Mathematical Programming Computation (2021) 13:583–644
https://doi.org/10.1007/s12532-021-00205-x

FULL LENGTH PAPER

An inexact proximal augmented Lagrangian framework with arbitrary linearly convergent inner solver for composite convex optimization

Fei Li[1] · **Zheng Qu**[1]

Received: 24 December 2019 / Accepted: 23 June 2021 / Published online: 17 July 2021

Abstract

We propose an inexact proximal augmented Lagrangian framework with explicit inner problem termination rule for composite convex optimization problems. We consider arbitrary linearly convergent inner solver including in particular stochastic algorithms, making the resulting framework more scalable facing the ever-increasing problem dimension. Each subproblem is solved inexactly with an explicit and self-adaptive stopping criterion, without requiring to set an a priori target accuracy. When the primal and dual domain are bounded, our method achieves $O(1/\sqrt{\epsilon})$ and $O(1/\epsilon)$ complexity bound in terms of number of inner solver iterations, respectively for the strongly convex and non-strongly convex case. Without the boundedness assumption, only logarithm terms need to be added and the above two complexity bounds increase respectively to $\tilde{O}(1/\sqrt{\epsilon})$ and $\tilde{O}(1/\epsilon)$, which hold both for obtaining ϵ-optimal and ϵ-KKT solution. Within the general framework that we propose, we also obtain $\tilde{O}(1/\epsilon)$ and $\tilde{O}(1/\epsilon^2)$ complexity bounds under relative smoothness assumption on the differentiable component of the objective function. We show through theoretical analysis as well as numerical experiments the computational speedup possibly achieved by the use of randomized inner solvers for large-scale problems.

Keywords Inexact augmented Lagrangian method · Large scale optimization · Randomized first-order method · Explicit inner termination rule · Relative smoothness condition

Fei Li was supported by Hong Kong PhD Fellowship Scheme No. PF15-16399. Zheng Qu was supported by Early Career Scheme from Hong Kong Research Grants Council No. 27302016. The computations were performed using research computing facilities offered by Information Technology Services, the University of Hong Kong.

✉ Zheng Qu
zhengqu@maths.hku.hk

Fei Li
lifei16@connect.hku.hk

[1] Department of Mathematics, The University of Hong Kong, Pokfulam Road, Pok Fu Lam, Hong Kong

Mathematics Subject Classification 90C06 · 90C25 · 49M37

1 Introduction

We consider the following optimization problem:

$$\min_{x\in\mathbb{R}^n} \quad f(x)+g(x)+h_1(p_1(x))+h_2(p_2(x)). \tag{1}$$

Here $g:\mathbb{R}^n\to\mathbb{R}\cup\{+\infty\}$, $h_1:\mathbb{R}^{d_1}\to\mathbb{R}$ are proper, convex and closed functions. The function $h_2:\mathbb{R}^{d_2}\to\mathbb{R}\cup\{+\infty\}$ is the indicator function of a convex and closed set $\mathcal{K}\subset\mathbb{R}^{d_2}$:

$$h_2(u_2)=\begin{cases}0 & \text{if } u_2\in\mathcal{K}\\ +\infty & \text{otherwise}\end{cases} \tag{2}$$

The function $f:\mathbb{R}^n\to\mathbb{R}\cup\{+\infty\}$ is convex and differentiable on an open set containing $\mathrm{dom}(g)$. The functions $p_1:\mathbb{R}^n\to\mathbb{R}^{d_1}$ and $p_2:\mathbb{R}^n\to\mathbb{R}^{d_2}$ are differentiable. In addition, we assume that g, h_1, h_2 are *simple* functions, in the sense that their proximal operator are easily computable. With some other standard assumptions stated in the later discussion, the model that we consider covers a wide range of optimization problems. As an example, the following linearly constrained convex optimization problem

$$\begin{aligned}\min_{x\in\mathbb{R}^n} \quad & f(x)+g(x)\\ \text{s.t.} \quad & Ax=b\end{aligned} \tag{3}$$

is a special case of (1) by letting $h_1\equiv 0$, $\mathcal{K}=\{b\}$ and $p_2(x)\equiv Ax$. Important applications of (3) include model predictive control [40] and basis pursuit problem [13]. When $h_1\equiv 0$, $\mathcal{K}$ is a closed convex cone in $\mathbb{R}^{d_2}$ and $p_2(\cdot)$ is convex with respect to $\mathcal{K}$, problem (1) reduces to the convex conic programming model [26,28] and in particular contains the constrained convex programming problem [38]:

$$\begin{aligned}\min_{x\in\mathbb{R}^n} \quad & f(x)+g(x)\\ \text{s.t.} \quad & f_1(x)\le 0,\ \ \ldots,\ f_m(x)\le 0.\end{aligned} \tag{4}$$

Apart from constrained programs, problem (1) also covers many popular models in machine learning, including the sparse-group LASSO [41], the fused LASSO [42], the square root LASSO [6], and the support vector machine problem [51].

In [38], Rockafellar built an inexact augmented Lagrangian method (ALM) framework for solving (4). At each iteration of the inexact ALM, one needs to solve a convex optimization problem (referred to as *inner problem*) presumed easier than the original constrained problem (4), up to a certain accuracy. Rockafellar [38] gave some stopping criteria for the test of the inner problem solution accuracy, as well as some

sufficient conditions guaranteeing the convergence of the inexact ALM method. In [31], Nesterov proposed a smoothing technique to deal with the unconstrained case ($h_2 \equiv 0$). The idea is again to replace the original problem by an easier subproblem and solve it up to a desired accuracy. Although existing work usually consider either $h_1 \equiv 0$ or $h_2 \equiv 0$ [7,11], we can treat them in a unified way because the augmented Lagrangian function corresponds to a smooth approximation of the function h_2. In fact, both Rockafellar's inexact ALM framework and Nesterov's smoothing technique are applications of the inexact proximal point method [5,39]. There is no essential difficulty in extending existing results from the case $h_1 \equiv 0$ or $h_2 \equiv 0$ to the general model (1). For this reason, in the following discussion, we do not make particular difference among the papers dealing with the two different cases (either $h_1 \equiv 0$ or $h_2 \equiv 0$).

We mainly consider two optimality criteria for the complexity analysis of inexact ALM. One is based on the primal feasibility and the primal value optimality gap and the other on the KKT-residual. A solution $x \in \mathrm{dom}(g)$ is said to be ϵ-optimal if [5,28,29,31,34,38,48]

$$|F(x) - F^\star| \leq \epsilon, \ \ \mathrm{dist}(p_2(x), \mathcal{K}) \leq \epsilon. \tag{5}$$

Here,

$$F(x) := f(x) + g(x) + h_1(p_1(x)), \ \ \forall x \in \mathbb{R}^n, \tag{6}$$

and $F^\star$ denotes the optimal value of (1). A solution $x \in \mathrm{dom}(g)$ is said to be ϵ-KKT optimal if there is $\lambda_1 \in \mathrm{dom}(h_1^*)$ and $\lambda_2 \in \mathrm{dom}(h_2^*)$ such that [22,26]

$$\begin{aligned} &\mathrm{dist}(0, \partial_x L(x, \lambda_1, \lambda_2)) \leq \epsilon, \\ &\mathrm{dist}(0, \partial_{\lambda_1} L(x, \lambda_1, \lambda_2)) \leq \epsilon, \ \ \mathrm{dist}(0, \partial_{\lambda_2} L(x, \lambda_1, \lambda_2)) \leq \epsilon. \end{aligned} \tag{7}$$

Here,

$$\begin{gathered} L(x, \lambda_1, \lambda_2) := f(x) + g(x) + \langle \lambda_1, p_1(x) \rangle - h_1^*(\lambda_1) + \langle \lambda_2, p_2(x) \rangle - h_2^*(\lambda_2), \\ \forall x \in \mathbb{R}^n, \lambda_1 \in \mathbb{R}^{d_1}, \lambda_2 \in \mathbb{R}^{d_2}, \end{gathered}$$

denotes the Lagrangian function and $h_1^*(\cdot)$ (resp. $h_2^*(\cdot)$) denotes the Fenchel conjugate function of $h_1(\cdot)$ (resp. $h_2(\cdot)$). A different criterion which can be derived from (7) under the boundedness of $\mathrm{dom}(g)$ was used in [24]. Most of the previously cited papers studied the complexity bound of the inexact ALM, which is the number of inner iterations needed for computing an ϵ-optimal solution or an ϵ-KKT solution. The lowest known complexity bound is $O(1/\epsilon)$ for obtaining an ϵ-optimal solution [5,28,31,34,43,48], and $\tilde{O}(1/\epsilon)$ for obtaining an ϵ-KKT solution [26].

There are some variants of inexact ALM which avoid the solution of inner problems, including the linearized ALM [48] and linearized ADMM [33] as well as their stochastic extensions [10,47,49]. These inner problem free methods are widely used in practice thanks to their simple implementation form and good practical convergence

behavior. However, $O(1/\epsilon)$ complexity bound of these methods are established only in an ergodic sense, not in the last iterate. In [44], an accelerated smooth gap reduction method (ASGARD) was developed with a non-ergodic $O(1/\epsilon)$ complexity bound and showed superior numerical performance than linearized ADMM, for the case when $p(\cdot)$ is an affine function. The method has been extended to a stochastic block coordinate update version in [1], called SMART-CD. In practice, it was observed that appropriately restarting ASGARD or SMART-CD can further speed up the convergence. In [43], the authors analyzed a double-loop ASGARD (ASGARD-DL) which achieves the non-ergodic complexity bound $O(1/\epsilon)$ and has similar practical convergence behavior as ASGARD with restart. ASGARD-DL [43] can be seen as an inexact ALM. However, in contrast to a series of work on inexact ALM [5,28,31,34,48], ASGARD-DL has an explicit inner termination rule and does not require the boundedness of $\mathrm{dom}(g)$.

The boundedness assumption of $\mathrm{dom}(g)$ seems to be crucial in the existing analysis for inexact ALM since it allows to directly control the number of iterations needed for the solution of each inner problem, using deterministic first-order solvers such as the accelerated proximal gradient (APG) [4]. It was argued that such boundedness assumption is mild because in some cases it is possible to find a bounded set including the optimal solution [24]. Nevertheless, removing this assumption from the complexity analysis of inexact ALM seems to be challenging and requires different approaches from existing ones. ASGARD-DL [43] is among the first which achieve the best complexity bound $O(1/\epsilon)$ without making the compactness assumption. However, their analysis builds on a very special property of APG and thus excludes the possibility of other inner solvers.

Allowing more flexible choice of inner solver is a very important feature in the large-scale setting. It is recognized that some randomized first-order methods can be more efficient than APG when the problem dimension is high. This includes for example the randomized coordinate descent variant of APG (*a.k.a.* APPROX) [17] and the stochastic variance reduced variant of APG (*a.k.a.* Katyusha) [2]. Compared with their deterministic origin APG, APPROX can reduce the computation load when the number of coordinates n is large, while Katyusha is more efficient when the number of constraints m (in (4)) is large. With the ever-increasing scale of the problems to be solved, it is necessary to employ randomized methods for solving the inner problems. However, the complexity analysis of inexact ALM with randomized inner solvers seems not to have been fully investigated.

In this paper, we develop an inexact proximal ALM which does not require the boundedness of $\mathrm{dom}(g)$ for the inner termination rule, and analyze its total complexity bound for any linearly convergent inner solver. Since randomized inner solvers are included, we will only require the optimality criteria (5) and (7) to be achieved in expectation, see (31) and (85). In addition, the complexity bound that we provide is an upper bound on the expectation of the number of total inner iterations. We summarize below our contributions.

1. We give a stochastic extension of Rockafellar's inexact proximal ALM framework, see Algorithm 1. The difference with the original framework lies in the inner

problem stopping criteria, which only asks the inner optimality gap to be smaller than a certain threshold in expectation.

2. For any linearly convergent inner solver $\mathcal{A}$, we give an upper bound on the number of inner iterations required to satisfy the stopping criteria, see (38). In contrast to related work [22,24,26,28,29,34,48], the upper bound computed by (38) does not depend on the diameter of $\mathrm{dom}(g)$ and in particular does not need to assume the boundedness of $\mathrm{dom}(g)$. Instead, our upper bound is adaptively computed based on the previous and current primal and dual iterates, as well as the linear convergence rate of the inner solver $\mathcal{A}$.
3. Based on the explicit upper bound computed by (38), we propose an inexact proximal ALM with an explicit inner termination rule, see Algorithm 2. Compared with the previously mentioned work, our termination rule
   - **does not** require the desired accuracy ϵ to be set a priori;
   - **does not** need to assume the boundedness of $\mathrm{dom}(g)$.
4. We show that the complexity bound of Algorithm 2 is $\tilde{O}(1/\epsilon^{\ell})$ to obtain an ϵ-optimal solution where $\ell > 0$ is a constant determined by the convergence rate of the inner solver $\mathcal{A}$, see Theorem 2. Our approach can be easily extended to obtain $\tilde{O}(1/\epsilon^{\ell})$ complexity bound for ϵ-KKT solution, see Sect. 6.2. When both the primal and dual domains are bounded, the bound $\tilde{O}(1/\epsilon^{\ell})$ can be improved to $O(1/\epsilon^{\ell})$ for obtaining an ϵ-optimal solution, see Sect. 6.3.
5. We show how to apply Theorem 2 under different problem structures and assumptions. When $p_1(\cdot)$ and $p_2(\cdot)$ are linear, under the same assumptions as [22,24,26,28,29,34] but without the boundedness of $\mathrm{dom}(g)$, we obtain $\tilde{O}(1/\epsilon)$ and $\tilde{O}(1/\sqrt{\epsilon})$ complexity bound respectively for the non-strongly convex and strongly convex case, see Corollary 5 and 6 . We also consider the case when f is only relatively smooth, and establish $\tilde{O}(1/\epsilon)$ and $\tilde{O}(1/\epsilon^2)$ complexity bound respectively for the non-strongly convex and strongly convex case, see Corollary 7.
6. We provide theoretical justification to support the use of randomized solvers in large-scale setting, see Table 1. We give numerical evidence to show that with appropriate choice of inner solver, our algorithm outperforms ASGARD-DL and SMART-CD, see Figs. 1, 2, 3, 4, 5. Moreover, compared with CVX, our algorithm often obtains a solution with medium accuracy within less computational time, see Tables 3, 4, 5.

Notations For any two vectors $\lambda_1 \in \mathbb{R}^{d_1}$ and $\lambda_2 \in \mathbb{R}^{d_2}$ we denote by $(\lambda_1; \lambda_2)$ the vector in $\mathbb{R}^{d_1+d_2}$ obtained by concatenating λ_1 and λ_2. Inversely, for any $\lambda \in \mathbb{R}^{d_1+d_2}$ we denote by $\lambda_1 \in \mathbb{R}^{d_1}$ the vector containing the first d_1 components of λ and $\lambda_2 \in \mathbb{R}^{d_2}$ the vector containing the last d_2 components of λ. We use $\|\cdot\|$ to denote the standard Euclidean norm for vector and spectral norm for matrix. For any matrix A, $A_{i,i}$ is the ith diagonal element of A. We denote by $e_i \in \mathbb{R}^n$ the ith standard basis vector in $\mathbb{R}^n$. For proper, closed and convex function $h(\cdot)$, $h^*(\cdot)$ denotes its Fenchel conjugate function. For any $x \in \mathbb{R}^{d_2}$, $\mathrm{dist}(x, \mathcal{K})$ denotes the distance from x to $\mathcal{K}$. For any integer n we denote by $[n]$ the set $\{1, 2, \cdots, n\}$.

Organization In Sect. 2, we study an inexact proximal ALM framework with expected inexactness condition. In Sect. 3, we give an upper bound on the number

of the inner iterations and obtain an instantiation of the general inexact proximal ALM. In Sect. 4, we briefly recall several first order methods and their respective convergence rate. In Sect. 5, we apply our main results to different structured problems. In Sect. 6, we discuss some extension of our work. In Sect. 7, we present numerical experiments. In Sect. 8, we make some concluding remarks. Background knowledge used and missing proofs can be found in the "Appendix".

2 Preliminaries

2.1 Problem and assumptions

For ease of presentation we rewrite (1) as

$$\min_{x\in\mathbb{R}^n} f(x)+g(x)+h(p(x)), \tag{8}$$

where

$$h((u_1;u_2)) := h_1(u_1)+h_2(u_2),\ \ u_1\in\mathbb{R}^{d_1}, u_2\in\mathbb{R}^{d_2},$$

and

$$p(x) := (p_1(x);\, p_2(x)),\ \ x\in\mathbb{R}^n.$$

Define the Lagrangian function

$$L(x,\lambda) := f(x)+g(x)+\langle\lambda,\, p(x)\rangle - h^*(\lambda), \tag{9}$$

and consider the Lagrange dual problem:

$$\max_{\lambda\in\mathbb{R}^d}\left[D(\lambda)\equiv\inf_x L(x,\lambda)\right]. \tag{10}$$

We shall call problem (8) the *primal problem* and (10) the *dual problem*. Apart from the structures mentioned in the very beginning of Sect. 1, we make the following additional assumptions throughout the paper.

Assumption 1 (a) h_1 is L_{h_1}-Lipschitz continuous.
(b) for any $x, y\in\mathbb{R}^n$, $u_1, v_1\in\mathbb{R}^{d_1}$ and $\alpha\in(0,1)$

$$\begin{aligned} &h_1\left(p_1(\alpha x+(1-\alpha)y)-\alpha u_1-(1-\alpha)v_1\right)\\ &\quad\le \alpha h_1(p_1(x)-u_1)+(1-\alpha)h_1(p_1(y)-v_1).\end{aligned}$$

(c) for any $x, y\in\mathbb{R}^n$, $u_2, v_2\in\mathbb{R}^{d_2}$ and $\alpha\in(0,1)$ such that $p_2(x)-u_2\in\mathcal{K}$ and $p_2(y)-v_2\in\mathcal{K}$, it holds that

$$p_2\left(\alpha x+(1-\alpha)y)\right)-\alpha u_2-(1-\alpha)v_2\in\mathcal{K}.$$

(d) both the primal and the dual problem have optimal solution and the strong duality holds, i.e., there is $x^\star \in \mathrm{dom}(g)$ and $\lambda^\star \in \mathrm{dom}(h^*)$ such that $p_2(x^\star) \in \mathcal{K}$ and

$$F(x^\star) = L(x^\star; \lambda^\star) = D(\lambda^\star). \tag{11}$$

If $p_1(\cdot) : \mathbb{R}^n \to \mathbb{R}^{d_1}$ is affine, then Assumption (b) holds. Otherwise, (b) holds if there is a partial order $\preceq_{\mathcal{C}_1}$ on $\mathbb{R}^{d_1}$ induced by a closed convex cone $\mathcal{C}_1 \subset \mathbb{R}^{d_1}$ (i.e. $x \preceq_{\mathcal{C}_1} y$ if and only if $y - x \in \mathcal{C}_1$) such that the function $p_1(\cdot)$ is convex with respect to the order $\preceq_{\mathcal{C}_1}$, i.e.,

$$p_1(\alpha x + (1-\alpha)y) \preceq_{\mathcal{C}_1} \alpha p_1(x) + (1-\alpha) p_1(y), \tag{12}$$

and the function $h_1(\cdot)$ is order preserving with respect to $\preceq$, i.e.,

$$u_1 \preceq_{\mathcal{C}_1} v_1 \Longrightarrow h_1(u_1) \leq h_1(v_1). \tag{13}$$

Similarly, if $p_2(\cdot) : \mathbb{R}^n \to \mathbb{R}^{d_2}$ is affine, then Assumption (c) holds. Otherwise, (c) holds if there is a partial order $\preceq_{\mathcal{C}_2}$ on $\mathbb{R}^{d_2}$ induced by a closed convex cone $\mathcal{C}_2$ such that the function $p_2(\cdot)$ is convex with respect to the order $\preceq_{\mathcal{C}_2}$, i.e.,

$$p_2(\alpha x + (1-\alpha)y) \preceq_{\mathcal{C}_2} \alpha p_2(x) + (1-\alpha) p_2(y), \tag{14}$$

and the set $\mathcal{K}$ is such that $u_2 + z_2 \in \mathcal{K}$ for any $u_2 \in \mathcal{K}$ and $z_2 \preceq_{\mathcal{C}_2} 0$.

Remark 1 For example, consider the partial order $\preceq$ induced by the nonnegative orthant $\mathbb{R}^{d_1}_+$, then (12) is satisfied if $p_1(x) = (q_1(x), \cdots, q_{d_1}(x))^\top$ with each $q_i : \mathbb{R}^n \to \mathbb{R}$ being convex. If the partial order $\preceq$ is induced by the cone of positive semidefinite matrices S^m_+, then (12) holds if $p_1(x) = \sum_{i=1}^t B_i q_i(x)$ with $B_1, \ldots, B_t \in S^m_+$ and each $q_i : \mathbb{R}^n \to \mathbb{R}$ being convex, see, e.g. [3]. The same class of examples apply to (14).

Remark 2 A special case when (13) holds is when h_1 is the support function of some bounded set included in the dual cone of $\mathcal{C}_1$, i.e.,

$$h_1(x) \equiv \sup\{\langle y, x\rangle : y \in \mathbb{B} \cap \mathcal{C}_1^*\},$$

where $\mathbb{B}$ is a bounded set and $\mathcal{C}_1^* := \{y : \langle y, x\rangle \geq 0, \ \ \forall x \in \mathcal{C}_1\}$ is the dual cone of $\mathcal{C}_1$. For example, when $\mathbb{B}$ is the unit ball with respect to the standard Euclidean norm and $\mathcal{C}_1 = \mathbb{R}^{d_1}_+$ is the nonnegative orthant, then $h_1(x) = \|\max(x, 0)\|$, see e.g. [18].

Let $d = d_1 + d_2$. Condition (b) and (c) in Assumption 1 imply that for any $x, y \in \mathbb{R}^n$, $u, v \in \mathbb{R}^d$ and $\alpha \in (0, 1)$

$$h(z) \leq \alpha h(p(x) - u) + (1-\alpha) h(p(y) - v), \tag{15}$$

where $z = p(\alpha x + (1-\alpha)y) - \alpha u - (1-\alpha)v$. The latter condition guarantees the convexity of $h(p(\cdot)) : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$.

2.2 Proximal ALM revisited

Let any $\lambda \in \mathbb{R}^d$ and $\beta > 0$. Define

$$h(u;\lambda,\beta) := \max_{v\in\mathbb{R}^d}\left\{\langle v,u\rangle - h^*(v) - \frac{\beta}{2}\|v-\lambda\|^2\right\}, \tag{16}$$

and

$$\Lambda(u;\lambda,\beta) := \arg\max_{v\in\mathbb{R}^d}\left\{\langle v,u\rangle - h^*(v) - \frac{\beta}{2}\|v-\lambda\|^2\right\}. \tag{17}$$

The function $h(\cdot;\lambda,\beta)$ is known as an approximate smooth function of the possibly nonsmooth function $h(\cdot)$ with parameter λ and β. We next recall some results needed later about the smooth function $h(\cdot;\lambda,\beta)$.

Lemma 1 [5,19,31]

1. *The function $h(u;\lambda,\beta)$ is convex and differentiable with respect to u. Denote by $\nabla_1 h(u;\lambda,\beta)$ the gradient with respect to the variable u, then we have*

$$\nabla_1 h(u;\lambda,\beta) = \Lambda(u;\lambda,\beta) \tag{18}$$

$$\|\nabla_1 h(u;\lambda,\beta) - \nabla_1 h(v;\lambda,\beta)\| \le \beta^{-1}\|u-v\| \tag{19}$$

2. *For any $u,\lambda\in\mathbb{R}^d$ and $\beta>0$, we have*

$$u - \beta(\Lambda(u;\lambda,\beta)-\lambda) \in \partial h^*(\Lambda(u;\lambda,\beta)). \tag{20}$$

3. *For any $u,\lambda\in\mathbb{R}^d$ and $\beta>0$ we have*

$$h(u;\lambda,\beta) = \min_w\left\{h(u-w) + \frac{1}{2\beta}\|w\|^2 + \langle w,\lambda\rangle\right\} \le h(u) \tag{21}$$

In addition, the optimal solution $w^\star$ for (21) is given by

$$w^\star = \beta(\Lambda(u;\lambda,\beta)-\lambda), \tag{22}$$

and

$$h(u;\lambda,\beta) = h(u-\beta(\Lambda(u;\lambda,\beta)-\lambda)) + \frac{\beta}{2}\|\Lambda(u;\lambda,\beta)\|^2 - \frac{\beta}{2}\|\lambda\|^2 \tag{23}$$

Define

$$\begin{aligned} &L(x;y,\lambda,\beta)\\ &:= \max_v\left\{f(x)+g(x)+\langle v,p(x)\rangle - h^*(v) - \frac{\beta}{2}\|v-\lambda\|^2\right\} + \frac{\beta}{2}\|x-y\|^2 \\ &= f(x)+g(x)+h(p(x);\lambda,\beta)+\frac{\beta}{2}\|x-y\|^2. \end{aligned} \tag{24}$$

Remark 3 If $h(\cdot)$ is the indicator function of $\mathbb{R}^d_-$, then (24) recovers the augmented Lagrangian function defined by [38, Equation 5.1]. Hence, the function defined by (24) can be seen as a generalization of the classical augmented Lagrangian function.

Lemma 2 *Fix any $\lambda \in \mathbb{R}^d$ and $\beta > 0$. Define*

$$\tilde{\psi}(x) := h(p(x); \lambda, \beta)$$

Then $\tilde{\psi} : \mathbb{R}^n \to \mathbb{R}$ is a convex and differentiable function with $\nabla\tilde{\psi}(x) = \nabla p(x) \Lambda(p(x); \lambda, \beta)$.

It follows from Lemma 2 that for any $\beta > 0$ the function $L(x; y, \lambda, \beta)$ is strongly convex with respect to the variable x.

Let $\{\beta_s : s \geq 0\}$ and $\{\epsilon_s : s \geq 0\}$ be two sequence of positive numbers. We recall the inexact proximal augmented Lagrangian framework in Algorithm 1. The objective function at outer iteration s is denoted by $H_s(\cdot)$ and $H_s^\star := \min_x H_s(x)$. A small difference with the classical inexact proximal ALM in [38] is that we only require to control the expectation of the subproblem objective value gap. In particular, note that the iterates $\{(x^s, \lambda^s)\}$ in Algorithm 1 are random variables. We denote by $\mathcal{F}_s$ the σ-algebra generated by $\{x^t : t \leq s\} \cup \{\lambda^t : t \leq s+1\}$.

Algorithm 1 IPALM (compare with [38])

Parameters: $\{\epsilon_s\}, \{\beta_s\}$;
Initialize: $x^{-1} \in \mathrm{dom}(g)$, $\lambda^0 \in \mathrm{dom}(h^*)$;
1: **for** $s = 0, 1, \ldots$ **do**
2: Find $x^s \simeq \arg\min H_s(x) \equiv L(x; x^{s-1}, \lambda^s, \beta_s)$ satisfying $\mathbb{E}[H_s(x^s) - H_s^\star] \leq \epsilon_s$
3: $\lambda^{s+1} \leftarrow \Lambda(p(x^s); \lambda^s, \beta_s)$
4: **end for**

We now recall a few known results about inexact proximal ALM. Note that we are in a slightly more general setting than [38] due to our problem formulation (1) and the expected inexactness condition in Algorithm 1. A modification of the original proof is needed to obtain the desired results. For completeness proofs can be found in Appendix D.1. Hereinafter, $x^\star$ is an arbitrary optimal solution of the primal problem (8) and $\lambda^\star$ is an arbitrary optimal solution of the dual problem (10) such that $(0, 0) \in \partial L(x^\star, \lambda^\star)$ where $L(x, \lambda)$ is the Lagrangian function defined in (9).

Lemma 3 (Compare with [39]) *Let $\{x^s, \lambda^s\}$ be the sequence generated by Algorithm 1. Then for any $s \geq 0$,*

$$\mathbb{E}\left[\left\|(x^s, \lambda^{s+1}) - (x^{s-1}, \lambda^s)\right\|\right] \leq \left\|(x^{-1}, \lambda^0) - (x^\star, \lambda^\star)\right\| + \sum_{i=0}^{s} \sqrt{2\epsilon_i/\beta_i} \tag{25}$$

$$\mathbb{E}\left[\left\|(x^s, \lambda^{s+1}) - (x^\star, \lambda^\star)\right\|^2\right] \leq \left(\left\|(x^{-1}, \lambda^0) - (x^\star, \lambda^\star)\right\| + \sum_{i=0}^{s} \sqrt{2\epsilon_i/\beta_i}\right)^2 \tag{26}$$

Lemma 3 allows us to give a bound on the produced primal dual sequence $\{(x^s, \lambda^s)\}_s$.

Corollary 1 *Consider Algorithm 1 with $\beta_s = \beta_0 \rho^s$ and $\epsilon_s = \epsilon_0 \eta^s$ for some $0 < \eta < \rho < 1$. Define*

$$c_0 := 2\left(\left\|(x^{-1}, \lambda^0) - (x^\star, \lambda^\star)\right\| + \frac{2\sqrt{\epsilon_0/\beta_0}}{1-\sqrt{\eta/\rho}}\right)^2 + 2\|\lambda^\star\|^2 + 2\|x^\star\|^2. \tag{27}$$

Then for all $s \geq 0$, we have

$$\begin{aligned} &\max\left(\mathbb{E}\left[\|\lambda^s\|^2\right], \mathbb{E}\left[\|x^s\|^2\right], \mathbb{E}\left[\left\|x^s - x^\star\right\|^2\right]\right) \leq c_0, \\ &\mathbb{E}\left[\|\lambda^{s+1} - \lambda^s\|\right] \leq \sqrt{c_0}. \end{aligned}$$

Theorem 1 (Compare with [38]) *Consider Algorithm 1. We have the following bounds:*

$$\begin{aligned} F(x^s) - F^\star \leq\ & H_s(x^s) - H_s^\star + L_{h_1}\beta_s\|\lambda_1^{s+1} - \lambda_1^s\| \\ & + \frac{\beta_s}{2}(\|\lambda^s\|^2 - \|\lambda^{s+1}\|^2) + \frac{\beta_s}{2}\|x^\star - x^{s-1}\|^2, \end{aligned} \tag{28}$$

$$F(x^s) - F^\star \geq -\beta_s\|\lambda_2^\star\|\|\lambda_2^{s+1} - \lambda_2^s\|, \tag{29}$$

$$\operatorname{dist}(p_2(x^s), \mathcal{K}) \leq \beta_s\|\lambda_2^{s+1} - \lambda_2^s\|. \tag{30}$$

Corollary 2 *Consider Algorithm 1 with $\beta_s = \beta_0 \rho^s$ and $\epsilon_s = \epsilon_0 \eta^s$ for some $0 < \eta < \rho < 1$. Then to obtain a solution x^s such that*

$$\left|\mathbb{E}[F(x^s) - F^\star]\right| \leq \epsilon,\ \ \mathbb{E}[\operatorname{dist}(p_2(x^s), \mathcal{K})] \leq \epsilon, \tag{31}$$

if suffices to run Algorithm 1 for

$$s \geq \frac{\ln(c_1/\epsilon)}{\ln 1/\rho} \tag{32}$$

number of outer iterations where

$$c_1 := \max(\epsilon_0 + 2L_{h_1}^2\beta_0 + c_0\beta_0, \beta_0\|\lambda_2^\star\|\sqrt{c_0}, \beta_0\sqrt{c_0}) \tag{33}$$

with c_0 defined in (27).

3 Recursive relation of inexactness

The main objective of this section is to show that the initial error $H_{s+1}(x^s) - H_{s+1}^\star$ of the inner problem at iteration $s+1$ in Algorithm 1 can be upper bounded using the last step error $H_s(x^s) - H_s^\star$ and some computable quantities. The bound will yield a

way to control the number of inner iterations. The key proposition of this section is as follows.

Proposition 1 *Consider Algorithm 1. If $\beta_s \geq \beta_{s+1} > \beta_s/2$, then*

$$\begin{aligned} H_{s+1}(x^s) - H_{s+1}^\star \leq\ & 2(H_s(x^s) - H_s^\star) + \frac{\beta_s - \beta_{s+1}}{2}\|\Lambda(p(x^s); \lambda^{s+1}, \beta_{s+1}) - \lambda^{s+1}\|^2 \\ & + \beta_s\|\lambda^{s+1} - \lambda^s\|^2 + \frac{\beta_s^2}{2\beta_{s+1} - \beta_s}\|x^{s-1} - x^s\|^2 + \|\lambda^{s+1} - \lambda^s\| \\ & \times \sqrt{\left((\beta_s + \beta_{s+1})L_{h_1} + \|\beta_s\lambda_1^s - \beta_{s+1}\lambda_1^{s+1}\|\right)^2 + \|\beta_s\lambda_2^s - \beta_{s+1}\lambda_2^{s+1}\|^2} \end{aligned} \tag{34}$$

We defer the proof of Proposition 1 in Sect. D.2. In the next section we show how to make use of Proposition 1 to get an implementable form of Algorithm 1. Hereinafter we assume that we have at our disposal an algorithm $\mathcal{A}$ suitable for solving each inner problem in Algorithm 1:

$$\min_x H_s(x). \tag{35}$$

Denote by $\mathcal{A}(x, k, H_s)$ the output obtained by running k iterations of Algorithm $\mathcal{A}$ on problem (35) starting with initial solution x. We only consider those inner solvers $\mathcal{A}$ satisfying the following requirement.

Assumption 2 *(Linearly Convergent Inner Solver)* For any outer iteration $s \in \{0, 1, \dots\}$ of Algorithm 1, there is $K_s \geq 1$ such that for any $x \in \mathrm{dom}(g)$,

$$\mathbb{E}\left[H_s\left(\mathcal{A}(x, k, H_s)\right) - H_s^\star | \mathcal{F}_{s-1}\right] \leq 2^{-\lfloor k/K_s \rfloor}\left(H_s(x) - H_s^\star\right). \tag{36}$$

We will give in Sect. 4 some examples of algorithms satisfying these properties.

3.1 Inner iteration complexity control for ALM

In this section we apply Proposition 1 to derive an implementable form of Algorithm 1.

Corollary 3 *Consider Algorithm 1 with $\beta_s \geq \beta_{s+1} > \beta_s/2$. Denote*

$$\delta_s := \sqrt{\left((\beta_s + \beta_{s+1})L_{h_1} + \|\beta_s\lambda_1^s - \beta_{s+1}\lambda_1^{s+1}\|\right)^2 + \|\beta_s\lambda_2^s - \beta_{s+1}\lambda_2^{s+1}\|^2}\ ,$$

and

$$\begin{aligned} M_s :=\ & \beta_s\|\lambda^{s+1} - \lambda^s\|^2 + \tfrac{\beta_s - \beta_{s+1}}{2}\|\Lambda(p(x^s); \lambda^{s+1}, \beta_{s+1}) - \lambda^{s+1}\|^2 \\ & + \tfrac{\beta_s^2}{2\beta_{s+1} - \beta_s}\|x^{s-1} - x^s\|^2 + \|\lambda^{s+1} - \lambda^s\|\delta_s\ . \end{aligned} \tag{37}$$

Let $m_{s+1} > 0$ be an integer satisfying

$$2\epsilon_s + M_s \leq 2^{\lfloor m_{s+1}/K_{s+1} \rfloor}\epsilon_{s+1}/2\,. \tag{38}$$

If $\mathbb{E}\left[H_s(x^s) - H_s^\star\right] \le \epsilon_s$, *then*

$$\mathbb{E}\left[H_{s+1}(x^{s+1}) - H_{s+1}^\star\right] \le \epsilon_{s+1}, \tag{39}$$

is guaranteed by letting

$$x^{s+1} = \mathcal{A}(x^s, m_{s+1}, H_{s+1}).$$

Remark 4 If the value of K_{s+1} is known, then all the values involved in (38) are computable. For conditions on the functions f, g, h and p which guarantee Assumption 2 and the computability of K_s, see Sect. 5.

An instantiation of Algorithm 1 with inner solver $\mathcal{A}$ and explicit number of inner iterations is given in Algorithm 2.

Algorithm 2 IPALM($\mathcal{A}$)

Parameters: $\beta_0 > 0, \epsilon_0 > 0, \rho \in (1/2, 1), \eta \in (0, 1)$
Initialize: $x^{-1} \in \mathrm{dom}(g)$, $\lambda^0 \in \mathrm{dom}(h^*)$, $\mathcal{L}_0 \le H_0^\star$
$m_0 = \lceil \max\left(K_0 \log_2\left((H_0(x^{-1}) - \mathcal{L}_0)/\epsilon_0\right), 0\right)\rceil$
$x^0 \leftarrow \mathcal{A}(x^{-1}, m_0, H_0)$
for $s = 0, 1, 2, \ldots$ **do**
  $\lambda^{s+1} \leftarrow \Lambda(p(x^s); \lambda^s, \beta_s)$
  $\beta_{s+1} = \rho\beta_s$
  $\epsilon_{s+1} = \eta\epsilon_s$
  choose m_{s+1} to be the smallest integer satisfying (38)
  $x^{s+1} \leftarrow \mathcal{A}(x^s, m_{s+1}, H_{s+1})$
end for

The initialization step of Algorithm 2 requires the knowledge of a lower bound $\mathcal{L}_0$ of $H_0^\star$. In Sect. 6.1 the readers can find examples when such lower bound is computable. In view of Assumption 2, the integer m_0 defined in the first step of Algorithm 2 guarantees that

$$\mathbb{E}\left[H_0\left(\mathcal{A}(x^{-1}, m_0, H_0)\right) - H_0^\star\right] \le \frac{\epsilon_0}{H_0(x^{-1}) - \mathcal{L}_0}\left(H_0(x^{-1}) - H_0^\star\right) \le \epsilon_0.$$

Remark 5 The stopping criteria of Algorithm 1 are

$$\mathbb{E}[H_s(x^s) - H_s^\star] \le \epsilon_s, \ \ \forall s \ge 0, \tag{40}$$

which involve estimation of the unknow values $H_s^\star$ for every iteration $s \ge 0$. In many situations, there is a computable function $U_s(\cdot)$ such that

$$H_s(x) - H_s^\star \le U_s(x), \ \ \forall s \ge 0, \ \ \forall x \in \mathbb{R}^n. \tag{41}$$

See a detailed discussion in Sect. 6.1. In the case of (41), if $\{x^s\}_{s\geq 0}$ satisfy

$$U_s(x^s) \leq \epsilon_s, \ \ \forall s \geq 0, \tag{42}$$

with probability one, then (40) is guaranteed. Although (42) provides with us a checkable condition for finding acceptable solutions for inner problems, it imposes the tightness of the bounds in the following sense:

$$\min_x U_s(x) \leq \epsilon_s, \ \ \forall s \geq 0. \tag{43}$$

On the contrary, an arbitrary lower bound of $H_0^\star$ is sufficient to implement Algorithm 2.

Even if (43) holds, there may be some issues to implement Algorithm 1 using (42). From the practical aspect, it may require much longer time to find $\{x^s\}_{s\geq 0}$ which satisfy condition (42), leading to a bad practical performance. From the theoretical aspect, the complexity analysis of finding $\{x^s\}_{s\geq 0}$ which satisfy (42) can be difficult for some bounds such as the primal dual gap presented later in Sect. 6.1.

3.2 Overall iteration complexity bound

To analyze the total complexity of Algorithm 2, we will evaluate $\mathbb{E}[m_s]$ for $s \geq 1$. The key step is to show that the expectation of the quantity M_s defined in (37) can be bounded by some constant times β_s, provided that the primal and dual sequence $\{(x^s, \lambda^s)\}$ is bounded.

Lemma 4 *Consider Algorithm 2. If there is a constant $c > 0$ such that*

$$\max(\mathbb{E}[\|\lambda^s - \lambda^{s+1}\|^2], \mathbb{E}[\|x^{s-1} - x^s\|^2], \mathbb{E}[\|\lambda^s\|^2]) \leq c, \tag{44}$$

then

$$\mathbb{E}[M_s] \leq \beta_s \left((11 + 2\rho^{-2})(L_{h_1}^2 + c) + (2\rho - 1)^{-1}c\right).$$

To ensure condition (44), we can rely on the result from Corollary 1.

Proposition 2 *Consider Algorithm 2 with parameters satisfying $\eta < \rho$. Then,*

$$\sum_{t=1}^{s} \mathbb{E}[m_t] \leq s + \sum_{t=1}^{s} K_t \left(t \log_2 \frac{\rho}{\eta} + c_2\right) \leq \left(1 + \log_2 \frac{\rho}{\eta} + c_2\right) s \sum_{t=1}^{s} K_t, \tag{45}$$

where

$$c_2 := \log_2 \left(\frac{4}{\eta} + \frac{2\beta_0 \left((11 + 2\rho^{-2})(L_{h_1}^2 + 4c_0) + 4(2\rho - 1)^{-1}c_0\right)}{\epsilon_0 \eta}\right) + 1. \tag{46}$$

with c_0 is defined as in (27).

Theorem 2 *Consider Algorithm 2 with parameters satisfying $\eta < \rho$. If there are three constants $\varsigma \geq 0$, $\omega > 0$ and $\ell > 0$ such that*

$$K_s \leq \frac{\omega}{\beta_s^\ell} + \varsigma, \ \forall s \geq 1. \tag{47}$$

Let $\epsilon \leq \epsilon_0$. Then to obtain a solution x^s such that

$$\left|\mathbb{E}[F(x^s) - F^\star]\right| \leq \epsilon, \ \mathbb{E}[\text{dist}(p_2(x^s), \mathcal{K})] \leq \epsilon, \tag{48}$$

the total expected number of iterations of Algorithm $\mathcal{A}$ is bounded by

$$\sum_{t=0}^{s} \mathbb{E}[m_t] \leq m_0 + \frac{c_3}{\epsilon^\ell} \ln \frac{c_1}{\epsilon\rho}, \tag{49}$$

where c_1 is defined in (33) and

$$c_3 := \frac{1 + \log_2(\rho/\eta) + c_2}{\ln(1/\rho)} \left(\frac{\varsigma c_1^\ell}{\rho^\ell \ell \ln(1/\rho)} + \frac{\omega c_1^\ell}{\beta_0^\ell (1 - \rho^\ell)\rho^\ell} \right), \tag{50}$$

with c_2 defined in (46).

To facilitate the comparison of complexity of different inner solvers, hereinafter we hide the logarithm terms and those constants independent with the inner solver into the $\tilde{O}$ notation. We also hide the constant ℓ since we only compare inner solvers with the same order ℓ.

Corollary 4 *Under the premise of Theorem 2, to obtain an ϵ-optimal solution in the sense of (48), the number of iterations of the inner solver $\mathcal{A}$ is bounded by*

$$\tilde{O}\left(\frac{\omega + \varsigma}{\epsilon^\ell}\right),$$

where the $\tilde{O}$ hides logarithm terms, and constants related to the inner solver convergence order ℓ and other inner solver independent constants $c_1, c_2, \rho, \eta, \beta_0, \epsilon_0$ and m_0.

4 Inner solvers

In this section we recall some algorithms satisfying Assumption 2 so that they can be used as inner solvers. Note that due to space limit we do not give the explicit form of the algorithms and refer the readers to the given references for details. This section is independent with the previous sections.

Consider the following convex minimization problem:

$$G^\star := \min_{x\in\mathbb{R}^n} [G(x) \equiv \phi(x) + P(x)], \tag{51}$$

where $P : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ is a convex, proper and closed function and $\phi : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ is a convex function differentiable on an open set containing $\mathrm{dom}(P)$. For any differentiable point $x \in \mathrm{dom}(\phi)$ and any $y \in \mathrm{dom}(\phi)$, we denote by $D_\phi(x; y)$ the Bregman distance from x to y with respect to the function ϕ:

$$D_\phi(y; x) := \phi(y) - \phi(x) - \langle \nabla\phi(x), y - x\rangle.$$

4.1 Accelerated proximal gradient

Assume that there is $L > 0$ such that

$$D_\phi(y; x) \le \frac{L}{2}\|x - y\|^2. \tag{52}$$

In addition, assume that there is $\mu > 0$ such that for any $y \in \mathrm{dom}(P)$ there is $y^\star \in \arg\min\{G(y) : y \in \mathbb{R}^n\}$ satisfying

$$G(y) - G^\star \ge \frac{\mu}{2}\|y - y^\star\|^2.$$

The accelerated proximal gradient (APG) method [4,30,45] can be applied to solve problem (51). If $\{x^k\}$ is the output after k iterations of APG starting with x^0 as initial solution, then

$$G(x^k) - G^\star \le \frac{1}{2}\left(G(x^0) - G^\star\right), \quad \forall k \ge 2\sqrt{2L/\mu},$$

see e.g. [16,27].

4.2 Accelerated randomized coordinate descent

There exist some variants of APG which may be more efficient when the problem dimension is high and the objective function has certain separability. If P is separable, i.e.,

$$P(x) \equiv \sum_{i=1}^{n} P_i(x_i),$$

then the randomized coordinate extension of APG, known as APPROX [17], can also be applied to solve (51). At each iteration, APPROX only updates a randomly selected set of coordinates. For simplicity let us consider the case when one coordinate is chosen

uniformly at each iteration. In this case denote by $v_i > 0$ the constant satisfying the following condition:

$$D_\phi(x + he_i; x) \leq \frac{v_i}{2}h^2, \ \forall x \in \mathrm{dom}(P), i \in [n], x + he_i \in \mathrm{dom}(P). \tag{53}$$

In addition, assume that there is $\mu > 0$ such that for any $y \in \mathrm{dom}(P)$ there is $y^\star \in \arg\min\{G(y) : y \in \mathbb{R}^n\}$ satisfying

$$G(y) - G^\star \geq \frac{\mu}{2}\|y - y^\star\|^2.$$

If $\{x^k\}$ is the output after k iterations of APPROX starting with x^0 as initial solution, then

$$\mathbb{E}\left[G(x^k) - G^\star\right] \leq \frac{1}{2}\left(G(x^0) - G^\star\right), \ \forall k \geq 2n\sqrt{2\max_i v_i/\mu + 2},$$

see e.g. [15]. Note that when carefully implemented APPROX could have significantly reduced per-iteration cost than its deterministic origin APG, see [17]. The total computational saving is more important when number of coordinates n is large.

4.3 Accelerated stochastic variance reduced method

If P is μ-strongly convex and ϕ is written as a large sum of convex functions, for example when

$$\phi(x) = \frac{1}{m}\sum_{j=1}^{m}\phi^j(x),$$

where each ϕ^j is convex and there is $L_j > 0$ such that

$$D_{\phi^j}(y; x) \leq \frac{L_i}{2}\|x - y\|^2, \ \forall x, y \in \mathrm{dom}(P).$$

Then the accelerated stochastic variance reduced methods, known as Katyusha [2,35], can be applied to solve (51). Katyusha combined the techniques from stochastic gradient descent, variance reduction and Nesterov's acceleration method. In particular, at each step, Katyusha randomly select a subset $S \subset [m]$ and use $\{\nabla\phi_j(\cdot) : j \in S\}$ to form a stochastic estimator of the gradient $\nabla\phi(\cdot)$. The convergence rate depends on the way we choose S, as shown in [35]. We will apply L-Katyusha[1] using nonuniform sampling with replacement [35]. More precisely, we use the following stochastic

[1] L-Katyusha stands for Loopless Katyusha. The algorithm Katyusha was first proposed by Allen-Zhu [2]. The loopless variants [21,35] have the same complexity order as the original one but has simpler implementation form and improved practical efficiency.

gradient estimator:

$$\sum_{j=1}^{\tau} p_j^{-1} \nabla \phi_{\sigma_j}(\cdot),$$

where σ_j is a random integer equal to j with probability $p_j := L_j/(L_1 + \ldots L_m)$. Here $\tau \in [m]$ is the batch size. We shall consider the case when $\tau \leq \sqrt{m}$, for which there is linear speedup with respect to the increasing batch size. In this case, if $\{x^k\}$ is the output after k iterations of L-Katyusha starting with x^0 as initial solution, then we know from [35] that

$$\mathbb{E}\left[G(x^k) - G^\star\right] \leq \frac{1}{2}\left(G(x^0) - G^\star\right), \ \ \forall k \geq \frac{10}{\tau} \max\left(m, \sqrt{(L_1 + \cdots + L_m)/\mu}\right). \tag{54}$$

Similarly, L-Katyusha becomes more efficient than APG when m is large. Moreover it enjoys linear speedup with increasing batch size τ.

4.4 Bregman proximal gradient

In this section, we recall the Bregman proximal gradient method for solving (51). This algorithm is an extension of the classical proximal gradient method in the case when ϕ does not have a Lipschitz continuous gradient but satisfies the so-called relative smoothness condition [8,25]. The latter means the existence of a convex function $\xi(\cdot)$ differentiable on $\mathrm{dom}(P)$ and $L > 0$ such that

$$D_\phi(y; x) \leq L D_\xi(y; x), \ \ \forall x, y \in \mathrm{dom}(P).$$

In addition, assume that there is $\mu > 0$ such that for any $y \in \mathrm{dom}(P)$, there is $y^\star \in \arg\min_y \{G(y) : y \in \mathbb{R}^n\}$ satisfying

$$G(y) - G^\star \geq \mu D_\xi(y; y^\star).$$

Let $\{x^k\}$ be the output after k iterations of the Bregman proximal gradient method starting with x^0 as initial solution. Then by [25, Theorem 3.1], we have

$$G(x^k) - G^\star \leq \frac{1}{2}\left(G(x^0) - G^\star\right), \ \ \forall k \geq 2L/\mu.$$

Note that this method requires that the following problem

$$\arg\min\{P(y) + \langle \nabla \phi(x), y - x\rangle + L D_\xi(y; x) : y \in \mathbb{R}^n\},$$

is easily solvable for any $x \in \mathrm{dom}(P)$.

Before we end this section, we note that the above four methods, with appropriate restart if necessary, are linearly convergent.

5 Applications

In this section we apply Algorithm 2 in different circumstances using the inner solvers discussed in Sect. 4. We denote by $\mu_g \geq 0$ the strong convexity parameter of the function g. Recall that the objective function to be minimized at outer iteration s is:

$$H_s(x) \equiv f(x) + g(x) + h(p(x); \lambda^s, \beta_s) + \frac{\beta_s}{2}\|x - x^{s-1}\|^2, \tag{55}$$

which can be written in the form of (51) as follows:

$$H_s(x) = \phi_s(x) + P_s(x), \tag{56}$$

with

$$\phi_s(x) \equiv f(x) + h(p(x); \lambda^s, \beta_s), \quad P_s(x) \equiv g(x) + \frac{\beta_s}{2}\|x - x^{s-1}\|^2. \tag{57}$$

Note that due to Lemma 2, we have

$$\begin{aligned} H_s(x) - H_s^\star \geq \frac{\beta_s + \mu_g}{2}\|x - y^\star\|^2 + D_f(x; y^\star), \\ \forall x \in \mathrm{dom}(g), \quad y^\star = \arg\min_y H_s(y). \end{aligned} \tag{58}$$

5.1 Composition with linear functions

Throughout this subsection we consider the special case when $p(x)$ is a linear function. More precisely we focus on the following problem:

$$\begin{aligned} \min_{x \in \mathbb{R}^n} \quad & F(x) \equiv f(x) + g(x) + h_1(A_1 x) \\ \text{s.t.} \quad & A_2 x \in \mathcal{K} \end{aligned} \tag{59}$$

where $A_1 \in \mathbb{R}^{d_1 \times n}$ and $A_2 \in \mathbb{R}^{d_2 \times n}$. Recall that in this special case condition (b) and (c) automatically holds. In addition we have

$$\phi_s(x) \equiv f(x) + h(Ax; \lambda^s, \beta_s), \tag{60}$$

where $A =: \begin{pmatrix} A_1 \\ A_2 \end{pmatrix}$. In view of Lemma 2 and (19), the function $\phi_s(\cdot)$ in (60) is differentiable with respect to x and

$$D_{\phi_s}(y; x) \leq \frac{\|A\|^2}{2\beta_s}\|x - y\|^2 + D_f(y; x), \quad \forall x, y \in \mathrm{dom}(g). \tag{61}$$

We next consider three subcases based on three different assumptions on the functions f and h.

5.1.1 APG as inner solver

In this subsection we consider the case when the function f satisfies the following additional assumption.

Assumption 3 There is $L > 0$ such that

$$D_f(y; x) \leq \frac{L}{2}\|x - y\|^2, \ \ \forall x, y \in \mathrm{dom}(g).$$

Under Assumption 3, it is clear from (61) that

$$D_{\phi_s}(y; x) \leq \frac{L + \beta_s^{-1}\|A\|^2}{2}\|x - y\|^2, \ \ \forall x, y \in \mathrm{dom}(g).$$

Together with (58), we know from Sect. 4.1 that in this case APG [4,30,45] can be used as an inner solver with

$$K_s \leq 2\sqrt{\frac{2(L + \beta_s^{-1}\|A\|^2)}{\mu_g + \beta_s}} + 1. \tag{62}$$

Then the following result follows directly from Corollary 4.

Corollary 5 *Consider problem* (59) *under Assumption* 1 *and* 3 *. Let us apply Algorithm* 2 *with APG* [4,30,45] *as inner solver* $\mathcal{A}$*. Then to obtain an* ϵ*-solution in the sense of* (48)*, the expected number of APG iterations is bounded by*

$$\begin{cases} \tilde{O}\left(\frac{\sqrt{L\beta_0 + \|A\|^2} + \sqrt{\mu_g}}{\sqrt{\mu_g \epsilon}}\right) & \text{if } \mu_g > 0 \\ \tilde{O}\left(\frac{\sqrt{L\beta_0 + \|A\|^2}}{\epsilon}\right) & \text{if } \mu_g = 0 \end{cases} \tag{63}$$

5.1.2 Large scale structured problem

In this subsection we consider the following structured special case of (59).

$$\begin{aligned} \min_{x \in \mathbb{R}^n} \ \ & F(x) \equiv \sum_{j=1}^{m_1} f_j(B_j x) + \sum_{i=1}^{n} g_i(x^i) + \sum_{j=m_1+1}^{m_2} \psi_j(B_j x) \\ \text{s.t.} \ \ & B_j x \in \mathcal{K}_j, \ \ j \in \{m_2 + 1, \ldots, m\} \end{aligned} \tag{64}$$

Here $\{B_j : j \in [m]\}$ are matrices/vectors of appropriate dimensions. In addition we make the following assumption.

Assumption 4 The functions g_i, ψ_j are all convex, proper closed and simple functions. The sets $\mathcal{K}_j$ are all convex, closed and simple sets. Moreover, for each $j \in [m_1]$, the function f_j is convex and

$$D_{f_j}(y;x) \leq \frac{1}{2}\|x-y\|^2, \ \ \forall x, y \in \mathrm{dom}(g).$$

For each $j \in [m_2]$, the function ψ_j is Lipschitz continuous.

In this case, the function ϕ_s defined as in (60) can be written in the form of finite sum problem:

$$\phi_s(x) \equiv \frac{1}{m}\sum_{j=1}^{m}\phi_s^j(x).$$

Here, for each $j \in [m_1]$, ϕ_s^j is a composite function of m-Lipschitz convex function and linear operator B_j so that

$$D_{\phi_s^j}(x+he_i;x) \leq \frac{m(B_j^\top B_j)_{i,i}}{2}h^2, \ \ \forall x \in \mathrm{dom}(g), i \in [n], x+he_i \in \mathrm{dom}(g), \tag{65}$$

$$D_{\phi_s^j}(y;x) \leq \frac{m\|B_j\|^2}{2}\|x-y\|^2, \ \ \forall x, y \in \mathrm{dom}(g). \tag{66}$$

For each $j \in \{m_1+1, \ldots, m\}$, ϕ_s^j is a composite function of $m\beta_s^{-1}$-Lipschitz convex function (due to Lemma 1) and linear operator B_j,

$$D_{\phi_s^j}(x+he_i;x) \leq \frac{m(B_j^\top B_j)_{i,i}}{2\beta_s}h^2, \ \ \forall x \in \mathrm{dom}(g), i \in [n], x+he_i \in \mathrm{dom}(g), \tag{67}$$

$$D_{\phi_s^j}(y;x) \leq \frac{m\|B_j\|^2}{2\beta_s}\|x-y\|^2, \ \ \forall x, y \in \mathrm{dom}(g). \tag{68}$$

Combining (65) and (67) we get

$$D_{\phi_s}(x+he_i;x) \leq \frac{\sum_{j=1}^{m_1}(B_j^\top B_j)_{i,i} + \beta_s^{-1}\sum_{j=m_1+1}^{m}(B_j^\top B_j)_{i,i}}{2}h^2,$$
$$\forall x \in \mathrm{dom}(g), i \in [n], x+he_i \in \mathrm{dom}(g).$$

In view of Sect. 4.2, APPROX can be used as inner solver with

$$K_s \leq 2n\sqrt{\frac{2\max_i\left(\sum_{j=1}^{m_1}(B_j^\top B_j)_{i,i} + \beta_s^{-1}\sum_{j=m_1+1}^{m}(B_j^\top B_j)_{i,i}\right)}{\mu_g+\beta_s}} + 2 + 1.$$

In view of (66), (68) and Sect. 4.3, L-Katyusha with batch size $\tau \leq \sqrt{m}$ can also be used as inner solver with

$$K_s \leq 10 \max\left(m, \sqrt{\frac{m\sum_{j=1}^{m_1}\|B_j\|^2 + m\beta_s^{-1}\sum_{j=m_1+1}^{m}\|B_j\|^2}{\mu_g+\beta_s}}\right)/\tau + 1.$$

Corollary 6 *Consider problem* (64) *under Assumption* 1 *and* 4. *Let us apply Algorithm* 2 *with restart APPROX* [15] *as inner solver $\mathcal{A}$. Then to obtain an ϵ-solution in the sense of* (48)*, the expected number of APPROX iterations is bounded by*

$$\begin{cases} \tilde{O}\left(\frac{n\sqrt{\max_i\left(\beta_0\sum_{j=1}^{m_1}(B_j^\top B_j)_{i,i}+\sum_{j=m_1+1}^{m}(B_j^\top B_j)_{i,i}\right)}+n\sqrt{\mu_g}}{\sqrt{\mu_g\epsilon}}\right) & \text{if } \mu_g > 0 \\ \tilde{O}\left(\frac{n\sqrt{\max_i\left(\beta_0\sum_{j=1}^{m_1}(B_j^\top B_j)_{i,i}+\sum_{j=m_1+1}^{m}(B_j^\top B_j)_{i,i}\right)}+n}{\epsilon}\right) & \text{if } \mu_g = 0 \end{cases} \tag{69}$$

If we apply Algorithm 2 *with L-Katyusha* [35] *as inner solver $\mathcal{A}$ and mini batch size $\tau \leq \sqrt{m}$, then to obtain an ϵ-solution in the sense of* (48)*, the expected number of L-Katyusha iterations is bounded by*

$$\begin{cases} \tilde{O}\left(\frac{\sqrt{m\beta_0\sum_{j=1}^{m_1}\|B_j\|^2+m\sum_{j=m_1+1}^{m}\|B_j\|^2}+m\sqrt{\mu_g}}{\tau\sqrt{\mu_g\epsilon}}\right) & \text{if } \mu_g > 0 \\ \tilde{O}\left(\frac{\sqrt{m\beta_0\sum_{j=1}^{m_1}\|B_j\|^2+m\sum_{j=m_1+1}^{m}\|B_j\|^2}+m}{\tau\epsilon}\right) & \text{if } \mu_g = 0 \end{cases} \tag{70}$$

Since (64) is a special case of (59), we can also use APG as inner solver and apply Corollary 5. However, note that the bounds provided by (63), (69) and (70) are not directly comparable since the iteration cost of APG, APPROX and L-Katyusha are different. When carefully implemented, n iterations of APPROX or m/τ iterations of L-Katyusha with mini-batch size τ has the same order of computational complexity as one iteration of APG. Indeed, n iterations of APPROX or m/τ iterations of L-Katyusha is in expectation equivalent to one full gradient evaluation (i.e., computation of the gradient of ϕ_s), which is required in every iteration of APG. We provide in Table 1 a comparison of the three inner solvers in terms of *batch complexity*, i.e. the number of full gradient evaluation. To simplify we consider the case when $\|B_j\|^2 = 1$ for all $j \in [m]$ and let $\beta_0 = 1$ and let

$$\mathcal{B} := \begin{pmatrix} B_1^\top & \cdots & B_m^\top \end{pmatrix} \begin{pmatrix} B_1 \\ \vdots \\ B_m \end{pmatrix} \in \mathbb{R}^{n\times n}. \tag{71}$$

Note that

$$\begin{aligned}&\lambda_{\max}(\mathcal{B}) \geq \max_i \lambda_{\max}(B_i^\top B_i) = \max_i \|B_i\|^2 = 1,\\ &\lambda_{\max}(\mathcal{B}) \geq \max_i \mathcal{B}_{i,i} \geq \tfrac{\mathrm{trace}(\mathcal{B})}{n} = \tfrac{1}{n}\textstyle\sum_{i=1}^m \mathrm{trace}(B_i^\top B_i) \geq \tfrac{1}{n}\sum_{i=1}^m \|B_i\|^2 = \tfrac{m}{n}.\end{aligned} \tag{72}$$

In addition, note that the bound in (72) is conservative. Indeed, the maximal eigenvalue is often much larger than the maximal diagonal element. We then draw the following conclusion from Table 1.

1. Using APPROX or L-Katyusha as inner solver yields better batch complexity bound than using APG as inner solver.
2. When $m \gg n$, using L-Katyusha as inner solver yields better batch complexity bound than using APPROX as inner solver.

Remark 6 (Parallel Linear Speedup) Note that the batch complexity bound for L-Katyusha in Table 1 is independent of the mini batch size τ. This means that Algorithm 2 with L-Katyusha as inner solver enjoys a parallel linear speedup when $\tau \leq \sqrt{m}$.

Remark 7 When we compare the bounds in Table 1 with other related work, some additional transformation is needed due to different problem formulation. Here we provide one example of comparing the bounds of our Table 1 with the complexity bound established in [23] for one special case of problem (64) when $m_1 = 0$ and $m_2 = m$. Consider the following regularized empirical risk minimization model with $\mu_g > 0$:

$$\min_{x\in\mathbb{R}^n} \quad F(x) \equiv \sum_{i=1}^n g_i(x^i) + \frac{1}{m}\sum_{j=1}^m m\psi_j(B_j x) \tag{73}$$

which corresponds to problem (1.2) in [23]. W.l.o.g. we assume that each ψ_j in is 1-Lipschitz continuous so that we know $L_{h_1} \leq \sqrt{m}$. Then by [23, Corollary 3], the number of iterations of Algorithm 4 in [23] is bounded by $O\left(m\sqrt{\frac{m}{\mu_g\epsilon}}\right)$, which corresponds to a batch complexity bound $O\left(\sqrt{\frac{m}{\mu_g\epsilon}}\right)$. The $\tilde{O}$ in Table 1 hides the constant c_3 defined in (50) which is proportional to $\sqrt{c_1}$ when $\mu_g > 0$. Recall the definition of c_1 in (33), which is bounded by $O(L_{h_1}^2) = O(m)$. Hence the batch complexity bound of our Algorithm 2 with L-Katyusha as inner solver for problem (73) is $\tilde{O}\left(\sqrt{\frac{m}{\mu_g\epsilon}}\right)$, which differs from the bound of [23] by a logarithm term. Nevertheless, note that our Algorithm 2 with L-Katyusha can enjoy a linear speedup up to $\tau \leq \sqrt{m}$ if parallel implementation is used, see Remark 6.

Table 1 Comparison of batch complexity bounds of Algorithm 2 applied on problem (64) using different inner solvers

Inner solver	Strongly convex case ($\mu_g > 0$)
APG (Corollary 5)	$\tilde{O}\left(\frac{\sqrt{\lambda_{\max}(\mathcal{B})}}{\sqrt{\mu_g \epsilon}}\right)$
APPROX (Corollary 6)	$\tilde{O}\left(\frac{\sqrt{\max_i \mathcal{B}_{i,i}}}{\sqrt{\mu_g \epsilon}}\right)$
L-Katyusha (Corollary 6)	$\tilde{O}\left(\frac{1}{\sqrt{\mu_g \epsilon}}\right)$
	non-strongly convex case ($\mu_g = 0$)
APG (Corollary 5)	$\tilde{O}\left(\frac{\sqrt{\lambda_{\max}(\mathcal{B})}}{\epsilon}\right)$
APPROX (Corollary 6)	$\tilde{O}\left(\frac{\sqrt{\max_i \mathcal{B}_{i,i}}}{\epsilon}\right)$
L-Katyusha (Corollary 6)	$\tilde{O}\left(\frac{1}{\epsilon}\right)$

Here we consider the special case when $\|B_j\|^2 = 1$ for all $j \in [m]$ and let $\beta_0 = 1$. The matrix $\mathcal{B}$ is defined as in (71) and $\mathcal{B}_{i,i}$ denotes the ith diagonal element of $\mathcal{B}$

5.1.3 Bregman proximal gradient as inner solver

In this subsection we consider the case when f is relatively smooth.

Assumption 5 There is a convex function ξ differentiable on an open set containing $\mathrm{dom}(g)$ and $L > \mu > 0$ such that

$$\mu D_\xi(y;x) \le D_f(y;x) \le L D_\xi(y;x), \quad \forall x, y \in \mathrm{dom}(g).$$

Moreover, for any $\alpha, \beta > 0$, $x \in \mathrm{dom}(\xi)$ and $x' \in \mathbb{R}^n$ the problem

$$\min_y \left\{ g(y) + \frac{\beta}{2}\|y - x'\|^2 + \alpha D_\xi(y;x) \right\},$$

is easily solvable.

In this case, by (61) we know that

$$\begin{aligned} D_{\phi_s}(y;x) &\le \frac{\beta_s^{-1}\|A\|^2}{2}\|x-y\|^2 + L D_\xi(y;x) \\ &\le \max\left(\beta_s^{-1}\|A\|^2, L\right)\left(\frac{1}{2}\|x-y\|^2 + D_\xi(y;x)\right) \quad \forall x, y \in \mathrm{dom}(g). \end{aligned}$$

Moreover, by (58) we know that

$$\begin{aligned} H_s(x) - H_s^\star &\ge \frac{\beta_s + \mu_g}{2}\|x - y^\star\|^2 + \mu D_\xi(x; y^\star) \\ &\ge \min\left(\beta_s + \mu_g, \mu\right)\left(\frac{1}{2}\|x - y^\star\|^2 + D_\xi(x; y^\star)\right), \end{aligned}$$

$$\forall x \in \mathrm{dom}(g), \quad y^\star \in \arg\min_y H_s(y).$$

Therefore, based on Sect. 4.4, the Bregman proximal gradient can be used as an inner solver with

$$K_s \leq \frac{2\max\left(\beta_s^{-1}\|A\|^2, L\right)}{\min\left(\beta_s + \mu_g, \mu\right)} + 1.$$

Corollary 7 *Consider problem (59) under Assumption 1 and 5. Let us apply Algorithm 2 with Bregman proximal gradient [8,25] as inner solver $\mathcal{A}$. Then to obtain an ϵ-solution in the sense of (48), the expected number of Bregman proximal gradient iterations is bounded by*

$$\begin{cases} \tilde{O}\left(\frac{\max\left(\|A\|^2, L\beta_0\right)+\min(\mu_g,\mu)}{\min(\mu_g,\mu)\epsilon}\right) & \text{if } \min(\mu_g, \mu) > 0 \\ \tilde{O}\left(\frac{\max\left(\|A\|^2, L\beta_0\right)}{\epsilon^2}\right) & \text{if } \min(\mu_g, \mu) = 0 \end{cases}$$

5.2 Composition with nonlinear functions

In this section we consider the general case when $p(x)$ is possibly nonlinear.

Assumption 6 There is $M_{p_2} > 0$, $M_{\nabla p} > 0$ and $L_{\nabla p} > 0$ such that

$$\|p_2(x)\| \leq M_{p_2}, \quad \forall x \in \mathrm{dom}(g), \tag{74}$$
$$\|\nabla p(x)\| \leq M_{\nabla p}, \quad \forall x \in \mathrm{dom}(g), \tag{75}$$
$$\|\nabla p(x) - \nabla p(y)\| \leq L_{\nabla p}\|x - y\|, \quad \forall x, y \in \mathrm{dom}(g). \tag{76}$$

Note that the same type of assumptions was used in [26, Section 2.4]. In particular as mentioned in [26], if the domain of g is compact then Assumption 6 holds. Assumption 6 is made in order to obtain the smoothness of the function $\nabla\phi_s$. Recall from (57) and Lemma 2 that

$$\nabla\phi_s(x) = \nabla f(x) + \nabla p(x)\Lambda(p(x); \lambda^s, \beta_s).$$

Lemma 5 *Under Assumption 6, $\forall x, y \in \mathrm{dom}(g)$,*

$$\|\nabla\phi_s(x) - \nabla\phi_s(y)\| \leq \|\nabla f(x) - \nabla f(y)\| + \left(L_{\nabla p}\left(L_{h_1} + \beta_s^{-1}d_s\right) + M_{\nabla p}^2\beta_s^{-1}\right)\|x - y\|,$$

where

$$d_s := \max_y \min_x \{\|x - y\| : x \in \mathcal{K}, \|y\| \leq M_{p_2} + \beta_s\|\lambda_2^s\|\} < +\infty.$$

Further, let Assumption 3 hold. Then Lemma 5 implies

$$D_{\phi_s}(y;x) \leq \frac{\left(L + L_{\nabla p}\left(L_{h_1} + \beta_s^{-1} d_s\right) + M_{\nabla p}^2 \beta_s^{-1}\right)}{2} \|x - y\|^2, \quad \forall x, y \in \mathrm{dom}(g).$$

Together with (58), we know from Sect. 4.1 that in this case APG [4,30,45] can be used as an inner solver with

$$K_s \leq 2\sqrt{\frac{2\left(L + L_{\nabla p}\left(L_{h_1} + \beta_s^{-1} d_s\right) + M_{\nabla p}^2 \beta_s^{-1}\right)}{\mu_g + \beta_s}} + 1. \tag{77}$$

Corollary 8 *Consider problem* (1) *under Assumption* 1, 3 *and* 6. *Let us apply Algorithm* 2 *with APG* [4,30,45] *as inner solver* $\mathcal{A}$. *Then to obtain an* ϵ*-solution in the sense of* (48), *the expected number of APG iterations is bounded by*

$$\begin{cases} \tilde{O}\left(\frac{\sqrt{L\beta_0 + L_{\nabla p}(L_{h_1}\beta_0 + d_s) + M_{\nabla p}^2} + \sqrt{\mu_g}}{\sqrt{\mu_g \epsilon}}\right) & \text{if } \mu_g > 0 \\ \tilde{O}\left(\frac{\sqrt{L\beta_0 + L_{\nabla p}(L_{h_1}\beta_0 + d_s) + M_{\nabla p}^2}}{\epsilon}\right) & \text{if } \mu_g = 0 \end{cases}$$

Remark 8 Corollary 8 recovers Corollary 5 as a special case with $L_{\nabla p} = 0$ and $M_{\nabla p} = \|A\|$.

Similarly, we could consider the large-scale structured problem as (64) but with nonlinear composite terms, or the relatively smooth assumption as in Sect. 5.1.3 instead of Assumption 3,. The same order of iteration complexity bound as Corollary 6 and 7 can be derived for the nonlinear composite case under Assumption 6.

6 Further discussion

6.1 Efficient inner problem stopping criteria

In Algorithm 2, we provide an upper bound on the number of inner iterations m_s needed in order to obtain an solution x^s such that

$$\mathbb{E}[H_s(x^s) - H_s^\star] \leq \epsilon_s.$$

In some cases, it is possible to have a computable upper bound $U_s(x^s)$ such that $U_s(x^s) \geq H_s(x^s) - H_s^\star$. Then we can check the value of $U_s(x^s)$ and stop the inner solve either when $U_s(x^s) \leq \epsilon_s$ or when the number of inner iterations exceeds m_s. Note that the solution x^s obtained in this way satisfies $\mathbb{E}[H_s(x^s) - H_s^\star] \leq 2\epsilon_s$, which is equivalent to a change from ϵ_0 to $2\epsilon_0$ in the previous analysis and hence all the previous

complexity bounds apply. Below we discuss two cases when such computable upper bound exists.

Recall from (56) that $H_s(x)$ can be written in the form of $H_s(x) = \phi_s(x) + P_s(x)$. Suppose that there is $L_s > 0$ such that

$$D_{\phi_s}(y; x) \leq \frac{L_s}{2}\|x - y\|^2, \ \ \forall x, y \in \mathrm{dom}(g). \tag{78}$$

Let

$$T(x) := \arg\min_{y\in\mathbb{R}^n} \left\{ \langle \nabla\phi_s(x), y - x\rangle + \frac{L_s}{2}\|y - x\|^2 + P_s(y) \right\}.$$

A basic property about proximal gradient step implies that

$$H_s(T(x)) - H_s^\star \leq \frac{8L_s^2\,\|x - T(x)\|^2}{\beta_s}, \ \ \forall x \in \mathrm{dom}\, g, \tag{79}$$

see e.g. [16, Propostion 4]. Note that in Sects. 5.1.1 and 5.2., we have given suitable conditions guaranteeing (78).

In the special case of the structured problem (64), the inner problem takes the following form

$$\min_{x\in\mathbb{R}^n} \left[F(x) \equiv \Psi(x) + \sum_{j=1}^m \Phi_j(B_j x) \right], \tag{80}$$

to which we can associate the following dual problem:

$$\max_{y\in\mathbb{R}^m} \left[\mathcal{D}(y) \equiv -\Psi^*(-B^\top y) - \sum_{j=1}^m \Phi_j^*(y_j) \right], \tag{81}$$

where $B =: \begin{pmatrix} B_1 \\ \vdots \\ B_m \end{pmatrix}$. In this case, when Ψ^* and Φ_j^* are easy to compute, a computable upper bound is given by $F(x^s) - \mathcal{D}(y^s)$ where y^s is a dual feasible solution constructed from x^s.

6.2 KKT solution

The convergence of ALM can also be measured through the KKT residual. Recall that a solution is said to be an ϵ-KKT solution if there exists $(u, v) \in \partial L(x, \lambda)$ such that $\|u\| \leq \epsilon$ and $\|v\| \leq \epsilon$, see e.g. [26]. Due to the possible randomness of the iterates in our algorithm, we shall measure the expected distance of the partial gradient of the Lagrangian to 0.

Algorithm 3 IPALM_KKT($\mathcal{A}$)

Parameters: $\beta_0 > 0, \epsilon_0 > 0, \rho \in (1/2, 1), \eta \in (0, \rho^3]$, $\{L_s\}_{s\geq 0}$ satisfying (78)
Initialize: $x^{-1} \in \mathrm{dom}(g)$, $\lambda^0 \in \mathrm{dom}(h^*)$, $\mathcal{L}_0 \leq H_0^\star$
1: $m_0 = \lceil \max\left(K_0 \log_2\left((H_0(x^{-1}) - \mathcal{L}_0)/\epsilon_0\right), 0\right)\rceil$
2: $\tilde{x}^0 \leftarrow \mathcal{A}(x^{-1}, m_0, H_0)$
3: **for** $s = 0, 1, 2, \ldots$ **do**
4: $\quad x^s \leftarrow \arg\min_{y\in\mathbb{R}^n}\left\{\langle \nabla\phi_s(\tilde{x}^s), y - \tilde{x}^s\rangle + \frac{L_s}{2}\|y - \tilde{x}^s\|^2 + \frac{\beta_s}{2}\|y - x^{s-1}\|^2 + g(y)\right\}$
5: $\quad \lambda^{s+1} \leftarrow \Lambda(p(x^s); \lambda^s, \beta_s)$
6: $\quad \beta_{s+1} = \rho\beta_s$
7: $\quad \epsilon_{s+1} = \eta\epsilon_s$
8: $\quad$ choose m_{s+1} to be the smallest integer satisfying (38)
9: $\quad \tilde{x}^{s+1} \leftarrow \mathcal{A}(x^s, m_{s+1}, H_{s+1})$
10: **end for**

For simplicity we restrict the discussion for the case when for any outer iteration s there is a constant $L_s > 0$ such that (78) holds. We modify slightly Algorithm 2 by adding one additional proximal gradient step (Line 5 in Algorithm 3) into each outer iteration. In addition, in Algorithm 3 we require η to be smaller than ρ^3. Since the proximal gradient step is guaranteed to decrease the objective value, we have

$$\mathbb{E}[H_s(x^s) - H_s^\star] \leq \mathbb{E}[H_s(\tilde{x}^s) - H_s^\star] \leq \epsilon_s, \quad \forall s \geq 0. \tag{82}$$

Hence Algorithm 3 falls into the class of Algorithm 1 and all the results in Sect. 2.2 can be applied. Moreover, in analogue to Theorem 1, we have the following bounds for the KKT residual.

Theorem 3 *Consider Algorithm 3. For any $s \geq 0$ we have*

$$\mathrm{dist}(0, \partial_x L(x^s, \lambda^{s+1})) \leq \sqrt{16L_s\left(H_s(\tilde{x}^s) - H_s^\star\right) + 2\beta_s^2\|x^s - x^{s-1}\|^2}\,, \tag{83}$$

$$\mathrm{dist}(0, \partial_\lambda L(x^s, \lambda^{s+1})) \leq \beta_s\|\lambda^{s+1} - \lambda^s\|. \tag{84}$$

Corollary 9 *Consider Algorithm 3. Assume that there is $\gamma > 0$ such that $L_s \leq \gamma\beta_s^{-1}$. Then to obtain a solution such that*

$$\mathbb{E}\left[\mathrm{dist}(0, \partial_x L(x^s, \lambda^{s+1}))\right] \leq \epsilon, \ \ \mathbb{E}\left[\mathrm{dist}(0, \partial_\lambda L(x^s, \lambda^{s+1}))\right] \leq \epsilon, \tag{85}$$

it suffices to run Algorithm 3 for

$$s \geq \frac{\ln(c_4/\epsilon)}{\ln(1/\rho)}, \tag{86}$$

number of outer iterations where

$$c_4 := \max\left(\sqrt{16\gamma\epsilon_0/\beta_0 + 8c_0\beta_0}, \beta_0\sqrt{c_0}\right).$$

Note that the outer iteration bound (86) for the KKT convergence (85) only differs from the bound for the objective value convergence (32) by a constant in the logarithm term. For each outer iteration, Algorithm 3 has one more proximal gradient step to execute than Algorithm 2 and this will only add a term with logarithm dependence with respect to ϵ into the total complexity bound. In particular, we can derive $\tilde{O}(1/\epsilon)$ complexity bound to obtain ϵ-KKT convergence in the sense of (85), and $\tilde{O}(1/\sqrt{\epsilon})$ if the function g is strongly convex. For brevity we omit the details which are highly similar to Sect. 5.

6.3 Bounded primal and dual domain

The bound $\tilde{O}(1/\epsilon^\ell)$ can be improved to $O(1/\epsilon^\ell)$ if both the primal and dual domain are bounded. Indeed, we require $\eta < \rho$ in Algorithm 2 to ensure the boundedness (in expectation) of the sequence $\{(x^s, \lambda^s)\}$. If the domain of g is bounded and there is no constraint, i.e., $\mathcal{K} = \mathbb{R}^{d_2}$, then $\{(x^s, \lambda^s)\}$ of Algorithm 1 is bounded for any choice of $\{\epsilon_s\}$ and $\{\beta_s\}$. In this case we can let $\eta = \rho$ and the bound in (45) can be improved to

$$\sum_{t=1}^{s} \mathbb{E}[m_t] \le s + c_2 \sum_{t=1}^{s} K_t.$$

Consequently, the bound in (49) can be improved to

$$\sum_{t=0}^{s} \mathbb{E}[m_t] \le m_0 + \frac{c_1^\ell}{\epsilon^\ell \rho^\ell \ell \ln(1/\rho)} + c_2 \left(\frac{\varsigma c_1^\ell}{\rho^\ell \ell \ln(1/\rho)} + \frac{\omega c_1^\ell}{\beta_0^\ell (1-\rho^\ell)} \right) \frac{1}{\epsilon^\ell},$$

and we get the $O(1/\epsilon^\ell)$ iteration complexity bound for an ϵ-solution in the sense of (48). However, for the ϵ-KKT solution in the sense of (85) we still only have $\tilde{O}(1/\epsilon^\ell)$ iteration complexity bound.

7 Numerical experiments

We will test the performance of Algorithm 2 with APPROX and L-Katyusha as inner solver, which are referred to as IPALM-APPROX and IPALM-Katyusha. We mainly compare with first-order primal dual solvers Linearized ADMM [33], ASGARG-DL [43] and SMART-CD [1].

Since all the four algorithms depends on the choice of β_0, we test $\beta_0 \in \{10^{-2}, 10^{-1}, 1, 10, 100\}$ and choose the best result to compare. (Note that the problem data used are all scaled so that the row vectors all have norm 1.) For IPALM, the default setting is $\rho = 0.9$ and $\eta = 0.95$ in all the experiments. The choice of ϵ_0 varies with instances and can be found in the readme file in the code. We also run CVX with its default solver SDPT3 to obtain a good approximation of the optimal value $F^\star$, which

is needed in the computation of the error term:

$$\log_{10}\left|\frac{F(x)-F^\star}{F^\star}\right|. \tag{87}$$

However, due to the large-scale problem that we solve, CVX may return inaccurate solution or even fail. To solve the issue on unknown $F^\star$, note that either CVX or our algorithm can provide a lower bound F_l and an upper bound F_u so that $F^\star \in [F_l, F_u]$. Define

$$\epsilon_c := (F_u - F_l)/F_l,$$

as the confidence error level. Then for any x such that $F(x) = (1+\epsilon)F_u$ for some $1 > \epsilon > \epsilon_c$, we have

$$\epsilon = \frac{F(x)-F_u}{F_u} \le \frac{F(x)-F^\star}{F^\star} \le \frac{F(x)-F_l}{F_l} = \epsilon_c + (1+\epsilon_c)\epsilon < 3\epsilon.$$

So we use $\frac{F(x)-F_u}{F_u}$ as an approximation of $\frac{F(x)-F^\star}{F^\star}$ for those x such that $F(x) > (1+\epsilon_c)F_u$.

The code for reproducing the results in this section is downloadable from https://zenodo.org/badge/latestdoi/329807599.

7.1 Least absolute deviation

The first problem we solve is of the form:

$$\min_{x\in\mathbb{R}^n} \|Ax-b\|_1 + \lambda\|x\|_1.$$

which is also know as Least Absolute Deviation (LAD) problem [46]. We use training data of four different datasets from libsvm [12] as A and modify b such that $Ax = b$ has a sparse solution. We set $\lambda = 0.01$. The details about the datasets are given in Table 2.

The result is shown in Fig. 1. We also compare the time of CVX with IPALM-APPROX to get a mid-level accurate solution in Table 3.

As we can see form Fig. 1, IPALM-APPROX has the best performance after accuracy 10^{-3}. ASGARD-DL works with full dimensional variables and therefore has slow convergence in time. Linearized ADMM and SMART-CD have similar performance as IPALM-APPROX but tend to be slower for obtaining more accurate solution. From Table 3 we can see to get a mid-level accurate solution, IPALM-APPROX significantly outperforms CVX for these four datasets.

Table 2 Datasets from libsvm

Dataset	Training size (m)	Number of features (n)
news20scale	15,935	62,061
rcv1	20,242	47,236
rcv1mc	15,564	47,236
news20binary	19,996	1,355,191

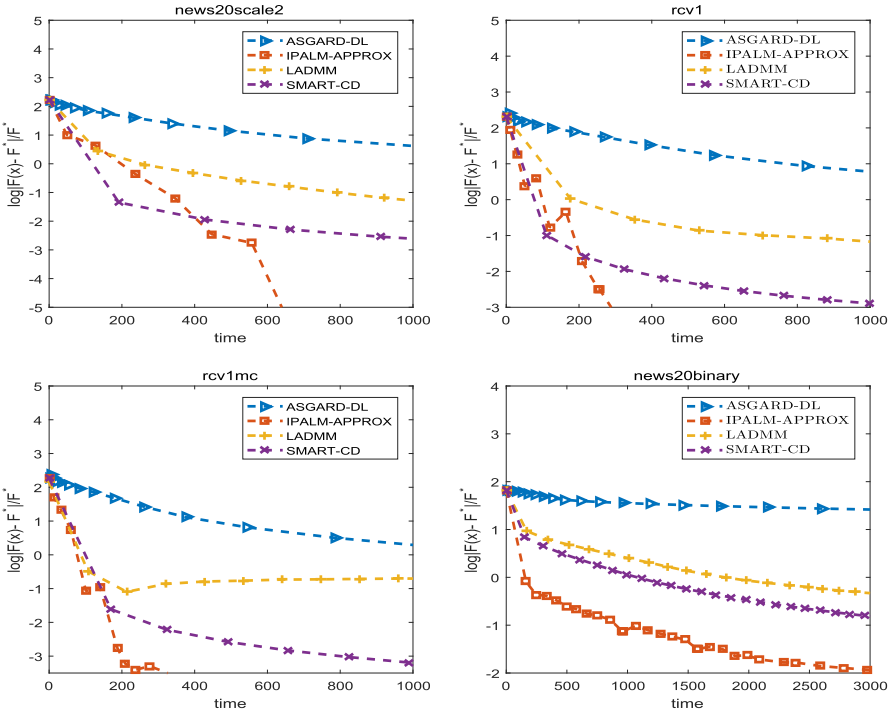


Fig. 1 Comparison of four algorithms for LAD problem on four datasets. The x-axis is time and y-axis is $\log((F(x) - F^\star)/F^\star)$. Here we use the result of CVX as an approximation of $F^\star$

Table 3 Running time of CVX and IPALM-APPROX for Least absolute deviation problem on four datasets

Dataset	Accuracy (87)	CVX time (s)	IPALM-APPROX time (s)
news20scale	10^{-5}	$\sim$4200	$\sim$ 500
rcv1	10^{-3}	$\sim$4000	$\sim$400
rcv1mc	10^{-3}	$\sim$1800	$\sim$300
news20binary	10^{-1}	$\sim$1400	$\sim$ 1000

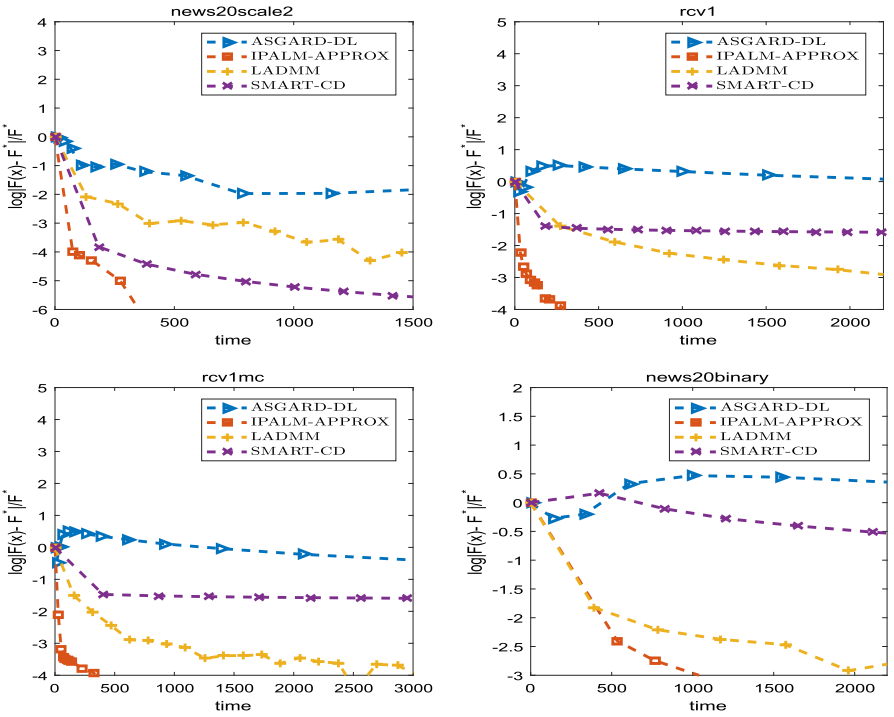


Fig. 2 Comparison of four algorithms for basis pursuit problem on four datasets. The x-axis is time and y-axis is $\log((F(x) - F^\star)/F^\star)$. Here we use the result of CVX as an approximation of $F^\star$

7.2 Basis pursuit

The second problem we solve is of the form:

$$\begin{aligned} \min_{x\in\mathbb{R}^n} \quad & \|x\|_1 \\ \text{s.t.} \quad & Ax = b \end{aligned}$$

which is known as basis pursuit problem [13]. The datasets used are shown in Table 2 and we modify b for each dataset to make sure that the problem is feasible.

The results are shown in Fig. 2 for the objective value gap and in Fig. 3 for the infeasibility gap. We also compare the time of CVX with IPALM-APPROX to get a mid-level accurate solution in Table 4.

As we can see from Figs. 2 and 3, IPALM-APPROX works well both in objective value and feasibility. Since SMART-CD reduces β much faster than IPALM-APPROX, it has fast convergence at the beginning, but small β leading to small stepsize and slow convergence in objective value for high accuracy. From Table 4, we see the difference between IPALM-APPROX and CVX if only medium accuracy is required.

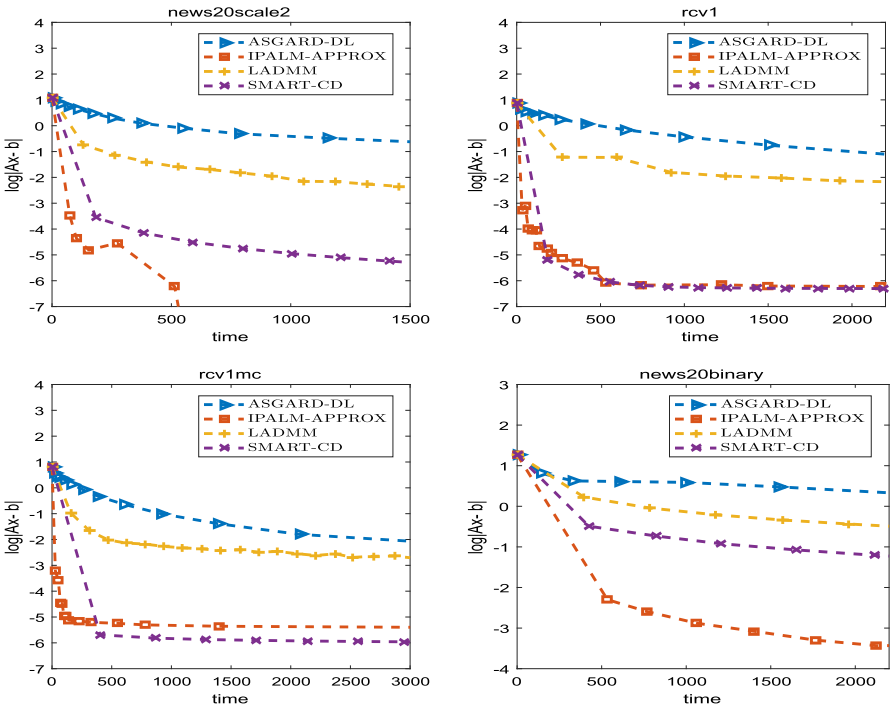


Fig. 3 Comparison of four algorithms for basis pursuit problem on four datasets. The x-axis is time and y-axis is infeasibility error $\log \|Ax - b\|$

Table 4 Running time of CVX and IPALM-APPROX for basis pursuit problem on four datasets

Dataset	Accuracy (87)	CVX time (s)	IPALM-APPROX time (s)
news20scale	10^{-6}	~4200	~500
rcv1	10^{-4}	~3600	~200
rcv1mc	10^{-4}	~1800	~500
news20binary	10^{-3}	~1800	~1000

7.3 Fused Lasso

The third problem we solve is of the form:

$$\min_{x\in\mathbb{R}^n} \frac{1}{2}\|Ax-b\|_2^2 + \lambda r\|x\|_1 + \lambda(1-r)\sum_i |x_i - x_{i+1}|.$$

which is known as Fused Lasso problem [42]. The datasets used are shown in Table 2 and we set $\lambda r = \lambda(1-r) = 0.01$.

The results are shown in Fig. 4 and Table 5. For this problem, we tested both IPALM-APPROX and IPALM-Katyusha. Note that for the datasets in Table 2, we

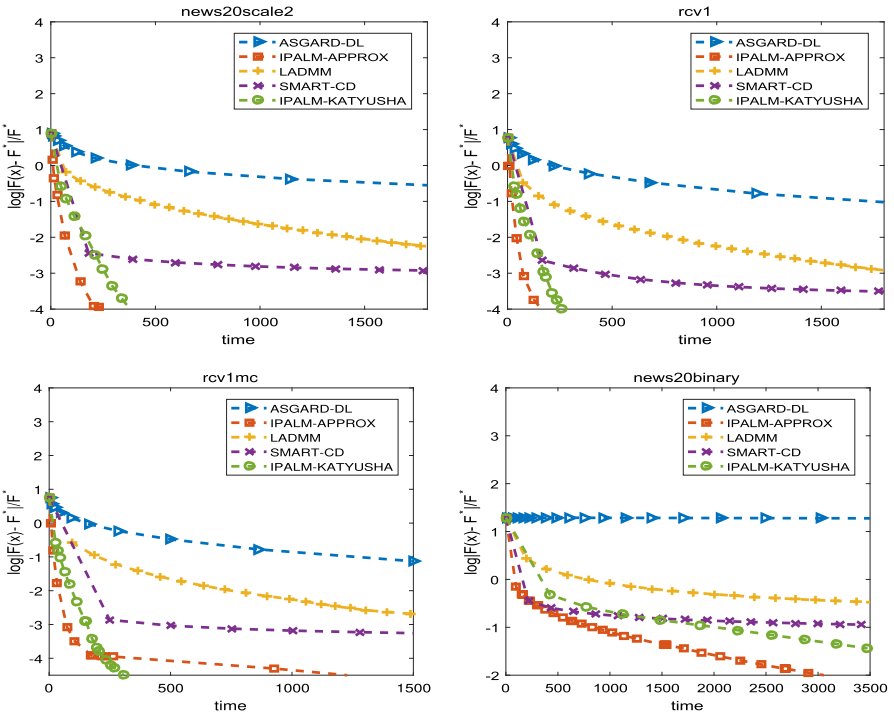


Fig. 4 Comparison of five algorithms for Fused Lasso problem on four datasets. The x-axis is time and y-axis is $\log((F(x) - F^\star)/F^\star)$. Here we use the result of CVX as an approximation of $F^\star$

have $n \leq m \leq 2n$ where m is the problem size in (64). According to Table 1, we should expect IPALM-Katyusha to work similarly as IPALM-APPROX, which is indeed observed in practice. Note that in our implementation we used $\tau = \sqrt{m}$ with single processor. Hence the computational time of IPALM-Katyusha can be further reduced when multi-processor and parallel implementation is used.

As we can see form Fig. 4, IPALM-APPROX and IPALM-Katyusha both perform better than linearized ADMM, ASGARD and SMART-CD. From Table 5, IPALM-Katyusha significantly outperforms CVX to get a mid-level accurate solution for these three datasets. CVX fails to solve the dataset news20binary since out of memory and the comparison is not included in Table 5.

7.4 Soft margin SVM

The forth problem we solve is of the form:

$$\min_{x\in\mathbb{R}^n,\omega\in\mathbb{R}} \lambda \|x\|_1 + \frac{1}{m}\sum_{i=1}^{m} \max\left(0, 1 - b_i\left(\langle a_i, x\rangle - \omega\right)\right) \tag{88}$$

Table 5 Running time of CVX, IPALM-APPROX and IPALM-Katyusha for Fused Lasso on three datasets

Dataset	Accuracy	CVX time (s)	IPALM-APPROX time (s)	IPALM-Katyusha time (s)
news20scale	10^{-4}	$\sim$1600	$\sim$1700	$\sim$ 400
rcv1	10^{-4}	$\sim$7000	$\sim$300	$\sim$ 300
rcv1mc	10^{-4}	$\sim$5500	$\sim$400	$\sim$ 300

Table 6 Datasets from libsvm

Dataset	Training size (m)	Number of features (n)
w4a	7366	300
a7a	16,100	123
a8a	22,696	123
a9a	32,561	123
w6a	17,188	300
w7a	24,692	300
w8a	49,479	300
ijcnn1	49,990	22
covtype	581,012	54
real-sim	72,309	209,58

Table 7 Test accuracy of LIBSVM and IPALM-Katyusha on seven datasets

Dataset	Time	Test accuracy	
		LIBSVM (%)	IPALM-Katyusha (%)
a7a	9.35 s	84.58	**84.84**
a8a	19.28 s	85.01	**85.13**
a9a	41.39 s	84.82	**84.96**
w6a	1.63 s	97.21	**98.67**
w7a	3.36 s	97.34	**98.67**
w8a	13.18 s	97.44	**98.64**
ijcnn1	31.20 s	**92.78**	92.16

which is known as l_1 regularized soft margin support vector machine problem [51]. Here $a_i \in \mathbb{R}^n$ are feature vectors and $b_i \in \{-1, 1\}$ are labels for $i = 1, \ldots, m$. We use ten different datasets from libsvm [12]. The details about the datasets are given in Table 6.

Since here $m \geq n$, we expect IPALM-Katyusha to converge faster than IPALM-APPROX, as indicated by our theoretical bounds given in Table 1. Indeed, as we observe from Figs. 5 and 6, IPALM-Katyusha has the best performance for all datasets. For most datasets in our experiment, the difference of IPALM-Katyusha and IPALM-APPROX is small. But for covtype, IPALM-Katyusha significantly outperforms IPALM-APPROX, as well as linearized ADMM, SMART-CD and ASGARD-DL, as shown in the last figure of Fig. 6. Note that for this type of problem CVX fails for most of the datasets so we do not compare the running time with CVX. Instead we compare the performance of IPALM-Katyusha with the package LIBSVM[2].

In Table 7, we compare the test accuracy of LIBSVM and IPALM-Katyusha. For LIBSVM, we use the default parameters and record its running time. Then we apply IPALM-Katyusha to solve problem (88) within the same time and compute the test accuracy with the output (x, ω). It can be seen that the performance of IPALM-

[2] https://www.csie.ntu.edu.tw/~cjlin/libsvm/.

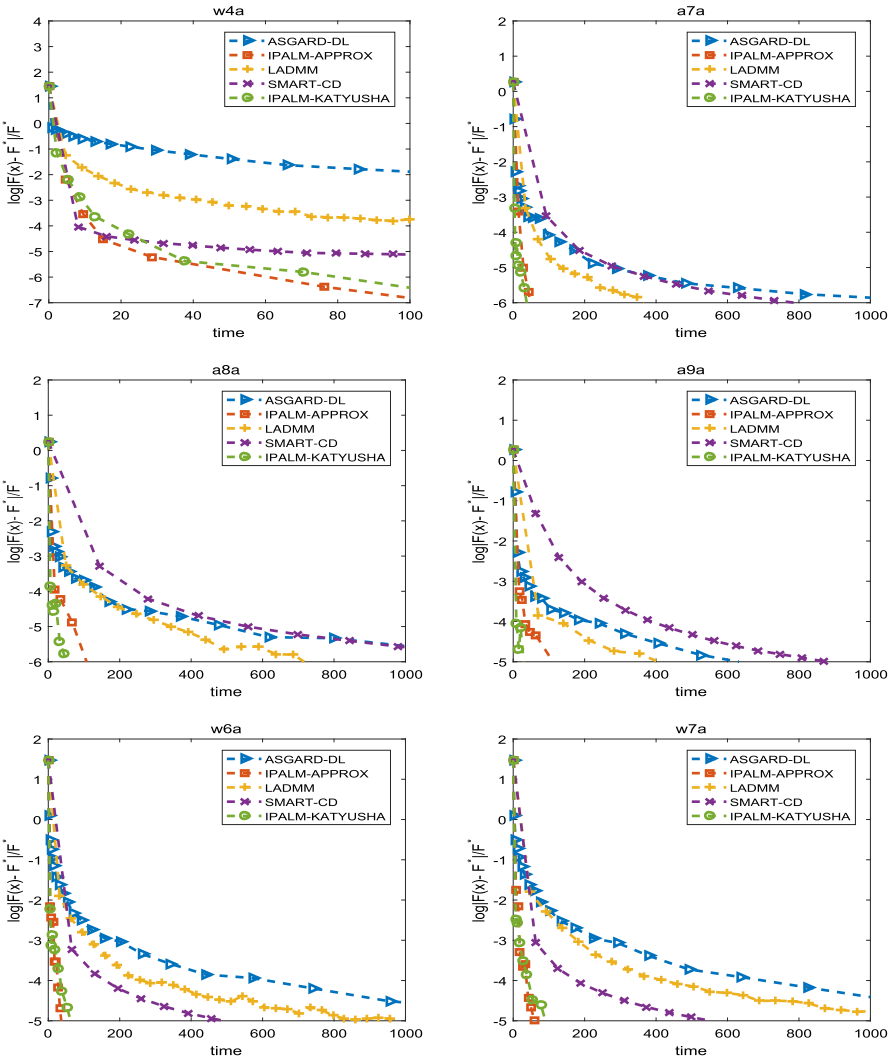


Fig. 5 Comparison of five algorithms for SVM problem on the datasets w4a, a7a, a8a, a9a, w6a, w7a. The x-axis is time and y-axis is $\log((F(x) - F^\star)/F^\star)$. Here we use the best result of these five algorithms as an approximation of $F^\star$

Katyusha is comparable with and most often slightly better than LIBSVM on these datasets.

7.5 Quadratically constrained quadratic programming

In this section we test the algorithms for solving the convex quadratically constrained quadratic program (QCQP):

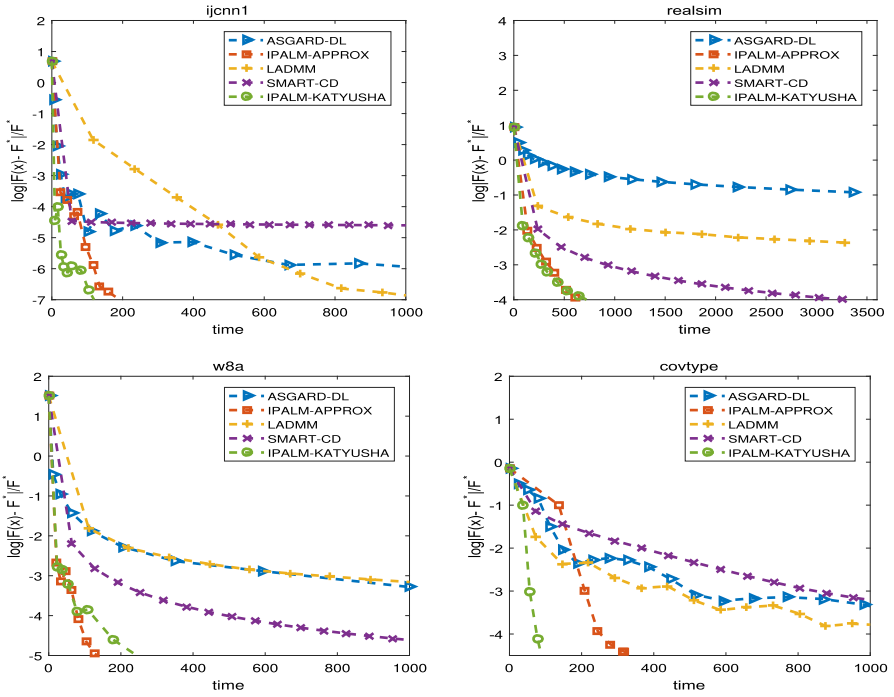


Fig. 6 Comparison of five algorithms for SVM problem on the datasets ijcnn1, realsim, w8a and covtype. The x-axis is time and y-axis is $\log((F(x) - F^\star)/F^\star)$. Here we use the best result of these five algorithms as an approximation of $F^\star$

$$\begin{aligned} \min_{x\in\mathbb{R}^n} \quad & \frac{1}{2}x^T Q_0 x + c_0^T x + d_0 \\ \text{s.t.} \quad & \frac{1}{2}x^T Q_j x + c_j^T x + d_j \le 0, \quad \forall j \in [m], \\ & l_i \le x_i \le u_i, \quad \forall i \in [n]. \end{aligned} \tag{89}$$

where $Q_0, Q_1, \ldots, Q_m$ are all positive semidefinite matrices.

For simplicity, we let $d_j = -1$ for all $j \in [m]$ and $l_i = -1$, $u_i = 1$ for all $i \in [n]$. We randomly generate positive semidefinite matrices $Q_0, Q_1, \ldots, Q_m$ and vectors $c_0, c_1, \ldots, c_m$.

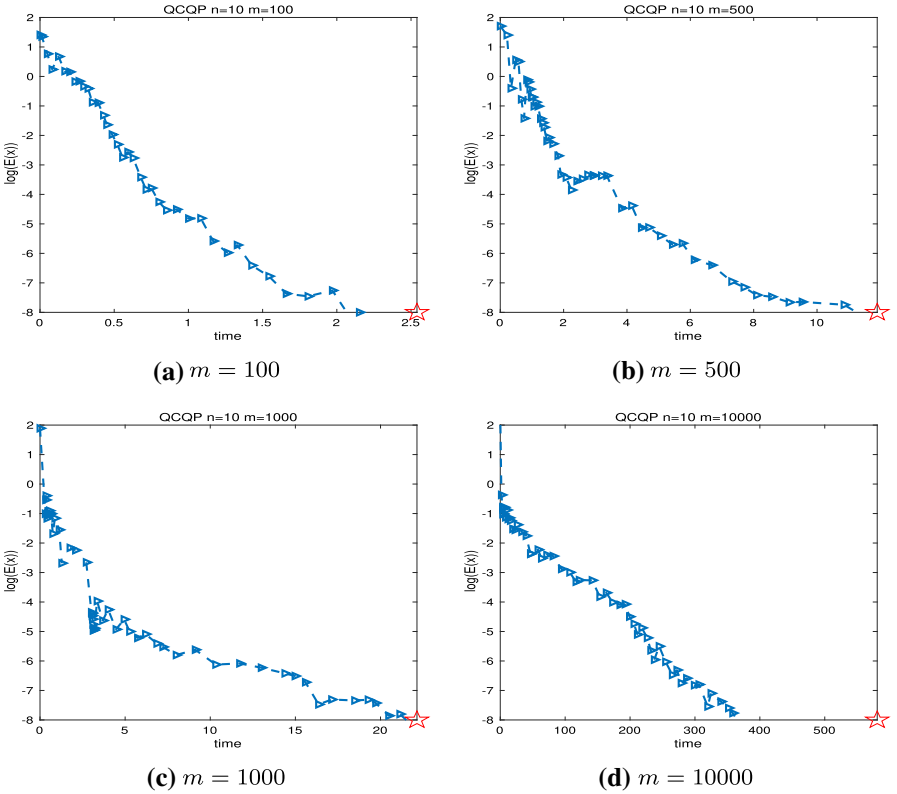
We fix the number of variables $n = 10$ and vary the number of contraints $m \in \{100, 500, 1000, 10000\}$. We report in Table 8 the running time of CVX for solving four instances of (89).

We denote by $E(x)$ the maximal of feasibility gap and rescaled objective value gap:

$$E(x) := \max\left(\frac{F(x) - F^\star}{F^\star}, \text{dist}(p(x), \mathcal{K})\right). \tag{90}$$

Table 8 CVX solution time for the four randome QCQP datasets

Dataset	Number of constraints (m)	CVX time (s)
qcqp1	100	2.54
qcqp2	500	11.90
qcqp3	1000	22.14
qcqp4	10,000	580.84



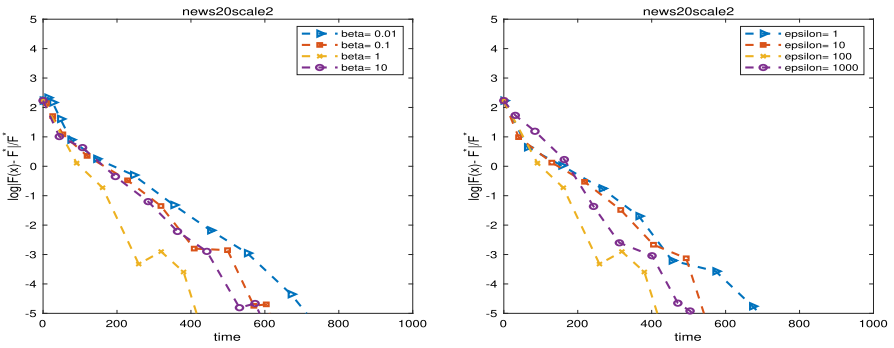
(a) $m = 100$ **(b)** $m = 500$

(c) $m = 1000$ **(d)** $m = 10000$

Fig. 7 Performance of IPALM-Katyusha on solving convex QCQP problem. The x axis is the running time and the y axis is $\log(E(x))$ where $E(x)$ is the error function defined by (90). The length of the x axis is the running time of CVX

where $F^\star$ is obtained from the solution returned by CVX. We plot in Fig. 7 the decrease of the error $E(x)$ versus the computational time of IPALM-Katyusha. The maximal x-axis length corresponds to the computational time of CVX, marked in red pentagram.

In veiw of Fig. 7, on these four instances, IPALM-Katyusha can solve up to accuracy 10^{-8} in time comparable with CVX running time. Moreover, the accuracy 10^{-6} can be reached in half of the running time of CVX, demonstrating the power of IPALM-Katyusha for obtaining solution of medium accuracy. We also observe that as the number of constraints increase to 10000, IPALM-Katyusha has better scalability than



(a) $\epsilon_0 = 100$ **(b)** $\beta_0 = 1$

Fig. 8 Comparison of different choices of β_0 and ϵ_0 of IPALM-APPROX on the least absolute deviation problem. The x-axis is time and y-axis is $\log((F(x) - F^\star)/F^\star)$ for both two figures. For **a** we fix $\epsilon_0 = 100$ and for **b** we fix $\beta_0 = 1$

the classical interior point method as the time for reaching 10^{-8} is less than the running time of CVX.

7.6 Further experiments

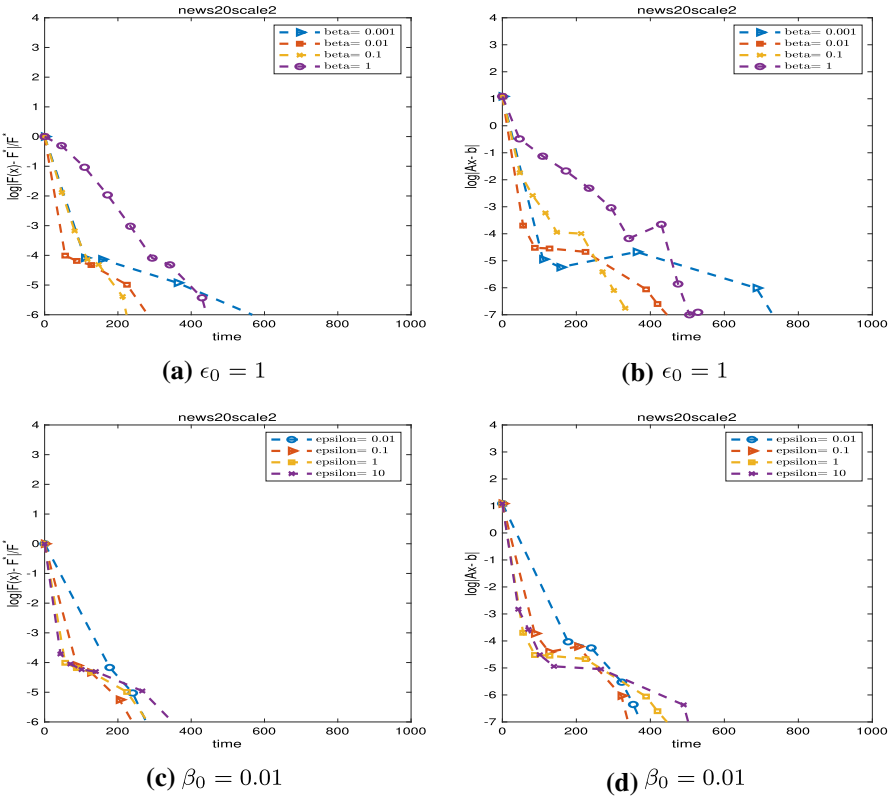
In this section, we report more experimental results to show how the choices of β_0 and ϵ_0 influence the performance of IPALM. In principle, if β_0 is small, it takes longer to solve the subproblems and more inner iterations will be needed. If β_0 is large, the gap between the original problem and the subproblem will be large and more outer iterations will be needed. Similarly, if ϵ_0 is large, the first few subproblems will not be solved with enough accuracy and more outer iterations will be needed. If instead ϵ_0 is small, then more number of inner iterations will be needed. Needless to say, the best choice of β_0 and ϵ_0 vary with difference problems and instances and a good determination rule is still missing. Here we report numerical results obtained from different choices of β_0 and ϵ_0.

From Fig. 8, it can be observed that when $\epsilon_0 = 100$, $\beta_0 = 1$ has better performance than $\beta_0 = 0.01, 0.1$ and 10. When $\beta_0 = 1$, $\epsilon_0 = 100$ has better performance than $\epsilon_0 = 1$, $\epsilon = 10$ and $\epsilon = 1000$. This matches the expectation that the best choice of ϵ_0 and β_0 should not be too large or too small. From Figs. 9, 10 and 11, we can observe similar results.

From the experiments, we observe that for a relatively wide range of choices, the influence of β_0 and ϵ_0 is moderate for the overall performance of our method.

8 Conclusion and future research

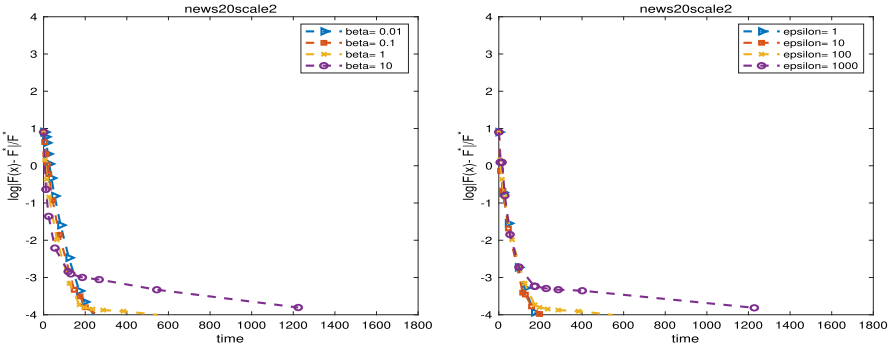
In this paper we consider a class of structured convex minimization problem and develop an inexact proximal augmented Lagrangian method with explicit inner termi-



(a) $\epsilon_0 = 1$ **(b)** $\epsilon_0 = 1$

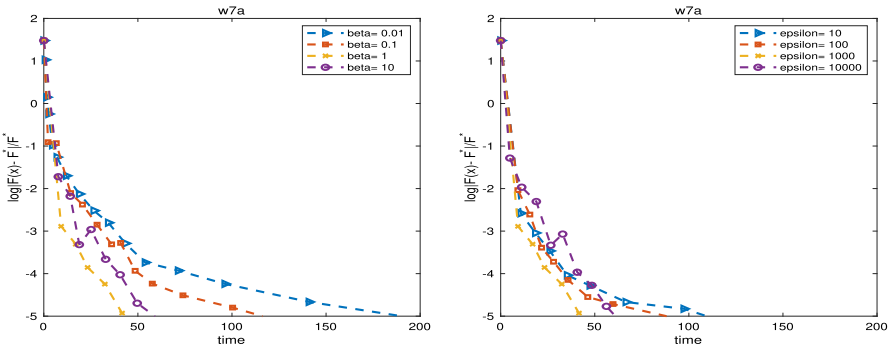
(c) $\beta_0 = 0.01$ **(d)** $\beta_0 = 0.01$

Fig. 9 Comparison of different choices of β_0 and ϵ_0 of IPALM-APPROX on the basis pursuit problem. The x-axis is time and y-axis is $\log((F(x) - F^\star)/F^\star)$ for **a**, **c** and $\log(|Ax - b|)$ for **b**, **d** . For **a**, **b** we fix $\epsilon_0 = 1$. For **c**, **d** we fix $\beta_0 = 0.01$



(a) $\epsilon_0 = 100$ **(b)** $\beta_0 = 1$

Fig. 10 Comparison of different choices of β_0 and ϵ_0 of IPALM-Katyusha on the fused Lasso problem. The x-axis is time and y-axis is $\log((F(x) - F^\star)/F^\star)$. For **a** we fix $\epsilon_0 = 100$. For **b** we fix $\beta_0 = 1$



(a) $\epsilon_0 = 1000$ **(b)** $\beta_0 = 1$

Fig. 11 Comparison of different choices of β_0 and ϵ_0 of IPALM-Katyusha on the svm problem. The x-axis is time and y-axis is $\log((F(x) - F^\star)/F^\star)$. For **a** we fix $\epsilon_0 = 1000$. For **b** we fix $\beta_0 = 1$

nation rule. Our framework allows arbitrary linearly convergent inner solver, including in particular many randomized first-order methods.

When $p(\cdot)$ is linear, under the same assumptions as [22,24,26,28,29,34] but without the boundedness of $\mathrm{dom}(g)$, we obtain nearly optimal $\tilde{O}(1/\epsilon)$ and $\tilde{O}(1/\sqrt{\epsilon})$ complexity bound respectively for the non-strongly convex and strongly convex case. The flexible inner solver choice allows us to deal with large-scale constrained problem more efficiently, with the aid of recent advances in randomized first-order methods for unconstrained problem. We provide numerical evidence showing the efficiency of our approach compared with existing ones when the problem dimension is high.

There are several interesting directions to exploit in the future.

1. The complexity bound established in this paper for non-strongly convex problem is $\tilde{O}(1/\epsilon)$. Throughout the paper we only rely on the fact that the sequence generated by PPA is bounded, whereas it is known that PPA can be linearly convergent if certain metric sub-regularity is satisfied, see e.g. [50]. We expect to obtain a linearly convergent rate under these conditions, see e.g. [23].
2. In numerical experiments, the choice of β_0 does influence the performance. Can a reasonable guess on β_0 be derived from the analysis?
3. When f is only relatively smooth (see Sect. 5.1.3), we only obtained $\tilde{O}(1/\epsilon^2)$ and $\tilde{O}(1/\epsilon)$ complexity bound for non-strongly convex and strongly convex case. Can we improve to $\tilde{O}(1/\epsilon)$ and $\tilde{O}(1/\sqrt{\epsilon})$?
4. Can this work be extended to saddle point problem? In particular, [20] discussed an inexact primal-dual method for nonbilinear saddle point problems with bounded $dom(g)$. Can we get rid of the boundedness assumption for saddle point problem?
5. Can this work be extended to weakly convex case as in [14,36]?

Acknowledgements We thank the three anonymous referees for their valuable comments for improving the paper.

A Proof of Lemma 1

Proof of Lemma 1 The first assertion follows from the proof of [31, Theorem 1]. See also [19, Theorem 2.1] and [5, Lemma 4.1]. The condition (20) is given by the first order optimality condition of (17). It implies

$$\Lambda(u;\lambda,\beta) \in \partial h\left(u - \beta(\Lambda(u;\lambda,\beta) - \lambda)\right). \tag{91}$$

The equality in (21) is a direct application of the Fenchel duality theorem [37]. See also [5, Equation 4.1 and 4.2]. The inequality in (21) follows by considering $w = 0$. The condition (22) follows from the first order optimality condition and (91). Finally (23) is obtained by plugging the optimal solution $w^\star$ in (22) into (21).

B Some useful lemmas

We first state two useful lemmas.

Lemma 6 *Let $\psi(\cdot) : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ be a convex function. Define:*

$$\tilde{\psi}(x) := \inf_w \{h(p(x) - w) + \psi(w)\},$$

Then condition (15) ensures the convexity of $\tilde{\psi}$.

Proof For any $x, y \in \mathbb{R}^n$ and $\alpha \in [0, 1]$, let $z = \alpha x + (1-\alpha)y$. By condition (15),

$$h\left(p(z) - \alpha u - (1-\alpha)v\right) \le \alpha h(p(x) - u) + (1-\alpha)h(p(y) - v), \quad \forall u, v \in \mathbb{R}^d.$$

It follows that

$$\begin{aligned}
\tilde{\psi}(z) &= \inf_{\omega} \{h(p(z) - \omega) + \psi(\omega)\} \\
&= \inf_{u,v} \{h\left(p(z) - \alpha u - (1-\alpha)v\right) + \psi(\alpha u + (1-\alpha)v)\} \\
&\le \inf_{u,v} \{\alpha h(p(x) - u) + (1-\alpha)h(p(y) - v) + \alpha\psi(u) + (1-\alpha)\psi(v)\} \\
&= \alpha \inf_{u} \{h(p(x) - u) + \psi(u)\} + (1-\alpha) \inf_{v} \{h(p(y) - v) + \psi(v)\} \\
&= \alpha\tilde{\psi}(x) + (1-\alpha)\tilde{\psi}(y).
\end{aligned}$$

□

Similarly, we can show the following result.

Lemma 7 *Let $\psi(\cdot) : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ be a convex function. Define:*

$$\tilde{\psi}(w) := \inf_x \{h(p(x) - w) + \psi(x)\},$$

Then condition (15) ensures the convexity of $\tilde{\psi}$.

C Inexact proximal point algorithm and inexact augmented Lagrangian method

C.1 Inexact proximal point method

Let $\mathcal{T} : \mathbb{R}^{n+d} \to \mathbb{R}^{n+d}$ be a maximal monotone operator and $\mathcal{J}_\rho = (\mathcal{I} + \rho\mathcal{T})^{-1}$ be the resolvent of $\mathcal{T}$, where $\mathcal{I}$ denotes the identity operator. Then for any $z^\star$ such that $0 \in \mathcal{T}(z^\star)$ [39],

$$\left\|\mathcal{J}_\rho(z) - z^\star\right\|^2 + \left\|\mathcal{J}_\rho(z) - z\right\|^2 \leq \left\|z - z^\star\right\|^2. \tag{92}$$

Algorithm 4 PPA

1: **Input:** z^0, $\{\varepsilon_s\}$, $\{\rho_s\}$.
2: **for** $k = 0, 1, \ldots$ **do**
3: Compute $z^{s+1} \approx \mathcal{J}_{\rho_s}(z^s)$ such that $\left\|z^{s+1} - \mathcal{J}_{\rho_s}(z^s)\right\| \leq \varepsilon_s$;
4: **end for**

Lemma 8 [39] *Let $\{z^s\}$ be the sequence generated by Algorithm 4. Then for any $z^\star$ such that $0 \in \mathcal{T}(z^\star)$,*

$$\left\|z^{s+1} - z^\star\right\| \leq \left\|z_0 - z^\star\right\| + \sum_{i=0}^{s} \varepsilon_i$$

$$\left\|z^{s+1} - z^s\right\| \leq \left\|z_0 - z^\star\right\| + \sum_{i=0}^{s} \varepsilon_i$$

We now give a stochastic generalization of Algorithm 4.

Algorithm 5 sPPA

1: **Input:** z^0, $\{\varepsilon_s\}$, $\{\rho_s\}$.
2: **for** $k = 0, 1, \ldots$ **do**
3: Compute $z^{s+1} \approx \mathcal{J}_{\rho_s}(z^s)$ such that $\mathbb{E}\left[\left\|z^{s+1} - \mathcal{J}_{\rho_s}(z^s)\right\|^2\right] \leq \varepsilon_s^2$;
4: **end for**

We then extend Lemma 8 for Algorithm 5.

Lemma 9 *Let $\{z^s\}$ be the sequence generated by Algorithm 5. Then for any $z^\star$ such that $0 \in \mathcal{T}(z^\star)$,*

$$\mathbb{E}\left[\left\|z^{s+1} - z^\star\right\|\right] \leq \left\|z_0 - z^\star\right\| + \sum_{i=0}^{s} \varepsilon_i$$

$$\mathbb{E}\left[\left\|z^{s+1}-z^{s}\right\|\right] \leq \left\|z_0-z^{\star}\right\| + \sum_{i=0}^{s} \varepsilon_i$$
$$\left(\mathbb{E}\left[\left\|z^{s+1}-z^{\star}\right\|^2\right]\right)^{1/2} \leq \left\|z_0-z^{\star}\right\| + \sum_{i=0}^{s} \varepsilon_i$$

Proof By (92), we know that for all $i \geq 0$

$$\begin{aligned}\left\|z^{i+1}-z^{\star}\right\| &\leq \left\|z^{i+1}-\mathcal{J}_{\rho_i}(z^i)\right\| + \left\|\mathcal{J}_{\rho_i}(z^i)-z^{\star}\right\| \\ &\leq \left\|z^{i+1}-\mathcal{J}_{\rho_i}(z^i)\right\| + \left\|z^{i}-z^{\star}\right\|.\end{aligned}$$

Taking expectation on both sides, we get

$$\mathbb{E}\left[\left\|z^{i+1}-z^{\star}\right\|\right] \leq \mathbb{E}\left[\left\|z^{i+1}-\mathcal{J}_{\rho_i}(z^i)\right\|\right] + \mathbb{E}\left[\left\|z^{i}-z^{\star}\right\|\right].$$

By the definition of z^s, we have $\left(\mathbb{E}\left\|z^{s+1}-\mathcal{J}_{\rho_s}(z^s)\right\|\right)^2 \leq \mathbb{E}\left\|z^{s+1}-\mathcal{J}_{\rho_s}(z^s)\right\|^2 \leq \varepsilon_s^2$ and therefore

$$\mathbb{E}\left[\left\|z^{i+1}-z^{\star}\right\|\right] \leq \epsilon_i + \mathbb{E}\left[\left\|z^{i}-z^{\star}\right\|\right].$$

The first estimates is derived by summing the above inequality from $i = 0$ to s.

By (92), we know that for all for all $s \geq 0$

$$\left\|z^{s+1}-z^{s}\right\| \leq \left\|z^{s+1}-\mathcal{J}_{\rho_s}(z^s)\right\| + \left\|\mathcal{J}_{\rho_s}(z^s)-z^{s}\right\| \leq \left\|z^{s+1}-\mathcal{J}_{\rho_s}(z^s)\right\| + \left\|z^{s}-z^{\star}\right\|.$$

Taking expectation on both sides,

$$\mathbb{E}\left[\left\|z^{s+1}-z^{s}\right\|\right] \leq \mathbb{E}\left[\left\|z^{s+1}-\mathcal{J}_{\rho_s}(z^s)\right\|\right] + \mathbb{E}\left[\left\|z^{s}-z^{\star}\right\|\right] \leq \epsilon_s + \mathbb{E}\left[\left\|z^{s}-z^{\star}\right\|\right].$$

Together with the first estimate, the second estimate is derived.

The third estimate is derived from (92):

$$\begin{aligned}0 &\leq \left\|\mathcal{J}_{\rho_s}-z^{s}\right\|^2 \leq \left\|z^{s}-z^{\star}\right\|^2 - \left\|\mathcal{J}_{\rho_s}(z^s)-z^{\star}\right\|^2 \\ &= \left\|z^{s}-z^{\star}\right\|^2 - \left\|\mathcal{J}_{\rho_s}(z^s)-z^{s+1}+z^{s+1}-z^{\star}\right\|^2 \\ &\leq \left\|z^{s}-z^{\star}\right\|^2 - \left\|z^{s+1}-z^{\star}\right\|^2 - \left\|\mathcal{J}_{\rho_s}(z^s)-z^{s+1}\right\|^2 \\ &\quad + 2\left\|\mathcal{J}_{\rho_s}(z^s)-z^{s+1}\right\|\left\|z^{s+1}-z^{\star}\right\|\end{aligned}$$

Taking expectation on both sides we have:

$$0 \leq \mathbb{E}\left[\left\|z^{s}-z^{\star}\right\|^2\right] - \mathbb{E}\left[\left\|z^{s+1}-z^{\star}\right\|^2\right]$$

$$
\begin{aligned}
&- \mathbb{E}\left[\left\|\mathcal{J}_{\rho_s}(z^s) - z^{s+1}\right\|^2\right] + 2\mathbb{E}\left[\left\|\mathcal{J}_{\rho_s}(z^s) - z^{s+1}\right\| \left\|z^{s+1} - z^\star\right\|\right] \\
&\le \mathbb{E}\left[\left\|z^s - z^\star\right\|^2\right] - \mathbb{E}\left[\left\|z^{s+1} - z^\star\right\|^2\right] \\
&\quad - \mathbb{E}\left[\left\|\mathcal{J}_{\rho_s}(z^s) - z^{s+1}\right\|^2\right] + 2\left(\mathbb{E}\left[\left\|\mathcal{J}_{\rho_s}(z^s) - z^{s+1}\right\|^2\right]\mathbb{E}\left[\left\|z^{s+1} - z^\star\right\|^2\right]\right)^{1/2} \\
&= \mathbb{E}\left[\left\|z^s - z^\star\right\|^2\right] - \left(\left(\mathbb{E}\left[\left\|z^{s+1} - z^\star\right\|^2\right]\right)^{1/2} - \left(\mathbb{E}\left[\left\|\mathcal{J}_{\rho_s}(z^s) - z^{s+1}\right\|^2\right]\right)^{1/2}\right)^2
\end{aligned}
$$

where the second inequality we use $\mathbb{E}[XY] \le (\mathbb{E}[X^2])^{1/2}(\mathbb{E}[Y^2])^{1/2}$. Therefore

$$
\begin{aligned}
\left(\mathbb{E}\left[\left\|z^{s+1} - z^\star\right\|^2\right]\right)^{1/2} - \varepsilon_s &\le \left(\mathbb{E}\left[\left\|z^{s+1} - z^\star\right\|^2\right]\right)^{1/2} - \left(\mathbb{E}\left[\left\|\mathcal{J}_{\rho_s}(z^s) - z^{s+1}\right\|^2\right]\right)^{1/2} \\
&\le \left(\mathbb{E}\left[\left\|z^s - z^\star\right\|^2\right]\right)^{1/2}
\end{aligned}
$$

Then summing up the latter inequalities from $s = 0$ we obtain the third inequality. □

C.2 Inexact ALM

We define the maximal monotone operator $\mathcal{T}_l$ as follows.

$$
\begin{aligned}
\mathcal{T}_l(x;\lambda) &= \{(v;u) : (v;-u) \in \partial L(x;\lambda)\} \\
&= \left\{\begin{pmatrix} \nabla f(x) + \partial g(x) + \nabla p(x)\lambda \\ -p(x) + \partial h^*(\lambda) \end{pmatrix}\right\}
\end{aligned}
$$

In the following we denote

$$
\begin{aligned}
&L^\star(y,\lambda,\beta) := \min_x L(x;y,\lambda,\beta), \\
&x^\star(y,\lambda,\beta) := \arg\min_x L(x;y,\lambda,\beta), \quad p^\star(y,\lambda,\beta) := p(x^\star(y,\lambda,\beta)).
\end{aligned} \tag{93}
$$

We further let $\Lambda^\star(y,\lambda,\beta) := \Lambda(p^\star(y,\lambda,\beta);\lambda,\beta)$. By first order optimality condition and (18), we know that

$$
\begin{aligned}
0 \in\ & \nabla f(x^\star(y,\lambda,\beta)) + \partial g(x^\star(y,\lambda,\beta)) \\
&+ \nabla p(x^\star(y,\lambda,\beta))\Lambda^\star(y,\lambda,\beta) + \beta(x^\star(y,\lambda,\beta) - y)
\end{aligned}
$$

Secondly we know from (20) that

$$
p^\star(y,\lambda,\beta) - \beta(\Lambda^\star(y,\lambda,\beta) - \lambda) \in \partial h^*(\Lambda^\star(y,\lambda,\beta)).
$$

It follows that

$$
(\mathcal{I} + \beta^{-1}\mathcal{T}_l)^{-1}(y;\lambda) = (x^\star(y,\lambda,\beta);\Lambda^\star(y,\lambda,\beta)) \tag{94}
$$

Lemma 10 *For any $x \in \mathbb{R}^n$ we have,*

$$L(x; y, \lambda, \beta) - L^\star(y, \lambda, \beta) \geq \frac{\beta}{2}\|x - x^\star(y, \lambda, \beta)\|^2 + \frac{\beta}{2}\|\Lambda(p(x); \lambda, \beta) - \Lambda(p^\star(y, \lambda, \beta); \lambda, \beta)\|^2. \tag{95}$$

Proof In this proof we fix $y \in \mathbb{R}^n$, $\lambda \in \mathbb{R}^d$ and $\beta > 0$. Recall the definitions in (93). Define

$$L(x, w; y, \lambda, \beta) := f(x) + g(x) + h(p(x) - w) + \frac{1}{2\beta}\|w\|^2 + \langle w, \lambda\rangle + \frac{\beta}{2}\|x - y\|^2 - \frac{\beta}{2}\|x - x^\star(y, \lambda, \beta)\|^2.$$

Then by (21),

$$\min_w L(x, w; y, \lambda, \beta) = L(x; y, \lambda, \beta) - \frac{\beta}{2}\|x - x^\star(y, \lambda, \beta)\|^2. \tag{96}$$

Since $L(x; y, \lambda, \beta) - \frac{\beta}{2}\|x - x^\star(y, \lambda, \beta)\|^2$ is a convex function with $x^\star(y, \lambda, \beta)$ being a critical point, it follows that

$$\min_x \min_w L(x, w; y, \lambda, \beta) = L^\star(y, \lambda, \beta). \tag{97}$$

Denote

$$H(w; y, \lambda, \beta) := \min_x L(x, w; y, \lambda, \beta). \tag{98}$$

In view of (22),

$$L(x; y, \lambda, \beta) - \frac{\beta}{2}\|x - x^\star(y, \lambda, \beta)\|^2 = L(x, \beta(\Lambda(p(x); \lambda, \beta) - \lambda); y, \lambda, \beta) \overset{(98)}{\geq} H(\beta(\Lambda(p(x); \lambda, \beta) - \lambda); y, \lambda, \beta). \tag{99}$$

Note that

$$\min_w H(w; y, \lambda, \beta) = \min_w \min_x L(x, w; y, \lambda, \beta) = \min_x \min_w L(x, w; y, \lambda, \beta) \overset{(97)}{=} L^\star(y, \lambda, \beta). \tag{100}$$

Denote $\Lambda^\star(y, \lambda, \beta) = \Lambda(p^\star(y, \lambda, \beta); \lambda, \beta)$. It follows that,

$$H(\beta(\Lambda^\star(y, \lambda, \beta) - \lambda); y, \lambda, \beta)$$

$$\geq \min_w H(w; y, \lambda, \beta) \overset{(100)}{=} L^\star(y, \lambda, \beta) = L(x^\star(y, \lambda, \beta); y, \lambda, \beta).$$

Using again (99) with $x = x^\star(y, \lambda, \beta)$ we deduce

$$H(\beta(\Lambda^\star(y, \lambda, \beta) - \lambda); y, \lambda, \beta) = \min_w H(w; y, \lambda, \beta). \tag{101}$$

Moreover, it follows from Lemma 7 that $H(w; y, \lambda, \beta)$ is $1/\beta$-strongly convex with respect to w. Thus,

$$\begin{aligned} &L(x; y, \lambda, \beta) - L^\star(y, \lambda, \beta) - \frac{\beta}{2}\|x - x^\star(y, \lambda, \beta)\|^2 \\ &\overset{(99)+(100)}{\geq} H(\beta(\Lambda(p(x); \lambda, \beta) - \lambda); y, \lambda, \beta) - \min_w H(w; y, \lambda, \beta) \\ &\overset{(101)}{\geq} \frac{1}{2\beta}\|\beta(\Lambda(p(x); \lambda, \beta) - \lambda) - \beta(\Lambda^\star(y, \lambda, \beta) - \lambda)\|^2 \\ &= \frac{\beta}{2}\|\Lambda(p(x); \lambda, \beta) - \Lambda^\star(y, \lambda, \beta)\|^2. \end{aligned}$$

□

We can then establish the following well known link between inexact ALM and inexact PPA.

Proposition 3 (Compare with [39]) *Algorithm 1 is a special case of Algorithm 5 with* $\mathcal{T} = \mathcal{T}_l$, $\rho_s = 1/\beta_s$ *and* $\varepsilon_s = \sqrt{2\epsilon_s/\beta_s}$.

Proof This follows from (94) and Lemma 10. □

D Missing proofs

D.1 Proofs in Section 2.2

Proof of Lemma 2 The convexity of $\tilde{\psi}$ follows from (21) and Lemma 6 with $\psi(w) := \frac{1}{2\beta}\|w\|^2 + \langle w, \lambda\rangle$. The gradient formula follows from (18).

Proof of Lemma 3 This is a direct consequence of Proposition 3 and Lemma 9.

Proof of Corollary 1 By Lemma 3, we have

$$\mathbb{E}\left[\left\|(x^s, \lambda^{s+1}) - (x^{s-1}, \lambda^s)\right\|\right] \leq \left\|(x^{-1}, \lambda^0) - (x^\star, \lambda^\star)\right\| + \frac{2\sqrt{\epsilon_0/\beta_0}}{1 - \sqrt{\eta/\rho}}, \quad \forall s \geq 0,$$

and

$$\mathbb{E}\left[\left\|(x^s, \lambda^{s+1}) - (x^\star, \lambda^\star)\right\|^2\right] \leq \left(\left\|(x^{-1}, \lambda^0) - (x^\star, \lambda^\star)\right\| + \frac{2\sqrt{\epsilon_0/\beta_0}}{1 - \sqrt{\eta/\rho}}\right)^2, \quad \forall s \geq 0.$$

Consequently,

$$\mathbb{E}\left[\left\|\lambda^{s+1}-\lambda^{s}\right\|\right] \leq \left\|(x^{-1},\lambda^{0})-(x^{\star},\lambda^{\star})\right\| + \frac{2\sqrt{\epsilon_0/\beta_0}}{1-\sqrt{\eta/\rho}},\ \ \forall s\geq 0,$$

and

$$\begin{aligned}&\max\left(\mathbb{E}\left[\left\|x^{s}-x^{\star}\right\|^{2}\right],\mathbb{E}\left[\left\|\lambda^{s+1}-\lambda^{\star}\right\|^{2}\right]\right)\\ &\leq\left(\left\|(x^{-1},\lambda^{0})-(x^{\star},\lambda^{\star})\right\|+\frac{2\sqrt{\epsilon_0/\beta_0}}{1-\sqrt{\eta/\rho}}\right)^{2},\ \ \forall s\geq 0.\end{aligned}$$

We then conclude.

Proof of Theorem 1 First,

$$\begin{aligned}h_1(p_1(x^s))-h(p(x^s);\lambda^s,\beta_s) &\overset{(23)}{=} h_1(p_1(x^s))-h_1(p_1(x^s)-\beta_s(\lambda_1^{s+1}-\lambda_1^s))\\ &\quad -\tfrac{\beta_s}{2}(\|\lambda^{s+1}\|^2-\|\lambda^s\|^2)\\ &\leq L_{h_1}\beta_s\|\lambda_1^{s+1}-\lambda_1^s\|+\tfrac{\beta_s}{2}(\|\lambda^s\|^2-\|\lambda^{s+1}\|^2).\end{aligned}\tag{102}$$

Then we know that

$$\begin{aligned}F(x^s)-L(x^s;x^{s-1},\lambda^s,\beta_s) &= h_1(p_1(x^s))-h(p(x^s);\lambda^s,\beta_s)-\frac{\beta_s}{2}\|x^s-x^{s-1}\|^2\\ &\overset{(102)}{\leq} L_{h_1}\beta_s\|\lambda_1^{s+1}-\lambda_1^s\|+\frac{\beta_s}{2}(\|\lambda^s\|^2-\|\lambda^{s+1}\|^2)\\ &\quad -\frac{\beta_s}{2}\|x^s-x^{s-1}\|^2.\end{aligned}$$

Since $H_s(\cdot)$ is β_s-strongly convex, we know that

$$\begin{aligned}L^{\star}(x^{s-1},\lambda^s,\beta_s) &\leq L(x^{\star};x^{s-1},\lambda^s,\beta^s)-\frac{\beta_s}{2}\|x^{\star}-x^{\star}(x^{s-1},\lambda^s,\beta_s)\|^2\\ &\overset{(21)}{\leq} F^{\star}+\frac{\beta_s}{2}\|x^{\star}-x^{s-1}\|^2-\frac{\beta_s}{2}\|x^{\star}-x^{\star}(x^{s-1},\lambda^s,\beta_s)\|^2.\end{aligned}$$

Combining the latter two bounds we get

$$\begin{aligned}F(x^s)-F^{\star} &\leq L(x^s;x^{s-1},\lambda^s,\beta_s)-L^{\star}(x^{s-1},\lambda^s,\beta_s)+L_{h_1}\beta_s\|\lambda_1^{s+1}-\lambda_1^s\|\\ &\quad+\frac{\beta_s}{2}(\|\lambda^s\|^2-\|\lambda^{s+1}\|^2)+\frac{\beta_s}{2}\|x^{\star}-x^{s-1}\|^2\\ &\quad-\frac{\beta_s}{2}\|x^{\star}-x^{\star}(x^{s-1},\lambda^s,\beta_s)\|^2-\frac{\beta_s}{2}\|x^s-x^{s-1}\|^2.\end{aligned}$$

Furthermore, by convexity of $h_1(\cdot)$,

$$\inf_x F(x)+\langle\lambda_2^{\star},p_2(x)\rangle-h_2^*(\lambda_2^{\star})\geq\inf_x f(x)+g(x)+\langle\lambda^{\star},p(x)\rangle-h^*(\lambda^{\star})=D(\lambda^{\star}).$$

Now we apply the strong duality assumption (11) to obtain:

$$F(x^s) + \langle \lambda_2^\star, p_2(x^s)\rangle - h_2^*(\lambda_2^\star) \geq \inf_x F(x) + \langle \lambda_2^\star, p_2(x)\rangle - h_2^*(\lambda_2^\star) \geq F^\star.$$

Consequently,

$$\begin{aligned} F(x^s) - F^\star &\geq \langle \lambda_2^\star, -p_2(x^s)\rangle + h_2^*(\lambda_2^\star) \\ &\geq \sup_v \langle \lambda_2^\star, v - p_2(x^s)\rangle - h_2(v) \geq -\|\lambda_2^\star\| \operatorname{dist}(p_2(x^s), \mathcal{K}). \end{aligned}$$

From (20) we know

$$p_2(x^s) - \beta_s(\lambda_2^{s+1} - \lambda_2^s) \in \mathcal{K},$$

and thus

$$\operatorname{dist}(p_2(x^s), \mathcal{K}) \leq \beta_s \|\lambda_2^{s+1} - \lambda_2^s\|.$$

Proof of Corollary 2 Taking expectation on both sides of the bounds in Theorem 1 we have:

$$\begin{aligned} \mathbb{E}\left[F(x^s) - F^\star\right] &\leq \epsilon_s + L_{h_1}\beta_s \left(\mathbb{E}\left\|\lambda_1^{s+1}\right\| + \mathbb{E}\left\|\lambda_1^s\right\|\right) \\ &\quad + \frac{\beta_s}{2}\mathbb{E}\left[\left\|\lambda^s\right\|^2\right] + \frac{\beta_s}{2}\mathbb{E}\left[\left\|x^s - x^{s-1}\right\|^2\right], \\ \mathbb{E}\left[F(x^s) - F^\star\right] &\geq -\beta_s\|\lambda_2^\star\|\mathbb{E}\left[\left\|\lambda^{s+1} - \lambda^s\right\|\right], \\ \mathbb{E}[\operatorname{dist}(p_2(x^s), K)] &\leq \beta_s \mathbb{E}\left[\left\|\lambda^{s+1} - \lambda^s\right\|\right]. \end{aligned}$$

By condition (a) in Assumption 1, we have for all $s \geq 0$, $\lambda_1^s \in \operatorname{dom}(h_1^*)$ and $\left\|\lambda_1^s\right\| \leq L_{h_1}$ due to [9, Proposition 4.4.6]. Then using Corollary 1, the above bounds can be relaxed as:

$$\begin{aligned} \mathbb{E}[F(x^s) - F^\star] &\leq \epsilon_s + 2L_{h_1}^2\beta_s + c_0\beta_s \\ \mathbb{E}[F(x^s) - F^\star] &\geq -\beta_s\|\lambda_2^\star\|\sqrt{c_0}, \\ \mathbb{E}[\operatorname{dist}(p_2(x^s), K)] &\leq \beta_s\sqrt{c_0}. \end{aligned}$$

We then conclude by noting that (32) guarantees

$$\max(\epsilon_0 + 2L_{h_1}^2\beta_0 + c_0\beta_0, \beta_0\|\lambda_2^\star\|\sqrt{c_0}, \beta_0\sqrt{c_0}) \leq \epsilon\rho^{-s}. \tag{103}$$

D.2 Proof of Proposition 1

This section is devoted to the proof of Proposition 1.

Lemma 11 *For any $x \in \mathbb{R}^n$, $\lambda, \lambda' \in \mathbb{R}^d$ and $\beta, \beta' \in \mathbb{R}_+$ we have,*

$$\begin{aligned} &L(x; y, \lambda, \beta) - L(x; y', \lambda', \beta') + \frac{\beta}{2}\|\Lambda(p(x); \lambda, \beta) - \lambda\|^2 \\ &\quad - \frac{\beta'}{2}\|\Lambda(p(x); \lambda', \beta') - \lambda'\|^2 \\ &\leq \langle \Lambda(p(x); \lambda, \beta) - \Lambda(p(x); \lambda', \beta'), \beta'(\Lambda(p(x); \lambda', \beta') - \lambda')\rangle \\ &\quad + \frac{\beta}{2}\|x - y\|^2 - \frac{\beta'}{2}\|x - y'\|^2, \end{aligned} \tag{104}$$

and

$$\begin{aligned} &L(x; y, \lambda, \beta) - L(x; y', \lambda', \beta') + \frac{\beta}{2}\|\Lambda(p(x); \lambda, \beta) - \lambda\|^2 \\ &\quad - \frac{\beta'}{2}\|\Lambda(p(x); \lambda', \beta') - \lambda'\|^2 \\ &\geq \langle \Lambda(p(x); \lambda, \beta) - \Lambda(p(x); \lambda', \beta'), \beta(\Lambda(p(x); \lambda, \beta) - \lambda)\rangle \\ &\quad + \frac{\beta}{2}\|x - y\|^2 - \frac{\beta'}{2}\|x - y'\|^2. \end{aligned} \tag{105}$$

Proof By the definitions (24), (16) and (17), we have

$$\begin{aligned} &L(x; y, \lambda, \beta) - L(x; y', \lambda', \beta') + \frac{\beta}{2}\|\Lambda(p(x); \lambda, \beta) - \lambda\|^2 - \frac{\beta'}{2}\|\Lambda(p(x); \lambda', \beta') - \lambda'\|^2 \\ &= \langle \Lambda(p(x); \lambda, \beta) - \Lambda(p(x); \lambda', \beta'), p(x)\rangle - h^*(\Lambda(p(x); \lambda, \beta)) \\ &\quad + h^*(\Lambda(p(x); \lambda', \beta')) + \frac{\beta}{2}\|x - y\|^2 - \frac{\beta'}{2}\|x - y'\|^2. \end{aligned}$$

Next we apply (20) to get

$$\begin{aligned} h^*(\Lambda(p(x); \lambda', \beta')) \geq\ & h^*(\Lambda(p(x); \lambda, \beta)) \\ &+ \langle \Lambda(p(x); \lambda, \beta) - \Lambda(p(x); \lambda', \beta'), \beta(\Lambda(p(x); \lambda, \beta) - \lambda) - p(x)\rangle, \end{aligned}$$

and

$$\begin{aligned} h^*(\Lambda(p(x); \lambda, \beta)) \geq\ & h^*(\Lambda(p(x); \lambda', \beta')) \\ &+ \langle \Lambda(p(x); \lambda, \beta) \\ &- \Lambda(p(x); \lambda', \beta'), p(x) - \beta'(\Lambda(p(x); \lambda', \beta') - \lambda')\rangle. \end{aligned}$$

□

Lemma 12 *Consider any* $u, \lambda, \lambda' \in \mathbb{R}^d$ *and* $\beta, \beta' \in \mathbb{R}_+$. *Condition (a) in Assumption 1 ensures:*

$$\begin{aligned}&\|\beta(\Lambda(u;\lambda,\beta)-\lambda)-\beta'(\Lambda(u;\lambda',\beta')-\lambda')\|\\&\quad\le\sqrt{((\beta+\beta')L_{h_1}+\|\beta\lambda_1-\beta'\lambda_1'\|)^2+\|\beta\lambda_2-\beta'\lambda_2'\|^2}.\end{aligned}\tag{106}$$

Proof Denote

$$\Lambda_i(u_i;\lambda_i,\beta):=\arg\max_{\xi_i}\left\{\langle\xi_i,u_i\rangle-h_i^*(\xi_i)-\frac{\beta}{2}\|\xi_i-\lambda_i\|^2\right\},\ \ i=1,2,\tag{107}$$

so that $\Lambda(u;\lambda,\beta)=(\Lambda_1(u_1;\lambda_1,\beta);\Lambda_2(u_2;\lambda_2,\beta))$. We can then decompose (20) into two independent conditions:

$$\Lambda_i(u_i;\lambda_i,\beta)\in\partial h_i(u_i-\beta(\Lambda_i(u_i;\lambda_i,\beta)-\lambda_i)),\ \ i=1,2.\tag{108}$$

By condition (a) in Assumption 1,

$$\|\Lambda_1(u_1;\lambda_1,\beta)\|\le L_{h_1}\tag{109}$$

which yields directly

$$\|\beta(\Lambda_1(u_1;\lambda_1,\beta)-\lambda_1)-\beta'(\Lambda_1(u_1;\lambda_1',\beta')-\lambda_1')\|\le(\beta+\beta')L_{h_1}+\|\beta\lambda_1-\beta'\lambda_1'\|.\tag{110}$$

On the other hand, since h_2 is an indicator function, ∂h_2 is a cone and (108) implies

$$\beta\Lambda_2(u_2;\lambda_2,\beta)\in\partial h_2(u_2-\beta(\Lambda_2(u_2;\lambda_2,\beta)-\lambda_2)).\tag{111}$$

The latter condition further leads to

$$\begin{aligned}&\langle\beta\Lambda_2(u_2;\lambda_2,\beta)-\beta'\Lambda_2(u_2;\lambda_2',\beta'),\beta(\Lambda_2(u_2;\lambda_2,\beta)-\lambda_2)\\&\quad-\beta'(\Lambda_2(u_2;\lambda_2',\beta')-\lambda_2')\rangle\le 0,\end{aligned}$$

which by Cauchy-Schwartz inequality implies

$$\|\beta(\Lambda_2(u_2;\lambda_2,\beta)-\lambda_2)-\beta'(\Lambda_2(u_2;\lambda_2',\beta')-\lambda_2')\|\le\|\beta\lambda_2-\beta'\lambda_2'\|.$$

Then (106) is obtained by simple algebra. □

Remark 9 If

$$h(u)=\begin{cases}0 & \text{if } u=b\\ +\infty & \text{otherwise}\end{cases}$$

for some constant vector $b \in \mathbb{R}^d$, then by (20) we have

$$u - \beta(\Lambda(u; \lambda, \beta) - \lambda) = b,$$

for any $u, \lambda \in \mathbb{R}^d$ and $\beta \geq 0$. In this special case a refinement of Lemma 12 can be stated as follows:

$$\|\beta(\Lambda(u; \lambda, \beta) - \lambda) - \beta'(\Lambda(u; \lambda', \beta') - \lambda')\| = 0.$$

Lemma 13 *Consider any $0 < \beta/2 < \beta'$ and any $w, w', y, y' \in \mathbb{R}^n$. We have*

$$\begin{aligned} &-\frac{\beta}{2}\|w' - w\|^2 + \frac{\beta}{2}\|w' - y\|^2 - \frac{\beta'}{2}\|w' - y'\|^2 \\ &\quad \leq \frac{\beta}{2}\|w - y'\|^2 + \frac{\beta(2\beta' + \beta)}{2(2\beta' - \beta)}\|y - y'\|^2. \end{aligned} \tag{112}$$

Proof We first recall the following basic inequality:

$$\|u + v\|^2 \leq (1 + a)\|u\|^2 + (1 + 1/a)\|v\|^2, \ \ \forall u, v \in \mathbb{R}^n, a > 0. \tag{113}$$

In view of (113) and the fact that $\beta' > \beta/2$, we know that

$$\begin{aligned} &-\frac{\beta}{2}\|w' - w\|^2 \leq \frac{\beta}{2}\|w - y'\|^2 - \frac{\beta}{4}\|w' - y'\|^2, \\ &-\frac{\beta' + \beta/2}{2}\|w' - y'\|^2 \leq \frac{\beta(2\beta' + \beta)}{2(2\beta' - \beta)}\|y - y'\|^2 - \frac{\beta}{2}\|w' - y\|^2. \end{aligned}$$

Combining the latter two inequalities we get (112). □

Using the above four lemmas, we establish a relation between $L(x; y', \lambda', \beta') - L^\star(y', \lambda', \beta')$ and $L(x; y, \lambda, \beta) - L^\star(y, \lambda, \beta)$.

Proposition 4 *For any $x, y, y' \in \mathbb{R}^n$, $\lambda, \lambda' \in \mathbb{R}^d$ and $0 < \beta/2 < \beta'$, we have*

$$\begin{aligned} &L(x; y', \lambda', \beta') - L^\star(y', \lambda', \beta') - \big(L(x; y, \lambda, \beta) - L^\star(y, \lambda, \beta)\big) \\ &\leq \|\lambda - \lambda'\|\sqrt{((\beta + \beta')L_{h_1} + \|\beta\lambda_1 - \beta'\lambda_1'\|)^2 + \|\beta\lambda_2 - \beta'\lambda_2'\|^2} \\ &\quad + \beta\|\lambda - \lambda'\|^2 + \frac{\beta - \beta'}{2}\|\Lambda(p(x); \lambda', \beta') - \lambda'\|^2 \\ &\quad + \frac{\beta' - \beta}{2}\|\Lambda(p^\star(y', \lambda', \beta'); \lambda, \beta) - \lambda\|^2 \\ &\quad + \frac{\beta}{2}\|\Lambda(p^\star(y, \lambda, \beta); \lambda, \beta) - \Lambda(p(x); \lambda, \beta)\|^2 \\ &\quad + \frac{\beta}{2}\|x^\star(y, \lambda, \beta) - y'\|^2 + \frac{\beta(2\beta' + \beta)}{2(2\beta' - \beta)}\|y - y'\|^2 \end{aligned}$$

$$-\frac{\beta}{2}\|x-y\|^2+\frac{\beta'}{2}\|x-y'\|^2. \tag{114}$$

Proof We first separate $L(x;y',\lambda',\beta')-L^\star(y',\lambda',\beta')$ into four parts:

$$\begin{aligned}
&L(x;y',\lambda',\beta')-L^\star(y',\lambda',\beta')\\
&\quad=\underbrace{L(x;y,\lambda,\beta)-L^\star(y,\lambda,\beta)}_{\Delta_1}+\underbrace{L(x;y',\lambda',\beta')-L(x;y,\lambda,\beta)}_{\Delta_2}\\
&\qquad+\underbrace{L(x^\star(y',\lambda',\beta');y,\lambda,\beta)-L^\star(y',\lambda',\beta')}_{\Delta_3}\\
&\quad+\underbrace{L^\star(y,\lambda,\beta)-L(x^\star(y',\lambda',\beta');y,\lambda,\beta)}_{\Delta_4}.
\end{aligned}$$

By Lemma 11,

$$\begin{aligned}
\Delta_2\le{}&\frac{\beta}{2}\|\Lambda(p(x);\lambda,\beta)-\lambda\|^2-\frac{\beta'}{2}\|\Lambda(p(x);\lambda',\beta')-\lambda'\|^2-\frac{\beta}{2}\|x-y\|^2\\
&+\frac{\beta'}{2}\|x-y'\|^2+\langle\Lambda(p(x);\lambda',\beta')-\Lambda(p(x);\lambda,\beta),\beta(\Lambda(p(x);\lambda,\beta)-\lambda)\rangle,
\end{aligned}$$

and

$$\begin{aligned}
\Delta_3\le{}&\frac{\beta'}{2}\|\Lambda(p^\star(y',\lambda',\beta');\lambda',\beta')-\lambda'\|^2-\frac{\beta}{2}\|\Lambda(p^\star(y',\lambda',\beta');\lambda,\beta)-\lambda\|^2\\
&+\langle\Lambda(p^\star(y',\lambda',\beta');\lambda,\beta)-\Lambda(p^\star(y',\lambda',\beta');\lambda',\beta'),\\
&\beta'(\Lambda(p^\star(y',\lambda',\beta');\lambda',\beta')-\lambda')\rangle\\
&+\frac{\beta}{2}\|x^\star(y',\lambda',\beta')-y\|^2-\frac{\beta'}{2}\|x^\star(y',\lambda',\beta')-y'\|^2.
\end{aligned}$$

We then get

$$\begin{aligned}
\Delta_2+\Delta_3\le{}&-\frac{\beta}{2}\|\Lambda(p(x);\lambda,\beta)-\lambda\|^2-\frac{\beta'}{2}\|\Lambda(p(x);\lambda',\beta')-\lambda'\|^2\\
&+\langle\Lambda(p(x);\lambda',\beta')-\lambda',\beta(\Lambda(p(x);\lambda,\beta)-\lambda)\rangle\\
&-\frac{\beta'}{2}\|\Lambda(p^\star(y',\lambda',\beta');\lambda',\beta')-\lambda'\|^2\\
&-\frac{\beta}{2}\|\Lambda(p^\star(y',\lambda',\beta');\lambda,\beta)-\lambda\|^2\\
&+\langle\Lambda(p^\star(y',\lambda',\beta');\lambda,\beta)-\lambda,\beta'(\Lambda(p^\star(y',\lambda',\beta');\lambda',\beta')-\lambda')\rangle\\
&+\langle\lambda-\lambda',\beta'(\Lambda(p^\star(y',\lambda',\beta');\lambda',\beta')-\lambda')-\beta(\Lambda(p(x);\lambda,\beta)-\lambda)\rangle\\
&-\frac{\beta}{2}\|x-y\|^2+\frac{\beta'}{2}\|x-y'\|^2+\frac{\beta}{2}\|x^\star(y',\lambda',\beta')-y\|^2\\
&-\frac{\beta'}{2}\|x^\star(y',\lambda',\beta')-y'\|^2\\
\le{}&\frac{\beta-\beta'}{2}\|\Lambda(p(x);\lambda',\beta')-\lambda'\|^2+\frac{\beta'-\beta}{2}\|\Lambda(p^\star(y',\lambda',\beta');\lambda,\beta)-\lambda\|^2
\end{aligned}$$

$$
\begin{aligned}
&+\langle \lambda-\lambda', \beta'(\Lambda(p^\star(y',\lambda',\beta');\lambda',\beta')-\lambda')-\beta(\Lambda(p(x);\lambda,\beta)-\lambda)\rangle\\
&-\frac{\beta}{2}\|x-y\|^2+\frac{\beta'}{2}\|x-y'\|^2+\frac{\beta}{2}\|x^\star(y',\lambda',\beta')-y\|^2\\
&-\frac{\beta'}{2}\|x^\star(y',\lambda',\beta')-y'\|^2,
\end{aligned}
$$

where the last inequality simply relies on $2\langle x, y\rangle \le \|x\|^2+\|y\|^2$. Further, according to Lemma 10,

$$
\begin{aligned}
\Delta_4 \le &-\frac{\beta}{2}\|\Lambda(p^\star(y',\lambda',\beta');\lambda,\beta)-\Lambda(p^\star(y,\lambda,\beta);\lambda,\beta)\|^2\\
&-\frac{\beta}{2}\|x^\star(y',\lambda',\beta')-x^\star(y,\lambda,\beta)\|^2.
\end{aligned}
$$

Therefore,

$$
\begin{aligned}
&\Delta_2+\Delta_3+\Delta_4-\frac{\beta-\beta'}{2}\|\Lambda(p(x);\lambda',\beta')-\lambda'\|^2-\frac{\beta'-\beta}{2}\|\Lambda(p^\star(y',\lambda',\beta');\lambda,\beta)-\lambda\|^2\\
&\le \langle \lambda-\lambda', \beta'(\Lambda(p^\star(y',\lambda',\beta');\lambda',\beta')-\lambda')-\beta(\Lambda(p(x);\lambda,\beta)-\lambda)\rangle\\
&\quad-\frac{\beta}{2}\|\Lambda(p^\star(y',\lambda',\beta');\lambda,\beta)-\Lambda(p^\star(y,\lambda,\beta);\lambda,\beta)\|^2\\
&\quad-\frac{\beta}{2}\|x^\star(y',\lambda',\beta')-x^\star(y,\lambda,\beta)\|^2-\frac{\beta}{2}\|x-y\|^2+\frac{\beta'}{2}\|x-y'\|^2\\
&\quad+\frac{\beta}{2}\|x^\star(y',\lambda',\beta')-y\|^2-\frac{\beta'}{2}\|x^\star(y',\lambda',\beta')-y'\|^2\\
&=\langle \lambda-\lambda', \beta'(\Lambda(p^\star(y',\lambda',\beta');\lambda',\beta')-\lambda')-\beta(\Lambda(p^\star(y',\lambda',\beta');\lambda,\beta)-\lambda)\rangle\\
&\quad+\beta\langle \lambda-\lambda', \Lambda(p^\star(y,\lambda,\beta);\lambda,\beta)-\Lambda(p(x);\lambda,\beta)\rangle\\
&\quad+\beta\langle \lambda-\lambda', \Lambda(p^\star(y',\lambda',\beta');\lambda,\beta)-\Lambda(p^\star(y,\lambda,\beta);\lambda,\beta)\rangle\\
&\quad-\frac{\beta}{2}\|\Lambda(p^\star(y',\lambda',\beta');\lambda,\beta)-\Lambda(p^\star(y,\lambda,\beta);\lambda,\beta)\|^2\\
&\quad-\frac{\beta}{2}\|x^\star(y',\lambda',\beta')-x^\star(y,\lambda,\beta)\|^2-\frac{\beta}{2}\|x-y\|^2+\frac{\beta'}{2}\|x-y'\|^2\\
&\quad+\frac{\beta}{2}\|x^\star(y',\lambda',\beta')-y\|^2-\frac{\beta'}{2}\|x^\star(y',\lambda',\beta')-y'\|^2\\
&\le \|\lambda-\lambda'\|\sqrt{((\beta+\beta')L_{h_1}+\|\beta\lambda_1-\beta'\lambda_1'\|)^2+\|\beta\lambda_2-\beta'\lambda_2'\|^2}+\beta\|\lambda-\lambda'\|^2\\
&\quad+\frac{\beta}{2}\|\Lambda(p^\star(y,\lambda,\beta);\lambda,\beta)-\Lambda(p(x);\lambda,\beta)\|^2-\frac{\beta}{2}\|x^\star(y',\lambda',\beta')-x^\star(y,\lambda,\beta)\|^2\\
&\quad-\frac{\beta}{2}\|x-y\|^2+\frac{\beta'}{2}\|x-y'\|^2+\frac{\beta}{2}\|x^\star(y',\lambda',\beta')-y\|^2-\frac{\beta'}{2}\|x^\star(y',\lambda',\beta')-y'\|^2,
\end{aligned}
\tag{115}
$$

where the last inequality follows from Lemma 12 and Cauchy Schwartz inequality. Now we apply Lemma 13 with $w = x^\star(y,\lambda,\beta)$ and $w' = x^\star(y',\lambda',\beta')$ to obtain:

$$
-\frac{\beta}{2}\|x^\star(y',\lambda',\beta')-x^\star(y,\lambda,\beta)\|^2+\frac{\beta}{2}\|x^\star(y',\lambda',\beta')
$$

$$
\begin{aligned}
&- y\|^2 - \frac{\beta'}{2}\|x^\star(y', \lambda', \beta') - y'\|^2 \\
&\leq \frac{\beta}{2}\|x^\star(y, \lambda, \beta) - y'\|^2 + \frac{\beta(2\beta' + \beta)}{2(2\beta' - \beta)}\|y - y'\|^2. \qquad (116)
\end{aligned}
$$

Plugging (116) into (115) with we derive (114). □

Now we are ready to give a proof for Proposition 1.

Proof of Proposition 1 We apply Proposition 4 with $\lambda = \lambda^s$, $\lambda' = \lambda^{s+1}$, $\beta = \beta_s$, $\beta' = \beta_{s+1}$, $x = x^s$, $y = x^{s-1}$ and $y' = x^s$ to obtain

$$
\begin{aligned}
&H_{s+1}(x^s) - H^\star_{s+1} - \left(H_s(x^s) - H^\star_s\right) \\
&\leq \|\lambda^s - \lambda^{s+1}\|\sqrt{\left((\beta_s + \beta_{s+1})L_{h_1} + \|\beta_s\lambda_1^s - \beta_{s+1}\lambda_1^{s+1}\|\right)^2 + \|\beta_s\lambda_2^s - \beta_{s+1}\lambda_2^{s+1}\|^2} \\
&\quad + \beta_s\|\lambda^s - \lambda^{s+1}\|^2 + \frac{\beta_s - \beta_{s+1}}{2}\|\Lambda(p(x^s); \lambda^{s+1}, \beta^{s+1}) - \lambda^{s+1}\|^2 \\
&\quad + \frac{\beta_{s+1} - \beta_s}{2}\|\Lambda(p^\star(x^s, \lambda^{s+1}, \beta_{s+1}); \lambda^s, \beta_s) - \lambda^s\|^2 \\
&\quad + \frac{\beta_s}{2}\|\Lambda(p^\star(x^{s-1}, \lambda^s, \beta_s); \lambda^s, \beta_s) - \Lambda(p(x^s); \lambda^s, \beta_s)\|^2 \\
&\quad + \frac{\beta_s}{2}\|x^\star(x^{s-1}, \lambda^s, \beta_s) - x^s\|^2 + \frac{\beta_s(2\beta_{s+1} + \beta_s)}{2(2\beta_{s+1} - \beta_s)}\|x^{s-1} - x^s\|^2 - \frac{\beta_s}{2}\|x^s - x^{s-1}\|^2.
\end{aligned}
$$

We apply Lemma 10 with $x = x^s$, $y = x^{s-1}$, $\lambda = \lambda^s$ and $\beta = \beta_s$ and get:

$$
\begin{aligned}
&\frac{\beta_s}{2}\|\Lambda(p^\star(x^{s-1}, \lambda^s, \beta_s); \lambda^s, \beta_s) - \Lambda(p(x^s); \lambda^s, \beta_s)\|^2 + \frac{\beta_s}{2}\|x^\star(x^{s-1}, \lambda^s, \beta_s) - x^s\|^2 \\
&\quad \leq H_s(x^s) - H^\star_s.
\end{aligned}
$$

Furthermore, since $\beta_{s+1} \leq \beta_s$ we have,

$$
\frac{\beta_{s+1} - \beta_s}{2}\|\Lambda(p^\star(x^s, \lambda^{s+1}, \beta_s); \lambda^s, \beta_s) - \lambda^s\|^2 \leq 0.
$$

We then derive (34) by the latter three bounds.

Remark 10 If

$$
h(u) = \begin{cases} 0 & \text{if } u = b \\ +\infty & \text{otherwise} \end{cases}
$$

for some constant vector $b \in \mathbb{R}^d$, for the reason stated in Remark 9, the number of inner iterations m_{s+1} in Algorithm 2 can be taken as the smallest integer satisfying

$$
\begin{aligned}
&2\epsilon_s + \beta_s\|\lambda^{s+1} - \lambda^s\|^2\frac{\beta_s - \beta_{s+1}}{2}\|\Lambda(p(x^s); \lambda^{s+1}, \beta_{s+1}) - \lambda^{s+1}\|^2 + \frac{\beta_s^2}{2\beta_{s+1} - \beta_s}\|x^{s-1} - x^s\|^2 \\
&\leq 2^{\lfloor m_{s+1}/K_{s+1}\rfloor}\epsilon_{s+1}/2.
\end{aligned}
$$

D.3 Proofs in Section 3.1

Proof of Corollary 3 By (36), we have

$$\mathbb{E}\left[H_{s+1}\left(x^{s+1}\right)-H_{s+1}^{\star}|\mathcal{F}_s\right]\leq 2^{-\lfloor m_{s+1}/K_{s+1}\rfloor}\left(H_{s+1}(x^s)-H_{s+1}^{\star}\right).$$

Then we apply Proposition 1 and obtain

$$\begin{aligned}\mathbb{E}\left[H_{s+1}\left(x^{s+1}\right)-H_{s+1}^{\star}|\mathcal{F}_s\right]\leq 2^{1-\lfloor m_{s+1}/K_{s+1}\rfloor}\left(H_s(x^s)-H_s^{\star}\right)\\ +2^{-\lfloor m_{s+1}/K_{s+1}\rfloor}M_s.\end{aligned}\tag{117}$$

If (38) holds, then

$$2^{-\lfloor m_{s+1}/K_{s+1}\rfloor}\leq\frac{\epsilon_{s+1}}{4\epsilon_s},\ \ 2^{-\lfloor m_{s+1}/K_{s+1}\rfloor}M_s\leq\frac{\epsilon_{s+1}}{2}.$$

It follows that

$$\mathbb{E}\left[H_{s+1}\left(x^{s+1}\right)-H_{s+1}^{\star}|\mathcal{F}_s\right]\leq\frac{\epsilon_{s+1}}{2\epsilon_s}\left(H_s(x^s)-H_s^{\star}\right)+\frac{\epsilon_{s+1}}{2}.$$

Then (39) is guaranteed by taking expectation on both sides of the last inequality.

D.4 Proofs in Section 3.2

Proof of Lemma 4 We first bound

$$\begin{aligned}&\mathbb{E}[\left((\beta_s+\beta_{s+1})L_{h_1}+\|\beta_s\lambda_1^s-\beta_{s+1}\lambda_1^{s+1}\|\right)^2+\|\beta_s\lambda_2^s-\beta_{s+1}\lambda_2^{s+1}\|^2]\\&\leq 2(\beta_s+\beta_{s+1})^2L_{h_1}^2+2\mathbb{E}[\|\beta_s\lambda^s-\beta_{s+1}\lambda^{s+1}\|^2]\\&\leq 2(\beta_s+\beta_{s+1})^2L_{h_1}^2+4(\beta_s^2+\beta_{s+1}^2)c\\&\leq 4(\beta_s^2+\beta_{s+1}^2)(L_{h_1}^2+c).\end{aligned}$$

Since

$$\lambda^{s+1}=\Lambda(p(x^s);\lambda^s,\beta_s),$$

by Lemma 12 we have

$$\|\beta_{s+1}\left(\Lambda(p(x^s);\lambda^{s+1},\beta_{s+1})-\lambda^{s+1}\right)-\beta_s(\lambda^{s+1}-\lambda^s)\|\tag{118}$$

$$\leq\sqrt{\left((\beta_s+\beta_{s+1})L_{h_1}+\|\beta_s\lambda_1^s-\beta_{s+1}\lambda_1^{s+1}\|\right)^2+\|\beta_s\lambda_2^s-\beta_{s+1}\lambda_2^{s+1}\|^2}\ .\tag{119}$$

Therefore,

$$\begin{aligned}&\|\Lambda(p(x^s);\lambda^{s+1},\beta_{s+1})-\lambda^{s+1}\|\\&\quad\le\beta_{s+1}^{-1}\beta_s\|\lambda^{s+1}-\lambda^s\|\\&\qquad+\beta_{s+1}^{-1}\sqrt{\left((\beta_s+\beta_{s+1})L_{h_1}+\|\beta_s\lambda_1^s-\beta_{s+1}\lambda_1^{s+1}\|\right)^2+\|\beta_s\lambda_2^s-\beta_{s+1}\lambda_2^{s+1}\|^2}.\end{aligned}$$

If follows that

$$\mathbb{E}[\|\Lambda(p(x^s);\lambda^{s+1},\beta_{s+1})-\lambda^{s+1}\|^2]\le 2\beta_{s+1}^{-2}\beta_s^2c+8\beta_{s+1}^{-2}(\beta_s^2+\beta_{s+1}^2)(L_{h_1}^2+c) \tag{120}$$

By $\mathbb{E}[XY]\le(\mathbb{E}[X^2])^{1/2}(\mathbb{E}[Y^2])^{1/2}$, we get

$$\begin{aligned}&\mathbb{E}\left[\|\lambda^{s+1}-\lambda^s\|\sqrt{\left((\beta_s+\beta_{s+1})L_{h_1}+\|\beta_s\lambda_1^s-\beta_{s+1}\lambda_1^{s+1}\|\right)^2+\|\beta_s\lambda_2^s-\beta_{s+1}\lambda_2^{s+1}\|^2}\right]\\&\quad\le\sqrt{4c(\beta_s^2+\beta_{s+1}^2)(L_{h_1}^2+c)}.\end{aligned} \tag{121}$$

Combining (44), (120) and (121), we then get an upper bound for $\mathbb{E}[M_s]$:

$$\begin{aligned}\mathbb{E}[M_{s+1}]&\le\beta_sc+\tfrac{\beta_s-\beta_{s+1}}{2}\left(2\beta_{s+1}^{-2}\beta_s^2c+8\beta_{s+1}^{-2}(\beta_s^2+\beta_{s+1}^2)(L_{h_1}^2+c)\right)\\&\quad+\tfrac{\beta_s^2}{2\beta_{s+1}-\beta_s}c+\sqrt{4c(\beta_s^2+\beta_{s+1}^2)(L_{h_1}^2+c)}\\&\le\beta_sc+\beta_s\left(\beta_{s+1}^{-2}\beta_s^2c+4\beta_{s+1}^{-2}(\beta_s^2+\beta_{s+1}^2)(L_{h_1}^2+c)\right)\\&\quad+\tfrac{\beta_s^2}{2\beta_{s+1}-\beta_s}c+2\beta_s\sqrt{c(1+\beta_{s+1}^2\beta_s^{-2})(L_{h_1}^2+c)}\\&\le 2\beta_sc+\beta_s\left(\beta_{s+1}^{-2}\beta_s^2c+(5+4\beta_{s+1}^{-2}\beta_s^2+\beta_{s+1}^2\beta_s^{-2})(L_{h_1}^2+c)\right)+\tfrac{\beta_s^2}{2\beta_{s+1}-\beta_s}c\end{aligned},$$

where the last inequality used $2\sqrt{ab}\le a+b$ for any $a,b>0$. Next we plug in $\beta_s=\beta_0\rho^s$ to obtain

$$\begin{aligned}\mathbb{E}[M_s]&\le\beta_s\left(2c+\rho^{-2}c+(9+\rho^{-2})(L_{h_1}^2+c)+(2\rho-1)^{-1}c\right)\\&\le\beta_s\left((11+2\rho^{-2})(L_{h_1}^2+c)+(2\rho-1)^{-1}c\right).\end{aligned}$$

Proof of Proposition 2 Since Algorithm 2 is a special case of Algorithm 1 with $\beta_s=\beta_0\rho^s$ and $\epsilon_s=\epsilon_0\eta^s$, we know from Corollary 1 that (44) holds with $c=4c_0$. Applying Lemma 4 we know that

$$\mathbb{E}[M_s]\le C\beta_s,$$

with $C=(11+2\rho^{-2})(L_{h_1}^2+4c_0)+4(2\rho-1)^{-1}c_0$. If m_{s+1} is the smallest integer satisfying (38), then

$$m_{s+1}\le K_{s+1}\left(\log_2\left(4\epsilon_s\epsilon_{s+1}^{-1}+2M_s\epsilon_{s+1}^{-1}\right)+1\right)+1. \tag{122}$$

By the concavity of $\log_2$ function we get

$$\begin{aligned}\mathbb{E}[m_{s+1}] &\le K_{s+1}\left(\log_2\left(4\epsilon_s\epsilon_{s+1}^{-1}+2C\beta_s\epsilon_{s+1}^{-1}\right)+1\right)+1\\ &= K_{s+1}\left(\log_2\left(4\eta^{-1}+2C\beta_0\epsilon_0^{-1}\eta^{-1}\rho^s\eta^{-s}\right)+1\right)+1.\end{aligned}$$

Since $\rho > \eta$, we get

$$\begin{aligned}\mathbb{E}[m_{s+1}] &\le K_{s+1}\left(\log_2\left(\left(4\eta^{-1}+2C\beta_0\epsilon_0^{-1}\eta^{-1}\right)\rho^s\eta^{-s}\right)+1\right)+1\\ &= K_{s+1}\left(\log_2\left(4\eta^{-1}+2C\beta_0\epsilon_0^{-1}\eta^{-1}\right)+1+\log_2\left(\rho^s\eta^{-s}\right)\right)+1\\ &= K_{s+1}\left(s\log_2\left(\rho\eta^{-1}\right)+c_2\right)+1.\end{aligned}$$

Proof of Theorem 2 By Corollary 2, (48) holds if

$$s \ge \frac{\ln(c_1/\epsilon)}{\ln(1/\rho)}.$$

Thus (48) is true for some integer s satisfying

$$s \le \frac{\ln(c_1/\epsilon)}{\ln(1/\rho)}+1 = \frac{\ln(c_1/(\epsilon\rho))}{\ln(1/\rho)}. \tag{123}$$

Since $\epsilon \le \epsilon_0$, we know that $\epsilon \le c_1$ and

$$s \le \frac{\ln(c_1/(\epsilon\rho))}{\ln(1/\rho)} = \frac{\ln(c_1^\ell/(\epsilon^\ell\rho^\ell))}{\ell\ln(1/\rho)} \le \frac{c_1^\ell}{\epsilon^\ell\rho^\ell\ell\ln(1/\rho)}, \tag{124}$$

where in the last inequality we used $\ln a \le a$ for any $a \ge 1$. In view of (47), we have

$$\begin{aligned}\sum_{t=1}^{s}K_t &\le \varsigma s+\frac{\omega}{\beta_0^\ell}\sum_{t=1}^{s}\rho^{-\ell t} \le \varsigma s+\frac{\omega\rho^{-\ell(s+1)}}{\beta_0^\ell(\rho^{-\ell}-1)} \overset{(123)}{\le} \varsigma s+\frac{\omega c_1^\ell}{\beta_0^\ell(1-\rho^\ell)\rho^\ell\epsilon^\ell}\\ &\overset{(124)}{\le}\left(\frac{\varsigma c_1^\ell}{\rho^\ell\ell\ln(1/\rho)}+\frac{\omega c_1^\ell}{\beta_0^\ell(1-\rho^\ell)\rho^\ell}\right)\frac{1}{\epsilon^\ell}.\end{aligned}$$

Then we apply Proposition 2 to obtain

$$\begin{aligned}\sum_{t=1}^{s}\mathbb{E}[m_t] &\le s\left(1+\log_2(\rho/\eta)+c_2\right)\left(\frac{\varsigma c_1^\ell}{\rho^\ell\ell\ln(1/\rho)}+\frac{\omega c_1^\ell}{\beta_0^\ell(1-\rho^\ell)\rho^\ell}\right)\frac{1}{\epsilon^\ell}\\ &\overset{(123)}{\le}\frac{1+\log_2(\rho/\eta)+c_2}{\ln(1/\rho)}\left(\frac{\varsigma c_1^\ell}{\rho^\ell\ell\ln(1/\rho)}+\frac{\omega c_1^\ell}{\beta_0^\ell(1-\rho^\ell)\rho^\ell}\right)\frac{1}{\epsilon^\ell}\ln\frac{c_1}{\epsilon\rho}.\end{aligned}$$

D.5 Proof in Section 5.1

Proof of Corollary 5 If K_s satisfies (62), then

$$K_s \leq 2\sqrt{\frac{2(L\beta_0 + \|A\|^2)}{\mu_g \beta_s + \beta_s^2}} + 1 \leq \begin{cases} \frac{2\sqrt{2(L\beta_0+\|A\|^2)/\mu_g}}{\sqrt{\beta_s}} + 1 & \text{if } \mu_g > 0 \\ \frac{2\sqrt{2(L\beta_0+\|A\|^2)}}{\beta_s} + 1 & \text{if } \mu_g = 0 \end{cases}$$

We then apply Corollary 4.

The proof of Corollary 6 and 7 are similar.

D.6 Proofs in Section 5.2

We first state a useful Lemma.

Lemma 14 *For any $u, \lambda \in \mathbb{R}^d$, $\beta > 0$,*

$$\|\Lambda(u; \lambda, \beta)\| \leq L_{h_1} + \beta^{-1} \operatorname{dist}(u_2 + \beta\lambda_2, \mathcal{K}) \tag{125}$$

Proof From (21),

$$h(u; \lambda, \beta) = \min_z \left\{ h(z) + \frac{1}{2\beta}\|u + \beta\lambda - z\|^2 - \frac{\beta}{2}\|\lambda\|^2 \right\} \tag{126}$$

with optimal solution

$$z^\star = u + \beta\lambda - \beta\Lambda(u; \lambda, \beta).$$

In particular, $\operatorname{dist}(u_2 + \beta\lambda_2, \mathcal{K})^2 = \beta^2\|\Lambda_2(u_2; \lambda_2, \beta)\|^2$. Together with (109) we obtain the desired bound. □

Proof of Lemma 5

$$\begin{aligned}
&\|\nabla p(x)\Lambda(p(x); \lambda^s, \beta_s) - \nabla p(y)\Lambda(p(y); \lambda^s, \beta_s)\| \\
&\leq \|\nabla p(x) - \nabla p(y)\|\|\Lambda(p(x); \lambda^s, \beta_s)\| + \|\nabla p(y)\|\|\Lambda(p(x); \lambda^s, \beta_s) \\
&\quad -\Lambda(p(y); \lambda^s, \beta_s)\| \\
&\overset{(125)+(19)}{\leq} L_{\nabla p}\|x - y\| \left(L_{h_1} + \beta_s^{-1} \operatorname{dist}(p_2(x) + \beta_s\lambda_2^s, \mathcal{K}) \right) \\
&\quad + M_{\nabla p}\|p(x) - p(y)\|\beta_s^{-1} \\
&\leq \left(L_{\nabla p} \left(L_{h_1} + \beta_s^{-1} \operatorname{dist}(p_2(x) + \beta_s\lambda_2^s, \mathcal{K}) \right) + M_{\nabla p}^2 \beta_s^{-1} \right) \|x - y\|.
\end{aligned}$$

Note that by (74) and the definition of d_s,

$$\operatorname{dist}(p_2(x) + \beta_s\lambda_2^s, \mathcal{K}) \leq d_s.$$

D.7 Proofs in Section 6.2

Proof of Theorem 3 We know from the basic property of proximal gradient step [32] that

$$\|x^s - \tilde{x}^s\|^2 \leq 2\left(H_s(\tilde{x}^s) - H_s^\star\right)/L_s.$$

By Line 4 in Algorithm 3,

$$0 \in \nabla\phi_s(\tilde{x}^s) + L_s(x^s - \tilde{x}^s) + \beta_s(x^s - x^{s-1}) + \partial g(x^s).$$

Therefore,

$$\begin{aligned}
&\operatorname{dist}(0, \nabla\phi_s(x^s) + \partial g(x^s)) \\
&\leq L_s\|\tilde{x}^s - x^s\| + \|\nabla\phi_s(x^s) - \nabla\phi_s(\tilde{x}^s)\| + \beta_s\|x^s - x^{s-1}\| \\
&\leq 2L_s\|\tilde{x}^s - x^s\| + \beta_s\|x^s - x^{s-1}\|
\end{aligned}$$

Combining the last two bounds and (18) we get $\nabla\phi_s(x^s) = \nabla f(x^s) + \nabla p(x^s)\lambda^{s+1}$ and

$$\operatorname{dist}(0, \nabla f(x^s) + \nabla p(x^s)\lambda^{s+1} + \partial g(x^s))^2 \leq 16L_s\left(H_s(\tilde{x}^s) - H_s^\star\right) + 2\beta_s^2\|x^s - x^{s-1}\|^2.$$

Secondly we know from (20) that

$$p(x^s) - \beta_s(\lambda^{s+1} - \lambda^s) \in \partial h^*(\lambda^{s+1}).$$

It follows that

$$\operatorname{dist}(0, p(x^s) - \partial h^*(\lambda^{s+1})) \leq \beta_s\|\lambda^{s+1} - \lambda^s\|.$$

Proof of Corollary 9 Due to (82), we can have the same bound (in expectation) of the sequence $\{(\tilde{x}^s, x^s, \lambda^s)\}$ as Corollary 1. Hence,

$$\begin{aligned}
&\mathbb{E}\left[\operatorname{dist}(0, \partial_x L(x^s, \lambda^{s+1}))\right] \leq \sqrt{16L_s\epsilon_s + 8c_0\beta_s^2} \leq \sqrt{16\gamma\epsilon_0/\beta_0 + 8c_0\beta_0}\rho^s, \\
&\mathbb{E}\left[\operatorname{dist}(0, \partial_\lambda L(x^s, \lambda^{s+1}))\right] \leq \beta_0\sqrt{c_0}\rho^s.
\end{aligned}$$

References

1. Alacaoglu, A., Tran-Dinh, Q., Fercoq, O., Cevher, V.: Smooth primal-dual coordinate descent algorithms for nonsmooth convex optimization. In: Advances in Neural Information Processing Systems, pp. 5852–5861 (2017)
2. Allen-Zhu, Z.: Katyusha: the first direct acceleration of stochastic gradient methods. J. Mach. Learn. Res. **18**(1), 8194–8244 (2017)

3. Auslender, A., Teboulle, M.: Interior projection-like methods for monotone variational inequalities. Math. Program. **104**(1), 39–68 (2005). https://doi.org/10.1007/s10107-004-0568-x
4. Beck, A., Teboulle, M.: A fast iterative shrinkage-thresholding algorithm for linear inverse problems. SIAM J. Imaging Sci. **2**(1), 183–202 (2009)
5. Beck, A., Teboulle, M.: Smoothing and first order methods: a unified framework. SIAM J. Optim. **22**(2), 557–580 (2012). https://doi.org/10.1137/100818327
6. Belloni, A., Chernozhukov, V., Wang, L.: Square-root lasso: pivotal recovery of sparse signals via conic programming. SSRN Electron. J. (2011). https://doi.org/10.2139/ssrn.1910753
7. Bertsekas, D.P.: Constrained Optimization and Lagrange Multiplier Methods. Academic Press, Cambridge (2014)
8. Bolte, J.H., Bauschke, J., Teboulle, M.: A descent lemma beyond Lipschitz gradient continuity: first-order methods revisited and applications. Math. Oper. Res. **42**, 330–348 (2016). https://doi.org/10.1287/moor.2016.0817
9. Borwein, J.M., Vanderwerff, J.D., et al.: Convex Functions: Constructions, Characterizations and Counterexamples, vol. 109. Cambridge University Press, Cambridge (2010)
10. Chambolle, A., Ehrhardt, M.J., Richtárik, P., Schonlieb, C.B.: Stochastic primal-dual hybrid gradient algorithm with arbitrary sampling and imaging applications. SIAM J. Optim. **28**(4), 2783–2808 (2018)
11. Chambolle, A., Pock, T.: A first-order primal-dual algorithm for convex problems with applications to imaging. J. Math. Imaging Vis. **40**(1), 120–145 (2011)
12. Chang, C.C., Lin, C.J.: LIBSVM: a library for support vector machines. ACM Trans. Intell. Syst. Technol. **2**(3), 27 (2011)
13. Chen, S., Donoho, D., Saunders, M.: Atomic decomposition by basis pursuit. SIAM J. Sci. Comput. **20**(1), 33–61 (1998). https://doi.org/10.1137/S1064827596304010
14. Drusvyatskiy, D., Paquette, C.: Efficiency of minimizing compositions of convex functions and smooth maps. Math. Program. **178**, 503–558 (2019)
15. Fercoq, O., Qu, Z.: Restarting the accelerated coordinate descent method with a rough strong convexity estimate. arXiv:1803.05771 (2018)
16. Fercoq, O., Qu, Z.: Adaptive restart of accelerated gradient methods under local quadratic growth condition. IMA J. Numer. Anal. (2019). https://doi.org/10.1093/imanum/drz007
17. Fercoq, O., Richtárik, P.: Accelerated, parallel and proximal coordinate descent. SIAM J. Optim. **25**(4), 1997–2023 (2015)
18. Friedlander, M.P., Goh, G.: Efficient evaluation of scaled proximal operators. Electron. Trans. Numer. Anal. **46**, 1–22 (2017)
19. Bauschke, H.H., Combettes, P.: The baillon-haddad theorem revisited. J. Convex Anal. **17**, 1–7 (2009)
20. Hien, L.T.K., Zhao, R., Haskell, W.B.: An inexact primal-dual smoothing framework for large-scale non-bilinear saddle point problems. arXiv preprint arXiv:1711.03669 (2017)
21. Kovalev, D., Horváth, S., Richtárik, P.: Don't jump through hoops and remove those loops: SVRG and Katyusha are better without the outer loop (2019)
22. Lan, G., Monteiro, R.D.: Iteration-complexity of first-order augmented Lagrangian methods for convex programming. Math. Program. **155**(1–2), 511–547 (2016). https://doi.org/10.1007/s10107-015-0861-x
23. Li, H., Lin, Z.: On the complexity analysis of the primal solutions for the accelerated randomized dual coordinate ascent. arXiv preprint arXiv:1807.00261 (2018)
24. Liu, Y., Liu, X., Ma, S.: On the nonergodic convergence rate of an inexact augmented lagrangian framework for composite convex programming. Math. Oper. Res. **44**(2), 632–650 (2019). https://doi.org/10.1287/moor.2018.0939
25. Lu, H., Freund, R., Nesterov, Y.: Relatively smooth convex optimization by first-order methods, and applications. SIAM J. Optim. **28**(1), 333–354 (2018). https://doi.org/10.1137/16M1099546
26. Lu, Z., Zhou, Z.: Iteration-complexity of first-order augmented Lagrangian methods for convex conic programming. arXiv preprint arXiv:1803.09941 (2018)
27. Necoara, I., Nesterov, Y., Glineur, F.: Linear convergence of first order methods for non-strongly convex optimization. Math. Program. (2018). https://doi.org/10.1007/s10107-018-1232-1
28. Necoara, I., Patrascu, A., Glineur, F.: Complexity of first-order inexact Lagrangian and penalty methods for conic convex programming. Optim. Methods Softw. **34**(2), 305–335 (2019). https://doi.org/10.1080/10556788.2017.1380642

29. Nedelcu, V., Necoara, I., Tran-Dinh, Q.: Computational complexity of inexact gradient augmented Lagrangian methods: application to constrained MPC. SIAM J. Control Optim. **52**(5), 3109–3134 (2014). https://doi.org/10.1137/120897547
30. Nesterov, Y.: A method of solving a convex programming problem with convergence rate $O(1/k^2)$. Sov. Math. Doklady **27**(2), 372–376 (1983)
31. Nesterov, Y.: Smooth minimization of non-smooth functions. Math. Program. **103**(1), 127–152 (2005). https://doi.org/10.1007/s10107-004-0552-5
32. Nesterov, Y., et al.: Gradient methods for minimizing composite objective function (2007)
33. Ouyang, Y., Chen, Y., Lan, G., Pasiliao, E., Jr.: An accelerated linearized alternating direction method of multipliers. SIAM J. Imaging Sci. **8**(1), 644–681 (2015)
34. Patrascu, A., Necoara, I., Tran-Dinh, Q.: Adaptive inexact fast augmented Lagrangian methods for constrained convex optimization. Optim. Lett. **11**, 609–626 (2015). https://doi.org/10.1007/s11590-016-1024-6
35. Qian, X., Qu, Z., Richtárik, P.: L-SVRG and L-Katyusha with arbitrary sampling. arXiv:1906.01481 (2019)
36. Rafique, H., Liu, M., Lin, Q., Yang, T.: Non-convex min-max optimization: Provable algorithms and applications in machine learning. arXiv preprint arXiv:1810.02060 (2018)
37. Rockafellar, R.T.: Convex Analysis. Princeton Mathematical Series, Princeton University Press, Princeton (1970)
38. Rockafellar, R.T.: Augmented Lagrangians and applications of the proximal point algorithm in convex programming. Math. Oper. Res. **1**(2), 97–116 (1976)
39. Rockafellar, R.T.: Monotone operators and the proximal point algorithm. SIAM J. Control Optim. **14**(5), 877–898 (1976)
40. Scokaert, P.O.M., Mayne, D.Q., Rawlings, J.B.: Suboptimal model predictive control (feasibility implies stability). IEEE Trans. Autom. Control **44**(3), 648–654 (1999). https://doi.org/10.1109/9.751369
41. Simon, N., Friedman, J., Hastie, T., Tibshirani, R.: A sparse-group lasso. J. Comput. Graph. Stat. **22**, 231–245 (2013)
42. Tibshirani, R., Saunders, M., Rosset, S., Zhu, J., Knight, K.: Sparsity and smoothness via the fused lasso. J. R. Stat. Soc. Ser. B **67**, 91–108 (2005)
43. Tran-Dinh, Q., Alacaoglu, A., Fercoq, O., Cevher, V.: An adaptive primal-dual framework for non-smooth convex minimization. Math. Program. Comput. (2019). https://doi.org/10.1007/s12532-019-00173-3
44. Tran-Dinh, Q., Fercoq, O., Cevher, V.: A smooth primal-dual optimization framework for nonsmooth composite convex minimization. SIAM J. Optim. **28**(1), 96–134 (2018)
45. Tseng, P.: On accelerated proximal gradient methods for convex-concave optimization. Submitted to SIAM Journal on Optimization (2008)
46. Wang, H., Li, G., Jiang, G.: Robust regression shrinkage and consistent variable selection through the lad-lasso. J. Bus. Econ. Stat. **25**(3), 347–355 (2007)
47. Xu, Y.: First-order methods for constrained convex programming based on linearized augmented Lagrangian function. arXiv preprint arXiv:1711.08020 (2017)
48. Xu, Y.: Iteration complexity of inexact augmented Lagrangian methods for constrained convex programming. arXiv:1711.05812 (2017)
49. Xu, Y., Zhang, S.: Accelerated primal-dual proximal block coordinate updating methods for constrained convex optimization. Comput. Optim. Appl. **70**(1), 91–128 (2018)
50. Yuan, X., Zeng, S., Zhang, J.: Discerning the linear convergence of ADMM for structured convex optimization through the lens of variational analysis. optimization-online (2018)
51. Zhu, J., Rosset, S., Hastie, T., Tibshirani, R.: 1normm support vector machines. In: Proceedings of the 16th International Conference on Neural Information Processing Systems, NIPS'03, pp. 49–56. MIT Press, Cambridge, MA, USA (2003). http://dl.acm.org/citation.cfm?id=2981345.2981352