background material for describing the ICG methods. Section 3 is split into three subsections. The first one precisely describes the problem of interest, while the last two present the IA-ICG and DA-ICG methods. Section 4 describes an efficient way of solving problem (2) by modifying a solution of problem (3). Section 5 presents some numerical results. Section 6 establishes the iteration complexity of the ICG methods. Finally, some auxiliary results are presented in Appendices A to D.

### 1.1 Notation and Basic Definitions

This subsection provides some basic notation and definitions.

The set of real numbers is denoted by $\mathbb{R}$. The set of non-negative real numbers and the set of positive real numbers is denoted by $\mathbb{R}_+$ and $\mathbb{R}_{++}$ respectively. The set of natural numbers is denoted by $\mathbb{N}$. The set of complex numbers is $\mathbb{C}$. The set of unitary matrices of size $n$–by–$n$ is $\mathcal{U}^n$. For $t > 0$, define $\log_1^+(t) := \max\{1, \log(t)\}$. Let $\mathbb{R}^n$ denote a real–valued $n$–dimensional Euclidean space with norm $\|\cdot\|$. Given a linear operator $A : \mathbb{R}^n \mapsto \mathbb{R}^p$, the operator norm of $A$ is denoted by $\|A\| := \sup\{\|Az\|/\|z\| : z \in \mathbb{R}^n, z \neq 0\}$. Using the asymptotic notation $\mathcal{O}$, we denote $\mathcal{O}_1(\cdot) \equiv \mathcal{O}(1 + \cdot)$.

Let $(m, n) \in \mathbb{N}^2$ and let $r = \min\{m, n\}$. Given matrices $X \in \mathbb{R}^{m\times n}$ and $Y \in \mathbb{R}^{n\times n}$, let the quantities $\sigma(X)$ and $\lambda(Y)$ denote the singular values and eigenvalues of $X$ and $Y$, respectively, in nonincreasing order. Let $\mathrm{dg} : \mathbb{R}^r \mapsto \mathbb{R}^{r\times r}$ and $\mathrm{Dg} : \mathbb{R}^{m\times n} \mapsto \mathbb{R}^r$ be given pointwise by

$$[\mathrm{dg}\, z]_{ij} = \begin{cases} z_i, & \text{if } i = j, \\ 0, & \text{otherwise,} \end{cases} \qquad [\mathrm{Dg}\, Z]_i = Z_{ii},$$

for every $z \in \mathbb{R}^r, Z \in \mathbb{R}^{m\times n}$, and $(i, j) \in \{1, ..., r\}^2$.

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

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

and we denote $\partial\psi \equiv \partial_0\psi$. The set of proper, lower semi-continuous, convex functions is denoted by $\overline{\text{Conv}}\ \mathcal{Z}$. The convex conjugate $\psi$ is denoted by $\psi^*$. The linear approximation of $\psi$ at a point $z_0 \in \operatorname{dom}\psi$ is denoted by $\ell_\psi(\cdot; z_0) := \psi(z_0) + \langle \nabla\psi(z_0), \cdot - z_0\rangle$. The indicator of a closed convex set $C \subseteq \mathcal{Z}$ at a point $z \in \mathcal{Z}$ is denoted by $\delta_C(z)$, which is 1 if $z \in C$ and $\infty$ otherwise. The local Lipschitz constant of $\nabla\psi$ at two points $u, z \in \mathcal{Z}$ is denoted by

$$L_\psi(x, y) = \begin{cases} \frac{\|\nabla\psi(x) - \nabla\psi(y)\|}{\|x - y\|}, & x \neq y, \\ 0, & x = y, \end{cases} \quad \forall x, y \in \operatorname{dom}\psi. \tag{5}$$

### 1.2 Motivating Applications

This subsection lists some motivating applications that are of the form in (1). Throughout this subsection, we will assume that we have two sparsity-inducing regularizers $\mathcal{R} = \mathcal{R}_s + \mathcal{R}_n$ and $\mathcal{P}$, where $\mathcal{R}_s$ and $\mathcal{P}$ are continuously differentiable functions with Lipschitz continuous gradients and $\mathcal{R}_n$ is a proper, lower semicontinuous, and convex function.

#### 1.2.1 Matrix Completion

Let $A \in \mathbb{R}^{m\times n}$ be a given data matrix and let $r = \min\{m, n\}$. Moreover, let $\Omega$ denote a subset of the indices of $A$. The goal of the general matrix completion problem is to find a low rank approximation of $A$ that is close to $A$ in some sense.

A nonconvex formulation (see, for example, Yao and Kwok, 2017) of this problem is

$$\min_{X\in\mathbb{R}^{m\times n}} \left\{ \frac{1}{2}\|P_\Omega(X - A)\|_F^2 + (\mathcal{R} \circ \sigma)(X) \right\},$$

where $P_\Omega$ is the function that zeros out the entries of its input that are not in $\Omega$.

#### 1.2.2 Phase Retrieval

Given a vector $x \in \mathbb{R}^n$, let $x[\omega]$ denote its discrete Fourier transform for some frequency $\omega$. Moreover, for some unknown noisy signal $\tilde{x} \in \mathbb{R}^n$ and a frequency set $\Omega \subseteq \mathbb{R}_+$, suppose that we are given measurements $\{|\tilde{x}[\omega]|\}_{\omega\in\Omega}$ and vectors $a_\omega \in \mathbb{C}^n$ such that $|\langle a_\omega, \tilde{x}\rangle| = |\tilde{x}[\omega]|$ for every $\omega \in \Omega$. The goal of the phase retrieval problem is to recover an approximation $x$ of $\tilde{x}$ such that $|\langle a_\omega, x\rangle|^2 \approx |\langle a_\omega, \tilde{x}\rangle|^2$ for every $\omega \in \Omega$.

A nonconvex formulation of this problem is

$$\min_{X\in\mathbb{R}^{|\Omega|\times|\Omega|}} \left\{ \frac{1}{2}\|\mathcal{A}(X) - b\|^2 + (\mathcal{R} \circ \lambda)(X) : X \succeq 0 \right\},$$

where $\lambda$ denotes the function that maps matrices to their eigenvalue vector and the quantities $\mathcal{A} : \mathbb{R}^{|\Omega|\times|\Omega|} \mapsto \mathbb{R}^{|\Omega|}$ and $b \in \mathbb{R}^{|\Omega|}$ are given by

$$[\mathcal{A}(X)]_\omega = \operatorname{tr}(a_\omega a_\omega^* X), \quad b_\omega = |\tilde{x}[\omega]|^2, \quad \forall (X, \omega) \in \mathbb{R}^{|\Omega|\times|\Omega|} \times \Omega.$$

In particular, this formulation is a generalization of the one in (Candes et al., 2015) where the convex function $\operatorname{tr} X$ is replaced with the nonconvex function $\mathcal{R}$.

#### 1.2.3 Robust Principal Component Analysis

Let $\widehat{M} \in \mathbb{R}^{m\times n}$ be a given data matrix and let $r = \min\{m, n\}$. The goal of the robust principal component analysis problem is to find an approximation $M + E$ of $\widehat{M}$ where $M$ is low-rank and $E$ is sparse.

A nonconvex formulation of this problem is

$$\min_{M,E\in\mathbb{R}^{m\times n}} \left\{\frac{1}{2}\|\widehat{M} - (M + E)\|_F^2 + (\mathcal{R} \circ \sigma)(M) + \mathcal{P}(E)\right\}.$$

In particular, this formulation is a instance of the one in (Wen et al., 2019) where more structure is imposed on the functions $\mathcal{R}$ and $\mathcal{P}$.

## 2. Background Material

Recall from Section 1 that our interest is in solving (1) by repeated solving a sequence of prox subproblems as in (2). This section presents some background material regarding (2).

This section considers the nonconvex composite optimization (NCO) problem

$$\min_{u\in\mathcal{Z}} \{\psi(u) := \psi_s(u) + \psi_n(u)\}, \tag{6}$$

where $\mathcal{Z}$ is a finite dimensional inner product space and the functions $\psi_s$ and $\psi_n$ are assumed to satisfy the following assumptions:
(B1) $\psi_n \in \overline{\text{Conv}}\ \mathcal{Z}$;
(B2) $\psi_s$ is continuously differentiable on $\mathcal{Z}$ and satisfies

$$\frac{\mu}{2}\|u - y\|^2 \le \psi_s(u) - [\psi_s(y) + \langle\nabla\psi_s(y), u - y\rangle] \le \frac{M}{2}\|u - y\|^2$$

for some $(\mu, M) \in \mathbb{R}^2_+$ and every $u, y \in \mathcal{Z}$.

Clearly, problems (1) and (2) are special cases of (6), and hence any definition or result that is stated in the context of (6) applies to (1) and/or (2).

An important notion of an approximate solution of (6) is as follows: given $\hat{\rho} > 0$, a pair $(y_r, v_r)$ is said to be a $\hat{\rho}$–approximate solution of (6) if

$$v_r \in \nabla\psi_s(y_r) + \partial\psi_n(y_r), \quad \|v_r\| \le \hat{\rho}. \tag{7}$$

In Section 3, we develop prox-type methods for finding $\hat{\rho}$–approximate solutions of (1) that repeatedly solve (2) inexactly by taking advantage of its spectral decomposition.

We now discuss the inexactness criterion under which the subproblems (2) are solved. Again, the criterion is described in the context of (6) as follows.

**Problem $\mathcal{A}$** : Given $(\mu, \sigma) \in \mathbb{R}^2_{++}$ and $z_0 \in \mathcal{Z}$, find $(y, v, \varepsilon) \in \operatorname{dom}\psi \times \mathcal{Z} \times \mathbb{R}_+$ such that

$$v \in \partial_\varepsilon\left(\psi - \frac{\mu}{2}\|\cdot - y\|^2\right)(y), \quad \|v\|^2 + 2\varepsilon \le \sigma^2\|y - z_0\|^2. \tag{8}$$

We begin by making three remarks about the above problem. First, if $(y, v, \varepsilon)$ solves Problem $\mathcal{A}$ with $\sigma = 0$, then $(v, \varepsilon) = (0, 0)$, and $z$ is an exact solution of (6). Hence, the output $(y, v, \varepsilon)$ of Problem $\mathcal{A}$ can be viewed as an inexact solution of (6) when $\sigma \in \mathbb{R}_{++}$. Second, the input $z_0$ is arbitrary for the purpose of this section. However, the two methods described in Section 3 for solving (1) repeatedly solve (2) according to Problem $\mathcal{A}$ with the input $z_0$ at the $k^{\text{th}}$ iteration determined by the iterates generated at the $(k-1)^{\text{th}}$ iteration. Third, defining the function

$$\Delta_\mu(u; y, v) := \psi(y) - \psi(u) - \langle v, u - y\rangle + \frac{\mu}{2}\|u - y\|^2 \quad \forall u \in \operatorname{dom}\psi, \tag{9}$$

another way to express the inclusion in (8) is $\Delta_\mu(u; y, v) \le \varepsilon$ for every $u \in \operatorname{dom}\psi$. Finally, the R-ACG algorithm presented later in this subsection will be shown to solve Problem $\mathcal{A}$ when $\psi_s$ is convex. Moreover, it solves a weaker version of Problem $\mathcal{A}$ involving $\Delta_\mu$ (see Problem $\mathcal{B}$ later on) whenever $\psi_s$ is not convex and as long as some key inequalities are satisfied during its execution.

A technical issue in our analysis in this paper lies in the ability of refining the output of Problem $\mathcal{A}$ to an approximate solution $(y_r, v_r)$ of (6), i.e., one satisfying the inclusion in (7), in which $\|v_r\|$ is nicely bounded. We now present a refinement procedure that addresses this issue.

---

**Refinement Procedure**

---

**Input**: a triple $(M, \psi_s, \psi_n)$ satisfying (B1)–(B2) and a pair $(y, v) \in \operatorname{dom}\psi_n \times \mathcal{Z}$;
**Output**: a pair $(y_r, v_r)$ satisfying the inclusion in (7);
Step 1 set the quantities

$$y_r = \operatorname*{argmin}_{u \in \mathcal{Z}} \left\{ \langle \nabla\psi_s(y) - v, u\rangle + \frac{M}{2}\|u - y\|^2 + \psi_n(u) \right\}, \tag{10}$$

$$v_r = v + M(y - y_r) + \nabla\psi_s(y_r) - \nabla\psi_s(y), \tag{11}$$

and output $(y_r, v_r)$.

---

The result below presents the key properties of the above procedure. For the sake of brevity, we write $(y_r, v_r) = RP(y, v)$ to indicate that the pair $(y_r, v_r)$ is the output of the above procedure with inputs $(M, \psi_s, \psi_n)$ and $(y, v)$.

**Proposition 1.** *Let $(M, \psi_s, \psi_n)$ satisfying assumptions (B1)–(B2) and a triple $(y, v, \varepsilon) \in \operatorname{dom}\psi_n \times \mathcal{Z} \in \mathbb{R}_+$ be given. Moreover, let $(y_r, v_r) = RP(y, v)$, denote $L_{\psi_s}(\cdot, \cdot)$ simply by $L(\cdot, \cdot)$ where $L_{\psi_s}(\cdot, \cdot)$ is as in (5), and let $\Delta_\mu$ be as in (9). Then, the following statements hold:*

*(a) $v_r \in \nabla\psi_s(y_r) + \partial\psi_n(y_r)$;*

*(b) for every $s \in \operatorname{dom}\psi_n$ we have $\Delta_\mu(u; y, v) \ge 0$ and, in particular,*

$$\Delta_\mu(y_r; y, v) \ge \frac{M}{2}\|y_r - y\|^2; \tag{12}$$

*(c) if* $\Delta_\mu(y_r; y, v) \le \varepsilon$ *and* $(y, v, \varepsilon)$ *satisfies the inequality in* (8)*, then*

$$\|v_r\| \le \sigma \left[1 + \frac{M + L(y, y_r)}{\sqrt{M}}\right] \|y - z_0\|; \tag{13}$$

*(d) if* $(y, v, \varepsilon)$ *solves Problem* $\mathcal{A}$*, then* $\Delta_\mu(u; y, v) \le \varepsilon$ *for every* $u \in \operatorname{dom} \psi_n$*, and, as a consequence, bound* (13) *holds.*

*Proof.* (a) Using the definition of $v_r$ and the optimality of $y_r$, we have that

$$v_r = v + M(y - y_r) + \nabla\psi_s(y_r) - \nabla\psi_s(y) \in \nabla\psi_s(y_r) + \partial\psi_n(y_r).$$

(b) The fact that $\Delta_\mu(u; y, v) \ge 0$ for every $u \in \operatorname{dom} \psi_n$ follows from the optimality of $y_r$ and the fact that $\psi_s \le \ell_{\psi_s}(\cdot; y) + M\|\cdot - y\|^2/2$. The bound (12) follows from Proposition 19 with $(L, g, h) = (M, \psi_s - \langle v, \cdot\rangle, \psi_n)$.

(c) Using the assumption that $\Delta_\mu(y_r; y, v) \le \varepsilon$, part (b), and the inequality in (8), we have that

$$\|y - y_r\| \le \sqrt{\frac{2\Delta_\mu(y_r; y, v)}{M}} \le \sqrt{\frac{2\varepsilon}{M}} \le \frac{\sigma}{\sqrt{M}}\|y - z_0\|. \tag{14}$$

Using the triangle inequality, the definition of $L(\cdot, \cdot)$, (14) and the inequality in (8) again, we conclude that

$$\|v_r\| \le \|v\| + [M + L(y, y_r)]\|y - y_r\| \le \sigma \left[1 + \frac{M + L(y, y_r)}{\sqrt{M}}\right] \|y - z_0\|.$$

(d) The fact that $\Delta_\mu(u; y, v) \le \varepsilon$ for every $u \in \operatorname{dom} \psi_n$ follows immediately from the inclusion in (8) and the definition of $\Delta_\mu$ in (9). The fact that (13) holds now follows from part (c). ∎

We make a few remarks about Proposition 1. First, it follows from (a) that $(y_r, v_r)$ satisfies the inclusion in (7). Second, it follows from (a) and (c) that if $\sigma = 0$, then $(y_r, v_r) = (0, 0)$, and hence $y_r$ is an exact stationary point of (6). In general, (13) implies that the residual $\|v_r\|$ is directly proportional to $\|y - w\|$, and hence, becomes smaller as this quantity approaches *zero*.

Inequality (13) plays an important technical role in the complexity analysis of the two prox-type methods of Section 3. Sufficient conditions for its validity are provided in (c) and (d), with (c) being the weaker one, in view of (d). When $\psi_s$ is convex, it is shown that every iterate of the R-ACG algorithm presented below always satisfies the inclusion in (8), and hence, verifying the the validity of the sufficient condition in (c) amounts to simply checking whether the inequality in (8) holds. When $\psi_s$ is not convex, verification of the inclusion in (8), and hence the sufficient condition in (d), is generally not possible, while the one in (c) is. This is a major advantage of the sufficient condition in (c), which is exploited in this paper towards the development of adaptive prox-type methods which attempt to approximately solve (6) when $\psi_s$ is not convex.

For the sake of future reference, we now state the following problem for finding a triple $(y, v, \varepsilon)$ satisfying the sufficient condition in Proposition 1(c). Its statement relies on the refinement procedure preceding Proposition 1.

**Problem $\mathcal{B}$** : Given the same inputs as in Problem $\mathcal{A}$, find $(y, v, \varepsilon) \in \operatorname{dom}\psi \times \mathcal{Z} \times \mathbb{R}_+$ satisfying the inequality in (8) and

$$\Delta_\mu(y_r; y, v) \leq \varepsilon, \tag{15}$$

where $\Delta_\mu(\cdot;\cdot,\cdot)$ is as in (9) and $y_r$ is the first component of the refined pair $(y_r, v_r) = RP(y, v)$.

We now state the aforementioned R-ACG algorithm which solves Problem $\mathcal{A}$ when $\psi_s$ is convex and solves Problem $\mathcal{B}$ whenever $\psi_s$ is not convex and two key inequalities are satisfied, one at every iteration (i.e., (16)) and one at the end of its execution.

**R-ACG Algorithm**

**Input**: a quadruple $(\mu, M, \psi_s, \psi_n)$ satisfying (B1)–(B2) and a pair $(\sigma, z_0)$;
**Output**: a triple $(y, v, \varepsilon)$ that solves Problem $\mathcal{B}$ or a *failure* status;
Step 0 define $\psi := \psi_s + \psi_n$ and set $z_0^c = z_0$, $B_0 = 0$, $\Gamma_0 \equiv 0$, and $j = 1$;
Step 1 compute the iterates

$$\begin{aligned}
\xi_{j-1} &= \frac{1 + \mu B_{j-1}}{M - \mu}, \quad b_{j-1} = \frac{\xi_{j-1} + \sqrt{\xi_{j-1}^2 + 4\xi_{j-1}B_{j-1}}}{2}, \\
B_j &= B_{j-1} + b_{j-1}, \quad \tilde{z}_{j-1} = \frac{B_{j-1}}{B_j} z_{j-1} + \frac{b_{j-1}}{B_j} z_{j-1}^c, \\
z_j &= \operatorname*{argmin}_{u \in \mathcal{Z}} \left\{ l_{\psi_s}(u; \tilde{z}_{j-1}) + \psi_n(u) + \frac{M}{2}\|u - \tilde{z}_{j-1}\|^2 \right\}, \\
z_j^c &= \frac{1}{1 + \mu B_j} \left[ z_{j-1}^c - \frac{b_{j-1}}{M - \mu}(\tilde{z}_{j-1} - z_j) + \mu(B_{j-1} z_{j-1}^c + a_{j-1} z_j) \right];
\end{aligned}$$

Step 2 compute the quantities

$$\begin{aligned}
\tilde{\gamma}_j &= l_{\psi_s}(\cdot; \tilde{z}_{j-1}) + \psi_n + \frac{\mu}{2}\|\cdot - \tilde{z}_{j-1}\|^2 \\
\gamma_j &= \tilde{\gamma}_j(z_j) + \frac{1}{M - \mu}\langle \tilde{z}_{j-1} - z_j, \cdot - z_j \rangle + \frac{\mu}{2}\|\cdot - z_j\|^2, \\
\Gamma_j &= \frac{B_{j-1}}{B_j}\Gamma_{j-1} + \frac{b_{j-1}}{B_j}\gamma_{j-1}, \quad r_j = \frac{z_0^c - z_j^c}{B_j} + \mu(z_j^c - z_j), \\
\eta_j &= \psi(z_j) - \Gamma_j(z_j^c) - \left\langle r_j, z_j - z_j^c \right\rangle + \frac{\mu}{2}\|z_j - z_j^c\|^2.
\end{aligned}$$

Step 3 if the inequality

$$\|B_j r_j + z_j - z_0\|^2 + 2B_j\eta_j \leq \|z_j - z_0\|^2 \tag{16}$$

holds, then go to step 4; otherwise, **stop** with a *failure* status;
Step 4 if the inequality

$$\|r_j\|^2 + 2\eta_j \leq \sigma^2\|z_j - z_0\|^2, \tag{17}$$

holds, then go to step 5; otherwise, go to step 1;

Step 5 set $(y, v, \varepsilon) = (z_j, r_j, \eta_j)$ and compute $(y_r, v_r) = RP(z_j, r_j)$; if the condition

$$\Delta_\mu(y_r; y, v) \leq \varepsilon,$$

holds then **stop** with a *success* status and **output** the triple $(y, v, \varepsilon)$; otherwise, **stop** with a *failure* status;

---

It is well-known that the scalar $B_j$ updated in step 1 satisfies

$$B_j \geq \frac{1}{M} \max\left\{\frac{j^2}{4}, \left(1 + \sqrt{\frac{\mu}{4M}}\right)^{2(j-1)}\right\} \quad \forall j \geq 1. \tag{18}$$

The next result presents the key properties about the R-ACG algorithm.

**Proposition 2.** *The R-ACG algorithm has the following properties:*

*(a) it stops with either failure or success in at most*

$$\left\lceil 1 + \left(1 + 2\sqrt{\frac{M}{\mu}}\right) \log_1^+\left(K_\sigma \sqrt{2M}\right)\right\rceil \tag{19}$$

*iterations, where $K_\sigma := 1 + \sqrt{2}/\sigma$;*

*(b) if it stops with success, then its output $(y, v, \varepsilon)$ solves Problem $\mathcal{B}$;*

*(c) if $\psi_s$ is convex then it always stops with success and its output $(y, v, \varepsilon)$ solves Problem $\mathcal{A}$.*

*Proof.* (a) See Appendix B.

(b) This follows from the successful checks in step 4 and 5 of the algorithm.

(c) The fact that the algorithm never stops with failure follows from Proposition 20(c)–(d) in Appendix B. The fact that the the algorithm stops with success follows the previous statement, the successful checks in step 4 and 5 of the algorithm, and the fact that the algorithm stops in a finite number of iterations in part (a). ■

## 3. Inexact Composite Gradient Methods

This section presents the ICG methods and the general problem that they solve. It contains three subsections. The first one presents Problem of interest and gives a general outline of the ICG methods, the second one presents the IA-ICG method, and the third one presents the DA-ICG method.

For the ease of presentation, the proofs of this section are deferred to Section 6.

### 3.1 Problem of Interest and Outline of the Methods

This subsection describes Problem that the ICG methods solve and outlines their structure.

The ICG methods consider the NCO problem

$$\min_{u \in \mathcal{Z}} [\phi(u) := f_1(u) + f_2(u) + h(u)] \tag{20}$$

where $\mathcal{Z}$ is an finite dimensional inner product space and the functions $f_1, f_2$, and $h$ are assumed to satisfy the following assumptions:

(A1) $h \in \overline{\text{Conv}}\ \mathcal{Z}$;

(A2) $f_1, f_2$ are continuously differentiable functions and there exists $(m_1, M_1) \in \mathbb{R}^2$ and $(m_2, M_2) \in \mathbb{R}^2$ such that, for $i \in \{1, 2\}$, we have

$$-\frac{m_i}{2}\|u - y\|^2 \leq f_i(u) - \ell_{f_i}(u; y) \leq \frac{M_i}{2}\|u - y\|^2 \quad \forall u, y \in \operatorname{dom} h; \tag{21}$$

(A3) for $i \in \{1, 2\}$, we have

$$\|\nabla f_i(u) - \nabla f_i(y)\| \leq L_i\|u - y\| \quad \forall u, y \in \operatorname{dom} h,$$

where $L_i := \max\{|m_i|, |M_i|\}$;

(A4) $\phi_* := \inf_{u \in \mathcal{Z}} \phi(u) > -\infty$.

Note that assumption (A2) implies that assumption (A3) holds when the interior of $\operatorname{dom} h$ is nonempty. Under the above assumptions, the ICG methods find an approximate solution $(\hat{y}, \hat{v})$ of (20) as in (7) with $\psi_s = f_1 + f_2$ and $\psi_n = h$, i.e.

$$\hat{v} \in \nabla f_1(\hat{y}) + \nabla f_2(\hat{y}) + \partial h(\hat{y}), \quad \|\hat{v}\| \leq \hat{\rho}. \tag{22}$$

We now outline the ICG methods. Given a starting point $y_0 \in \operatorname{dom} \psi_n$ and a special stepsize $\lambda > 0$, each method continually calls the R-ACG algorithm of Section 2 to find an approximate solution of a prox-linear form of (20). More specifically, each R-ACG call is used to tentatively find an approximate solution of

$$\min_{u \in \mathcal{Z}} \left[\psi(u) = \lambda\left[\ell_{f_1}(u; z_0) + f_2(u) + h(u)\right] + \frac{1}{2}\|u - z_0\|^2\right], \tag{23}$$

for some reference point $z_0$. For the IA-ICG method, the point $z_0$ is $y_0$ for the first R-ACG call and is the last obtained approximate solution for the other R-ACG calls. For the DA-ICG method, the point $z_0$ is chosen in an accelerated manner.

From the output of the $k^{\text{th}}$ R-ACG call, a refined pair $(\hat{y}, \hat{v}) = (\hat{y}_k, \hat{v}_k)$ is generated which: (i) always satisfies the inclusion of (22); and (ii) is such that $\min_{i \leq k} \|\hat{v}_i\| \to 0$ as $k \to \infty$. More specifically, this refined pair is generated by applying the refinement procedure of Section 2 and adding some adjustments to the resulting output to conform with our goal of finding an approximate solution as in (22). For the ease of future reference, we now state this specialized refinement procedure. Before proceeding, we introduce the shorthand notation

$$M_i^+ := \max\{M_i, 0\}, \quad m_i^+ := \max\{m_i, 0\}, \quad L_i(x, y) := L_{f_i}(x, y),$$

for $i \in \{1, 2\}$, to keep its presentation (and future results) concise.

---

**Specialized Refinement Procedure**

---

**Input**: a quadruple $(M_2, f_1, f_2, h)$ satisfying (A1)–(A2), a scalar $\lambda > 0$, and a triple $(y, v, z_0) \in \operatorname{dom} \psi_n \times \mathcal{Z} \times \mathcal{Z}$;

**Output**: a pair $(\hat{y}, \hat{v})$ satisfying the inclusion of (22);

Step 1 compute $(\hat{y}, v_r) = RP(y, v)$ using the refinement procedure in Section 2 with

$$M = \lambda M_2^+ + 1, \quad \psi_s = \lambda\left[\ell_{f_1}(\cdot; z_0) + f_2\right] + \frac{1}{2}\|\cdot - z_0\|^2, \quad \psi_n = \lambda h;$$

Step 2 compute the residual

$$\hat{v} = \frac{1}{\lambda}(v_r + z_0 - y) + \nabla f_1(\hat{y}) - \nabla f_1(z_0),$$

and output $(\hat{y}, \hat{v})$.

---

The result below states some properties about the above procedure. For the sake of brevity, we write $(\hat{y}, \hat{v}) = SRP(y, v, y_0)$ to indicate that the pair $(\hat{y}, \hat{v})$ is the output of the above procedure with inputs $(M_2, f_1, f_2, h)$, $\lambda$, and $(y, v, z_0)$.

**Lemma 3.** *Let $(m_1, M_1)$, $(m_2, M_2)$, and $(f_1, f_2, h)$ satisfying assumptions (A1)–(A3) and a quadruple $(z_0, y, v, \varepsilon) \in \mathcal{Z} \times \operatorname{dom}\psi_n \times \mathcal{Z} \in \mathbb{R}_+$ be given. Moreover, let $(\hat{y}, \hat{v}) = SRP(y, v, y_0)$ and define*

$$C_\lambda(x, y) := \frac{1 + \lambda\left[M_2^+ + L_1(x, y) + L_2(x, y)\right]}{\sqrt{1 + \lambda M_2^+}}, \tag{24}$$

*for every $x, y \in \mathcal{Z}$. Then, the following statements hold:*

*(a) $\hat{v} \in \nabla f_1(\hat{y}) + \nabla f_2(\hat{y}) + \partial h(\hat{y})$;*

*(b) if $(y, v, \varepsilon)$ solves Problem $\mathcal{B}$ with $(\mu, \psi_s, \psi_n)$ as in (26), then*

$$\|\hat{v}\| \le \left[L_1(y, w) + \frac{2 + \sigma C_\lambda(y, \hat{y})}{\lambda}\right]\|y - z_0\|.$$

It is worth recalling from Section 1 that in the applications we consider, the cost of the R-ACG call is small compared to SVD computation performed that is performed before solving each subproblem as in (23). Hence, in the analysis that follows, we present complexity results related to the number of subproblems solved rather than the total number of R-ACG iterations. We do note, however, that the number of R-ACG iterations per subproblem is finite in view of Proposition 2(a).

### 3.2 IA-ICG Method

This subsection presents the static IA-ICG method and its (titular) dynamic variant.

We first state the static IA-ICG method.

---

**Static IA-ICG Method**

---

**Input**: function triple $(f_1, f_2, h)$ and scalar quadruple $(m_1, M_1, m_2, M_2) \in \mathbb{R}^4$ satisfying (A1)–(A4), tolerance $\hat{\rho} > 0$, initial point $y_0 \in \operatorname{dom} h$, and scalar pair $(\lambda, \sigma) \in \mathbb{R}_{++} \times (0, 1)$ satisfying

$$\lambda M_1 + \sigma^2 \le \frac{1}{2}; \tag{25}$$

**Output**: a pair $(\hat{y}, \hat{v})$ satisfying (22) or a *failure* status;

Step 0 let $\Delta_1(\cdot;\cdot,\cdot)$ be as in (9) with $\mu = 1$, and set $k = 1$;

Step 1 use the R-ACG algorithm to tentatively solve Problem $\mathcal{B}$ associated with (23), i.e., with inputs $(\mu, M, \psi_s, \psi_n)$ and $(\sigma, z_0)$ where the former is given by

$$\begin{aligned} \mu = 1, \quad M = \lambda M_2^+ + 1, \\ \psi_s = \lambda\left[\ell_{f_1}(\cdot; z_0) + f_2\right] + \frac{1}{2}\|\cdot - z_0\|^2, \quad \psi_n = \lambda h, \end{aligned} \tag{26}$$

and $z_0 = y_{k-1}$; if the R-ACG stops with *failure*, then **stop** with a *failure* status; otherwise, let $(y_k, v_k, \varepsilon_k)$ denote its output and go to step 2;

Step 2 if the inequality $\Delta_1(y_{k-1}; y_k, v_k) \leq \varepsilon_k$ holds, then go to step 3; otherwise, **stop** with a *failure* status;

Step 3 set $(\hat{y}_k, \hat{v}_k) = SRP(y_k, v_k, y_{k-1})$; if $\|\hat{v}_k\| \leq \hat{\rho}$ then **stop** with a *success* status and **output** $(\hat{y}, \hat{v}) = (\hat{y}_k, \hat{v}_k)$; otherwise, update $k \leftarrow k + 1$ and go to step 1.

---

Note that the static IA-ICG method may fail without obtaining a pair satisfying (22). In Proposition 4(c) below, we state that a sufficient condition for the method to stop successfully is that $f_2$ be convex. This property will be important when we present the (dynamic) IA-ICG method, which: (i) repeatedly calls the static method; and (ii) incrementally transfers convexity from $f_1$ to $f_2$ between each call until a successful termination is achieved.

We now make some additional remarks about the above method. First, it performs two kinds of iterations, namely, ones that are indexed by $k$ and ones that are performed by the R-ACG algorithm. We refer to the former kind as outer iterations and the latter kind as inner iterations. Second, in view of (25), if $M_1 > 0$ then $0 < \lambda < (1 - 2\sigma^2)/(2M_1)$ whereas if $M_1 \leq 0$ then $0 < \lambda < \infty$.

The next result summarizes some facts about the static IA-ICG method. Before proceeding, we first define some useful quantities. For $\lambda > 0$ and $u, w \in \mathcal{Z}$, define

$$\widetilde{\ell}_\phi(u; w) := \ell_{f_1}(u; w) + f_2(u) + h(u), \quad \overline{C}_\lambda := \frac{1 + \lambda(M_2^+ + L_1 + L_2)}{\sqrt{1 + \lambda M_2^+}}. \tag{27}$$

**Theorem 4.** *The following statements hold about the static IA-ICG method:*

*(a) it stops in*

$$\mathcal{O}_1\left(\left[\sqrt{\lambda}L_1 + \frac{1 + \sigma\overline{C}_\lambda}{\sqrt{\lambda}}\right]^2 \left[\frac{\phi(z_0) - \phi_*}{\hat{\rho}^2}\right]\right) \tag{28}$$

*outer iterations, where $\phi_*$ is as in (A4);*

*(b) if it stops with success, then its output pair $(\hat{y}, \hat{v})$ is a $\hat{\rho}$–approximate solution of (20);*

*(c) if $f_2$ is convex, then it always stops with success.*

We now make three remarks about the above results. First, if $\sigma = \mathcal{O}(1/\overline{C}_\lambda)$ then (28) reduces to

$$\mathcal{O}_1\left(\left[\sqrt{\lambda}L_1 + \frac{1}{\sqrt{\lambda}}\right]^2 \left[\frac{\phi(z_0) - \phi_*}{\hat{\rho}^2}\right]\right). \tag{29}$$

Moreover, comparing the above complexity to the iteration complexity of the ECG method described in Section 1, which is known (see, for example, Monteiro et al., 2012) to obtain

an approximate solution of (20) in

$$\mathcal{O}_1\left(\left[\sqrt{\lambda}(L_1+L_2)+\frac{1}{\sqrt{\lambda}}\right]^2\left[\frac{\phi(z_0)-\phi_*}{\hat{\rho}^2}\right]\right) \tag{30}$$

iterations, we see that (29) is smaller than (30) in magnitude when $L_2$ is large. Second, Theorem 4(b) shows that if the method stops with success, regardless of the convexity of $f_2$, then its output pair $(\hat{y},\hat{v})$ is always an approximate solution of (20). Third, in view of Proposition 10, the quantities $L_1$ and $\overline{C}_\lambda$ in all of the previous complexity results can be replaced by their averaged counterparts in (43). As these averaged quantities only depend on $\{(y_i,\hat{y}_i)\}_{i=1}^k$, we can infer that the static IA-ICG method adapts to the local geometry of its input functions.

We now state the (titular) dynamic IA-ICG method that resolves the issue of failure in the static IA-ICG method.

**IA-ICG Method**

**Input**: the same as the static IA-ICG method but with an additional parameter $\xi_0>0$;
**Output**: a pair $(\hat{y},\hat{v})$ satisfying (22);
Step 0 set $\xi=\xi_0$, $\ell=1$, and

$$\begin{gathered} f_1=f_1-\frac{\xi}{2}\|\cdot\|^2,\quad f_2=f_2+\frac{\xi}{2}\|\cdot\|^2,\\ m_1=m_1+\xi,\quad M_1=M_1-\xi,\quad m_2=m_2-\xi,\quad M_2=M_2+\xi; \end{gathered} \tag{31}$$

Step 1 call the static IA-ICG method with inputs $(f_1,f_2,h)$, $(m_1,M_1,m_2,M_2)$, $\hat{\rho}$, $y_0$, and $(\lambda,\sigma)$;
Step 2 if the static IA-ICG call stops with a *failure* status, then set $\xi=2\xi$, update the quantities in (31) with the new value of $\xi$, increment $\ell=\ell+1$, and go to step 1; otherwise, let $(\hat{y},\hat{v})$ be the output pair returned by the static IA-ICG call, **stop**, and **output** this pair.

Some remarks about the above method are in order. First, in view of (25) and the fact that $M_1$ is monotonically decreasing, the parameter $\lambda$ does not need to be changed for each IA-ICG call. Second, in view of assumption (A2) and Theorem 4(c), the IA-ICG call in step 1 always terminates with success whenever $m_2\leq 0$. As a consequence, the total number of IA-ICG calls is at most $\left\lceil\log(2m_2^+/\xi_0)\right\rceil$. Third, in view of the second remark and Theorem 4(b), the methods always obtains a $\hat{\rho}$–approximate solution of (20) in a finite number of IA-ICG outer iterations. Finally, in view of second remark again, the total number of IA-ICG outer iterations is as in Theorem 4(a) but with: (i) an additional multiplicative factor of $\left\lceil\log(2m_2^+/\xi_0)\right\rceil$; and (ii) the constants $m_1$ and $M_2$ replaced with $(m_1+2m_2^+)$ and $(M_2+2m_2^+)$, respectively. It is worth mentioning that a more refined analysis, such as the one in (Kong et al., 2020), can be applied in order to remove the factor of $\left\lceil\log(2m_2^+/\xi_0)\right\rceil$ from the previously mentioned complexity.

### 3.3 DA-ICG Method

This subsection presents the static DA-ICG method, but omits its (titular) dynamic variant for the sake of brevity. We do argue, however, that the dynamic variant can be stated in the same way as the (dynamic) IA-ICG method of Subsection 6.1 but with the call to the static IA-ICG method replaced with a call to the static DA-ICG method of this subsection.

We start by stating some additional assumptions. It is assumed that:

(i) the set $\operatorname{dom} h$ is closed;

(ii) there exists a bounded set $\Omega \supseteq \operatorname{dom} h$ for which a projection oracle exists.

We now state the static DA-ICG method.

---

**Static DA-ICG Method**

---

**Input**: function triple $(f_1, f_2, h)$ and scalar quadruple $(m_1, M_1, m_2, M_2) \in \mathbb{R}^4$ satisfying (A1)–(A4), tolerance $\hat{\rho} > 0$, initial point $y_0 \in \operatorname{dom} h$, and scalar pair $(\lambda, \sigma) \in \mathbb{R}_{++} \times (0, 1)$ satisfying

$$\lambda M_1 + \sigma^2 \leq \frac{1}{2}; \tag{32}$$

**Output**: a pair $(\hat{y}, \hat{v})$ satisfying (22) or a *failure* status;

Step 0 let $\Delta_1(\cdot;\cdot,\cdot)$ be as in (9) with $\mu = 1$, and set $A_0 = 0$, $x_0 = y_0$, and $k = 1$;

Step 1 compute the quantities

$$\begin{aligned} a_{k-1} &= \frac{1 + \sqrt{1 + 4A_{k-1}}}{2}, \quad A_k = A_{k-1} + a_{k-1}, \\ \tilde{x}_{k-1} &= \frac{A_{k-1} y_{k-1} + a_{k-1} x_{k-1}}{A_k}; \end{aligned} \tag{33}$$

Step 2 use the R-ACG algorithm to tentatively solve Problem $\mathcal{B}$ associated with (23), i.e., with inputs $(\mu, M, \psi_s, \psi_n)$ and $(\sigma, z_0)$ where the former is as in (26) and $z_0 = \tilde{x}_{k-1}$; if the R-ACG stops with *success*, then let $(y_k^a, v_k, \varepsilon_k)$ denote its output and go to step 3; otherwise, **stop** with a *failure* status;

Step 3 if the inequality $\Delta_1(y_{k-1}; y_k^a, v_k) \leq \varepsilon_k$ holds, then go to step 4; otherwise, **stop** with a *failure* status;

Step 4 set $(\hat{y}_k, \hat{v}_k) = SRP(y_k^a, v_k, \tilde{x}_{k-1})$ where $SRP(\cdot,\cdot,\cdot)$ is described in Subsection 3.1; if $\|\hat{v}_k\| \leq \hat{\rho}$ then **stop** with a *success* status and **output** $(\hat{y}, \hat{v}) = (\hat{y}_k, \hat{v}_k)$; otherwise, compute

$$\begin{aligned} x_k &= \operatorname*{argmin}_{u \in \Omega} \frac{1}{2} \left\| u - \left[ x_{k-1} - a_{k-1} \left( v_k + \tilde{x}_{k-1} - y_k^a \right) \right] \right\|^2, \\ y_k &= \operatorname*{argmin}_{u \in \{y_{k-1}, y_k^a\}} \left[ f_1(u) + f_2(u) + h(u) \right], \end{aligned} \tag{34}$$

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

---

Note that, similar to the static IA-ICG method, the static DA-ICG method may fail without obtaining a pair satisfying (22). Proposition 5(c) shows that a sufficient condition for the method to stop successfully is that $f_2$ be convex. Using arguments similar to the ones

employed to derive the dynamic IA-ICG method, a dynamic version of DA-ICG method can also be developed that repeatedly invokes the static DA-ICG in place of the static IA-ICG.

We now make some additional remarks about the above method. First, it performs two kinds of iterations, namely, ones that are indexed by $k$ and ones that are performed by the R-ACG algorithm. We refer to the former kind as outer iterations and the latter kind as inner iterations. Second, in view of the update for $y_k$ in (34), the collection of function values $\{\phi(y_i)\}_{i=0}^k$ is non-increasing. Third, in view of (32), if $M_1 > 0$ then $0 < \lambda < (1-2\sigma^2)/(2M_1)$ whereas if $M_1 \leq 0$ then $0 < \lambda < \infty$.

It is worth mentioning that the outer iteration scheme of the DA-ICG method is a monotone and inexact generalization of the accelerated gradient (AG) method in (Ghadimi and Lan, 2016). More specifically, the AG method can be viewed as a version of the DA-ICG method where: (i) $\sigma = 0$; (ii) the R-ACG algorithm in step 2 is replaced by an exact solver of (23); and (iii) the update of $x_k$ in (34) is replaced by an update involving prox evaluation of the function $a_{k-1}(f_2 + h)$. Hence, the DA-ICG method can be significantly more efficient when its R-ACG call is more efficient than an exact solver of (23) and/or when the projection onto $\Omega$ is more efficient than evaluating the prox of $a_{k-1}(f_2 + h)$.

The next result summarizes some facts about the DA-ICG method. Before proceeding, we introduce the useful constants

$$\begin{aligned} D_h &:= \sup_{u,z\in \operatorname{dom} h} \|u - z\|, \quad D_\Omega := \sup_{u,z\in\Omega} \|u - z\|, \quad \Delta_\phi^0 := \phi(y_0) - \phi_*, \\ d_0 &:= \inf_{u^*\in\mathcal{Z}} \{\|y_0 - u^*\| : \phi(u^*) = \phi_*\}, \quad E_{\lambda,\sigma} := \sqrt{\lambda} L_1 + \frac{1 + \sigma \overline{C}_\lambda}{\sqrt{\lambda}}. \end{aligned} \tag{35}$$

**Theorem 5.** *The following statements hold about the static DA-ICG method:*

*(a) it stops in*

$$\mathcal{O}_1 \left( \frac{E_{\lambda,\sigma}^2 [m_1^+ D_h^2 + \Delta_\phi^0]}{\hat{\rho}^2} + \frac{E_{\lambda,\sigma} [m_1^+ + 1/\lambda]^{1/2} D_\Omega}{\hat{\rho}} \right) \tag{36}$$

*outer iterations;*

*(b) if it stops with success, then its output pair $(\hat{y}, \hat{v})$ is a $\hat{\rho}$–approximate solution of (20);*

*(c) if $f_2$ is convex, then it always stops with success in*

$$\mathcal{O}_1 \left( \frac{E_{\lambda,\sigma}^2 m_1^+ D_h^2}{\hat{\rho}^2} + \frac{E_{\lambda,\sigma} [m_1^+]^{1/2} D_\Omega}{\hat{\rho}} + \frac{E_{\lambda,\sigma}^{2/3} d_0^{2/3} \lambda^{-1/3}}{\hat{\rho}^{2/3}} \right) \tag{37}$$

*outer iterations.*

We now make three remarks about the above results. First, in the "best" scenario of $\max\{m_1, m_2\} \leq 0$, we have that (37) reduces to

$$\mathcal{O}_1 \left( \left[ L_1 + \frac{1}{\lambda} \right]^{2/3} \left[ \frac{d_0^{2/3}}{\hat{\rho}^{2/3}} \right] \right),$$

which has a smaller dependence on $\hat{\rho}$ when compared to (29). In the "worst" scenario of $\min\{m_1, m_2\} > 0$, if we take $\sigma = \mathcal{O}(1/\overline{C}_\lambda)$, then (36) reduces to

$$\mathcal{O}_1\left(\left[\sqrt{\lambda}L_1 + \frac{1}{\sqrt{\lambda}}\right]^2\left[\frac{m_1^+ D_h^2 + \phi(y_0) - \phi_*}{\hat{\rho}^2}\right]\right),$$

which has the same dependence on $\hat{\rho}$ as in (29). Second, part (c) shows that if the method stops with an output pair $(\hat{y}, \hat{v})$, regardless of the convexity of $f_2$, then that pair is always an approximate solution of (20). Third, in view of Proposition 18, the quantities $L_1$ and $\overline{C}_\lambda$ in all of the previous complexity results can be replaced by their averaged counterparts in (57). As these averaged quantities only depend on $\{(y_i^a, \hat{y}_i, \tilde{x}_{i-1})\}_{i=1}^k$, we can infer that the static DA-ICG method, like the static IA-ICG method of the previous subsection, also adapts to the local geometry of its input functions.

## 4. Exploiting the Spectral Decomposition

Recall that at every outer iteration of the ICG methods in Section 3, a call to the R-ACG algorithm is made to tentatively solve Problem $\mathcal{B}$ (see Subsection 3.1) associated with (23). Our goal in this section is to present a significantly more efficient algorithm (based on the idea outlined in Section 1) for solving the same problem when the underlying problem of interest is (1).

The content of this section is divided into two subsections. The first one presents the aforementioned algorithm, whereas the second one proves its key properties.

### 4.1 Spectral R-ACG Algorithm

This subsection presents an efficient algorithm for solving Problem $\mathcal{B}$ associated with (23). Throughout our presentation, we let $Z_0$ represent the starting point given to the R-ACG algorithm by the two ICG methods.

We first state the aforementioned efficient algorithm.

---

**Spectral R-ACG Algorithm**

---

**Input**: a quadruple $(M_2, f_1, f_2^{\mathcal{V}}, h^{\mathcal{V}})$ satisfying (A1)–(A3) with $(f_2, h) = (f_2^{\mathcal{V}}, h^{\mathcal{V}})$ and a triple $(\lambda, \sigma, Z_0)$;

**Output**: a triple $(Y, V, \varepsilon)$ that solves Problem $\mathcal{B}$ associated with (23) or a *failure* status;

Step 1 compute

$$Z_0^\lambda := Z_0 - \lambda \nabla f_1(Z_0), \tag{38}$$

and a pair $(P, Q) \in \mathcal{U}^m \times \mathcal{U}^n$ satisfying $Z_0^\lambda = P[\operatorname{dg}\sigma(Z_0^\lambda)]Q^*$;

Step 2 use the R-ACG algorithm to tentatively solve Problem $\mathcal{B}$ associated with (3), i.e., with inputs $(\mu, M, \psi_s^{\mathcal{V}}, \psi_n^{\mathcal{V}})$ and $(\sigma, z_0)$ where the former is given by

$$\begin{aligned} \mu = 1, \quad M := \lambda M_2^+ + 1, \\ \psi_s^{\mathcal{V}} := \lambda f_2^{\mathcal{V}} - \langle \sigma(Z_0^\lambda), \cdot \rangle + \frac{1}{2}\|\cdot\|^2, \quad \psi_n^{\mathcal{V}} := \lambda h^{\mathcal{V}}, \end{aligned} \tag{39}$$

and $z_0 = \operatorname{Dg}(P^* Z_0 Q)$; if the R-ACG stops with *success*, then let $(y, v, \varepsilon)$ denote its output and go to step 3; otherwise, **stop** with a *failure* status;

Step 3 set $Y = P(\operatorname{dg} y)Q^*$ and $V = P(\operatorname{dg} v)Q^*$, and output the triple $(Y, V, \varepsilon)$.

---

We now make three remarks about the above algorithm. First, the matrices $P$ and $Q$ in step 1 can be obtained by computing an SVD of $Z_0^\lambda$. Second, in view of Proposition 20(a) and the fact that $(\mu, M)$ in (39) and (26) are the same, the iteration complexity is the same as the vanilla R-ACG algorithm. Finally, because the functions $\psi_s^{\mathcal{V}}$ and $\psi_n^{\mathcal{V}}$ in (39) have vector inputs over $\mathbb{R}^r$, the steps in the spectral R-ACG algorithm are significantly less costly than the ones in the R-ACG algorithm, which use functions with matrix inputs over $\mathbb{R}^{m\times n}$.

The following result, whose proof is in the next subsection, presents the key properties of this algorithm.

**Proposition 6.** *The spectral R-ACG algorithm has the following properties:*

*(a) if it stops with success, then its output triple $(Y, V, \varepsilon)$ solves Problem $\mathcal{B}$ associated with (23);*

*(b) if $f_2$ is convex, then it always stops with success and its output $(Y, V, \varepsilon)$ solves Problem $\mathcal{A}$ associated with (23).*

### 4.2 Proof of Proposition 6

For the sake of brevity, let $(\psi_s, \psi_n)$ be as in (26) and, using $P$ and $Q$ from the spectral R-ACG algorithm, define for every $(u, U) \in \mathbb{R}^r \times \mathbb{R}^{m\times n}$, the functions

$$\mathcal{M}(u) := P(\operatorname{dg} u)Q^*, \quad \mathcal{V}(U) := \operatorname{Dg}(P^*UQ),$$
$$\psi(U) := \psi_s(U) + \psi_n(U), \quad \psi^{\mathcal{V}}(u) := \psi_s^{\mathcal{V}}(u) + \psi_n^{\mathcal{V}}(u).$$

The first result relates $(\psi_s, \psi_n)$ to $(\psi_s^{\mathcal{V}}, \psi_n^{\mathcal{V}})$.

**Lemma 7.** *Let $(y, v, \varepsilon)$ and $(Y, V)$ be as in the spectral R-ACG algorithm. Then, the following properties hold:*

*(a) we have*

$$\psi_n^{\mathcal{V}}(y) = \psi_n(Y), \quad \psi_s^{\mathcal{V}}(y) + B_0^\lambda = \psi_s(Y),$$

*where $B_0^\lambda := \lambda f_1(Z_0) - \lambda\langle \nabla f_1(Z_0), Z_0\rangle + \|Z_0\|_F^2/2$;*

*(b) we have*

$$V \in \partial_\varepsilon \left(\psi - \frac{1}{2}\|\cdot - Y\|_F^2\right)(Y) \iff v \in \partial_\varepsilon \left(\psi^{\mathcal{V}} - \frac{1}{2}\|\cdot - y\|^2\right)(y). \tag{40}$$

*Proof.* (a) The relationship between $\psi_n^{\mathcal{V}}$,and $\psi_n$ is immediate. On the other hand, using the definitions of $Y, f_2$, and $B_0^\lambda$, we have

$$\begin{aligned}\psi_s^{\mathcal{V}}(y) + B_0^\lambda &= \lambda f_2(Y) - \langle Z_0^\lambda, Y\rangle + \frac{1}{2}\|Y\|_F^2 + B_0^\lambda \\ &= \lambda\left[f_2(Y) + f_1(Z_0) + \langle \nabla f_1(Z_0), Y - Z_0\rangle\right] + \frac{1}{2}\|Y - Z_0\|_F^2 = \psi_s(Y).\end{aligned}$$

(b) Let $S_0 = V + Z_0^\lambda - Y$ and $s_0 = v + \sigma(Z_0^\lambda) - y$, and note that $S_0 = \mathcal{M}(s_0)$. Moreover, in view of part (a) and the definition of $\psi$, observe that the left inclusion in (40) is equivalent

to $S_0 \in \partial_\varepsilon(\lambda[f_2+h])(Y)$. Using this observation, the fact that $S_0$ and $Y$ have a simultaneous SVD, and Theorem 23 with $(S,s) = (S_0, s_0)$, $\Psi = \lambda[f_2 + h]$, and $\Psi^{\mathcal{V}} = \lambda[f_2^{\mathcal{V}} + h^{\mathcal{V}}]$, we have that the left inclusion in (40) is also equivalent to $s_0 \in \partial_\varepsilon(\lambda[f_2^{\mathcal{V}}+h^{\mathcal{V}}])(y)$. The conclusion now follows from the observing that the latter inclusion is equivalent to the the right inclusion in (40). ■

We are now ready to give the proof of Proposition 6.

*Proof of Proposition 6.* (a) Let $(y, v) = (\mathcal{V}(Y), \mathcal{V}(V))$ and remark that the successful termination of the algorithm implies that the inequality in (8) and (15) hold. Using this remark, the fact that $\|V\|_F^2 = \|v\|^2$, and the bound

$$\begin{aligned}\sigma^2\|z_j - z_0\|^2 &= \sigma^2\left(\|z_j\|^2 - 2\langle z_j, \mathcal{V}(z_0)\rangle + \|Z_0\|_F^2\right) + \sigma^2(\|\mathcal{V}(z_0)\|^2 - \|Z_0\|_F^2)\\ &\leq \sigma^2\left(\|Z_j\|^2 - 2\langle Z_j, Z_0\rangle + \|Z_0\|_F^2\right) = \sigma^2\|Z_j - Z_0\|_F^2, \end{aligned} \tag{41}$$

we then have that the inequality in (8) also holds with $(y, v) = (Y, V)$.

To show the corresponding inequality for (15), let $(Y_r, V_r) = RP(Y, V)$ using the refinement procedure in Section 2. Moreover, let $(y_r, v_r) = RP(y, v)$ and $\Delta_1^{\mathcal{V}}(\cdot;\cdot,\cdot)$ be as in (9), where $(\psi_s, \psi_n) = (\psi_s^{\mathcal{V}}, \psi_n^{\mathcal{V}})$. It now follows from (10), (11), Lemma 22 with $\Psi = \psi_n$ and $S = V + MY - \nabla\psi_s(Y)$, and Lemma 21(b) that $Y_r, Y, V$, and $V_r$ have a simultaneous SVD. As a consequence of this, the first remark, and Lemma 7(a), we have that

$$\begin{aligned}\varepsilon \geq \Delta_1^{\mathcal{V}}(y_r; y, v) &= \psi^{\mathcal{V}}(y) - \psi^{\mathcal{V}}(y_r) - \langle v, y_r - y\rangle + \frac{1}{2}\|y_r - y\|^2\\ &= \psi(Y) - \psi(Y_r) - \langle V, Y_r - Y\rangle + \frac{1}{2}\|Y_r - Y\|^2 = \Delta_1(Y_r; Y, V),\end{aligned}$$

and hence that (15) holds with $(y, v) = (Y, V)$.

(b) This follows from part (a), Proposition 2(c), and Lemma 7(b). ■

## 5. Computational Results

This section presents computational results that highlight the performance of the IA-ICG and DA-ICG methods, and it contains three subsections. The first one describes the implementation details, the second presents computational results related to a set of spectral composite problem, while the third gives some general comments about the computational results.

### 5.1 Implementation Details

This subsection precisely describes the implementation of the methods and experiments of this section.

We first describe some practical modifications to the IA-ICG method. Given $\lambda > 0$ and $(z_j, z_0) \in \mathcal{Z}^2$, denote

$$\Delta_\phi^\lambda = 4\lambda\left[\phi(z_0) - \widetilde{\ell}_\phi(z_j; z_0) - \frac{M_1}{2}\|z_j - z_0\|^2\right]$$

where $\widetilde{\ell}_\phi$ is as in (27). Motivated by the first inequality in the descent condition (42), we relax (17) in the R-ACG call to the three separate conditions: $\|z_j - z_0\|^2 \leq \Delta_\phi^\lambda$, $\|r_j\|^2 \leq \Delta_\phi^\lambda$, and $2\eta_j \leq \Delta_\phi^\lambda$.

We now describe some modifications and parameter choices that are common to both methods. First, both ICG methods use the spectral R-ACG algorithm of Subsection 4.1 in place of the R-ACG algorithm of Section 2. Moreover, this R-ACG variant uses a line search subroutine for estimating the upper curvature $M$ that is used during its execution. Second, when each of the dynamic ICG methods invoke their static counterparts, the parameters $A_0$ and $y_0$ are set to be the last obtained parameters of the previous invocation or the original parameters if it is the first invocation, i.e., we implement a warm–start strategy. Third, we adaptively update $\lambda$ at each outer iteration as follows: given the old value of $\lambda = \lambda_{\text{old}}$ at the $k^{\text{th}}$ outer iteration, the new value of $\lambda = \lambda_{\text{new}}$ at the $(k+1)^{\text{th}}$ iteration is given by

$$\lambda_{\text{new}} = \begin{cases} \lambda_{\text{old}}, & r_k \in [0.5, 2.0], \\ \lambda_{\text{old}} \cdot \sqrt{0.5}, & r_k < 0.5, \\ \lambda_{\text{old}} \cdot \sqrt{2}, & r_k > 2.0, \end{cases} \qquad r_k = \frac{\left[\lambda(M_2^+ + 2m_2^+) + 1\right] \|y_k - \hat{y}_k\|}{\|\hat{v}_k - \left[\lambda(M_2^+ + 2m_2^+) + 1\right](y_k - \hat{y}_k)\|}.$$

Fourth, we take $\mu = 1/2$ rather than $\mu = 1$ for each of R-ACG calls in order to reduce the possibility of a failure from the R-ACG algorithm. Fifth, in view of (41), we relax condition (17) in the vector-based R-ACG call of Subsection 4.1 to

$$\|r_j\|^2 + 2\eta_j \leq \sigma^2 \|z_j - z_0\|^2 + \tau,$$

where $\tau := \sigma^2(\|Z_0\|_F^2 - \|z_0\|^2) \geq 0$. Finally, both ICG methods choose the common hyper-parameters $(\xi_0, \lambda, \sigma) = (M_1, 5/M_1, 1/2)$ at initialization.

We now describe the two other benchmark methods considered. The first is the ECG method described in Section 1 with $\lambda = 1/M_1$. The second is a modification of the accelerated gradient (AG) method that was proposed and analyzed in (Ghadimi and Lan, 2016). More specifically, the implementation is a modification of Algorithm 2 in (Ghadimi and Lan, 2016, Section 2) in which $\alpha_k = 2/(k+1)$, $\beta_k = 1/[2(M_1 + M_2)]$, and $\lambda_k = k\beta_k/2$ for every $k \geq 1$.

Finally, we state some additional details about the numerical experiments. First, the problems considered are of the form in (1) and satisfy assumptions (A1)–(A4) with $f_2 = f_2^{\mathcal{V}} \circ \sigma$ and $h = h^{\mathcal{V}} \circ \sigma$. Second, given a tolerance $\hat{\rho} > 0$ and an initial point $Y_0 \in \operatorname{dom} h$, every method in this section seeks a pair $(\hat{Y}, \hat{V}) \in \operatorname{dom} h \times \mathbb{R}^{m \times n}$ satisfying

$$\hat{V} \in \nabla f_1(\hat{Y}) + \nabla(f_2^{\mathcal{V}} \circ \sigma)(\hat{Y}) + \partial(h^{\mathcal{V}} \circ \sigma)(\hat{Y}), \quad \frac{\|\hat{V}\|}{\|\nabla f_1(Y_0) + (f_2^{\mathcal{V}} \circ \sigma)(Y_0)\| + 1} \leq \hat{\rho}.$$

Finally, all described algorithms are implemented in MATLAB 2020a and are run on Linux 64-bit machines that contain at least 8 GB of memory.

### 5.2 Spectral Composite Problems

This subsection presents computational results of a set of spectral composite optimization problems and contains two sub-subsections. The first one examines a class of nonconvex matrix completion problems, while the second one examines a class of blockwise matrix completion problems.

| Name | $m$ | $n$ | % nonzero | $\min_{i,j} A_{ij}$ | $\max_{i,j} A_{ij}$ |
|---|---|---|---|---|---|
| Jester[1] | 24938 | 100 | 24.66% | -9.95 | 10 |
| Anime[2] | 506 | 9437 | 10.50% | 1 | 10 |
| MovieLens 100K[3] | 610 | 9724 | 1.70% | 0.5 | 5 |
| FilmTrust[4] | 1508 | 2071 | 1.14% | 0.5 | 8 |
| MovieLens 1M[5] | 6040 | 3952 | 4.19% | 1 | 5 |

Table 1: Description of the MC data matrices $A \in \mathbb{R}^{m\times n}$.

#### 5.2.1 Matrix completion

Given a quadruple $(\alpha, \beta, \mu, \theta) \in \mathbb{R}^4_{++}$, a data matrix $A \in \mathbb{R}^{m\times n}$, and indices $\Omega$, this subsection considers the following constrained matrix completion (MC) problem:

$$\begin{aligned} \min_{U\in\mathbb{R}^{m\times n}} \ & \frac{1}{2}\|P_\Omega(U-A)\|_F^2 + \kappa_\mu \circ \sigma(U) + \tau_\alpha \circ \sigma(U) \\ \text{s.t.} \ & \|U\|_F^2 \le \sqrt{mn}\cdot \max_{i,j} |A_{ij}|, \end{aligned}$$

where $P_\Omega$ is the linear operator that zeros out any entry that is not in $\Omega$ and

$$\kappa_\mu(z) = \frac{\mu\beta}{\theta}\sum_{i=1}^n \log\left(1+\frac{|z_i|}{\theta}\right), \quad \tau_\alpha(z) = \alpha\beta\left[1-\exp\left(-\frac{\|z\|_2^2}{2\theta}\right)\right]$$

for every $z \in \mathbb{R}^n$. Here, the function $\kappa_\mu + \tau_\alpha$ is a nonconvex generalization of the convex elastic net regularizer (see, for example, Sun and Zhang, 2012), and it is well-known (see, for example, Yao and Kwok, 2017) that the function $\kappa_\mu - \mu\|\cdot\|_*$ is concave, differentiable, and has a $(2\beta\mu/\theta^2)$-Lipschitz continuous gradient.

We now describe the different data matrices that are considered. Each matrix $A \in \mathbb{R}^{m\times n}$ is obtained from a different collaborative filtering system where each row represents a unique user, each column represents a unique item, and each entry represents a particular rating. Table 1 lists the names of each data set, where the data originates from (in the footnotes), and some basic statistics about the matrices.

We now describe the experiment parameters considered. First the starting point $Z_0$ is randomly generated from a shifted binomial distribution that closely follows the data matrix $A$. More specifically, the entries of $Z_0$ are distributed according to a $\textsc{Binomial}(n, \mu/n) - \underline{A}$ distribution, where $\mu$ is the sample average of the nonzero entries in $A$, the integer $n$ is the ceiling of the range of ratings in $A$, and $\underline{A}$ is the minimum rating in $A$. Second, the decomposition of the objective function is as follows

$$f_1 = \frac{1}{2}\|P_\Omega(\cdot - A)\|_F^2, \quad f_2^{\mathcal{V}} = \mu\left[\kappa_\mu(\cdot) - \frac{\beta}{\theta}\|\cdot\|_1\right] + \tau_\alpha(\cdot), \quad h^{\mathcal{V}} = \frac{\mu\beta}{\theta}\|\cdot\|_1 + \delta_{\mathcal{F}}(\cdot),$$

1. See the ratings in the file "jester_dataset_1_1.zip" from http://eigentaste.berkeley.edu/dataset/.
2. See a subset of the ratings from https://www.kaggle.com/CooperUnion/anime-recommendations-database where each user has rated at least 720 items.
3. See the ratings in the file "ml-latest-small.zip" from https://grouplens.org/datasets/movielens/.
4. See the ratings in the file "ratings.txt" under the FilmTrust section in https://www.librec.net/datasets.html.
5. See the ratings in the file "ml-1m.zip" from https://grouplens.org/datasets/movielens/.

where $\mathcal{F} = \{U \in \mathbb{R}^{m\times n} : \|U\|_F \le \sqrt{mn}\cdot \max_{i,j}|A_{ij}|\}$ is the set of feasible solutions. Third, in view of the previous decomposition, the curvature parameters are set to be

$$m_1 = 0, \quad M_1 = 1, \quad m_2 = \frac{2\beta\mu}{\theta^2} + \frac{2\alpha\beta}{\theta}\exp\left(\frac{-3\theta}{2}\right), \quad M_2 = \frac{\alpha\beta}{\theta},$$

where it can be shown that the smallest and largest eigenvalues of $\nabla^2\tau_\alpha(z)$ are bounded below and above by $-2\alpha\beta\exp(-3\theta/2)/\theta$ and $\alpha\beta/\theta$, respectively, for every $z \in \mathbb{R}^n$. Fourth, each problem instance uses a specific data matrix $A$ from Table 1, the hyperparameters $(\alpha, \beta, \mu, \theta) = (10, 20, 2, 1)$ and $\hat{\rho} = 10^{-6}$, and $\Omega$ to be the index set of nonzero entries in the chosen matrix $A$. Finally, a cutoff time of 10800 seconds is used for the MovieLens 1M data set and a cutoff time of 7200 seconds is used for the other data sets.

We now present the results. Figure 1 contains the plots of the log objective function value against the runtime, listed in increasing order of the smallest dimension in the data matrix.

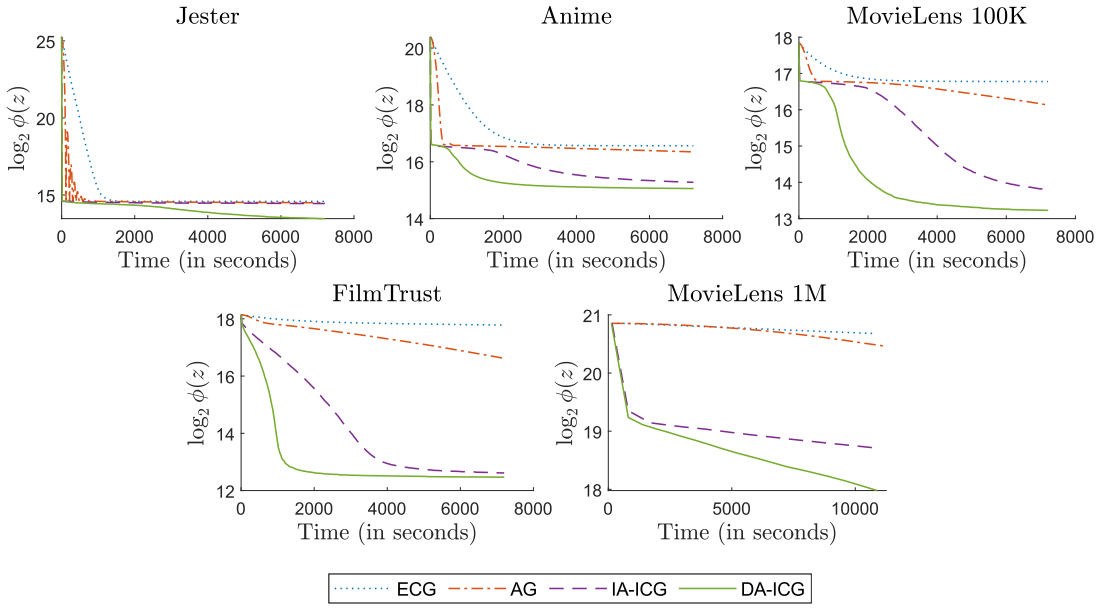


Figure 1: Function value vs. runtime for the MC problems.

### 5.2.2 Blockwise matrix completion

Given a quadruple $(\alpha, \beta, \mu, \theta) \in \mathbb{R}^4_{++}$, a block decomposable data matrix $A \in \mathbb{R}^{m\times n}$ with blocks $\{A_i\}_{i=1}^k \subseteq \mathbb{R}^{p\times q}$, and indices $\Omega$, this subsection considers the following constrained blockwise matrix completion (BMC) problem:

$$\begin{aligned}\min_{U\in\mathbb{R}^{m\times n}} \ & \frac{1}{2}\|P_\Omega(U - A)\|_F^2 + \sum_{i=1}^k [\kappa_\mu \circ \sigma(U_i) + \tau_\alpha \circ \sigma(U_i)] \\ \text{s.t.} \ & \|U\|_F^2 \le \sqrt{mn}\cdot \max_{i,j}|A_{ij}|,\end{aligned}$$

where $P_\Omega$, $\kappa_\mu$, and $\tau_\alpha$ are as in Subsection 5.2.1 and $U_i \in \mathbb{R}^{p\times q}$ is the $i^{\text{th}}$ block of $U$ with the same indices as $A_i$ with respect to $A$.

We now describe the two classes of data matrices that are considered. Every data matrix is a 5-by-5 block matrix consisting of 50-by-100 sized submatrices. Every submatrix contains only 25% nonzero entries and each data matrix generates its submatrix entries from different probability distributions. More specifically, for a sampled probability $p \sim \textsc{Uniform}[0,1]$ specific to a fixed submatrix, one class uses a $\textsc{Binomial}(n,p)$ distribution with $n = 10$, while the other uses a $\textsc{TruncatedNormal}(\mu,\sigma)$ distribution with $\mu = 10p$, $\sigma^2 = 10p(1-p)$, and upper and lower bounds 0 and 10, respectively.

We now describe the experiment parameters considered. First, the the decomposition of the objective function and the quantities $Z_0$, $(m_1, M_1)$, $(m_2, M_2)$, $\hat{\rho}$, and $\Omega$ are the same as in Subsection 5.2.1. Second, we fix $(\beta,\theta) = (20,1)$ and vary $(\alpha,\mu,A)$ across the different problem instances. Finally, a cutoff time of 7200 seconds is used for all of Problem instances tested.

We now present the results. Figure 2 contains the plots of the log objective function value against the runtime for the binomial data set, listed in increasing order of $M_2$. The corresponding plots for the truncated normal data set are similar to the binomial plots so we omit them for the sake of brevity. Tables 2 and 3 respectively contain the last function values of each algorithm for the binomial and truncated normal data sets, listed in increasing order of $M_2$. Moreover, each row of these tables corresponds to a different choice of $(\mu,\alpha)$ and the bolded numbers highlight which algorithm performed the best in terms of the last function value.

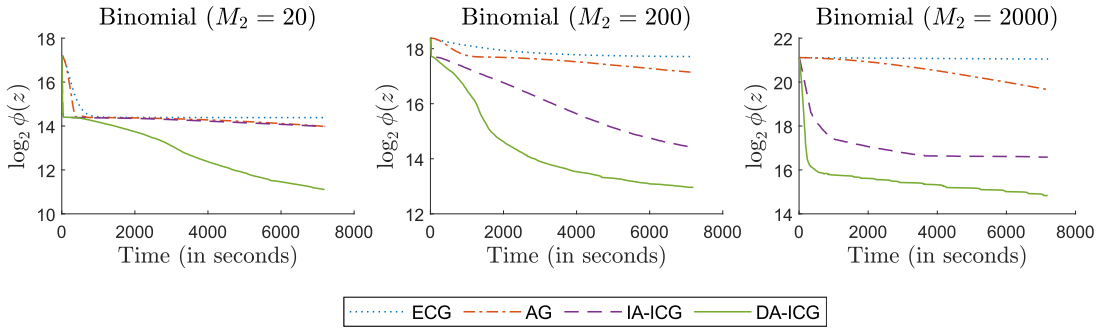


Figure 2: Function value vs. runtime for the binomial BMC problems.

| Parameters | | Last Function Value | | | |
|---|---|---|---|---|---|
| $(\mu,\alpha)$ | $M_2$ | ECG | AG | IA-ICG | DA-ICG |
| $(1, 0.2)$ | 20 | 2.13E+04 | 1.62E+04 | 1.61E+04 | **2.20E+03** |
| $(10, 2)$ | 200 | 2.15E+05 | 1.44E+05 | 2.19E+04 | **7.98E+03** |
| $(100, 20)$ | 2000 | 2.17E+06 | 8.24E+05 | 9.82E+04 | **2.92E+04** |

Table 2: Last function values for the binomial BMC problems.

| Parameters | | Last Function Value | | | |
|---|---|---|---|---|---|
| $(\mu, \alpha)$ | $M_2$ | ECG | AG | IA-ICG | DA-ICG |
| $(1, 0.2)$ | 20 | 2.14E+04 | 8.92E+03 | 1.26E+04 | **1.25E+03** |
| $(10, 2)$ | 200 | 2.21E+05 | 1.75E+05 | 3.29E+04 | **1.16E+04** |
| $(100, 20)$ | 2000 | 2.27E+06 | 1.71E+06 | 1.06E+05 | **4.50E+04** |

Table 3: Last function values for the truncated normal BMC problems.

### 5.3 General Comments

From the results of the previous subsection, we make a few comments. First, the DA-ICG and IA-ICG methods are generally more efficient than the AG and ECG methods. Second, the DA-ICG method appears to be able to escape local minima more quickly than the other methods. Third, the larger the constant $M_2$ is, the more efficient the ICG methods are compared to the benchmark methods. Finally, the larger the smallest dimension of the matrix space is, the more efficient the inexact methods are compared to the exact ones.

## 6. ICG Iteration Complexities

This section establishes the iteration complexities for each of the static ICG methods in Section 3.

### 6.1 IA-ICG Iteration Complexity

This subsection establishes the key properties of the static IA-ICG method.

**Lemma 8.** *Let $\{(y_i, \hat{y}_i, \hat{v}_i)\}_{i=1}^k$ be the collection of iterates generated by the static IA-ICG method. For every $i \geq 1$, we have*

$$\frac{1}{4\lambda}\|y_{i-1} - y_i\|^2 \leq \phi(y_{i-1}) - \widetilde{\ell}_\phi(y_i; y_{i-1}) - \frac{M_1}{2}\|y_i - y_{i-1}\|^2 \leq \phi(y_{i-1}) - \phi(y_i), \tag{42}$$

*where $\widetilde{\ell}_\phi$ is as in (27).*

*Proof.* Let $i \geq 1$ be fixed and let $(y_i, v_i, \varepsilon_i)$ be the point output by the $i^{\text{th}}$ successful call to the R-ACG algorithm. Moreover, let $\Delta_1(\cdot;\cdot,\cdot)$ be as in (9) with $(\psi_s, \psi_n)$ given by (26). Using the definition of $\widetilde{\ell}_\phi$, step 2 of the method, and fact that $(y^a, v, \varepsilon) = (y_i, v_i, \varepsilon_i)$ solves Problem $\mathcal{B}$ in Section 2 with $(\mu, \psi_s, \psi_n)$ as in (26), we have that

$$\varepsilon_i \geq \Delta_1(y_{i-1}; y_i, v_i) = \lambda\widetilde{\ell}_\phi(y_i; y_{i-1}) - \lambda\phi(y_{i-1}) - \langle v_i, y_i - y_{i-1}\rangle + \|y_i - y_{i-1}\|^2.$$

Rearranging the above inequality and using assumption (A2), (25), and the fact that $\langle a, b\rangle \geq -\|a\|^2/2 - \|b\|^2/2$ for every $a, b \in \mathcal{Z}$ yields

$$\begin{aligned}\lambda\phi(y_{i-1}) - \lambda\widetilde{\ell}_\phi(y_i; y_{i-1}) &\geq \langle v_i, y_{i-1} - y_i\rangle - \varepsilon_i + \|y_i - y_{i-1}\|^2 \\ &= \frac{1}{2}\|y_i - y_{i-1}\|^2 - \frac{1}{2}\left(\|v_i\|^2 + 2\varepsilon_i\right) \geq \left(\frac{1-\sigma^2}{2}\right)\|y_i - y_{i-1}\|^2\end{aligned}$$

$$= \frac{\lambda M_1}{2}\|y_i - y_{i-1}\|^2 + \left(\frac{1 - \lambda M_1 - \sigma^2}{2}\right)\|y_i - y_{i-1}\|^2$$
$$= \frac{\lambda M_1}{2}\|y_i - y_{i-1}\|^2 + \frac{1}{4}\|y_i - y_{i-1}\|^2.$$

Rearranging terms yields the first inequality of (42). The second inequality of (42) follows from the first inequality, the fact that $\widetilde{\ell}_\phi(y_i; y_{i-1}) + M_1\|y_i - y_{i-1}\|^2/2 \geq \phi(y_i)$ from assumption (A2), and the definition of $\widetilde{\ell}_\phi$. ■

The next results establish the rate at which the residual $\|\hat{v}_i\|$ tend to 0.

**Lemma 9.** *Let $p > 1$ be given. Then, for every $a, b \in \mathbb{R}^k$, we have*

$$\min_{1\leq i\leq k}\{|a_i b_i|\} \leq k^{-p}\,\|a\|_1\,\|b\|_{1/(p-1)}\,.$$

*Proof.* Let $p > 1$ and $a, b \in \mathbb{R}^k$ be fixed and let $q \geq 1$ be such that $p^{-1} + q^{-1} = 1$. Using the fact that $\langle x, y\rangle \leq \|x\|_p\|y\|_q$ for every $x, y \in \mathbb{R}^k$, and denoting $\tilde{a}$ and $\tilde{b}$ to be vectors with entries $|a_i|^{1/p}$ and $|b_i|^{1/p}$, respectively, we have that

$$k \min_{1\leq i\leq k}\{|a_i b_i|\}^{1/p} \leq \sum_{i=1}^k |a_i b_i|^{1/p}$$
$$\leq \|\tilde{a}\|_p\|\tilde{b}\|_q = \|a\|_1^{1/p}\left(\sum_{i=1}^k |b_i|^{q/p}\right)^{1/q} = \left(\|a\|_1\|b\|_{q/p}\right)^{1/p}.$$

Dividing by $k$, taking the $p^{\text{th}}$ power on both sides, and using the fact that $p/q = p - 1$, yields

$$\min_{1\leq i\leq k}\{|a_i b_i|\} \leq k^{-p}\|a\|_1\|b\|_{q/p} = k^{-p}\|a\|_1\|b\|_{1/(p-1)}.$$

■

**Proposition 10.** *Let $\{(y_i, \hat{y}_i, \hat{v}_i)\}_{i=1}^k$ be as in Lemma 8 and define the quantities*

$$L_{1,k}^{\text{avg}} := \frac{1}{k}\sum_{i=1}^k L_1(y_i, y_{i-1}), \quad C_{\lambda,k}^{\text{avg}} := \frac{1}{k}\sum_{i=1}^k C_\lambda(\hat{y}_i, y_i), \tag{43}$$

*where $C_\lambda(\cdot,\cdot)$ and $\overline{C}_\lambda$ are as in (24) and (27), respectively. Then, we have*

$$\min_{i\leq k}\|\hat{v}_i\| = \mathcal{O}_1\left(\left[\sqrt{\lambda}L_{1,k}^{\text{avg}} + \frac{1 + \sigma C_{\lambda,k}^{\text{avg}}}{\sqrt{\lambda}}\right]\left[\frac{\phi(z_0) - \phi_*}{k}\right]^{1/2}\right) + \frac{\hat{\rho}}{2}.$$

*Proof.* Using Lemma 3 with $(y, w) = (y_i, y_{i-1})$ and the fact that $C_\lambda(\cdot,\cdot) \leq \overline{C}_\lambda$ and $L_1(\cdot,\cdot) \leq L_1$, we have $\|\hat{v}_i\| \leq \mathcal{E}_i\|y_i - y_{i-1}\|$, for every $i \leq k$, where

$$\mathcal{E}_i := \frac{2 + \lambda L_1(y_i, y_{i-1}) + \sigma C_\lambda(\hat{y}_i, y_i)}{\lambda} \quad \forall i \geq 1.$$

As a consequence, using the sum of the second bound in Lemma 8 from $i = 1$ to $k$, the definitions in (43), and Lemma 9 with $p = 3/2$, $a_i = \mathcal{E}_i$, and $b_i = \|y_i - y_{i-1}\|$ for $i = 1$ to $k$, yields

$$\min_{i\le k} \|\hat{v}_i\| \le \min_{i\le k} \mathcal{E}_i \|y_i - y_{i-1}\| \le \frac{1}{k^{3/2}} \left(\sum_{i=1}^{k} \mathcal{E}_i\right) \left(\sum_{i=1}^{k} \|y_i - y_{i-1}\|^2\right)^{1/2}$$
$$= \mathcal{O}_1\left(\left[\sqrt{\lambda} L_{1,k}^{\text{avg}} + \frac{1 + \sigma C_{\lambda,k}^{\text{avg}}}{\sqrt{\lambda}}\right]\left[\frac{\phi(z_0) - \phi_*}{k}\right]^{1/2}\right).$$

■

We are now ready to give the proof of Theorem 4.

*Proof of Theorem 4.* (a) This follows from Proposition 10, the fact that $C_\lambda(\cdot,\cdot) \le \overline{C}_\lambda$ and $L_{f_1}(\cdot,\cdot) \le L_1$, and the stopping condition in step 3.

(b) The fact that $(\hat{y}, \hat{v}) = (\hat{y}_k, \hat{v}_k)$ satisfies the inclusion of (22) follows from Lemma 3 with $(y, v, w) = (y_k, v_k, y_{k-1})$. The fact that $\|\hat{v}\| \le \hat{\rho}$ follows from the stopping condition in step 3.

(c) This follows from Proposition 2(c) and the fact that method stops in finite number of iterations from part (a). ■

### 6.2 DA-ICG Iteration Complexity

This subsection establishes several key properties of static DA-ICG method.

To avoid repetition, we assume throughout this subsection that $k \ge 1$ denotes an arbitrary successful outer iteration of the DA-ICG method and let

$$\{(a_i, A_i, y_i, y_i^a, x_i, \tilde{x}_{i-1}, \hat{y}_i, \hat{v}_i, v_i, \varepsilon_i)\}_{i=1}^{k}$$

denote the sequence of all iterates generated by it up to and including the $k^{\text{th}}$ iteration. Observe that this implies that the $i^{\text{th}}$ DA-ICG outer iteration for any $1 \le i \le k$ is successful, i.e., the (only) R-ACG call in step 2 of the DA-ICG method does not stop with failure and $\Delta_1(y_{i-1}; y_i^a, v_i) \le \varepsilon_i$. Moreover, throughout this subsection we let

$$\widetilde{\gamma}_i(u) = \ell_{f_1}(u; \tilde{x}_{i-1}) + f_2(u) + h(u), \quad \gamma_i(u) = \widetilde{\gamma}_i(y_i^a) + \frac{1}{\lambda}\langle v_i + \tilde{x}_{i-1} - y_i^a, u - y_i^a\rangle. \tag{44}$$

The first set of results present some basic properties about the functions $\widetilde{\gamma}_i$ and $\gamma_i$ as well as the iterates generated by the method.

**Lemma 11.** *Let $\Delta_1(\cdot;\cdot,\cdot)$ be as in (9) with $(\psi_s, \psi_n)$ given by (26). Then, the following statements hold for any $s \in \operatorname{dom} h$ and $1 \le i \le k$:*

*(a)* $\gamma_i(y_i^a) = \widetilde{\gamma}_i(y_i^a)$;
*(b)* $x_i = \operatorname{argmin}_{u\in\Omega} \{\lambda a_{i-1}\gamma_i(u) + \|u - x_{i-1}\|^2/2\}$;
*(c)* $y_i^a - v_i = \operatorname{argmin}_{u\in\mathcal{Z}} \{\lambda\gamma_i(u) + \|u - \tilde{x}_{i-1}\|^2/2\}$;
*(d)* $-M_1\|u - \tilde{x}_{i-1}\|^2/2 \le \widetilde{\gamma}_i(u) - \phi(u) \le m_1\|u - \tilde{x}_{i-1}\|^2/2$;
*(e)* $\phi(y_{i-1}) \ge \phi(y_i)$ *and* $\phi(y_i^a) \ge \phi(y_i)$.

*Proof.* To keep the notation simple, denote

$$\begin{aligned}(y_+^a, y_+, y, \tilde{x}) = (y_i^a, y_i, y_{i-1}, \tilde{x}_{i-1}), \quad (x_+, x) = (x_i, x_{i-1}),\\ (A_+, A, a) = (A_i, A_{i-1}, a_{i-1}), \quad (v, \varepsilon) = (v_i, \varepsilon_i).\end{aligned} \tag{45}$$

(a) This is immediate from the definitions of $\gamma$ and $\widetilde{\gamma}$ in (44).

(b) Define $\widehat{x}_i := x_{k-1} - a_{k-1}\,(v_k + \tilde{x}_{k-1} - y_k^a)$. Using the definition of $\gamma$ in (44), we have that

$$\begin{aligned}&\operatorname*{argmin}_{u\in\Omega}\left\{\lambda a\gamma\left(u\right) + \frac{1}{2}\|u-x\|^2\right\} = \operatorname*{argmin}_{u\in\Omega}\left\{a\left\langle v+\tilde{x}-y_+^a, u-x\right\rangle + \frac{1}{2}\|u-x\|^2\right\}\\ &= \operatorname*{argmin}_{u\in\Omega}\frac{1}{2}\left\|u-\left(x-a\left[v+\tilde{x}-y_+^a\right]\right)\right\|^2 = \operatorname*{argmin}_{u\in\Omega}\frac{1}{2}\left\|u-\widehat{x}_+\right\|^2 = x_+.\end{aligned}$$

(c) Using the definition of $\gamma$ in (44), we have that

$$\lambda\nabla\gamma\left(y_+^a - v\right) + \left(y_+^a - v\right) - \tilde{x} = \left(v + \tilde{x} - y_+^a\right) + \left(y_+^a - v\right) - \tilde{x} = 0,$$

and hence, the point $y_+^a - v$ is the global minimum of $\lambda\gamma + \|\cdot - \tilde{x}\|^2/2$.

(d) This follows from inequality (21) with $i = 1$ and the definition of $\widetilde{\gamma}$ in (44).

(e) This follows immediately from the update rule of $y_i$ in (34). ∎

**Lemma 12.** *Let $w = \tilde{x}_{i-1}$, the pair $(\psi_n, \psi_s)$ be as in (26), and $\Delta_1(\cdot;\cdot,\cdot)$ be as in (9) with $(\psi_s, \psi_n)$ given by (26). Then, following statements hold:*

*(a) the triple $(y_i^a, v_i, \varepsilon_i)$ solves Problem $\mathcal{B}$ and satisfies $\Delta_1(y_{i-1}; y_i^a, v_i) \le \varepsilon$, and hence*

$$\|v_i\| + 2\varepsilon_i \le \sigma^2\|y_i^a - \tilde{x}_{i-1}\|^2, \quad \Delta_1(u; y_i^a, v_i) \le \varepsilon_i \quad \forall u \in \{\hat{y}_i, y_{i-1}\}, \tag{46}$$

*(b) if $f_2$ is convex, then $(y_i^a, v_i, \varepsilon_i)$ solves Problem $\mathcal{A}$;*

*(c) $\Delta_1(s; y_i^a, v_i) = \lambda[\gamma_i(s) - \widetilde{\gamma}_i(s)]$;*

*(d) $\Delta_1(y_i; y_i^a, v_i) \le \varepsilon$.*

*Proof.* (a) This follows from step 2 of the DA-ICG method and Proposition 2(b).

(b) This follows from steps 2 and 3 of the DA-ICG method, the fact that $h$ is convex, and Proposition 2(c) with $\psi_s = \widetilde{\gamma}_i + \|\cdot - \tilde{x}_{i-1}\|^2/2$.

(c) Using the definitions of $(\psi_s, \psi_n)$ and $(\gamma, \widetilde{\gamma})$ in (26) and (44), respectively, we have that

$$\begin{aligned}&\Delta_1(s; y_+^a, v) = (\psi_s + \psi_n)(y_+^a) - (\psi_s + \psi_n)(s) - \left\langle v, y_+^a - s\right\rangle + \frac{1}{2}\|s - y_+^a\|^2\\ &= \left[\lambda\widetilde{\gamma}(y_+^a) + \frac{1}{2}\|y_+^a - \tilde{x}\|^2\right] - \left[\lambda\widetilde{\gamma}(s) + \frac{1}{2}\|s-\tilde{x}\|^2\right] - \left\langle v, y_+^a - s\right\rangle + \frac{1}{2}\|s - y_+^a\|^2\\ &= \left[\lambda\gamma(s) + \frac{1}{2}\|s-\tilde{x}\|^2\right] - \left[\lambda\widetilde{\gamma}(s) + \frac{1}{2}\|s-\tilde{x}\|^2\right] = \lambda\gamma(s) - \lambda\widetilde{\gamma}(s).\end{aligned}$$

(d) If $y_i = y_{i-1}$, then this follows from step 3 of the method. On the other hand, if $y_i = y_i^a$, then this follows from part (c). ∎

We now state (without proof) some well-known properties of $A_i$ and $a_{i-1}$.

**Lemma 13.** *For every $1 \le i \le k$, we have that:*
*(a)* $a_{i-1}^2 = A_i$;
*(b)* $i^2/4 \le A_i \le i^2$.

The next two lemmas are technical results that are needed to establish the key inequality in Proposition 16.

**Lemma 14.** *For every $u \in \operatorname{dom} h$ and $1 \le i \le k$, we have that*

$$\frac{1}{2}\left(A_{i-1}\|y_{i-1} - \tilde{x}_{i-1}\|^2 + a_{i-1}\|u - \tilde{x}_{i-1}\|^2\right) \le 2D_\Omega^2 + a_{i-1}D_h^2.$$

*Proof.* Throughout the proof, we use the notation in (45). Using the relation $(p+q)^2 \le 2p^2 + 2q^2$ for every $p, q \in \mathbb{R}$, Lemma 13(a), the fact that $A \le A^+$, $x \in \Omega$, and $y \in \operatorname{dom} h$, and the definitions of $\tilde{x}$ in (33) and of $D_\Omega$ and $D_h$ in (35), we conclude that

$$\begin{aligned}
A\|y - \tilde{x}\|^2 + a\|u - \tilde{x}\|^2 &= A\left\|\frac{a}{A_+}(y - x)\right\|^2 + a\left\|\frac{A}{A_+}(u - y) + \frac{a}{A_+}(u - x)\right\|^2 \\
&\le \frac{A}{A_+}\left(\|(y - u) + (u - x)\|^2 + 2a\left[\frac{A^2}{A_+^2}\|u - y\|^2 + \frac{a^2}{A_+^2}\|u - x\|^2\right]\right) \\
&\le \frac{2A}{A^+}\left(\|u - y\|^2 + \|u - x\|^2\right) + 2a\|u - y\|^2 + \frac{2a}{A_+}\|u - x\|^2 \\
&\le 2\left[\|u - x\|^2 + (1 + a)\|u - y\|^2\right] \le 2[D_\Omega^2 + (1 + a)D_h^2].
\end{aligned}$$

The conclusion now follows from dividing both sides of the above inequalities by 2 and using the fact that $D_h \le D_\Omega$. ∎

**Lemma 15.** *For every $u \in \operatorname{dom} h$ and $1 \le i \le k$, we have that*

$$\begin{aligned}
&A_i\left[\phi(y_i) + \left(\frac{1 - \lambda M_1}{2\lambda}\right)\|y_i^a - \tilde{x}_{i-1}\|^2 - \frac{\|v_i\|^2}{2\lambda}\right] + \frac{1}{2\lambda}\|u - x_i\|^2 \\
&\le A_{i-1}\gamma_i(y_{i-1}) + a_{i-1}\gamma_i(u) + \frac{1}{2\lambda}\|u - x_{i-1}\|^2. \qquad (47)
\end{aligned}$$

*Proof.* Throughout the proof, we use the notation in (45). We first present two key expressions. First, using the definition of $\gamma$ in (44) and Lemma 11(c), it follows that

$$\begin{aligned}
\min_{u \in \mathcal{Z}}\left\{\lambda\gamma(u) + \frac{1}{2}\|u - \tilde{x}\|^2\right\} &= \lambda\widetilde{\gamma}(y_+^a) - \langle v + \tilde{x} - y_+^a, v\rangle + \frac{1}{2}\|v + \tilde{x} - y_+^a\|^2 \\
&= \lambda\widetilde{\gamma}(y_+^a) - \|v\|^2 - \langle v, \tilde{x} - y_+^a\rangle + \frac{1}{2}\|v + \tilde{x} - y_+^a\|^2 \\
&= \lambda\widetilde{\gamma}(y_+^a) - \frac{1}{2}\|v\|^2 + \frac{1}{2}\|\tilde{x} - y_+^a\|^2. \qquad (48)
\end{aligned}$$

Second, Lemma 11(b) and the fact that the function $a\gamma + \|\cdot - x\|^2/(2\lambda)$ is $(1/\lambda)$–strongly convex imply that

$$a\gamma(x_+) + \frac{1}{2\lambda}\|x_+ - x\|^2 \le a\gamma(u) + \frac{1}{2\lambda}\|u - x\|^2 - \frac{1}{2\lambda}\|u - x_+\|^2. \qquad (49)$$

Using (48), Lemma 11(d)–(e), Lemma 13(a), and the fact that $\gamma$ is affine, we have that

$$\begin{aligned}
&A_+\left[\phi(y_+)+\left(\frac{1-\lambda M_1}{2\lambda}\right)\|y_+^a-\tilde{x}\|^2\right] \le A_+\left[\widetilde{\gamma}\left(y_+^a\right)+\frac{1}{2\lambda}\|y_+^a-\tilde{x}\|^2\right]\\
&= A_+\left[\min_{u\in\mathcal{Z}}\left\{\gamma(u)+\frac{1}{2\lambda}\|u-\tilde{x}\|^2\right\}+\frac{\|v\|^2}{2\lambda}\right]\\
&\le A_+\left[\gamma\left(\frac{Ay+ax_+}{A_+}\right)+\frac{1}{2\lambda}\left\|\frac{Ay+ax_+}{A_+}-\frac{Ay+ax}{A_+}\right\|^2+\frac{\|v\|^2}{2\lambda}\right]\\
&= A\gamma(y)+a\gamma(x_+)+\frac{a^2}{2\lambda A_+}\|x-x_+\|^2+\frac{A_+}{2\lambda}\|v\|^2\\
&= A\gamma(y)+a\gamma(x_+)+\frac{1}{2\lambda}\|x-x_+\|^2+\frac{A_+}{2\lambda}\|v\|^2 \qquad (50)
\end{aligned}$$

The conclusion now follows from combining (49) with (50). ■

We now present an inequality that plays an important role in the analysis of the DA-ICG method.

**Proposition 16.** *Let $\Delta_1(\cdot;\cdot,\cdot)$ be as in (9) with $(\psi_s,\psi_n)$ as in (26), and define*

$$\theta_i(u) := A_i\left[\phi(y_i)-\phi(u)\right]+\frac{1}{2\lambda}\|u-x_i\|^2 \quad \forall i\ge 0. \qquad (51)$$

*For every $u\in\operatorname{dom} h$ satisfying $\Delta_1(u;y_i^a,v_i)\le\varepsilon$ and $1\le i\le k$, we have that*

$$\frac{A_i}{4\lambda}\|y_i^a-\tilde{x}_{i-1}\|^2 \le m_1^+\left(a_{i-1}D_h^2+2D_\Omega^2\right)+\theta_{i-1}(u)-\theta_i(u). \qquad (52)$$

*Proof.* Throughout the proof, we use the notation in (45) together with the notation $\theta=\theta_{i-1}$ and $\theta_+=\theta_i$. Let $u\in\operatorname{dom} h$ be such that $\Delta_1(u;y_+^a,v)\le\varepsilon$. Subtracting $A\phi(u)$ from both sides of the inequality in (47) and using the definition of $\theta_+$ we have

$$\begin{aligned}
&\frac{A_+}{2\lambda}\left[(1-\lambda M_1)\|y_+^a-\tilde{x}\|^2-\|v\|^2\right]+\theta_+(u)\\
&= \frac{A_+}{2\lambda}\left[(1-\lambda M_1)\|y_+^a-\tilde{x}\|^2-\|v\|^2\right]+A_+\left[\phi(y_+)-\phi(u)\right]+\frac{1}{2\lambda}\|u-y_+^a\|^2\\
&\le A\gamma(y)+a\gamma(u)-A\phi(u)+\frac{1}{2\lambda}\|u-x\|^2\\
&= a\left[\gamma(u)-\phi(u)\right]+A\left[\gamma(y)-\phi(y)\right]+\theta(u). \qquad (53)
\end{aligned}$$

Moreover, using Lemma 12(a) and (c), and with our assumption that $\Delta_1(u;y_+^a,v)\le\varepsilon$, we have that

$$\gamma(s)-\phi(s)=\widetilde{\gamma}(s)-\phi(s)+\frac{\Delta_1(s;y_+^a,v)}{\lambda}\le\frac{m_1^+}{2}\|s-\tilde{x}\|^2+\frac{\varepsilon}{\lambda} \quad \forall s\in\{u,y\}. \qquad (54)$$

Combining (53), (54), and Lemma 14 then yields

$$\frac{A_+}{2\lambda}\left[(1-\lambda M_1)\|y_+^a-\tilde{x}\|^2-\|v\|^2\right]+\theta_+(u)$$

$$\leq \frac{m_1^+}{2}\left[a\|u-\tilde{x}\|^2 + A\|y-\tilde{x}\|^2\right] + \frac{\varepsilon A_+}{\lambda} + \theta(u) \leq m_1^+\left(aD_h^2 + 2D_\Omega^2\right) + \frac{\varepsilon A_+}{\lambda} + \theta(u).$$

Re-arranging the above terms and using (32) together with the first inequality in (46), we conclude that

$$\begin{aligned} m_1^+\left(aD_h^2 + 2D_\Omega^2\right) + \theta(u) - \theta_+(u) &\geq \frac{A_+}{2\lambda}\left[(1-\lambda M_1)\|y_+^a - \tilde{x}\|^2 - \|v\|^2 - 2\varepsilon\right] \\ &\geq \frac{A_+(1-\lambda M_1 - \sigma^2)}{2\lambda}\|y_+^a - \tilde{x}\|^2 \geq \frac{A_+}{4\lambda}\|y_+^a - \tilde{x}\|^2. \end{aligned}$$

■

The following result describes some important technical bounds obtained by summing (52) for two different choices of $u$ (possibly changing with $i$) from $i = 1$ to $k$.

**Proposition 17.** *Let $\Delta_\phi^0$ and $d_0$ be as in (35) and define*

$$S_k := \frac{1}{4\lambda}\sum_{i=1}^k A_i\|y_i^a - \tilde{x}_{i-1}\|^2. \tag{55}$$

*Then, the following statements hold:*

*(a) $S_k = \mathcal{O}_1(k^2[m_1^+ D_h^2 + \Delta_\phi^0] + k[m_1^+ + 1/\lambda]D_\Omega^2)$;*

*(b) if $f_2$ is convex, then $S_k = \mathcal{O}_1(k^2 m_1^+ D_h^2 + k m_1^+ D_\Omega^2 + d_0^2/\lambda)$.*

*Proof.* (a) Let $\Delta_1(\cdot;\cdot,\cdot)$ be defined as in (9) with $(\psi_s, \psi_n)$ given by (26). Using (51), the fact that $x_i, y_i^a \in \Omega$, the fact that $A_i$ is nonnegative and increasing, and the definitions of $\theta_i$ and $D_\Omega$ in (51) and (35), respectively, we have that

$$\begin{aligned} \sum_{i=1}^k [\theta_{i-1}(y_i) - \theta_i(y_i)] &\leq \sum_{i=1}^k A_{i-1}\left[\phi(y_{i-1}) - \phi(y_i)\right] + \frac{1}{2\lambda}\sum_{i=1}^k \|y_i - x_{i-1}\|^2 \\ &\leq A_k \sum_{i=1}^k \left[\phi(y_{i-1}) - \phi(y_i)\right] + \frac{k}{2\lambda}D_\Omega^2 \leq A_k\left[\phi(y_0) - \phi_*\right] + \frac{k}{2\lambda}D_\Omega^2. \end{aligned} \tag{56}$$

Moreover, noting Lemma 12(d) and using Proposition 16 with $u = y_i$, we conclude that (52) holds with $u = y_i$ for every $1 \leq i \leq k$. Summing these $k$ inequalities and using (56), the definition of $S_k$ in (55), and Lemma 13(b) yields the desired conclusion.

(b) Assume now that $f_2$ is convex and let $y_*$ be a point such that $\phi(y_*) = \phi_*$ and $\|y_0 - y_*\| = d_0$. It then follows from Lemma 12(b) and Proposition 1(d) with $(y, v) = (y_i^a, v_i)$ that $\Delta_1(y_*; y_i^a, v_i) \leq \varepsilon$ for every $1 \leq i \leq k$. The conclusion now follows by using an argument similar to the one in (a) but which instead sums (52) with $u = y_*$ from $i = 1$ to $k$, and uses the fact that

$$\sum_{i=1}^k [\theta_{i-1}(y_*) - \theta_i(y_*)] = \theta_0(y_*) - \theta_k(y_*) \leq \frac{1}{2\lambda}\|y_0 - y_*\|^2 = \frac{d_0}{2\lambda},$$

where the inequality is due to the fact that $\theta_k(y_*) \geq 0$ (see Equation 51) and $A_0 = 0$. ■

We now establish the rate at which the residual $\|\hat{v}_i\|$ tends to 0.

**Proposition 18.** *Let $S_k$ be as in (55). Moreover, define the quantities*

$$L_{1,k}^{\text{avg}} := \frac{1}{k}\sum_{i=1}^{k} L_1(y_i^a, \tilde{x}_{i-1}), \quad C_{\lambda,k}^{\text{avg}} := \frac{1}{k}\sum_{i=1}^{k} C_\lambda(\hat{y}_i, y_i^a), \tag{57}$$

*where $C_\lambda(\cdot,\cdot)$ and $\overline{C}_\lambda$ are as in (24) and (27), respectively. Then, we have*

$$\min_{i\le k}\|\hat{v}_i\| = \mathcal{O}_1\left(\left[\sqrt{\lambda}L_{1,k}^{\text{avg}} + \frac{1+\sigma C_{\lambda,k}^{\text{avg}}}{\sqrt{\lambda}}\right]\left[\frac{S_k}{k^3}\right]^{1/2}\right) + \frac{\hat{\rho}}{2}.$$

*Proof.* Let $\ell = \lceil k/2 \rceil$. Using Lemma 3 with $(z,w) = (y_i^a, \tilde{x}_{i-1})$ and the bounds $C_\lambda(\cdot,\cdot) \le \overline{C}_\lambda$ and $L_1(\cdot,\cdot) \le L_1$ we have that $\|\hat{v}_i\| \le \mathcal{E}_i\|y_i^a - \tilde{x}_{i-1}\|$, for every $\ell \le i \le k$, where

$$\mathcal{E}_i = \frac{2 + \lambda L_1(y_i^a, \tilde{x}_{i-1}) + \sigma C_\lambda(\hat{y}_i, y_i^a)}{\lambda} \quad \forall i \ge 1.$$

As a consequence, using the definition of $S_k$ in (55), the definitions in (57), Lemma 9 with $p = 3/2$, $a_i = \mathcal{E}_i/\sqrt{A_i}$, and $b_i = \sqrt{A_i}\|y_i^a - \tilde{x}_{i-1}\|$ for $i \in \{\ell, ..., k\}$, Lemma 13(b), and the fact that $(k - \ell + 1) \ge k/2$, yields

$$\begin{aligned}
&\min_{\ell\le i\le k}\|\hat{v}_i\| \le \min_{\ell\le i\le k}\mathcal{E}_i\|y_i^a - \tilde{x}_{i-1}\| \\
&\le \frac{1}{(k-\ell+1)^{3/2}}\left(\sum_{i=\ell}^{k}\frac{\mathcal{E}_i}{\sqrt{A_i}}\right)\left(\sum_{i=\ell}^{k} A_i\|y_i^a - \tilde{x}_{i-1}\|^2\right)^{1/2} \\
&\le \frac{2^{3/2}}{k^{3/2}}\left(\frac{2}{k}\sum_{i=1}^{k}\mathcal{E}_i\right)(4\lambda S_k)^{1/2} = \mathcal{O}_1\left(\left[\sqrt{\lambda}L_{1,k}^{\text{avg}} + \frac{1+\sigma C_{\lambda,k}^{\text{avg}}}{\sqrt{\lambda}}\right]\left[\frac{S_k}{k^3}\right]^{1/2}\right).
\end{aligned}$$

■

We are now ready to prove Theorem 5.

*Proof of Theorem 5.* (a) This follows from Proposition 18, Proposition 17(a), the fact that $C_\lambda(\cdot,\cdot) \le \overline{C}_\lambda$ and $L_{f_1}(\cdot,\cdot) \le L_1$, and the termination condition in step 4.

(b) The fact that $(\hat{y}, \hat{v}) = (\hat{y}_k, \hat{v}_k)$ satisfies the inclusion of (22) follows from Lemma 3 with $(y, v, z_0) = (y_k^a, v_k, \tilde{x}_{k-1})$. The fact that $\|\hat{v}\| \le \hat{\rho}$ follows from the stopping condition in step 4.

(c) The fact that the method does not fail follows from Proposition 2(c). The bound in (37) follows from a similar argument as in part (a) except that Proposition 17(a) is replaced with Proposition 17(b). ■

## Acknowledgments

**Weiwei Kong**: This author acknowledges the support of the Natural Sciences and Engineering Research Council of Canada (NSERC), [funding reference number PGSD3-516700-2018], and the partial support of ONR Grant N00014-18-1-2077.

**Renato D.C. Monteiro**: The works of this author were partially supported by ONR Grant N00014-18-1-2077.

## Appendix A. General Refinement Procedures

The result below, whose proof is in (Kong et al., 2019, Lemma 19), presents properties of a general refinement procedure.

**Proposition 19.** *Let $h \in \overline{\text{Conv}}\ (\mathcal{Z})$, $z \in \operatorname{dom} h$, and $g$ be a differentiable function on $\operatorname{dom} h$ which satisfies $g(u) - \ell_g(u;z) \le L\|u-z\|^2/2$ for some $L \ge 0$ and every $u \in \operatorname{dom} g$. Moreover, define the quantities*

$$\hat{z} := \operatorname*{argmin}_u \left\{ \ell_g(u;z) + h(u) + \frac{L}{2}\|u-z\|^2 \right\}, \quad \hat{q} := L(z-\hat{z}),$$
$$\delta := h(z) - h(\hat{z}) - \langle r - \nabla g_\lambda(z), z-\hat{z}\rangle, \quad \Delta := (g+h)(z) - (g+h)(\hat{z})$$

*Then, it holds that*

$$\hat{q} \in \nabla g(z) + \partial h(\hat{z}), \quad \delta \ge 0, \quad \delta + \frac{1}{2L}\|\hat{q}\|^2 \le \Delta,$$
$$(\hat{q}, \delta) = \operatorname*{argmin}_{(r,\varepsilon)\in\mathcal{Z}\times\mathbb{R}_+} \left\{ \frac{1}{2L}\|r\|^2 + \varepsilon : r \in \nabla g(z) + \partial_\varepsilon h(z) \right\}.$$

## Appendix B. R-ACG Algorithm

This section presents technical results related to the R-ACG algorithm.

The first set of results describes some basic properties of the generated iterates.

**Proposition 20.** *If $\psi_s$ is $\mu$–strongly convex, then the following statements hold:*

*(a)* $z_j^c = \operatorname{argmin}_{u\in\mathcal{Z}} \{B_j\Gamma_j(u) + \|u - z_0^c\|^2/2\}$;

*(b)* $\Gamma_j \le \psi$ *and* $B_j\psi(z_j) \le \inf_{u\in\mathcal{Z}} \{B_j\Gamma_j(u) + \|u-z_0^c\|^2/2\}$;

*(c)* $\eta_j \ge 0$ *and* $r_j \in \partial_{\eta_j}\left(\psi - \mu\|\cdot - z_j\|^2/2\right)(z_j)$;

*(d)* $\|B_j r_j + z_j - z_0\|^2 + 2B_j\eta_j \le \|z_j - z_0\|^2$.

*Proof.* (a) See (Monteiro et al., 2016, Proposition 1).

(b) See (Monteiro et al., 2016, Proposition 1(b)).

(c) The optimality of $z_j^c$ in part (a), the $\mu$-strong convexity of $\Gamma_j$, and the definition of $r_j$ imply that

$$\begin{aligned} r_j &= \frac{z_0^c - z_j^c}{B_j} + \mu(z_j^c - z_j) \in \partial\left(\Gamma_j - \frac{\mu}{2}\|\cdot - z_j^c\|^2 + \mu\left\langle \cdot, z_j^c - z_j\right\rangle\right)(z_j^c) \\ &= \partial\left(\Gamma_j - \frac{\mu}{2}\|\cdot - z_j\|^2\right)(z_j^c). \end{aligned}$$

Using the above inclusion, the definition of $\eta_j$, the fact that $\Gamma_j - \mu\|\cdot\|^2/2$ is affine, and part (b), we now conclude that

$$\begin{aligned}\psi(z) - \frac{\mu}{2}\|z - z_j\|^2 \geq \Gamma_j(z) - \frac{\mu}{2}\|z - z_j\|^2 &= \Gamma_j(z_j^c) - \frac{\mu}{2}\|z_j^c - z_j\|^2 + \left\langle r_j, z - z_j^c \right\rangle \\ &= \psi(z_j) + \langle r_j, z - z_j\rangle - \eta_j,\end{aligned}$$

for every $z \in \operatorname{dom}\psi_n$, which is exactly the desired inclusion. The fact that $\eta_j \geq 0$ follows from the above inequality with $z = z_j$.

(d) It follows from parts (a)–(b) and the definition of $\eta_j$ that

$$\begin{aligned}\eta_j &\leq \Gamma_j(u) + \frac{1}{2B_j}\|u - z_0\|^2 - \psi(z_j) + \eta_j = -\frac{1}{B_j}\left\langle z_0 - z_j^c, z_j - z_j^c\right\rangle + \frac{1}{2B_j}\|z_j^c - z_0\|^2 \\ &= \frac{1}{2B_j}\|z_j - z_0\|^2 - \frac{1}{2B_j}\|z_j - z_j^c\|^2 = \frac{1}{2B_j}\|z_j - z_0\|^2 - \frac{1}{2B_j}\|B_j r_j + z_j - z_0\|^2.\end{aligned}$$

Multiplying both sides of the above inequality by $2B_j$ yields the desired conclusion. ■

The next result presents the general iteration complexity of the algorithm, i.e. Proposition 2(a).

*Proof of Proposition 2(a).* Let $\ell$ be the quantity in (19) and suppose that the R-ACG algorithm has not stopped with failure before iteration $\ell$. In view of step 2 of the algorithm, it follows that (17) holds for every $1 \leq j \leq \ell$. We now show that it must stop with success at the end of the $\ell^{\text{th}}$ iteration. Indeed, note that (18), (19), the fact that $K > 1$, and the fact $\log(1+t) \geq t/(1+t)$ for $t \geq 0$ imply that

$$B_\ell \geq \frac{1}{M}\left(1 + \sqrt{\frac{\mu}{4M}}\right)^{2(\ell-1)} \geq 2K_\sigma^2 > 2. \tag{58}$$

Combining the triangle inequality, (17), the fact that $2/B_\ell \leq 1/K_\sigma^2$ and $(2/B_\ell)^2 < 1$ from (58), and the relation $(a+b)^2 \leq 2a^2 + 2b^2$ for all $a, b \in \mathbb{R}$, we conclude that

$$\begin{aligned}\|r_\ell\|^2 + 2\eta_\ell &\leq \max\left\{1/B_\ell^2, 1/(2B_\ell)\right\}\left(\|B_\ell r_\ell\|^2 + 4B_\ell\eta_\ell\right) \\ &\leq \max\left\{1/B_\ell^2, 1/(2B_\ell)\right\}\left(2\|B_\ell r_\ell + z_\ell - z_0\|^2 + 2\|z_\ell - z_0\|^2 + 4B_\ell\eta_\ell\right) \\ &\leq \max\left\{(2/B_\ell)^2, 2/B_\ell\right\}\|z_\ell - z_0\|^2 \leq \frac{1}{K_\sigma^2}\|z_\ell - z_0\|^2 \leq \sigma^2\|z_\ell - z_0\|^2.\end{aligned}$$

■

## Appendix C. Refined ICG Points

This appendix presents technical results related to the refined points of the ICG methods.

The result below proves Lemma 3 from the main body of the paper.

*Proof of Lemma 3.* (a) Using Proposition 1(a), the definition of $\hat{v}$, and the definitions of $\psi_s$ and $\psi_n$ in (26), we have that

$$\hat{v} \in \frac{1}{\lambda}\left[\nabla\psi_s(\hat{y}) + \partial\psi_n(\hat{y}) + w - y\right] + \nabla f_1(\hat{y}) - \nabla f_1(w)$$

$$= \frac{1}{\lambda}\left[\lambda\nabla f_1(w) + \lambda f_2(\hat{y}) + (w-y) + \lambda\partial h(y)\right] + \nabla f_1(\hat{y}) - \nabla f_1(w)$$
$$= \nabla f_1(\hat{y}) + \nabla f_2(\hat{y}) + \partial h(\hat{y}),$$

(b) Using assumption (A3), Proposition 1(b), the choice of $M$ in (26), and the fact that $\Delta_\mu(y_r; y, v) \leq \varepsilon$, we first observe that

$$\begin{aligned}\|\nabla f_1(\hat{y}) - \nabla f_1(z_0)\| - L_1(y, z_0)\|y - z_0\| &\leq L_1(y,\hat{y})\|\hat{y} - y\| \\ &\leq \frac{L_1(y,\hat{y})\sqrt{2\Delta_\mu(y_r; y, v)}}{\sqrt{\lambda M_2^+ + 1}} \leq \frac{\sigma L_1(y,\hat{y})}{\sqrt{\lambda M_2^+ + 1}}\|y - z_0\|. \end{aligned} \tag{59}$$

Using now (59), the choice of $M$ in (26), Proposition 1(c) with $L(\cdot,\cdot) = \lambda L_2(\cdot,\cdot)$, the fact that $\sigma \leq 1$, and the definition of $C_\lambda(\cdot,\cdot)$, we conclude that

$$\begin{aligned}\|\hat{v}\| &\leq \frac{1}{\lambda}\|v_r\| + \frac{1}{\lambda}\|y - z_0\| + \|\nabla f_1(\hat{y}) - \nabla f_1(z_0)\| \\ &\leq \left[L_1(y, z_0) + \frac{1+\sigma}{\lambda} + \frac{\sigma\left[\lambda M_2^+ + 1 + \lambda L_1(y,\hat{y}) + \lambda L_2(y,\hat{y})\right]}{\lambda\sqrt{\lambda M_2^+ + 1}}\right]\|y - z_0\| \\ &\leq \left[L_1(y, z_0) + \frac{2 + \sigma C_\lambda(y,\hat{y})}{\lambda}\right]\|y - z_0\|.\end{aligned}$$

■

## Appendix D. Spectral Functions

This section presents some results about spectral functions as well as the proof of Propositions 6. It is assumed that the reader is familiar with the key quantities given in Subsection 4.1 (e.g., see Equations 38 and 39).

We first state two well-known results (see Lewis, 1995; Beck, 2017) about spectral functions.

**Lemma 21.** *Let $\Psi = \Psi^{\mathcal{V}} \circ \sigma$ for some absolutely symmetric function $\widetilde{\Psi} : \mathbb{R}^r \mapsto \mathbb{R}$. Then, the following properties hold:*

*(a) $\Psi^* = (\Psi^{\mathcal{V}} \circ \sigma)^* = (\Psi^{\mathcal{V}})^* \circ \sigma$;*

*(b) $\nabla\Psi = (\nabla\Psi^{\mathcal{V}}) \circ \sigma$;*

**Lemma 22.** *Let $(\Psi, \Psi^{\mathcal{V}})$ be as in Lemma 21, the pair $(S, Z) \in \mathcal{Z} \times \operatorname{dom}\Psi$ be fixed, and the decomposition $S = P[\operatorname{dg}\sigma(S)]Q^*$ be an SVD of $S$, for some $(P, Q) \in \mathcal{U}^m \times \mathcal{U}^n$. If $\Psi \in \overline{\operatorname{Conv}}\ \mathbb{R}^{m\times n}$ and $\Psi^{\mathcal{V}} \in \overline{\operatorname{Conv}}\ \mathbb{R}^r$, then for every $M > 0$, we have*

$$S \in \partial\left(\Psi + \frac{M}{2}\|\cdot\|_F^2\right)(Z) \iff \begin{cases}\sigma(S) \in \partial\left(\Psi^{\mathcal{V}} + \frac{M}{2}\|\cdot\|^2\right)(\sigma(Z)), \\ Z = P[\operatorname{dg}\sigma(Z)]Q^*.\end{cases}$$

We now present a new result about spectral functions.

**Theorem 23.** *Let $(\Psi, \Psi^{\mathcal{V}})$ be as in Lemma 21 and the point $Z \in \mathbb{R}^{m\times n}$ be such that $\sigma(Z) \in \operatorname{dom} \Psi^{\mathcal{V}}$. Then for every $\varepsilon \geq 0$, we have $S \in \partial_\varepsilon \Psi(Z)$ if and only if $\sigma(S) \in \partial_{\varepsilon(S)} \Psi^{\mathcal{V}}(\sigma(Z))$, where*

$$\varepsilon(S) := \varepsilon - [\langle \sigma(Z), \sigma(S)\rangle - \langle Z, S\rangle] \geq 0. \tag{60}$$

*Moreover, if $S$ and $Z$ have a simultaneous SVD, then $\varepsilon(S) = \varepsilon$.*

*Proof.* Using Lemma 21(a), (60), and the well-known fact that $S \in \partial_\varepsilon \Psi(Z)$ if and only if $\varepsilon \geq \Psi(Z) + \Psi^*(S) - \langle Z, S\rangle$, we have that $S \in \partial_\varepsilon \Psi(Z)$ if and only if

$$\begin{aligned}
\varepsilon(S) &= \varepsilon - [\langle \sigma(Z), \sigma(S)\rangle - \langle Z, S\rangle] \\
&\geq \Psi(Z) + \Psi^*(S) - \langle Z, S\rangle - [\langle \sigma(Z), \sigma(S)\rangle - \langle Z, S\rangle] \\
&= \Psi^{\mathcal{V}}(\sigma(Z)) + (\Psi^{\mathcal{V}})^*(\sigma(S)) - \langle \sigma(Z), \sigma(S)\rangle\,,
\end{aligned}$$

or, equivalently, $\sigma(S) \in \partial_{\varepsilon(S)} \Psi^{\mathcal{V}}(\sigma(Z))$ and $\varepsilon(S) \geq 0$. To show that the existence of a simultaneous SVD of $S$ and $Z$ implies $\varepsilon(S) = \varepsilon$ it suffices to show that $\langle \sigma(S), \sigma(Z)\rangle = \langle S, Z\rangle$. Indeed, if $S = P[\operatorname{dg}\sigma(S)]Q^*$ and $Z = P[\operatorname{dg}\sigma(Z)]Q^*$, for some $(P, Q) \in \mathcal{U}^m \times \mathcal{U}^n$, then we have

$$\langle S, Z\rangle = \langle \operatorname{dg}\sigma(S), P^*P[\operatorname{dg}\sigma(Z)]Q^*Q\rangle = \langle \operatorname{dg}\sigma(S), \operatorname{dg}\sigma(Z)\rangle = \langle \sigma(S), \sigma(Z)\rangle.$$

■

## References


Amir Beck. *First-order methods in optimization*, volume 25. SIAM, 2017.

Emmanuel J. Candes, Yonina C. Eldar, Thomas Strohmer, and Vladislav Voroninski. Phase retrieval via matrix completion. *SIAM review*, 57(2):225–251, 2015.

Yair Carmon, John C. Duchi, Oliver Hinder, and Aaron Sidford. Accelerated methods for nonconvex optimization. *SIAM Journal on Optimization*, 28(2):1751–1772, 2018.

Dmitriy Drusvyatskiy and Courtney Paquette. Efficiency of minimizing compositions of convex functions and smooth maps. *Mathematical Programming*, 178(1-2):503–558, 2019.

Saeed Ghadimi and Guanghui Lan. Accelerated gradient methods for nonconvex nonlinear and stochastic programming. *Math. Program.*, 156:59–99, 2016. ISSN 1436-4646.

Saeed Ghadimi, Guanghui Lan, and Hongchao Zhang. Generalized Uniformly Optimal Methods for Nonlinear Programming. *arXiv e-prints*, art. arXiv:1508.07384, August 2015.

Weiwei Kong, Jefferson G. Melo, and Renato D. C. Monteiro. Complexity of a quadratic penalty accelerated inexact proximal point method for solving linearly constrained nonconvex composite programs. *SIAM Journal on Optimization*, 29(4):2566–2593, 2019.

Weiwei Kong, Jefferson G. Melo, and Renato D. C. Monteiro. An efficient adaptive accelerated inexact proximal point method for solving linearly constrained nonconvex composite problems. *Comp. Opt. and Appl.*, 76(2):305–346, 2020.

Adrian S. Lewis. The convex analysis of unitarily invariant matrix functions. *Journal of Convex Analysis*, 2(1):173–183, 1995.

Jiaming Liang and Renato D. C. Monteiro. A Doubly Accelerated Inexact Proximal Point Method for Nonconvex Composite Optimization Problems. *arXiv e-prints*, art. arXiv:1811.11378, November 2018.

Jiaming Liang, Renato D. C. Monteiro, and Chee-Khian Sim. A FISTA-type accelerated gradient algorithm for solving smooth nonconvex composite optimization problems. *arXiv e-prints*, art. arXiv:1905.07010, May 2019.

Renato D. C. Monteiro, Camilo Ortiz, and Benar F. Svaiter. Gradient methods for minimizing composite functions. *Math. Program.*, pages 1–37, 2012.

Renato D. C. Monteiro, Camilo Ortiz, and Benar F. Svaiter. An adaptive accelerated first-order method for convex optimization. *Comput. Optim. Appl.*, 64:31–73, 2016.

Courtney Paquette, Hongzhou Lin, Dmitriy Drusvyatskiy, Julien Mairal, and Zaid Harchaoui. Catalyst Acceleration for Gradient-Based Non-Convex Optimization. *arXiv e-prints*, art. arXiv:1703.10993, March 2017.

Tingni Sun and Cun-Hui Zhang. Calibrated elastic regularization in matrix completion. In *Advances in Neural Information Processing Systems*, pages 863–871, 2012.

Fei Wen, Rendong Ying, Peilin Liu, and Robert C. Qiu. Robust pca using generalized nonconvex regularization. *IEEE Transactions on Circuits and Systems for Video Technology*, 2019.

Quanming Yao and James T. Kwok. Efficient learning with a family of nonconvex regularizers by redistributing nonconvexity. *The Journal of Machine Learning Research*, 18(1): 6574–6625, 2017.