

Retraction-based first-order feasible methods for difference-of-convex programs with smooth inequality and simple geometric constraints

Yongle Zhang* Guoyin Li† Ting Kei Pong‡ Shiqi Xu§

August 23, 2022

Abstract

In this paper, we propose first-order feasible methods for difference-of-convex (DC) programs with smooth inequality and simple geometric constraints. Our strategy for maintaining feasibility of the iterates is based on a “retraction” idea adapted from the literature of manifold optimization. When the constraints are convex, we establish the global subsequential convergence of the sequence generated by our algorithm under strict feasibility condition, and analyze its convergence rate when the objective is in addition convex according to the Kurdyka-Lojasiewicz (KL) exponent of the extended objective (i.e., sum of the objective and the indicator function of the constraint set). We also show that the extended objective of a large class of Euclidean norm (and more generally, group LASSO penalty) regularized convex optimization problems is a KL function with exponent $\frac{1}{2}$; consequently, our algorithm is locally linearly convergent when applied to these problems. We then extend our method to solve DC programs with a single specially structured nonconvex constraint. Finally, we discuss how our algorithms can be applied to solve two concrete optimization problems, namely, group-structured compressed sensing problems with Gaussian measurement noise and compressed sensing problems with Cauchy measurement noise, and illustrate the empirical performance of our algorithms.

Keywords: First-order feasible methods, retraction, difference-of-convex optimization, Kurdyka-Lojasiewicz exponents

1 Introduction

We consider the following difference-of-convex (DC) optimization problem with smooth inequality and geometric constraints:

$$\begin{aligned} \min_{x \in \mathbb{R}^n} \quad & P(x) := P_1(x) - P_2(x) \\ \text{s.t.} \quad & g_i(x) \leq 0, \quad i = 1, \dots, m, \\ & x \in C, \end{aligned} \tag{1.1}$$

*Department of Mathematics, Visual Computing and Virtual Reality Key Laboratory of Sichuan Province, Sichuan Normal University, Chengdu, People’s Republic of China. E-mail: yongle-zhang@163.com yongle-zhang@163.com

†Department of Applied Mathematics, University of New South Wales, Sydney, Australia. E-mail: g.li@unsw.edu.au aug.li@unsw.edu.au

‡Department of Applied Mathematics, the Hong Kong Polytechnic University, Hong Kong, People’s Republic of China. E-mail: tk.pong@polyu.edu.hk hktk.pong@polyu.edu.hk

§Department of Mathematics, Sichuan Normal University, Chengdu, People’s Republic of China. E-mail: xushiq98@163.com xushiq98@163.com

where $P_1 : \mathbb{R}^n \rightarrow \mathbb{R}$ and $P_2 : \mathbb{R}^n \rightarrow \mathbb{R}$ are finite-valued convex functions, $C \subseteq \mathbb{R}^n$ is a nonempty compact convex set, $g_i : \mathbb{R}^n \rightarrow \mathbb{R}$ is smooth with Lipschitz gradient on C for each i , and the feasible set $\mathcal{F} := \{x \in C : g_i(x) \leq 0, i = 1, \dots, m\}$ is nonempty.

The model problem (1.1) covers many important optimization problems arising from diverse areas. For example, one can model the compressed sensing (CS) problem with Cauchy noise by minimizing the difference of ℓ_1 -norm and ℓ_2 -norm [45] or its variants subject to an inequality constraint defined by the so-called Lorentzian norm [17], and the set C in (1.1) can be used for modeling other prior information such as bounds or nonnegativity; more discussions on this particular application can be found in Section 6.2 below. Problem (1.1) is typically challenging to solve because it is in general nonconvex and nonsmooth.

In this paper, we are interested in developing first-order feasible methods for solving problem (1.1) that rely on affine approximations to the smooth inequality constraints in the subproblems. Our methods utilize only first-order information, and the subproblems that arise are convex constrained optimization problems with the constraint sets being the intersection of C and a polyhedron. Moreover, it also has the desirable feature that all the iterates generated always belong to the feasible region. In the literature, the most widely used method for solving (1.1) when the objective function P is additionally smooth and also involves affine approximations to the constraint set is the so-called sequential quadratic programming (SQP) method: Indeed, this class of methods typically further requires that the objective function P and all the constraint functions g_i 's to be twice continuously differentiable. The underlying strategy of SQP methods is to construct an algorithm which, at each iteration, utilizes a convex quadratic programming problem with affine constraints to approximate the original nonlinear optimization problem. The literature is vast and we refer the readers to [2, 7, 11, 12, 24, 25, 28, 31, 35, 37, 40, 44]. Recent works on SQP-type methods include [7, 12], where the authors proposed and studied the global convergence of some general majorization minimization schemes which cover various SQP-type methods as special cases. However, these SQP-type methods therein are infeasible SQP-type methods, that is, they do not necessarily force all iterates to be feasible. There are also “feasible SQP methods” that perform arc-search along quadratic curves determined by two directions, namely a descent direction and a feasible direction (see, for example, [25, 31, 37]). However, these methods make use of the smoothness of the objective and constraint functions to construct the two directions, and it is unclear how their strategies can be adapted to problem (1.1) that has a possibly nonsmooth objective.

In this paper, we aim at developing feasible first-order methods for the nonconvex nonsmooth problem (1.1) under additional assumptions on g_i by making use of affine approximations for the constraints as in SQP-type methods, and establishing the global convergence of the proposed algorithms. Different from the SQP-type methods covered in [7, 12], our methods maintain feasibility of all iterates for solving (1.1), under additional assumptions on g_i . Our strategy for maintaining feasibility is different from those adopted in the literature of “feasible SQP methods” that are based on arc-search. Instead, we adapt the retraction strategy used in manifold optimization [19, 21] in our scheme to ensure descent property for the objective function while maintaining the feasibility of the iterates.

For nonconvex optimization problems with difference-of-convex structure, our method belongs to the so-called successive convex programs strategy which, in each subproblem, a convex approximation to the constraint is formed. In the literature, popular methods of this form include the standard difference-of-convex algorithm (DCA) and its variants (see [32] for a recent survey), and also the moving balls approximation methods [7, 8, 12, 48]. For these methods, their subproblems involve quadratic or general convex constraints which are, in general, harder to solve comparing to our proposed method because our method solves a linearly constrained problem in each subproblem. It is also worth mentioning that the technique in using affine approximation for the constraints

to construct numerical methods to solve possibly nonconvex problems has been exploited in the literature (for example, see [1, 2, 43]). However, our method differs from these methods in various important aspects. More explicitly, [1, 43] proposed the so-called supporting hyperplane methods which employ a cutting plane strategy. In particular, for the supporting hyperplane methods in [1, 43], as the algorithm progresses, the number of halfspaces involved in the subproblem *grows*, with many of them constructed based on information from the *past iterates*. In contrast, our method only requires a *fixed* number of halfspaces in each iteration, and the constraint functions are linearized only at the *current iterate*. Thus, each subproblem in our method bears a similar computational cost. While the method in [2] can make use of subproblems with a *fixed* number of constraints and is applicable to a wider class of problems (e.g., it does not require differentiability of the constraint function), the sequence generated and its cluster points are not guaranteed to be feasible; this is different from our method, which maintains the feasibility of all iterates. In addition, due to the specific construction of the retraction step, our method ensures the descent property of the objective value: an important aspect that was not exploited in [1, 2, 43].

More explicitly, the contributions of this paper are summarized as follows:

- We propose a retraction based first-order feasible method that utilizes affine approximations for the constraints in each subproblem for solving problem (1.1) in the case where the constraint functions g_i 's are convex under strict feasibility condition, and establish its global subsequential convergence. We also analyze the convergence rate in the fully convex setting, i.e., when P_2 is also zero, under suitable Kurdyka-Łojasiewicz (KL) assumptions.
- By assuming a stronger version of the strict feasibility condition, we then extend our method to solve (1.1) with one single specially structured nonconvex constraint function and establish the global subsequential convergence of the extended method. Our assumptions on the constraint function are general enough to allow the choice of the composition of the Lorentzian norm function with an affine function, a commonly used loss function for compressed sensing problems with Cauchy measurement noise [17].
- Finally, we examine the KL exponents of several structured optimization models of the form (1.1), an important quantity which quantifies the convergence rate of the proposed method in the convex setting. Specifically, we show that the extended objective¹ of a large class of Euclidean norm (and more generally, group LASSO penalty) regularized convex optimization problems is a KL function with exponent $\frac{1}{2}$. This complements the recent study in this direction [33, 46, 48, 50], and, in particular, gives rise to linear convergence of our method in solving these structured optimization models.

The rest of the paper is organized as follows. We introduce notation and preliminary materials in Section 2. Our retraction based first-order feasible method for solving (1.1) when g_i 's are convex is presented and analyzed in Section 3; we also analyze the convergence rate there in the fully convex setting. In Section 4, we discuss how our method can be extended to solving (1.1) with one single specially structured nonconvex constraint function. Explicit KL exponents of several structured optimization models in the form of (1.1) are derived in Section 5. Finally, in Section 6, we perform numerical experiments to compare the performance of our proposed methods with an adaptation of the method in [7] on solving two concrete optimization problems: group-structured CS problems with Gaussian measurement noise and CS problems with Cauchy measurement noise. For the readers' convenience, in the Appendix, we also provide explanations on how the subproblems are solved in our numerical experiments.

¹This refers to the sum of the objective and the indicator function of the constraint set.

2 Notation and preliminaries

In this paper, we use \mathbb{R}^n to denote the Euclidean space of dimension n and \mathbb{R}_+^n to denote the nonnegative orthant of \mathbb{R}^n . For two vectors x and $y \in \mathbb{R}^n$, we use $\langle x, y \rangle$ to denote the inner product. The Euclidean norm of x is denoted by $\|x\|$ and the ℓ_1 norm of x is denoted by $\|x\|_1$. For $x \in \mathbb{R}^n$ and $r \geq 0$, we let $B(x, r)$ denote the closed ball centered at x with radius r with respect to the Euclidean norm. We also use \mathbb{N}_+ to denote the set of positive integers.

For an extended-real-valued function $h : \mathbb{R}^n \rightarrow [-\infty, \infty]$, we denote its domain by $\text{dom } h = \{x \in \mathbb{R}^n : h(x) < \infty\}$. The function h is said to be proper if $h(x) > -\infty$ for all $x \in \mathbb{R}^n$ and $\text{dom } h \neq \emptyset$. Moreover, a proper function is said to be closed if it is lower semicontinuous. Given a proper closed function h , the regular subdifferential $\hat{\partial}h$ and the (limiting) subdifferential ∂h at $x \in \text{dom } h$ are given respectively by

$$\begin{aligned}\hat{\partial}h(x) &= \left\{ v \in \mathbb{R}^n : \liminf_{y \rightarrow x, y \neq x} \frac{h(y) - h(x) - \langle v, y - x \rangle}{\|y - x\|} \geq 0 \right\}, \\ \partial h(x) &= \left\{ v \in \mathbb{R}^n : \exists x^k \xrightarrow{h} x \text{ and } v^k \in \hat{\partial}h(x^k) \text{ with } v^k \rightarrow v \right\},\end{aligned}$$

where $x^k \xrightarrow{h} x$ means $x^k \rightarrow x$ and $h(x^k) \rightarrow h(x)$. By convention, if $x \notin \text{dom } h$, we set $\partial h(x) = \emptyset$. We also write $\text{dom } \partial h := \{x \in \mathbb{R}^n : \partial h(x) \neq \emptyset\}$. When h is continuously differentiable at x , we have $\partial h(x) = \{\nabla h(x)\}$ according to [39, Exercise 8.8(b)]. Moreover, when h is convex, from [39, Proposition 8.12], the above subdifferentials reduce to the classical subdifferential in convex analysis.

For a nonempty set $C \subseteq \mathbb{R}^n$, the support function is defined as $\sigma_C(x) := \sup_{y \in C} \langle x, y \rangle$ and the indicator function δ_C is defined as

$$\delta_C(x) := \begin{cases} 0 & \text{if } x \in C, \\ \infty & \text{otherwise.} \end{cases}$$

The normal cone of a closed set C at an $x \in C$ is defined by $\mathcal{N}_C(x) := \partial \delta_C(x)$. In addition, for a nonempty closed set $C \subseteq \mathbb{R}^n$, we denote the distance from a point $x \in \mathbb{R}^n$ to C by $\text{dist}(x, C) := \inf_{y \in C} \|x - y\|$.

We next recall the Kurdyka-Łojasiewicz property, which is satisfied by many functions such as proper closed semialgebraic functions, and is important for analyzing global convergence and local convergence rate of first-order methods; see, for example, [4–6, 13, 33].

Definition 2.1 (Kurdyka-Łojasiewicz property and exponent). *We say that a proper closed function $h : \mathbb{R}^n \rightarrow (-\infty, \infty]$ satisfies the Kurdyka-Łojasiewicz (KL) property at an $\hat{x} \in \text{dom } \partial h$ if there are $a \in (0, \infty]$, a neighborhood V of \hat{x} and a continuous concave function $\phi : [0, a) \rightarrow [0, \infty)$ with $\phi(0) = 0$ such that*

- (i) ϕ is continuously differentiable on $(0, a)$ with $\phi' > 0$ on $(0, a)$;
- (ii) for any $x \in V$ with $h(\hat{x}) < h(x) < h(\hat{x}) + a$, it holds that

$$\phi'(h(x) - h(\hat{x})) \text{dist}(0, \partial h(x)) \geq 1. \quad (2.1)$$

If h satisfies the KL property at $\hat{x} \in \text{dom } \partial h$ and the ϕ in (2.1) can be chosen as $\phi(\nu) = a_0 \nu^{1-\alpha}$ for some $a_0 > 0$ and $\alpha \in [0, 1)$, then we say that h satisfies the KL property at \hat{x} with exponent α .

A proper closed function h satisfying the KL property at every point in $\text{dom } \partial h$ is called a KL function, and a proper closed function h satisfying the KL property with exponent $\alpha \in [0, 1)$ at every point in $\text{dom } \partial h$ is called a KL function with exponent α .

Before ending this section, we recall the following standard constraint qualifications on the constraint set of (1.1) and the first-order necessary optimality conditions for (1.1).

Definition 2.2 (MFCQ). *We say that the Mangasarian-Fromovitz constraint qualifications for (1.1) holds at an $x \in \mathcal{F}$ if the following implication holds:*

$$\left. \begin{aligned} & -\sum_{i=1}^m \lambda_i \nabla g_i(x) \in \mathcal{N}_C(x) \\ & \lambda_i g_i(x) = 0, \quad i = 1, \dots, m \\ & \lambda_i \geq 0, \quad i = 1, \dots, m \end{aligned} \right\} \Rightarrow \lambda_i = 0, \quad i = 1, \dots, m.$$

It is known that if the g_i 's in (1.1) are in addition convex and the (generalized) Slater condition holds, i.e., $\{x \in C : g_i(x) < 0, \quad i = 1, \dots, m\} \neq \emptyset$, then the MFCQ holds at any point in \mathcal{F} ; indeed, suppose that g_i 's in (1.1) are in addition convex, $x \in \mathcal{F}$, $-\sum_{i=1}^m \lambda_i \nabla g_i(x) \in \mathcal{N}_C(x)$, $\lambda_i g_i(x) = 0$ and $\lambda_i \geq 0$ for all i , and let $\hat{x} \in C$ satisfy $\max_{1 \leq i \leq m} g_i(\hat{x}) \leq -\delta < 0$ for some $\delta > 0$. Then

$$0 \leq \sum_{i=1}^m \langle \lambda_i \nabla g_i(x), \hat{x} - x \rangle \leq \sum_{i=1}^m \lambda_i (g_i(\hat{x}) - g_i(x)) = \sum_{i=1}^m \lambda_i g_i(\hat{x}) \leq -\delta \sum_{i=1}^m \lambda_i,$$

where the first inequality holds thanks to $-\sum_{i=1}^m \lambda_i \nabla g_i(x) \in \mathcal{N}_C(x)$ and $\hat{x} \in C$, and the second inequality is due to convexity. The above display together with $\lambda_i \geq 0$ for all i shows that $\lambda_i = 0$ for all i .

Definition 2.3 (Critical point). *We say that $x \in \mathbb{R}^n$ is a critical point of (1.1) if there exists $\lambda = (\lambda_1, \lambda_2, \dots, \lambda_m) \in \mathbb{R}_+^m$ such that (x, λ) satisfies*

- (i) $g_i(x) \leq 0, \quad i = 1, \dots, m$, and $x \in C$;
- (ii) $\lambda_i g_i(x) = 0, \quad i = 1, \dots, m$;
- (iii) $0 \in \partial P_1(x) - \partial P_2(x) + \sum_{i=1}^m \lambda_i \nabla g_i(x) + \mathcal{N}_C(x)$.

If the MFCQ holds at every point in \mathcal{F} , by using similar arguments as in [48, Section 2], one can show that any local minimizer of (1.1) is a critical point of (1.1). We omit its proof for brevity.

3 Convex constraints with an explicit strict feasible point

In this section, we study (1.1) under the following additional assumption:

Assumption 3.1. *For each $i = 1, \dots, m$, the function g_i in (1.1) is convex. Moreover, the Slater condition holds and a Slater point is known, i.e., there exists $x^\odot \in C$ with $g_i(x^\odot) < 0$ for $i = 1, \dots, m$, and x^\odot is explicitly known.*

We now describe our method for solving (1.1) under Assumption 3.1 in Algorithm 1 below. In essence, our algorithm is an SQP-type method in the sense that, in each iteration, we replace the smooth functions g_i by their affine minorants in the subproblem (see (3.1)). On the other hand, we force our iterates to be feasible by performing a suitable convex combination with the Slater point x^\odot in Step 2a). A line-search scheme is adapted to guarantee sufficient descent in Step 2b). This strategy for maintaining feasibility is different from those adopted in feasible SQP methods, which perform arc-search along quadratic curves [25, 37]. Our strategy is more similar in spirit to the retraction strategy in manifold optimization [19, 21]: as we shall see in the convergence analysis,

Algorithm 1 FPA method with retraction for (1.1) under Assumption 3.1

Step 0. Choose $x^0 \in \mathcal{F}$, $0 < \underline{\beta} < \bar{\beta}$, $c > 0$ and $\eta \in (0, 1)$. Choose x^\odot as in Assumption 3.1 and set $k = 0$.

Step 1. Pick any $\xi^k \in \partial P_2(x^k)$. Choose $\beta_k^0 \in [\underline{\beta}, \bar{\beta}]$ arbitrarily and set $\tilde{\beta} = \beta_k^0$.

Step 2. Compute

$$\begin{aligned} \tilde{u} = \arg \min_{x \in \mathbb{R}^n} \quad & P_1(x) - \langle \xi^k, x - x^k \rangle + (2\tilde{\beta})^{-1} \|x - x^k\|^2 \\ \text{s.t.} \quad & g_i(x^k) + \langle \nabla g_i(x^k), x - x^k \rangle \leq 0, \quad i = 1, \dots, m, \\ & x \in C. \end{aligned} \quad (3.1)$$

Step 2a) If $\max_{1 \leq i \leq m} \{g_i(\tilde{u})\} \leq 0$, set $\tilde{x} = \tilde{u}$ and $\tilde{\tau} = 0$; else, find $\tilde{\tau} \in (0, 1)$ such that $\tilde{x} := (1 - \tilde{\tau})\tilde{u} + \tilde{\tau}x^\odot$ satisfies $\max_{1 \leq i \leq m} \{g_i(\tilde{x})\} = 0$.

Step 2b) If $P(\tilde{x}) \leq P(x^k) - \frac{c}{2} \|\tilde{u} - x^k\|^2$, go to **Step 3**; else, update $\tilde{\beta} \leftarrow \eta \tilde{\beta}$ and go to **Step 2**.

Step 3. Set $x^{k+1} = \tilde{x}$, $u^k = \tilde{u}$, $\beta_k = \tilde{\beta}$, $\tau_k = \tilde{\tau}$. Update $k \leftarrow k + 1$ and go to **Step 1**.

they induce similar descent conditions on the objective function values. Since our method makes use of affine approximations (hence, polyhedral) and leverages the idea of retraction, we shall refer to our algorithm as first-order polyhedral approximation (FPA) method with retraction.

We first show that Algorithm 1 is well defined. To this end, it suffices to show that, if $x^k \in \mathcal{F}$ for some $k \geq 0$, then the subproblem (3.1) has a unique solution, $\tilde{\tau}$ exists in Step 2a) and the line-search step in Step 2b) terminates in finitely many inner iterations, so that x^{k+1} can be generated. Note that such an x^{k+1} belongs to \mathcal{F} in view of Step 2a). This together with an induction argument would establish the well-definedness of Algorithm 1.

Theorem 3.1 (Well-definedness of Algorithm 1). *Consider (1.1) and suppose that Assumption 3.1 holds. Suppose that $x^k \in \mathcal{F}$ is generated at the beginning of the k -th iteration of Algorithm 1 for some $k \geq 0$. Then the following statements hold:*

- (i) *The convex subproblem (3.1) has a unique solution. Moreover, x^\odot from Assumption 3.1 is a Slater point of the feasible set of (3.1), that is, $x^\odot \in C$ and*

$$\max_{1 \leq i \leq m} \{g_i(x^k) + \langle \nabla g_i(x^k), x^\odot - x^k \rangle\} < 0.$$

- (ii) *If $\max_{1 \leq i \leq m} \{g_i(\tilde{u})\} > 0$, then there exists $\tilde{\tau} \in (0, 1)$ such that $\tilde{x} := (1 - \tilde{\tau})\tilde{u} + \tilde{\tau}x^\odot$ satisfies $\max_{1 \leq i \leq m} \{g_i(\tilde{x})\} = 0$. Moreover, in this case, for any such $\tilde{\tau}$, we have*

$$\tilde{\tau} \leq \frac{L \|\tilde{u} - x^k\|^2}{-2 \max_{1 \leq i \leq m} \{g_i(x^\odot)\}} \quad \text{and} \quad \|\tilde{x} - \tilde{u}\| \leq \frac{ML \|\tilde{u} - x^k\|^2}{-2 \max_{1 \leq i \leq m} \{g_i(x^\odot)\}}, \quad (3.2)$$

where $M := \sup_{x \in C} \|x - x^\odot\|$ and $L := \max_{1 \leq i \leq m} L_{g_i}$, with L_{g_i} being the Lipschitz continuity modulus of ∇g_i on C for each i .

- (iii) *Step 2 of algorithm 1 terminates in finitely many inner iterations.*

Thus, an $x^{k+1} \in \mathcal{F}$ can be generated at the end of the $(k+1)$ -th iteration of Algorithm 1.

Proof. (i): First, we note that the feasible region of (3.1) is closed and convex. Next, we show that this set is nonempty by showing that it contains the point $x^\odot \in C$. Indeed, since each g_i is convex, we have that for each i , $g_i(y) + \langle \nabla g_i(y), x^\odot - y \rangle \leq g_i(x^\odot) < 0$, for all $y \in \mathbb{R}^n$. This also shows that x^\odot is a Slater point of the feasible set of (3.1). Now, since P_1 is convex and continuous, we see that $x \mapsto P_1(x) - \langle \xi^k, x - x^k \rangle + \frac{1}{2\beta} \|x - x^k\|^2$ is strongly convex and continuous for any $\beta > 0$. Therefore, (3.1) is a convex optimization problem with a strongly convex objective function over a nonempty closed convex set, and hence, has a unique solution.

(ii): Suppose that $\max_{1 \leq i \leq m} \{g_i(\tilde{u})\} > 0$. By Assumption 3.1, we have $\max_{1 \leq i \leq m} \{g_i(x^\odot)\} < 0$. Let $g(x) = \max_{1 \leq i \leq m} g_i(x)$. Then, by the continuity of g , there exists $\tilde{\tau} \in (0, 1)$ such that $\tilde{x} := (1 - \tilde{\tau})\tilde{u} + \tilde{\tau}x^\odot$ satisfies $g(\tilde{x}) = \max_{1 \leq i \leq m} \{g_i(\tilde{x})\} = 0$.

Since g is convex, we have

$$0 = g(\tilde{x}) = \max_{1 \leq i \leq m} \{g_i(\tilde{x})\} \leq (1 - \tilde{\tau}) \max_{1 \leq i \leq m} \{g_i(\tilde{u})\} + \tilde{\tau} \max_{1 \leq i \leq m} \{g_i(x^\odot)\}.$$

Upon rearranging terms in the above inequality, we see that

$$\begin{aligned} \tilde{\tau} &\stackrel{(a)}{\leq} \frac{\max_{1 \leq i \leq m} \{g_i(\tilde{u})\}}{\max_{1 \leq i \leq m} \{g_i(\tilde{u})\} - \max_{1 \leq i \leq m} \{g_i(x^\odot)\}} \stackrel{(b)}{\leq} \frac{\max_{1 \leq i \leq m} \{g_i(\tilde{u})\}}{-\max_{1 \leq i \leq m} \{g_i(x^\odot)\}} \\ &\stackrel{(c)}{\leq} \frac{\max_{1 \leq i \leq m} \{g_i(x^k) + \langle \nabla g_i(x^k), \tilde{u} - x^k \rangle + (L_{g_i}/2) \|\tilde{u} - x^k\|^2\}}{-\max_{1 \leq i \leq m} \{g_i(x^\odot)\}} \stackrel{(d)}{\leq} \frac{L \|\tilde{u} - x^k\|^2}{-2 \max_{1 \leq i \leq m} \{g_i(x^\odot)\}}, \end{aligned} \quad (3.3)$$

where (a) and (b) hold because $\max_{1 \leq i \leq m} \{g_i(\tilde{u})\} > 0$ and $\max_{1 \leq i \leq m} \{g_i(x^\odot)\} < 0$, (c) is true because ∇g_i is Lipschitz with modulus L_{g_i} on the convex set C that contains both \tilde{u} and x^k , and (d) made use of $L := \max_{1 \leq i \leq m} L_{g_i}$ and the fact that \tilde{u} is feasible for (3.1). Hence,

$$\|\tilde{x} - \tilde{u}\| = \tilde{\tau} \|\tilde{u} - x^\odot\| \stackrel{(a)}{\leq} \tilde{\tau} M \stackrel{(b)}{\leq} \frac{ML}{-2 \max_{1 \leq i \leq m} \{g_i(x^\odot)\}} \|\tilde{u} - x^k\|^2,$$

where (a) holds because $\tilde{u} \in C$ and $M = \sup_{x \in C} \|x - x^\odot\|$, and (b) follows from (3.3).

(iii): Note that we have $\tilde{x} = (1 - \tilde{\tau})\tilde{u} + \tilde{\tau}x^\odot$ for some $\tilde{\tau} \in [0, 1]$, and hence

$$\begin{aligned} P(\tilde{x}) &= P_1(\tilde{x}) - P_2(\tilde{x}) \stackrel{(a)}{\leq} P_1(\tilde{x}) - P_2(x^k) - \langle \xi^k, \tilde{x} - x^k \rangle \\ &\stackrel{(b)}{\leq} (1 - \tilde{\tau})P_1(\tilde{u}) + \tilde{\tau}P_1(x^\odot) - P_2(x^k) - (1 - \tilde{\tau})\langle \xi^k, \tilde{u} - x^k \rangle - \tilde{\tau}\langle \xi^k, x^\odot - x^k \rangle \\ &= P_1(\tilde{u}) - \langle \xi^k, \tilde{u} - x^k \rangle + \tilde{\tau}(P_1(x^\odot) - P_1(\tilde{u}) - \langle \xi^k, x^\odot - \tilde{u} \rangle) - P_2(x^k) \\ &\stackrel{(c)}{\leq} P_1(x^k) - \frac{1}{2\beta} \|\tilde{u} - x^k\|^2 - P_2(x^k) + \tilde{\tau}(P_1(x^\odot) - P_1(\tilde{u}) - \langle \xi^k, x^\odot - \tilde{u} \rangle) \\ &\stackrel{(d)}{\leq} P(x^k) - \frac{1}{2\beta} \|\tilde{u} - x^k\|^2 + \tilde{\tau}M_1 \stackrel{(e)}{\leq} P(x^k) - \left(\frac{1}{2\beta} - \frac{M_1 L}{-2 \max_{1 \leq i \leq m} \{g_i(x^\odot)\}} \right) \|\tilde{u} - x^k\|^2, \end{aligned}$$

where (a) follows from the convexity of P_2 , (b) holds because P_1 is convex, (c) follows from the facts that \tilde{u} is the minimizer of (3.1) and x^k is feasible for (3.1) (since $x^k \in \mathcal{F}$), (d) holds

with $M_1 := \sup_{y, z \in C} \sup_{\xi \in \partial P_2(z)} \{P_1(x^\odot) - P_1(y) - \langle \xi, x^\odot - y \rangle\} \in [0, \infty)$ (note that $M_1 < \infty$ thanks to [42, Theorem 2.6] and the compactness of C , and $M_1 \geq 0$ as $x^\odot \in C$), and (e) follows from (3.2).

Therefore, as long as $\tilde{\beta} < \frac{1}{2} \left(\frac{c}{2} + \frac{M_1 L}{-2 \max_{1 \leq i \leq m} \{g_i(x^\odot)\}} \right)^{-1}$ (independent of k), we have $P(\tilde{x}) \leq P(x^k) - \frac{c}{2} \|\tilde{u} - x^k\|^2$. In view of Step 2b) of Algorithm 1, we conclude that the inner loop terminates finitely. This completes the proof. \square

Remark 3.1 (Uniform lower bound on β_k). *From the proof of Theorem 3.1(iii), we see that for the β_k in Algorithm 1, it holds that*

$$\inf_k \beta_k \geq \beta_{\min} := \min \left\{ \frac{\eta}{2} \left(\frac{c}{2} + \frac{M_1 L}{-2 \max_{1 \leq i \leq m} \{g_i(x^\odot)\}} \right)^{-1}, \underline{\beta} \right\},$$

where η , c and $\underline{\beta}$ are given in Step 0 of Algorithm 1, L is given in Theorem 3.1(ii) and $M_1 = \sup_{y, z \in C} \sup_{\xi \in \partial P_2(z)} \{P_1(x^\odot) - P_1(y) - \langle \xi, x^\odot - y \rangle\}$.

Remark 3.2 (Finding $\tilde{\tau}$). *In Step 2a) of Algorithm 1, when $\max_{1 \leq i \leq m} \{g_i(\tilde{u})\} > 0$, we need to find $\tilde{\tau} \in (0, 1)$ so that $\max_{1 \leq i \leq m} \{g_i((1 - \tilde{\tau})\tilde{u} + \tilde{\tau}x^\odot)\} = 0$; such a $\tilde{\tau}$ exists thanks to Theorem 3.1(ii). In fact, we only need to find $\tilde{\tau}_i \in (0, 1)$ so that $g_i((1 - \tilde{\tau}_i)\tilde{u} + \tilde{\tau}_i x^\odot) = 0$ for each $i \in I := \{i : g_i(\tilde{u}) > 0\}$. Then taking $\tilde{\tau} = \max_{i \in I} \{\tilde{\tau}_i\}$ and letting $\tilde{x} := (1 - \tilde{\tau})\tilde{u} + \tilde{\tau}x^\odot$, it can be verified that $\max_{1 \leq i \leq m} \{g_i(\tilde{x})\} = 0$. Since g_i is convex and smooth, $\tilde{\tau}_i$ can be found efficiently by solving the equation $g_i((1 - t)\tilde{u} + tx^\odot) = 0$ via the Newton's method, starting with $t = 0$. Moreover, in the case where g_i is a quadratic function (e.g., when g_i is the least squares loss function), $\tilde{\tau}_i$ admits a closed form solution.*

Remark 3.3. *Since the Slater condition holds for the convex subproblem (3.1) according to Theorem 3.1(i), from [38, Corollary 28.2.1, Theorem 28.3], we have that for each k , \tilde{u} is a minimizer of (3.1) if and only if there exists $\tilde{\lambda} = (\tilde{\lambda}_1, \dots, \tilde{\lambda}_m) \in \mathbb{R}_+^m$ (which is called a Lagrange multiplier) such that the following conditions hold:*

- (i) $g_i(x^k) + \langle \nabla g_i(x^k), \tilde{u} - x^k \rangle \leq 0$, $i = 1, \dots, m$, and $\tilde{u} \in C$;
- (ii) $\tilde{\lambda}_i(g_i(x^k) + \langle \nabla g_i(x^k), \tilde{u} - x^k \rangle) = 0$, $i = 1, \dots, m$;
- (iii) $0 \in \partial P_1(\tilde{u}) - \xi^k + \tilde{\beta}^{-1}(\tilde{u} - x^k) + \sum_{i=1}^m \tilde{\lambda}_i \nabla g_i(x^k) + \mathcal{N}_C(\tilde{u})$.

We next show that any cluster point of the sequence $\{x^k\}$ generated by Algorithm 1 is a critical point of (1.1) in the sense of Definition 2.3.

Theorem 3.2 (Subsequential convergence of Algorithm 1). *Consider (1.1) and suppose that Assumption 3.1 holds. Let $\{(x^k, u^k)\}$ be the sequence generated by Algorithm 1 and λ^k be a Lagrange multiplier of (3.1) with $\tilde{\beta} = \beta_k$. Then the following statements hold:*

- (i) *It holds that $\lim_{k \rightarrow \infty} \|u^k - x^k\| = 0$.*
- (ii) *The sequences $\{x^k\}$ and $\{\lambda^k\}$ are bounded.*
- (iii) *Any accumulation point of $\{x^k\}$ is a critical point of (1.1).*

Proof. Using Step 2b), we obtain $P(x^{k+1}) \leq P(x^k) - \frac{\varepsilon}{2} \|u^k - x^k\|^2$. Rearranging terms and summing both sides of the inequality from zero to infinity, we have

$$\frac{\varepsilon}{2} \sum_{k=0}^{\infty} \|u^k - x^k\|^2 \leq P(x_0) - \lim_{k \rightarrow \infty} P(x^k) \leq P(x_0) - \inf_{x \in C} P(x) < \infty,$$

where the second inequality holds because $\{x^k\} \subseteq C$ by construction, and the finiteness follows from the compactness of C and the continuity of P . Therefore, $\|u^k - x^k\| \rightarrow 0$. This proves (i).

We now prove (ii). Since $\{x^k\} \subseteq C$ and C is compact, $\{x^k\}$ is bounded. We next show that $\{\lambda^k\}$ is bounded. Suppose to the contrary that $\{\lambda^k\}$ is unbounded and let $\{\lambda^{k_j}\}$ be a subsequence of $\{\lambda^k\}$ such that $\|\lambda^{k_j}\| \rightarrow \infty$. In view of the compactness of C , by passing to a further subsequence if necessary, we may also assume that $\lim_{j \rightarrow \infty} x^{k_j} = x^*$ for some $x^* \in C$, and $\lim_{j \rightarrow \infty} \frac{\lambda^{k_j}}{\|\lambda^{k_j}\|} = \lambda^*$ for some $\lambda^* \in \mathbb{R}_+^m$ with $\|\lambda^*\| = 1$.

From (3.1) and the definition of Lagrange multipliers (see Remark 3.3), we have

$$0 \in \partial P_1(u^{k_j}) - \xi^{k_j} + \frac{1}{\beta_{k_j}}(u^{k_j} - x^{k_j}) + \sum_{i=1}^m \lambda_i^{k_j} \nabla g_i(x^{k_j}) + \mathcal{N}_C(u^{k_j}), \quad (3.4)$$

and

$$\lambda_i^{k_j} (g_i(x^{k_j}) + \langle \nabla g_i(x^{k_j}), u^{k_j} - x^{k_j} \rangle) = 0, \quad i = 1, \dots, m. \quad (3.5)$$

Also, we note from Remark 3.1, the compactness of C and the continuity of P_2 that the sequences $\{x^k\}$, $\{u^k\}$, $\{1/\beta_k\}$, $\{\xi^k\}$ are bounded. Furthermore, by [38, Theorem 24.7], the sets $\{\partial P_1(u^{k_j})\}$ are uniformly bounded thanks to the continuity of P_1 . Now, dividing $\|\lambda^{k_j}\|$ on both sides of (3.4) and (3.5) and passing to the limit as j goes to infinity, we have upon recalling $\lim_{j \rightarrow \infty} x^{k_j} = x^*$ and item (i), and invoking the closedness of \mathcal{N}_C that

$$0 \in \sum_{i=1}^m \lambda_i^* \nabla g_i(x^*) + \mathcal{N}_C(x^*) \quad \text{and} \quad \lambda_i^* g_i(x^*) = 0, \quad i = 1, \dots, m.$$

Since $\lambda^* \in \mathbb{R}_+^m$, the above relation together with Assumption 3.1 (which implies the MFCQ in Definition 2.2) and the fact that $\|\lambda^*\| = 1$ leads to a contradiction. Thus, $\{\lambda^k\}$ is bounded.

Finally, we prove (iii). Suppose \bar{x} is an accumulation point of $\{x^k\}$ with $\lim_{j \rightarrow \infty} x^{k_j} = \bar{x}$ for some convergent subsequence $\{x^{k_j}\}$. Since $\{\lambda^k\}$ is bounded thanks to item (ii), passing to a further subsequence if necessary, we may assume without loss of generality that $\lim_{j \rightarrow \infty} \lambda^{k_j} = \bar{\lambda}$ for some $\bar{\lambda} \in \mathbb{R}_+^m$. Since $\|u^k - x^k\| \rightarrow 0$ according to item (i), we also have $\lim_{j \rightarrow \infty} u^{k_j} = \bar{x}$. Using these together with the closedness of ∂P_1 , ∂P_2 and \mathcal{N}_C , the uniform boundedness of the sequences of sets $\{\partial P_1(u^{k_j})\}$ and $\{\partial P_2(x^{k_j})\}$, the Lipschitz continuity of ∇g and the boundedness of $\{1/\beta_k\}$ (see Remark 3.1), we have upon passing to the limit as j goes to infinity in item (iii) of Remark 3.3 with $(u^{k_j}, x^{k_j}, \xi^{k_j}, \lambda^{k_j})$ in place of $(\tilde{u}, x^k, \xi^k, \tilde{\lambda})$ that

$$0 \in \partial P_1(\bar{x}) - \partial P_2(\bar{x}) + \sum_{i=1}^m \bar{\lambda}_i \nabla g_i(\bar{x}) + \mathcal{N}_C(\bar{x}).$$

Similarly, using item (ii) of Remark 3.3 with $(u^{k_j}, x^{k_j}, \lambda^{k_j})$ in place of $(\tilde{u}, x^k, \tilde{\lambda})$ and the continuity of ∇g , we have upon passing to the limit that

$$\bar{\lambda}_i g_i(\bar{x}) = 0, \quad i = 1, \dots, m.$$

Finally, since for each i , g_i is continuous and $g_i(x^k) \leq 0$ thanks to Step 2a), we have $g_i(\bar{x}) \leq 0$. In addition, since $\{x^k\} \subseteq C$ and C is closed, we have $\bar{x} \in C$. Hence \bar{x} is a critical point of (1.1). This completes the proof. \square

3.1 Convergence rate analysis in convex setting

In this subsection, we study the global convergence of the sequence $\{x_k\}$ generated by Algorithm 1 and its convergence rate, under the following fully convex setting:

Assumption 3.2. Suppose that in (1.1), $P_2 = 0$.

Theorem 3.3 (Convergence rate of Algorithm 1 in convex setting). Consider (1.1) and suppose that Assumptions 3.1 and 3.2 hold. Let $\{x^k\}$ be the sequence generated by Algorithm 1. Then $\{x^k\}$ converges to a minimizer x^* of (1.1). If in addition $x \mapsto P_1(x) + \sum_{i=1}^m \delta_{g_i(\cdot) \leq 0}(x) + \delta_C(x)$ is a KL function with exponent $\alpha \in [0, 1)$, then the following statements hold:

(i) If $\alpha \in [0, \frac{1}{2}]$, then there exist $c_0 > 0$, $s \in (0, 1)$ and $k_0 \in \mathbb{N}_+$, such that

$$\|x^k - x^*\| \leq c_0 s^k \quad \text{for } k > k_0.$$

(ii) If $\alpha \in (\frac{1}{2}, 1)$, then there exist $c_0 > 0$, and $k_0 \in \mathbb{N}_+$, such that

$$\|x^k - x^*\| \leq c_0 k^{-\frac{1-\alpha}{2\alpha-1}} \quad \text{for } k > k_0.$$

Proof. Since C is a compact set and g is continuous, the feasible set \mathcal{F} of (1.1) is compact. Combining this with the fact that P_1 is continuous, we have $S := \underset{x \in \mathcal{F}}{\text{Arg min}} P_1(x) \neq \emptyset$. Let Υ be the set of accumulation point of $\{x^k\}$ for notational simplicity. Using Theorem 3.2 and [38, Theorem 28.3], we see that

$$\emptyset \neq \Upsilon \subseteq S.$$

From Remark 3.3, Assumptions 3.1 and 3.2, and the convexity of P_1 , we see that for each k , there exists $\lambda^k \in \mathbb{R}_+^m$ such that $\lambda_i^k(g_i(x^k) + \langle \nabla g_i(x^k), u^k - x^k \rangle) = 0$ for all $i = 1, \dots, m$ and u^k is a minimizer of the following function:

$$L_k(x) := P_1(x) + \delta_C(x) + \frac{1}{2\beta_k} \|x - x^k\|^2 + \sum_{i=1}^m \lambda_i^k (g_i(x^k) + \langle \nabla g_i(x^k), x - x^k \rangle).$$

Note that $x \mapsto L_k(x)$ is strongly convex with modulus $\frac{1}{\beta_k}$. Then we see that for any $x \in C$,

$$\begin{aligned} P_1(u^k) + \frac{1}{2\beta_k} \|u^k - x^k\|^2 &= L_k(u^k) \leq L_k(x) - \frac{1}{2\beta_k} \|x - u^k\|^2 \\ &= P_1(x) + \frac{1}{2\beta_k} \|x - x^k\|^2 + \sum_{i=1}^m \lambda_i^k (g_i(x^k) + \langle \nabla g_i(x^k), x - x^k \rangle) - \frac{1}{2\beta_k} \|x - u^k\|^2, \end{aligned} \tag{3.6}$$

where the first equality holds because $\lambda_i^k(g_i(x^k) + \langle \nabla g_i(x^k), u^k - x^k \rangle) = 0$ for $i = 1, \dots, m$.

Let $F(x) := P_1(x) + \delta_C(x) + \sum_i \delta_{g_i(\cdot) \leq 0}(x)$ for notational simplicity. For any $\bar{x} \in S$, from Step 2a) and Step 3 of Algorithm 1, we have $\tau_k \in [0, 1]$, $x^{k+1} = (1 - \tau_k)u_k + \tau_k x^\odot \in \mathcal{F}$ and

$$\begin{aligned} F(x^{k+1}) &= P_1(x^{k+1}) \stackrel{(a)}{\leq} P_1(u^k) + \tau_k (P_1(x^\odot) - P_1(u^k)) + \frac{1}{2\beta_k} \|u^k - x^k\|^2 \\ &\stackrel{(b)}{\leq} P_1(\bar{x}) + \frac{1}{2\beta_k} \|\bar{x} - x^k\|^2 + \sum_{i=1}^m \lambda_i^k (g_i(x^k) + \langle \nabla g_i(x^k), \bar{x} - x^k \rangle) - \frac{1}{2\beta_k} \|\bar{x} - u^k\|^2 \\ &\quad + \tau_k (P_1(x^\odot) - P_1(u^k)) \end{aligned}$$

$$\begin{aligned}
&\stackrel{(c)}{\leq} P_1(\bar{x}) + \frac{1}{2\beta_k} \|\bar{x} - x^k\|^2 - \frac{1}{2\beta_k} \|\bar{x} - u^k\|^2 + \tau_k(P_1(x^\odot) - P_1(u^k)) \\
&\stackrel{(d)}{\leq} P_1(\bar{x}) + \frac{1}{2\beta_k} \|\bar{x} - x^k\|^2 - \frac{1}{2\beta_k} \|\bar{x} - x^{k+1}\|^2 + M_2\tau_k + \frac{1}{2\beta_k} (\|\bar{x} - x^{k+1}\|^2 - \|\bar{x} - u^k\|^2), \quad (3.7)
\end{aligned}$$

where (a) holds because P_1 is convex, (b) follows from (3.6) with $x = \bar{x}$, (c) holds because $\lambda_i^k \geq 0$ and $g_i(x^k) + \langle \nabla g_i(x^k), \bar{x} - x^k \rangle \leq g_i(\bar{x}) \leq 0$ (thanks to $\bar{x} \in S \subseteq \mathcal{F}$), (d) holds with

$$M_2 := P_1(x^\odot) - \inf_{x \in C} P_1(x) \in [0, \infty),$$

where the finiteness follows from the continuity of P_1 and the compactness of C .

We now derive an upper bound for (3.7). Notice that

$$\begin{aligned}
\|\bar{x} - x^{k+1}\|^2 - \|\bar{x} - u^k\|^2 &= (\|\bar{x} - x^{k+1}\| - \|\bar{x} - u^k\|) (\|\bar{x} - x^{k+1}\| + \|\bar{x} - u^k\|) \\
&\stackrel{(a)}{\leq} \|x^{k+1} - u^k\| (\|\bar{x} - x^{k+1}\| + \|\bar{x} - u^k\|) \\
&\stackrel{(b)}{=} \tau_k \|x^\odot - u^k\| (\|\bar{x} - x^{k+1}\| + \|\bar{x} - u^k\|) \stackrel{(c)}{\leq} 2M_3^2\tau_k, \quad (3.8)
\end{aligned}$$

where (a) follows from the triangle inequality, (b) holds because $x^{k+1} - u^k = \tau_k(x^\odot - u^k)$ and $\tau_k \geq 0$, (c) holds with $M_3 := \sup_{x, y \in C} \|x - y\| < \infty$, where the finiteness is due to the compactness of C .

Now combining (3.7) and (3.8), we have

$$\begin{aligned}
F(x^{k+1}) &\leq P_1(\bar{x}) + \frac{1}{2\beta_k} \|\bar{x} - x^k\|^2 - \frac{1}{2\beta_k} \|\bar{x} - x^{k+1}\|^2 + \left(M_2 + \frac{M_3^2}{\beta_k}\right) \tau_k \\
&\stackrel{(a)}{\leq} P_1(\bar{x}) + \frac{1}{2\beta_k} \|\bar{x} - x^k\|^2 - \frac{1}{2\beta_k} \|\bar{x} - x^{k+1}\|^2 + \left(M_2 + \frac{M_3^2}{\beta_k}\right) \frac{L\|u^k - x^k\|^2}{-2 \max_{1 \leq i \leq m} \{g_i(x^\odot)\}} \\
&\stackrel{(b)}{\leq} P_1(\bar{x}) + \frac{1}{2\beta_k} \|\bar{x} - x^k\|^2 - \frac{1}{2\beta_k} \|\bar{x} - x^{k+1}\|^2 + M_4(P_1(x^k) - P_1(x^{k+1})),
\end{aligned}$$

where (a) follows from (3.2), (b) holds because $\|u^k - x^k\|^2 \leq \frac{2}{c}(P(x^k) - P(x^{k+1}))$ from Step 2b) of Algorithm 1 and we define $M_4 := \left(M_2 + \frac{M_3^2}{\beta_{\min}}\right) \frac{L}{-c \max_{1 \leq i \leq m} \{g_i(x^\odot)\}} \in (0, \infty)$, with β_{\min} given in Remark 3.1.

Rearranging terms in the above inequality and noting $\bar{F}^* := \inf F = P_1(\bar{x})$ whenever $\bar{x} \in S$, we have

$$\beta_k(F(x^{k+1}) - \bar{F}^*) \leq \frac{1}{2} \|\bar{x} - x^k\|^2 - \frac{1}{2} \|\bar{x} - x^{k+1}\|^2 + M_4\beta_k(F(x^k) - F(x^{k+1})).$$

Recall that $\beta_{\min} \leq \beta_k \leq \bar{\beta}$, where β_{\min} is defined in Remark 3.1 and $\bar{\beta}$ is a parameter in Step 0 of Algorithm 1. Then for any $\bar{x} \in S$, we have

$$\beta_{\min}(F(x^{k+1}) - \bar{F}^*) \leq \frac{1}{2} \|\bar{x} - x^k\|^2 - \frac{1}{2} \|\bar{x} - x^{k+1}\|^2 + M_4\bar{\beta}(F(x^k) - F(x^{k+1})).$$

Rearranging terms in the above inequality, we have

$$(\beta_{\min} + M_4\bar{\beta})(F(x^{k+1}) - \bar{F}^*) \leq \frac{1}{2} \|\bar{x} - x^k\|^2 - \frac{1}{2} \|\bar{x} - x^{k+1}\|^2 + M_4\bar{\beta}(F(x^k) - \bar{F}^*).$$

The desired results now follow from the same arguments as in the proof of [48, Theorem 3.12], starting from [48, Eq. (3.16)]. This completes the proof. \square

4 Extension: Structured non-convex constraint with strong strict feasibility

In this section, we consider DC optimization problem with structured nonconvex inequality constraint and geometric constraint. As we will see later, our setting covers, in particular, several important model problems such as the DC optimization problem with Lorentzian norm and box constraints. In order to simplify notation, we only consider the case where one single nonconvex inequality constraint is present in (1.1).

More explicitly, we consider the following special case of (1.1):

$$\begin{aligned} \min_{x \in \mathbb{R}^n} \quad & P(x) := P_1(x) - P_2(x) \\ \text{s.t.} \quad & \ell(Ax - b) \leq \sigma, \\ & x \in C, \end{aligned} \tag{4.1}$$

where P_1 and P_2 are finite-valued convex functions, C is a nonempty compact convex set, $A \in \mathbb{R}^{p \times n}$, $b \in \mathbb{R}^p$, $\sigma > 0$, the set $\mathcal{F}_1 := \{x \in C : \ell(Ax - b) \leq \sigma\} \neq \emptyset$, and ℓ also satisfies the following Assumption 4.1:

Assumption 4.1. *The function $\ell : \mathbb{R}^p \rightarrow \mathbb{R}$ is defined as*

$$\ell(u) := \sum_{i=1}^p \varphi(u_i^2),$$

where $\varphi : \mathbb{R}_+ \rightarrow \mathbb{R}_+$ satisfies the following conditions:

- (i) φ is a continuous concave function with $\varphi(0) = 0$;
- (ii) φ is continuously differentiable with positive derivative on $(0, \infty)$;
- (iii) The limit $\vartheta := \lim_{t \downarrow 0} \varphi'(t)$ exists, and the function $t \mapsto \varphi'_+(t)$ is locally Lipschitz on $[0, \infty)$.²

We present below some concrete examples of φ that satisfies the conditions in Assumption 4.1.

Example 4.1. *As in [3], for a given $a > 0$, we consider $\varphi : \mathbb{R}_+ \rightarrow \mathbb{R}_+$ given as follows:*

$$\varphi(u) = \frac{(a+1)u}{a+u}.$$

One can see that φ is a continuous concave function and is differentiable with positive derivative on $(0, \infty)$ as its first and second derivatives exist:

$$\varphi'(u) = \frac{a(a+1)}{(a+u)^2} \quad \text{and} \quad \varphi''(u) = -\frac{2a(a+1)}{(a+u)^3}.$$

From this we see further that φ' is Lipschitz continuous on $[0, \infty)$ with modulus $\frac{2(a+1)}{a^2}$. Thus φ satisfies the conditions in Assumption 4.1.

Example 4.2. *We consider the function $\varphi : \mathbb{R}_+ \rightarrow \mathbb{R}$ defined as follows for a given γ [16]:*

$$\varphi(u) = \log(u + \gamma^2) - \log(\gamma^2).$$

²Here, and throughout this paper, we use $\varphi'_+(t)$ to denote the right hand derivative of φ at t , which is defined as $\varphi'_+(t) = \lim_{h \rightarrow 0^+} \frac{\varphi(t+h) - \varphi(t)}{h}$.

One can see that φ satisfies the conditions in Assumption 4.1 and φ' is Lipschitz continuous on $[0, \infty)$ with modulus $\frac{1}{\gamma^4}$. In this case, we have

$$\ell(u) = \sum_{i=1}^p \log \left(1 + \frac{u_i^2}{\gamma^2} \right),$$

which is the Lorentzian norm: This function is used as the loss function for measurement noise following the Cauchy distribution; see [17, 18].

Recall that the Slater condition (i.e., Assumption 3.1) was instrumental in deriving the first-order optimality conditions of (1.1) and analyzing convergence of Algorithm 1. Here, for analyzing (4.1) and developing an analogue of Algorithm 1 for solving it, we will consider the following stronger strict feasibility assumption on the feasible set \mathcal{F}_1 of (4.1).

Assumption 4.2. *There exists $x^\circ \in C$ such that $Ax^\circ = b$ and x° is explicitly known.*

Before presenting our algorithm for solving (4.1) under Assumptions 4.1 and 4.2, to simplify the presentation of the algorithm, we first introduce some auxiliary notation.

Consider (4.1) and suppose Assumption 4.1 holds. For any $y \in \mathbb{R}^n$, let

$$\omega_i^y := \varphi'_+((a_i^T y - b_i)^2), \quad i = 1, \dots, p, \quad (4.2)$$

$$\tilde{\sigma}^y := \sigma - \ell(Ay - b) + \sum_{i=1}^p \omega_i^y (a_i^T y - b_i)^2, \quad (4.3)$$

and define the function $\ell^y : \mathbb{R}^p \rightarrow \mathbb{R}$ by

$$\ell^y(u) := \sum_{i=1}^p \omega_i^y u_i^2, \quad (4.4)$$

where A and b are as in (4.1) and a_i^T denotes the i th row of A . Note that under Assumption 4.1, the function φ is concave. Hence, for any $x, y \in \mathbb{R}^n$ and each $i = 1, \dots, p$, we have from the super-gradient inequality that

$$\varphi((a_i^T x - b_i)^2) \leq \varphi((a_i^T y - b_i)^2) + \omega_i^y [(a_i^T x - b_i)^2 - (a_i^T y - b_i)^2],$$

Summing both sides of the above inequality from $i = 1$ to p , we obtain that

$$\begin{aligned} \ell(Ax - b) &\leq \ell(Ay - b) + \sum_{i=1}^p \omega_i^y [(a_i^T x - b_i)^2 - (a_i^T y - b_i)^2] \\ &= \ell(Ay - b) + \ell^y(Ax - b) - \ell^y(Ay - b). \end{aligned} \quad (4.5)$$

We are now ready to present our algorithm for solving (4.1) under Assumptions 4.1 and 4.2, which is stated as Algorithm 2 below. This is again an SQP-type method and is also motivated by the retraction strategy. Indeed, the subproblem (4.6) takes the same form as the subproblem (3.1) in Algorithm 1, with $\ell(Ax - b)$ in place of $g(x)$. The key difference lies in Step 2a). In Algorithm 1, we “retract” by taking convex combinations so that \tilde{x} satisfies the convex constraint $\{x \in C : g_i(x) \leq 0, \quad i = 1, \dots, m\}$. However, this approach cannot be directly adapted for (4.1), because $\{x \in C : \ell(Ax - b) \leq \sigma\}$ is nonconvex. Fortunately, one can deduce from (4.2), (4.3), (4.4) and (4.5) that $\ell(Ax - b) - \sigma \leq \ell^y(Ax - b) - \tilde{\sigma}^y$ (see Lemma 4.1(v) below); moreover, the function ℓ^y is always convex (see Lemma 4.1(ii) below). Thus, in Step 2a) of our Algorithm 2, we “retract” so that \tilde{x} satisfies the *convex* constraint $\mathcal{F}_1^y := \{x \in C : \ell^y(Ax - b) \leq \tilde{\sigma}^y\}$ for some suitable $y \in \mathbb{R}^n$. Since ℓ^y is always nonnegative, for this retraction strategy to make sense, we have to at least show that $\tilde{\sigma}^y > 0$ at each iteration. We prove this together with some auxiliary facts concerning ℓ^y in the next lemma.

Algorithm 2 FPA with retraction for (4.1) under Assumptions 4.1 and 4.2

Step 0. Choose $x^0 \in \mathcal{F}_1$, $0 < \underline{\beta} < \bar{\beta}$, $c > 0$ and $\eta \in (0, 1)$. Choose x^\odot as in Assumption 4.2 and set $k = 0$.

Step 1. Pick any $\xi^k \in \partial P_2(x^k)$. Compute $\omega_i^{x^k}$ as in (4.2) and $\tilde{\sigma}^{x^k}$ as in (4.3). Choose $\beta_k^0 \in [\underline{\beta}, \bar{\beta}]$ arbitrarily and set $\tilde{\beta} = \beta_k^0$.

Step 2. Compute

$$\begin{aligned} \tilde{u} = \arg \min_{x \in \mathbb{R}^n} \quad & P_1(x) - \langle \xi^k, x - x^k \rangle + (2\tilde{\beta})^{-1} \|x - x^k\|^2 \\ \text{s.t.} \quad & \ell(Ax^k - b) + \langle A^T \nabla \ell(Ax^k - b), x - x^k \rangle \leq \sigma, \\ & x \in C. \end{aligned} \quad (4.6)$$

Step 2a) If $\ell^{x^k}(A\tilde{u} - b) \leq \tilde{\sigma}^{x^k}$, set $\tilde{x} = \tilde{u}$ and $\tilde{\tau} = 0$; else, find $\tilde{\tau} \in (0, 1)$ such that $\tilde{x} := (1 - \tilde{\tau})\tilde{u} + \tilde{\tau}x^\odot$ satisfies $\ell^{x^k}(A\tilde{x} - b) = \tilde{\sigma}^{x^k}$.

Step 2b) If $P(\tilde{x}) \leq P(x^k) - \frac{c}{2} \|\tilde{u} - x^k\|^2$, go to **Step 3**; else, update $\tilde{\beta} \leftarrow \eta \tilde{\beta}$ and go to **Step 2**.

Step 3. Set $x^{k+1} = \tilde{x}$, $u^k = \tilde{u}$, $\beta_k = \tilde{\beta}$, $\tau_k = \tilde{\tau}$. Update $k \leftarrow k + 1$ and go to **Step 1**.

Lemma 4.1. Consider (4.1) and let \mathcal{F}_1 denote its feasible set. Assume that Assumption 4.1 holds. Let $y \in \mathbb{R}^n$, and let $\tilde{\sigma}^y$ and ℓ^y be defined in (4.2) and (4.4) respectively. Then the following statements hold:

- (i) It holds that $\tilde{\sigma}_{\min} := \inf_{v \in \mathcal{F}_1} \tilde{\sigma}^v > 0$.
- (ii) $\ell^y(0) = 0$ and ℓ^y is convex.
- (iii) ℓ^y is smooth and there exists $L_\ell > 0$ (independent of y)³ such that for all $u, v \in \mathbb{R}^p$, we have
$$\|\nabla \ell^y(u) - \nabla \ell^y(v)\| \leq L_\ell \|u - v\|. \quad (4.7)$$
- (iv) $\ell(Ay - b) - \sigma = \ell^y(Ay - b) - \tilde{\sigma}^y$, and $\nabla \ell(Ay - b) = \nabla \ell^y(Ay - b)$.
- (v) $\ell(Ax - b) - \sigma \leq \ell^y(Ax - b) - \tilde{\sigma}^y$ for any $x \in \mathbb{R}^n$.

Proof. (i): Let $v \in \mathcal{F}_1$. Since $\varphi : \mathbb{R}_+ \rightarrow \mathbb{R}_+$ is a continuous concave function with $\varphi(0) = 0$ that is continuously differentiable with positive derivative on $(0, \infty)$, we have $\omega_i^v > 0$ for all $i \notin I_0$, where $I_0 = \{i : a_i^T v - b_i = 0\}$. Next, note that $\ell(Av - b) \leq \sigma$. In addition, if v satisfies $\ell(Av - b) = \sigma$, then there must exist i_0 such that $a_{i_0}^T v - b_{i_0} \neq 0$ (since $\ell(0) = 0 < \sigma$), hence

$$\tilde{\sigma}^v = \sigma - \ell(Av - b) + \sum_{i \notin I_0} \omega_i^v (a_i^T v - b_i)^2 > 0.$$

Finally, from (4.3), we have $z \mapsto \tilde{\sigma}^z$ is continuous. This together with the compactness of C shows that $\tilde{\sigma}_{\min} := \min_{v \in \mathcal{F}_1} \tilde{\sigma}^v > 0$.

(ii): From (4.4), one can see that $\ell^y(0) = 0$. The convexity of ℓ^y follows from (4.2) and Assumption 4.1.

³Indeed, one can take $L_\ell = 2\vartheta$ with ϑ as in Assumption 4.1.

(iii): By the definition of ℓ^y , we have $\nabla \ell^y(u) = (2\omega_i^y u_i)_{i=1}^p$, and $\nabla^2 \ell^y(u) = 2\text{Diag}(\omega^y)$.⁴ Now, according to [47, Lemma 2.2], for any y , we have that $0 \leq \omega_i^y \leq \lim_{t \downarrow 0} \varphi'(t) = \vartheta$. This implies that (4.7) holds with $L_\ell = 2\vartheta$.

(iv): The first equation follows from (4.3) and the definition of ℓ^y . As for the second equation, note that for any $y \in \mathbb{R}^n$, using the definitions of ℓ and ℓ^y , we have $\nabla \ell(u) = (2\varphi'_+(u_i^2)u_i)_{i=1}^p$ and $\nabla \ell^y(u) = (2\varphi'_+((a_i^T y - b_i)^2)u_i)_{i=1}^p$. Therefore,

$$\nabla \ell(Ay - b) = (2\varphi'_+((a_i^T y - b_i)^2)(a_i^T y - b_i))_{i=1}^p = \nabla \ell^y(Ay - b).$$

(v): From (4.5), we see that $\ell(Ax - b) - \ell^y(Ax - b) \leq \ell(Ay - b) - \ell^y(Ay - b)$ for any $x \in \mathbb{R}^n$. Combining this with item (iv), we deduce that $\ell(Ax - b) - \sigma \leq \ell^y(Ax - b) - \tilde{\sigma}^y$ for any $x \in \mathbb{R}^n$. This completes the proof. \square

Now, we prove the well-definedness of Algorithm 2. Recall that \mathcal{F}_1 denotes the feasible set of (4.1). To prove well-definedness, it suffices to show that, if an $x^k \in \mathcal{F}_1$ for some $k \geq 0$, then the subproblem (4.6) has a unique solution, $\tilde{\tau}$ exists in Step 2a) and the line-search step in Step 2b) terminates in finitely many inner iterations so that x^{k+1} can be generated: Note that such an x^{k+1} belongs to \mathcal{F}_1 in view of Step 2a) and Lemma 4.1(v). This together with an induction argument would establish well-definedness.

Theorem 4.1 (Well-definedness of Algorithm 2). *Consider (4.1) and suppose that Assumptions 4.1 and 4.2 hold. Suppose that an $x^k \in \mathcal{F}_1$ is generated at the beginning of the k -th iteration of Algorithm 2 for some $k \geq 0$. Then the following statements hold:*

- (i) *The convex subproblem (4.6) has a unique solution. Moreover, x^\odot from Assumption 4.2 is a Slater point of the feasible set of (4.6).*
- (ii) *If $\ell^{x^k}(A\tilde{u} - b) > \tilde{\sigma}^{x^k}$, then there exists $\tilde{\tau} \in (0, 1)$ such that $\tilde{x} := (1 - \tilde{\tau})\tilde{u} + \tilde{\tau}x^\odot$ with $\ell^{x^k}(A\tilde{x} - b) = \tilde{\sigma}^{x^k}$. Moreover, for any such $\tilde{\tau}$, it holds that*

$$\tilde{\tau} \leq \frac{L_\ell \|A\|^2}{2\tilde{\sigma}_{\min}} \|\tilde{u} - x^k\|^2 \quad \text{and} \quad \|\tilde{x} - \tilde{u}\| \leq \frac{ML_\ell \|A\|^2}{2\tilde{\sigma}_{\min}} \|\tilde{u} - x^k\|^2,$$

where $M := \sup_{x \in C} \|x - x^\odot\|$, and $\tilde{\sigma}_{\min}$ and L_ℓ are defined as in Lemma 4.1.

- (iii) *Step 2 of algorithm 2 terminates in finitely many inner iterations.*

- (iv) *For each $\tilde{\beta} > 0$, the subproblem (4.6) has a Lagrange multiplier $\tilde{\lambda} \geq 0$ and it satisfies*

$$0 \in \partial P_1(\tilde{u}) - \xi^k + \frac{1}{\tilde{\beta}}(\tilde{u} - x^k) + \tilde{\lambda} A^T \nabla \ell(Ax^k - b) + \mathcal{N}_C(\tilde{u}),$$

and

$$\tilde{\lambda}(\ell(Ax^k - b) + \langle A^T \nabla \ell(Ax^k - b), \tilde{u} - x^k \rangle - \sigma) = 0.$$

In particular, an $x^{k+1} \in \mathcal{F}_1$ can be generated at the end of the $(k+1)$ -th iteration of Algorithm 2.

Proof. (i): We first observe that the feasible region of (4.6) is closed and convex. We next show that it is nonempty by showing that x^\odot is a Slater point of the feasible set of (4.6). To this end, we note from the convexity of ℓ^{x^k} (see Lemma 4.1(ii)) that

$$\ell^{x^k}(Ax^k - b) + \langle A^T \nabla \ell^{x^k}(Ax^k - b), x^\odot - x^k \rangle \leq \ell^{x^k}(Ax^\odot - b) \stackrel{(a)}{=} \ell^{x^k}(0) \stackrel{(b)}{=} 0 \stackrel{(c)}{<} \tilde{\sigma}^{x^k}, \quad (4.8)$$

⁴Here, $\text{Diag}(\omega^y)$ is the diagonal matrix whose i -th diagonal entry is ω_i^y .

where (a) follows from Assumption 4.2, (b) follows from Lemma 4.1(ii), and (c) holds because of Lemma 4.1(i) and the fact that $x^k \in \mathcal{F}_1$. Hence,

$$\begin{aligned} & \ell(Ax^k - b) + \langle A^T \nabla \ell(Ax^k - b), x^\odot - x^k \rangle \\ &= \sigma + \ell^{x^k}(Ax^k - b) - \tilde{\sigma}^{x^k} + \langle A^T \nabla \ell^{x^k}(Ax^k - b), x^\odot - x^k \rangle < \sigma, \end{aligned}$$

where the equality follows from Lemma 4.1(iv) and the inequality holds thanks to (4.8). Since we also have $x^\odot \in C$ by assumption, we have shown that x^\odot is a Slater point for (4.6). Next, since P_1 is convex and continuous, it holds that $x \mapsto P_1(x) - \langle \xi^k, x - x^k \rangle + (2\tilde{\beta})^{-1} \|x - x^k\|^2$ is strongly convex and continuous, for any $\tilde{\beta} > 0$. Therefore, (4.6) has a unique solution.

(ii): Suppose that $\ell^{x^k}(A\tilde{u} - b) > \tilde{\sigma}^{x^k}$. By Assumption 4.2, Lemma 4.1(i) and (ii), and the fact that $x^k \in \mathcal{F}_1$, we have

$$\ell^{x^k}(Ax^\odot - b) - \tilde{\sigma}^{x^k} = -\tilde{\sigma}^{x^k} < 0. \quad (4.9)$$

Letting $x_\tau := (1 - \tau)\tilde{u} + \tau x^\odot$, then $\ell^{x^k}(Ax_1 - b) - \tilde{\sigma}^{x^k} < 0$ and $\ell^{x^k}(Ax_0 - b) - \tilde{\sigma}^{x^k} > 0$. Hence, there exists $\tilde{\tau} \in (0, 1)$ such that $\tilde{x} := (1 - \tilde{\tau})\tilde{u} + \tilde{\tau}x^\odot$ satisfies $\ell^{x^k}(A\tilde{x} - b) - \tilde{\sigma}^{x^k} = 0$.

Now, recall from Lemma 4.1(ii) that ℓ^{x^k} is convex and thus

$$0 = \ell^{x^k}(A\tilde{x} - b) - \tilde{\sigma}^{x^k} \leq (1 - \tilde{\tau}) \left(\ell^{x^k}(A\tilde{u} - b) - \tilde{\sigma}^{x^k} \right) + \tilde{\tau} \left(\ell^{x^k}(Ax^\odot - b) - \tilde{\sigma}^{x^k} \right).$$

Upon rearranging terms in the above inequality, we see that

$$\begin{aligned} \tilde{\tau} &\stackrel{(a)}{\leq} \frac{\ell^{x^k}(A\tilde{u} - b) - \tilde{\sigma}^{x^k}}{(\ell^{x^k}(A\tilde{u} - b) - \tilde{\sigma}^{x^k}) + \tilde{\sigma}^{x^k}} \stackrel{(b)}{\leq} \frac{\ell^{x^k}(A\tilde{u} - b) - \tilde{\sigma}^{x^k}}{\tilde{\sigma}^{x^k}} \\ &\stackrel{(c)}{\leq} \frac{\ell^{x^k}(Ax^k - b) + \langle A^T \nabla \ell^{x^k}(Ax^k - b), \tilde{u} - x^k \rangle - \tilde{\sigma}^{x^k} + (L_\ell \|A\|^2/2) \|\tilde{u} - x^k\|^2}{\tilde{\sigma}^{x^k}} \\ &\stackrel{(d)}{\leq} \frac{L_\ell \|A\|^2}{2\tilde{\sigma}_{\min}} \|\tilde{u} - x^k\|^2, \end{aligned} \quad (4.10)$$

where (a) and (b) hold thanks to $\ell^{x^k}(A\tilde{u} - b) - \tilde{\sigma}^{x^k} > 0$ and (4.9), (c) holds because ℓ^{x^k} is smooth with L_ℓ -Lipschitz gradient (see Lemma 4.1(iii)), and (d) holds in view of Lemma 4.1(iv) and the feasibility of \tilde{u} for (4.6), and the fact that $\tilde{\sigma}^{x^k} \geq \tilde{\sigma}_{\min} > 0$ (thanks to Lemma 4.1(i) and $x^k \in \mathcal{F}_1$). Hence,

$$\|\tilde{x} - \tilde{u}\| = \tilde{\tau} \|\tilde{u} - x^\odot\| \stackrel{(a)}{\leq} \tilde{\tau} M \stackrel{(b)}{\leq} \frac{ML_\ell \|A\|^2}{2\tilde{\sigma}_{\min}} \|\tilde{u} - x^k\|^2,$$

where $M := \sup_{x \in C} \|x - x^\odot\|$ and (a) holds because $\tilde{u} \in C$, and (b) follows from (4.10).

(iii): Note that $\tilde{x} = (1 - \tilde{\tau})\tilde{u} + \tilde{\tau}x^\odot$ for some $\tilde{\tau} \in [0, 1]$ and thus

$$\begin{aligned} P(\tilde{x}) &= P_1(\tilde{x}) - P_2(\tilde{x}) \stackrel{(a)}{\leq} P_1(\tilde{x}) - P_2(x^k) - \langle \xi^k, \tilde{x} - x^k \rangle \\ &\stackrel{(b)}{\leq} (1 - \tilde{\tau})P_1(\tilde{u}) + \tilde{\tau}P_1(x^\odot) - P_2(x^k) - (1 - \tilde{\tau})\langle \xi^k, \tilde{u} - x^k \rangle - \tilde{\tau}\langle \xi^k, x^\odot - x^k \rangle \\ &= P_1(\tilde{u}) - \langle \xi^k, \tilde{u} - x^k \rangle + \tilde{\tau}(P_1(x^\odot) - P_1(\tilde{u}) - \langle \xi^k, x^\odot - \tilde{u} \rangle) - P_2(x^k) \\ &\stackrel{(c)}{\leq} P_1(x^k) - \frac{1}{2\tilde{\beta}} \|\tilde{u} - x^k\|^2 - P_2(x^k) + \tilde{\tau}(P_1(x^\odot) - P_1(\tilde{u}) - \langle \xi^k, x^\odot - \tilde{u} \rangle) \\ &\stackrel{(d)}{\leq} P(x^k) - \frac{1}{2\tilde{\beta}} \|\tilde{u} - x^k\|^2 + \tilde{\tau}M_1 \stackrel{(e)}{\leq} P(x^k) - \left(\frac{1}{2\tilde{\beta}} - \frac{M_1 L_\ell \|A\|^2}{2\tilde{\sigma}_{\min}} \right) \|\tilde{u} - x^k\|^2, \end{aligned}$$

where (a) follows from the convexity of P_2 , (b) holds because P_1 is convex, (c) follows from the facts that \tilde{u} is the minimizer of (4.6) and x^k is feasible for (4.6) (since $x^k \in \mathcal{F}_1$), (d) holds

with $M_1 := \sup_{y, z \in C} \sup_{\xi \in \partial P_2(z)} \{P_1(x^\odot) - P_1(y) - \langle \xi, x^\odot - y \rangle\} \in [0, \infty)$ (note that $M_1 < \infty$ thanks to [42, Theorem 2.6] and the compactness of C , and $M_1 \geq 0$ since $x^\odot \in C$), (e) follows from item (ii).

Therefore, as long as $\tilde{\beta} < \frac{1}{2} \left(\frac{c}{2} + \frac{M_1 L_\ell \|A\|^2}{2\tilde{\sigma}_{\min}} \right)^{-1}$ (independent of k), we have $P(\tilde{x}) \leq P(x^k) - \frac{c}{2} \|\tilde{u} - x^k\|^2$. In view of Step 2b) of Algorithm 1, we conclude that the inner loop terminates finitely.

(iv): Since (4.6) is convex and the Slater condition holds by item (i), the desired conclusion follows from [38, Corollary 28.2.1] and [38, Theorem 28.3]. \square

Remark 4.1 (Uniform lower bound on β_k). From the proof of Theorem 4.1(iii), we see that for the β_k in Algorithm 2, it holds that

$$\inf_k \beta_k \geq \hat{\beta}_{\min} := \min \left\{ \frac{\eta}{2} \left(\frac{c}{2} + \frac{M_1 L_\ell \|A\|^2}{2\tilde{\sigma}_{\min}} \right)^{-1}, \underline{\beta} \right\},$$

where η , c and $\underline{\beta}$ are given in Step 0 of Algorithm 2, $\tilde{\sigma}_{\min}$ and L_ℓ are defined as in Lemma 4.1, and $M_1 := \sup_{y, z \in C} \sup_{\xi \in \partial P_2(z)} \{P_1(x^\odot) - P_1(y) - \langle \xi, x^\odot - y \rangle\}$.

Remark 4.2 (Finding $\tilde{\tau}$). In Step 2a) of Algorithm 2, when $\ell^{x^k}(A\tilde{u} - b) > \tilde{\sigma}^{x^k}$, we need to find $\tilde{\tau}$ so that $\ell^{x^k}(A((1 - \tilde{\tau})\tilde{u} + \tilde{\tau}x^\odot) - b) = \tilde{\sigma}^{x^k}$. We note that such a parameter $\tilde{\tau}$ exists thanks to Theorem 4.1(ii), and admits a closed form formula since ℓ^{x^k} is a quadratic function.

Before presenting the convergence result of Algorithm 2, we show that the MFCQ holds for (4.1), under Assumptions 4.1 and 4.2.

Proposition 4.1. Consider (4.1) and suppose that Assumptions 4.1 and 4.2 hold. Then the MFCQ for (4.1) holds at any $y \in \mathcal{F}_1$, i.e., for any $y \in \mathcal{F}_1$, the following implication holds:

$$\left. \begin{array}{l} -\lambda A^T \nabla \ell(Ay - b) \in \mathcal{N}_C(y) \\ \lambda(\ell(Ay - b) - \sigma) = 0 \\ \lambda \geq 0 \end{array} \right\} \Rightarrow \lambda = 0.$$

Proof. Suppose that $-\lambda A^T \nabla \ell(Ay - b) \in \mathcal{N}_C(y)$ and $\lambda(\ell(Ay - b) - \sigma) = 0$ but $\lambda > 0$. We will derive a contradiction.

To this end, recall from Lemma 4.1(iv) that $\ell(Ay - b) - \sigma = \ell^y(Ay - b) - \tilde{\sigma}^y$ and $\nabla \ell(Ay - b) = \nabla \ell^y(Ay - b)$. Thus, we have

$$-\lambda A^T \nabla \ell^y(Ay - b) \in \mathcal{N}_C(y) \quad \text{and} \quad \lambda(\ell^y(Ay - b) - \tilde{\sigma}^y) = 0. \quad (4.11)$$

Using the second relation above, the convexity of ℓ^y (see Lemma 4.1(ii)) and the assumption that $\lambda > 0$, we have

$$\langle \lambda A^T \nabla \ell^y(Ay - b), x^\odot - y \rangle \leq \lambda \ell^y(Ax^\odot - b) - \lambda \ell^y(Ay - b) = \lambda \ell^y(Ax^\odot - b) - \lambda \tilde{\sigma}^y = -\lambda \tilde{\sigma}^y < 0,$$

where the second equality follows from Assumption 4.2 and Lemma 4.1(ii), and the strict inequality holds in view of $\lambda > 0$, $y \in \mathcal{F}_1$ and Lemma 4.1(i). On the other hand, the first relation in (4.11) shows that $\langle -\lambda A^T \nabla \ell^y(Ay - b), x - y \rangle \leq 0$ for any $x \in C$, which contradicts the above display because $x^\odot \in C$. Thus, we must have $\lambda = 0$. This completes the proof. \square

We now present the main convergence result of Algorithm 2. The proof is almost identical to that of Theorem 3.2, except that Theorem 4.1(iv) is used in place of Remark 3.3, and Remark 4.1 is used in place of Remark 3.1. We omit its proof for brevity.

Theorem 4.2 (Subsequential convergence of Algorithm 2). *Consider (4.1) and suppose that Assumptions 4.1 and 4.2 hold. Let $\{(x^k, u^k)\}$ be the sequence generated by Algorithm 2 and λ_k be a Lagrange multiplier of (4.6) with $\beta = \beta_k$. Then the following statements hold:*

- (i) *It holds that $\lim_{k \rightarrow \infty} \|u^k - x^k\| = 0$.*
- (ii) *The sequences $\{x_k\}$ and $\{\lambda_k\}$ are bounded.*
- (iii) *Any accumulation point of $\{x^k\}$ is a critical point of (4.1).*

5 Explicit Kurdyka-Łojasiewicz exponent for some structured convex optimization models

In this section, we examine the KL exponent for some structured convex optimization model that arises from applications. Specifically, we consider the following constrained optimization problem:

$$\min_{x \in \mathbb{R}^n} F(x) := h(x) + \delta_{G(\cdot) \leq 0}(x), \quad (5.1)$$

where h is proper, closed and convex, the function $G(x) = (q_1(A_1x), \dots, q_m(A_mx))$ is continuous with each $q_i : \mathbb{R}^{s_i} \rightarrow \mathbb{R}$ being strictly convex and each $A_i \in \mathbb{R}^{s_i \times n}$, and $\{x \in \text{dom } h : G(x) \leq 0\} \neq \emptyset$. One can see that, when q_i 's are additionally Lipschitz differentiable and $h = P_1 + \delta_C$ for some continuous convex function P_1 and nonempty compact convex set C , problem (5.1) is a special case of (1.1) with $P_2 = 0$ and $g_i = q_i \circ A_i$ for $i = 1, \dots, m$.

The proof of the following theorem is similar to that of [48, Theorem 4.2]. We omit its proof for brevity.

Theorem 5.1. *Let F be as in (5.1) and $\bar{x} \in \text{Argmin } F$. Suppose the following conditions hold:*

- (i) *There exists a Lagrange multiplier $\bar{\lambda} \in \mathbb{R}_+^m$ for (5.1) and $x \mapsto h(x) + \langle \bar{\lambda}, G(x) \rangle$ is a KL function with exponent $\alpha \in (0, 1)$.*
- (ii) *The strict complementarity condition holds at $(\bar{x}, \bar{\lambda})$, i.e., for any i satisfying $\bar{\lambda}_i = 0$, it holds that $q_i(A_i\bar{x}) < 0$.*

Then F satisfies the KL property with exponent α at \bar{x} .

The next corollary deals with (5.1) with $m = 1$. Its proof follows a similar line of arguments as the proof of [48, Corollary 4.3]. We omit the proof for brevity.

Corollary 5.1. *Let F be as in (5.1) with $m = 1$. Suppose the following conditions hold:*

- (i) $\inf_{x \in \mathbb{R}^n} h(x) < \inf_{x \in \mathbb{R}^n} \{h(x) : q_1(A_1x) \leq 0\}$.
- (ii) *There exists a Lagrange multiplier $\bar{\lambda} \in \mathbb{R}_+$ for (5.1) and $x \mapsto h(x) + \bar{\lambda}q_1(A_1x)$ is a KL function with exponent $\alpha \in (0, 1)$.*

Then F is KL function with exponent α .

Theorem 5.1 and Corollary 5.1 give sufficient conditions for us to derive the KL exponent of F in (5.1) from that of its Lagrangian. In the following subsections, we will make use of Corollary 5.1 to derive the KL exponent of (5.1) with $m = 1$ and specific choices of h . We will need the following assumption, which is a commonly used sufficient condition for the existence of Lagrange multipliers.

Assumption 5.1. *For (5.1) with $m = 1$, there exists $x^\odot \in \text{dom } h$ with $q_1(A_1x^\odot) < 0$.*

5.1 Case 1: h is the sum of the ℓ_1 norm and the indicator function of a polyhedron containing the origin

Suppose that $h = \|\cdot\|_1 + \delta_C$ in (5.1), where C is polyhedron containing the origin. We deduce from [33, Corollary 5.1] and Corollary 5.1 that the KL exponent of the corresponding F in (5.1) is $\frac{1}{2}$ if $m = 1$, $q_1 : \mathbb{R}^{s_1} \rightarrow \mathbb{R}$ takes one of the following forms with $b \in \mathbb{R}^{s_1}$ and $\sigma > 0$ chosen so that the origin is not feasible and that Assumption 5.1 holds:

- (i) (Basis pursuit denoising [14]) $q_1(z) = \frac{1}{2}\|z - b\|^2 - \sigma$.
- (ii) (Logistic loss [30]) $q_1(z) = \sum_{i=1}^{s_1} \log(1 + \exp(b_i z_i)) - \sigma$ for some $b \in \{-1, 1\}^{s_1}$.
- (iii) (Poisson loss [52]) $q_1(z) = \sum_{i=1}^{s_1} (-b_i z_i + \exp(z_i)) - \sigma$.

5.2 Case 2: h is the sum of a closed gauge and the Euclidean norm

When $h = \|\cdot\| + \kappa$, where κ is a closed gauge, i.e., a nonnegative, positively homogeneous closed convex function that vanishes at the origin, the corresponding problem (5.1) with $m = 1$ and a suitable choice of q_1 was studied in [27, Section 6.1] as an approach to “regularize” an underlying gauge optimization problem. In what follows, we consider the following assumption on q_1 in (5.1) with $m = 1$.

Assumption 5.2. *For (5.1) with $m = 1$, the function q_1 is strongly convex on any compact convex set and is twice continuously differentiable. Moreover, $\inf q_1 > -\infty$.*

Conditions on q_1 similar to those in Assumption 5.2 can be found in [33, 51] and are standard assumptions when it comes to studying the KL property of functions of the form $\lambda q_1(A_1 x) + h(x)$, where $\lambda > 0$; these are Lagrangian functions of the F in (5.1) with $m = 1$. In the next theorem, we study the KL property of (5.1) with $m = 1$ under Assumptions 5.1 and 5.2. Our analysis makes use of Corollary 5.1, which means that we will first study the KL property of some suitable Lagrangian functions. The proof of item (ii) below follows closely the arguments in [46, Theorem 4.4] concerning the so-called C^2 -cone-reducible sets: Indeed, our argument is based on representing a certain set as the level set of a continuously differentiable function with Lipschitz continuous gradient.

Theorem 5.2. *Let $h = \|\cdot\| + \kappa$ for some closed gauge κ . Then the following statements hold:*

- (i) *For any $\zeta \in \mathbb{R}^n$, the set $(\partial h)^{-1}(\zeta)$ is a polyhedron.*
- (ii) *The mapping $x \mapsto \partial h(x)$ is metrically subregular at $x \in \text{dom } \partial h$ for any $\zeta \in \partial h(x)$.*
- (iii) *Consider (5.1) with $m = 1$ and h as above. If Assumptions 5.1 and 5.2 hold and the origin is not feasible, then the function F in (5.1) is a KL function with an exponent of $\frac{1}{2}$.*

Proof. (i): Since κ is a closed gauge, from [26, Proposition 2.1(iii)], we see that $\kappa(x) = \sigma_D(x)$, where $D := \{x \in \mathbb{R}^n : \kappa^\circ(x) \leq 1\}$ is a convex closed set, and κ° is the polar of the gauge κ . Combining this with $\|x\| = \sigma_B(x)$, where $B := B(0, 1)$, we have that

$$(\partial h)^{-1} = (\partial(\|\cdot\| + \kappa))^{-1} = (\partial\sigma_{B+D})^{-1} \stackrel{(a)}{=} \mathcal{N}_{B+D}, \quad (5.2)$$

where (a) follows from [39, Eq. 11(4)].

Define $\Theta : \mathbb{R}^n \rightarrow \mathbb{R}$ by $\Theta(v) := \text{dist}^2(v, D) - 1$. Then we have $\nabla \Theta(v) = 2(v - P_D(v))$, where $P_D(v)$ denotes the projection of v onto D , and

$$B + D = \{v : \text{dist}^2(v, D) - 1 \leq 0\} = \{v : \Theta(v) \leq 0\}.$$

Moreover, $\Theta(v) = 0 \Leftrightarrow \|v - P_D(v)\| = 1$. In addition, from [29, Section D: Theorem 1.3.5], we have for any $v \in \mathbb{R}^n$ that

$$\mathcal{N}_{B+D}(v) = \begin{cases} \{0\} & \text{if } \Theta(v) < 0, \\ \bigcup_{t \geq 0} t(v - P_D(v)) & \text{if } \Theta(v) = 0, \\ \emptyset & \text{otherwise.} \end{cases} \quad (5.3)$$

Using (5.3) together with (5.2), we conclude that $(\partial h)^{-1}(\zeta)$ is a polyhedron for any $\zeta \in \mathbb{R}^n$, which completes the proof of (i).

(ii): Fix any $x \in \text{dom } \partial h$ and any $\zeta \in \partial h(x)$. In order to prove that the mapping $x \mapsto \partial h(x)$ is metrically subregular at x for ζ , by [22, Theorem 3H.3], we only need to prove that the set-valued mapping $(\partial h)^{-1}$ is calm at ζ with respect to x . By (5.2), it suffices to show that the mapping $v \mapsto \mathcal{N}_{B+D}(v)$ is calm at ζ with respect to x , i.e., there exist $\delta, \rho, \eta > 0$, such that whenever $v \in B(\zeta, \rho)$,

$$\mathcal{N}_{B+D}(v) \cap B(x, \delta) \subseteq \mathcal{N}_{B+D}(\zeta) + \eta \|v - \zeta\| B(0, 1). \quad (5.4)$$

To this end, we note first from (5.2) and (5.3) that $x \in \mathcal{N}_{B+D}(\zeta)$, which implies in particular that $\zeta \in B + D$. Hence, $\Theta(\zeta) \leq 0$. Now, if $\Theta(\zeta) < 0$, in view of (5.3), we have $\mathcal{N}_{B+D}(\zeta) = \{0\}$. In addition, by the continuity of Θ , there exists $\rho_1 > 0$ such that $\Theta(v) < 0$ whenever $v \in B(\zeta, \rho_1)$, which means that $\mathcal{N}_{B+D}(v) = \{0\}$. Thus, (5.4) holds for $\rho = \rho_1 > 0$ and any $\delta > 0$ and $\eta > 0$ in this case.

Next, suppose that $\Theta(\zeta) = 0$. Then $\|\zeta - P_D(\zeta)\| = 1 > 0$. Set

$$\delta = 1, \quad \rho = 0.25 \quad \text{and} \quad \eta = 4(1 + \|x\|), \quad (5.5)$$

and fix any $v \in B(\zeta, \rho)$. If $\Theta(v) > 0$, then $\mathcal{N}_{B+D}(v) = \emptyset$ in view of (5.3). On the other hand, if $\Theta(v) < 0$, then $\mathcal{N}_{B+D}(v) = \{0\}$. Therefore, when $\Theta(v) \neq 0$, one can check that (5.4) holds with δ and η as in (5.5).

Now, suppose that $\Theta(v) = 0$. Then we have from (5.3) that $\mathcal{N}_{B+D}(v) = \mathbb{R}_+(v - P_D(v))$. Thus, for any $x_1 \in \mathcal{N}_{B+D}(v) \cap B(x, \delta)$, there exists $t \geq 0$ such that $x_1 = t(v - P_D(v))$ and $\|x_1 - x\| \leq \delta = 1$. Then

$$\begin{aligned} 1 &\geq \|x_1 - x\| = \|t(v - P_D(v)) - x\| \geq t\|v - P_D(v)\| - \|x\| \\ &\geq t\|\zeta - P_D(\zeta)\| - t\|v - P_D(v) - [\zeta - P_D(\zeta)]\| - \|x\| \\ &\stackrel{(a)}{\geq} t\|\zeta - P_D(\zeta)\| - 2t\|v - \zeta\| - \|x\| \stackrel{(b)}{\geq} t\|\zeta - P_D(\zeta)\| - 2t\rho - \|x\| = 0.5t - \|x\|, \end{aligned}$$

where (a) follows from the Lipschitz continuity of $P_D(\cdot)$, as D is a closed convex set, (b) holds because $v \in B(\zeta, \rho)$, and the last equality holds because $\|\zeta - P_D(\zeta)\| = 1$ and $\rho = 0.25$. Rearranging terms in the above inequality, we have $t \leq 2(1 + \|x\|)$. Hence,

$$\begin{aligned} x_1 &= t(v - P_D(v)) = t(\zeta - P_D(\zeta)) + t(v - P_D(v) - (\zeta - P_D(\zeta))) \\ &\stackrel{(a)}{\in} \mathcal{N}_{B+D}(\zeta) + t\|v - P_D(v) - (\zeta - P_D(\zeta))\| B(0, 1) \\ &\stackrel{(b)}{\subseteq} \mathcal{N}_{B+D}(\zeta) + 2t\|v - \zeta\| B(0, 1) \stackrel{(c)}{\subseteq} \mathcal{N}_{B+D}(\zeta) + \eta\|v - \zeta\| B(0, 1), \end{aligned}$$

where (a) follows from (5.3), (b) holds because of the Lipschitz continuity of $P_D(\cdot)$, and (c) follows from the fact that $t \leq 2(1 + \|x\|)$ and the definition of η in (5.5). Thus, we have shown that the mapping $v \mapsto \mathcal{N}_{B+D}(v)$ is calm at ζ with respect to x . This completes the proof of (ii).

(iii): Notice that $\arg \min h = \{0\}$ and the origin is not feasible for (5.1) by assumption. Thus, from Corollary 5.1, in order to prove that the function F in (5.1) with $m = 1$ is a KL function with exponent $\frac{1}{2}$, we only need to show that $F_{\bar{\lambda}} := h + \bar{\lambda} q_1 \circ A_1$ is a KL function with exponent $\frac{1}{2}$ for

all $\bar{\lambda} > 0$.⁵ To this end, according to [33, Theorem 4.1], it suffices to show that $F_{\bar{\lambda}}$ satisfies the Luo-Tseng error bound. Clearly, $\text{Arg min } F_{\bar{\lambda}}$ is compact and nonempty for any $\bar{\lambda} > 0$ because h is coercive and $\inf q_1 > -\infty$ by Assumption 5.2.

Fix any $\bar{\lambda} > 0$ and any $\bar{x} \in \text{Arg min } F_{\bar{\lambda}}$. Since q_1 is strictly convex and $\bar{\lambda} > 0$, we have that $A_1 x \equiv A_1 \bar{x}$ over $\text{Arg min } F_{\bar{\lambda}}$. Thus, we can derive that $\bar{\lambda} A_1^T \nabla q_1(A_1 x) \equiv \bar{\lambda} A_1^T \nabla q_1(A_1 \bar{x}) =: \bar{\zeta}$ over $\text{Arg min } F_{\bar{\lambda}}$. Moreover, in view of [39, Theorem 10.1] and [39, Exercise 8.8(c)], we have $-\bar{\zeta} \in \partial h(\bar{x})$. Now, as in [51, Section 3.3], we define

$$\Gamma_f(y) := \{x \in \mathbb{R}^n : A_1 x = y\} \text{ and } \Gamma_h(\zeta) := \{x \in \mathbb{R}^n : -\zeta \in \partial h(x)\}.$$

Then $\Gamma_h(\bar{\zeta}) = (\partial h)^{-1}(-\bar{\zeta})$ and we see from item (i) that this is a polyhedron. This together with [9, Corollary 3] shows that $\{\Gamma_f(A_1 \bar{x}), \Gamma_h(\bar{\zeta})\}$ is boundedly linearly regular. Next, since $-\bar{\zeta} \in \partial h(\bar{x})$, we have from item (ii) that the subdifferential mapping ∂h is metrically subregular at \bar{x} for $-\bar{\zeta}$. Thus, in view of [51, Theorem 2], [51, Corollary 1] and the arbitrariness of $\bar{x} \in \text{Arg min } F_{\bar{\lambda}}$, we conclude that $F_{\bar{\lambda}}$ satisfies the Luo-Tseng error bound. This completes the proof. \square

5.3 Case 3: h is the sum of group LASSO penalty and a block-structured closed gauge

In this section, we suppose that

$$h(x) = \sum_{J \in \mathcal{J}} (\|x_J\| + \kappa_J(x_J)), \quad (5.6)$$

where \mathcal{J} is a partition of $\{1, 2, \dots, n\}$, $x_J \in \mathbb{R}^{|J|}$ is the subvector of $x \in \mathbb{R}^n$ indexed by $J \in \mathcal{J}$, and $\kappa_J : \mathbb{R}^{|J|} \rightarrow \mathbb{R} \cup \{\infty\}$ is a closed gauge for each $J \in \mathcal{J}$. The group LASSO regularizer, which is convex but not polyhedral in general, is motivated by the desire to induce sparsity among variables at a group level and has found applications in signal processing and statistics; see, e.g., [23, 49].

Theorem 5.3. *Consider (5.1) with $m = 1$ and h defined as in (5.6). Suppose that Assumptions 5.1 and 5.2 hold and the origin is not feasible for (5.1). Then F is a KL function with exponent $\frac{1}{2}$.*

Proof. As in the proof of Theorem 5.2(iii), we again have $\text{Arg min } F_{\bar{\lambda}}$ being compact and nonempty for any $\bar{\lambda} > 0$, where $F_{\bar{\lambda}} := h + \bar{\lambda} q_1 \circ A_1$. Fix any $\bar{\lambda} > 0$ and any $\bar{x} \in \text{Arg min } F_{\bar{\lambda}}$, and note that $A_1 x \equiv A_1 \bar{x}$ over $\text{Arg min } F_{\bar{\lambda}}$. Hence, $\bar{\lambda} A_1^T \nabla q_1(A_1 x) \equiv \bar{\lambda} A_1^T \nabla q_1(A_1 \bar{x}) =: \bar{\zeta}$ over $\text{Arg min } F_{\bar{\lambda}}$. Moreover, we also have $-\bar{\zeta} \in \partial h(\bar{x})$ in view of [39, Theorem 10.1] and [39, Exercise 8.8(c)]. Then we define as before the sets

$$\Gamma_f(y) := \{x \in \mathbb{R}^n : A_1 x = y\} \text{ and } \Gamma_h(\zeta) := \{x \in \mathbb{R}^n : -\zeta \in \partial h(x)\}.$$

Since $\Gamma_h(\bar{\zeta}) = \prod_{J \in \mathcal{J}} \Gamma_{h_J}(\bar{\zeta}_J)$, and we have from Theorem 5.2(i) that $\Gamma_{h_J}(\bar{\zeta}_J)$ is a polyhedron for every $J \in \mathcal{J}$, we deduce that $\Gamma_h(\bar{\zeta})$ is polyhedral. Thus, by [9, Corollary 3], $\{\Gamma_f(A_1 \bar{x}), \Gamma_h(\bar{\zeta})\}$ is boundedly linearly regular.

Next, since $-\bar{\zeta} \in \partial h(\bar{x})$, we have $-\bar{\zeta}_J \in \partial h_J(\bar{x}_J)$ for each $J \in \mathcal{J}$. Now, Theorem 5.2(ii) states that ∂h_J is metrically subregular at \bar{x}_J for $-\bar{\zeta}_J$, i.e., for each $J \in \mathcal{J}$, there exist constants $\eta_J, \rho_J > 0$ such that

$$\text{dist}(x_J, (\partial h_J)^{-1}(-\bar{\zeta}_J)) \leq \eta_J \text{dist}(-\bar{\zeta}_J, \partial h_J(x_J)) \text{ for all } x_J \in B(\bar{x}_J, \rho_J).$$

⁵Indeed, Assumption (i) in Corollary 5.1 implies that any Lagrange multiplier (which exists thanks to Assumption 5.1 and the coerciveness of h) has to be positive.

Define $\rho = \min_{J \in \mathcal{J}} \rho_J$. Then for all $x \in B(\bar{x}, \rho)$, we have from the above display that

$$\begin{aligned} \text{dist}^2(x, (\partial h)^{-1}(-\bar{\zeta})) &= \sum_{J \in \mathcal{J}} \text{dist}^2(x_J, (\partial h_J)^{-1}(-\bar{\zeta}_J)) \\ &\leq \sum_{J \in \mathcal{J}} \eta_J^2 \text{dist}^2(-\bar{\zeta}_J, \partial h_J(x_J)) \leq \eta^2 \text{dist}^2(-\bar{\zeta}, \partial h(x)), \end{aligned}$$

where $\eta = \max_{J \in \mathcal{J}} \eta_J$. Thus, ∂h is metrically subregular at \bar{x} for $-\bar{\zeta}$.

Now, in view of [51, Theorem 2], [51, Corollary 1] and the arbitrariness of $\bar{x} \in \text{Arg min } F_{\bar{\lambda}}$, we conclude that $F_{\bar{\lambda}}$ satisfies the Luo-Tseng error bound. Then the desired conclusion follows from [33, Theorem 4.1] and Corollary 5.1. This completes the proof. \square

6 Numerical experiments

In this section, we consider two specific applications from signal processing that can be formulated as (1.1) and (4.1), respectively. We then illustrate our algorithms on solving random instances of these problems.

6.1 Group sparse signals with Gaussian noise

In signal processing, when the signal is contaminated with Gaussian noise, the least squares loss function can be used [15, 20]. If it is also known that the original signal belongs to a union of subspaces, the group LASSO penalty [49] can be used for inducing such structure in the recovered solution; see [23]. Here, motivated by the use of the ℓ_{1-2} regularizer in compressed sensing in [45], we consider a natural extension that subtracts a positive multiple of the norm function from the group LASSO penalty function:

$$\begin{aligned} \min_{x \in \mathbb{R}^n} \quad & \sum_{J \in \mathcal{J}} \|x_J\| - \mu \|x\| \\ \text{s.t.} \quad & \|Ax - b\| \leq \sigma, \end{aligned} \tag{6.1}$$

where $\mu \in (0, 1)$, $A \in \mathbb{R}^{p \times n}$ has full row rank, $b \in \mathbb{R}^p$, $\sigma \in (0, \|b\|)$, \mathcal{J} is a partition of $\{1, 2, \dots, n\}$, $x_J \in \mathbb{R}^{|J|}$ is the subvector of x indexed by $J \in \mathcal{J}$.

Since the feasible set of (6.1) is unbounded, Algorithm 1 cannot be directly applied to solving (6.1). Fortunately, one can show that (6.1) is equivalent to the following model for some $M > 0$:

$$\begin{aligned} \min_{x \in \mathbb{R}^n} \quad & \sum_{J \in \mathcal{J}} \|x_J\| - \mu \|x\| \\ \text{s.t.} \quad & \|Ax - b\|^2 \leq \sigma^2, \\ & \max_{J \in \mathcal{J}} \|x_J\| \leq M. \end{aligned} \tag{6.2}$$

Indeed, as A has full row rank, we know that the feasible set of (6.1) is nonempty because it contains $\tilde{x} := A^\dagger b$. Since the objective of (6.1) is also level-bounded, the solution set of (6.1), denoted by S , is nonempty. Hence,

$$S \subseteq \left\{ x \in \mathbb{R}^n : \sum_{J \in \mathcal{J}} \|x_J\| - \mu \|x\| \leq \sum_{J \in \mathcal{J}} \|\tilde{x}_J\| - \mu \|\tilde{x}\| \right\}.$$

Thus, $x \in S$ implies $(1 - \mu) \max_{J \in \mathcal{J}} \|x_J\| \leq (1 - \mu) \|x\| \leq \sum_{J \in \mathcal{J}} \|x_J\| - \mu \|x\| \leq \sum_{J \in \mathcal{J}} \|\tilde{x}_J\| - \mu \|\tilde{x}\|$. This shows that (6.2) has the same set of optimal solutions as (6.1) if $M \geq (1 - \mu)^{-1} [\sum_{J \in \mathcal{J}} \|\tilde{x}_J\| - \mu \|\tilde{x}\|]$.

With these choices of $M > 0$, problem (6.2) is a special case of (1.1) with $m = 1$, $P_1(x) = \sum_{J \in \mathcal{J}} \|x_J\|$, $P_2(x) = \mu\|x\|$, $g_1(x) = \|Ax - b\|^2 - \sigma^2$ and $C = \{x \in \mathbb{R}^n : \max_{J \in \mathcal{J}} \|x_J\| \leq M\}$. Moreover, Assumption 3.1 holds for (6.2) with $x^\odot = \tilde{x}$. Therefore, we can apply Algorithm 1 to (6.2). In our numerical experiments below, we will be solving random instances of (6.2) with

$$M = (1 - \mu)^{-1} \left[\sum_{J \in \mathcal{J}} \|\tilde{x}_J\| - \mu\|\tilde{x}\| \right].$$

6.2 Compressed sensing with Cauchy noise for complex signals

In compressed sensing for real signals, when the measurement (say, of dimension $2p$) is contaminated with Cauchy noise, the Lorentzian norm $\|y\|_{LL_2, \gamma} := \sum_{j=1}^{2p} \log(1 + \gamma^{-2} y_j^2)$ can be employed as the loss function [17, 18], where $\gamma > 0$ and $y \in \mathbb{R}^{2p}$. Here we explore the scenario when the signal and the measurement matrix have *complex* entries with the transmission contaminated by Cauchy noise. Specifically, in this setting, an unknown sparse vector $\mathbf{z}_0 \in \mathbb{C}^n$ is compressed via multiplication with a given matrix $\mathbf{A} \in \mathbb{C}^{p \times n}$ and the resulting $\mathbf{A}\mathbf{z}_0$ is transmitted. The target is to recover approximately the vector \mathbf{z}_0 from the received signal $\mathbf{b} := \mathbf{A}\mathbf{z}_0 + \varepsilon$, where $\varepsilon \in \mathbb{C}^p$ is the noise vector, with the real and imaginary parts of each entry following an i.i.d. Cauchy distribution with mean 0. In view of the structure of ε , we will view it as a random vector in \mathbb{R}^{2p} with i.i.d. Cauchy entries and consider the Lorentzian norm for this real vector in our compressed sensing model below.

To derive the model, we start by writing

$$\mathbf{A} := \mathbf{A}_{\text{re}} + i\mathbf{A}_{\text{im}}, \quad \mathbf{b} := \mathbf{b}_{\text{re}} + i\mathbf{b}_{\text{im}},$$

where $\mathbf{A}_{\text{re}}, \mathbf{A}_{\text{im}} \in \mathbb{R}^{p \times n}$, and $\mathbf{b}_{\text{re}}, \mathbf{b}_{\text{im}} \in \mathbb{R}^p$, and define

$$A := \begin{bmatrix} \mathbf{A}_{\text{re}} & -\mathbf{A}_{\text{im}} \\ \mathbf{A}_{\text{im}} & \mathbf{A}_{\text{re}} \end{bmatrix} \in \mathbb{R}^{2p \times 2n}, \quad b := \begin{bmatrix} \mathbf{b}_{\text{re}} \\ \mathbf{b}_{\text{im}} \end{bmatrix} \in \mathbb{R}^{2p}. \quad (6.3)$$

Then, notice that for any complex vector $\mathbf{z} = \mathbf{x} + i\mathbf{y} \in \mathbb{C}^n$, where \mathbf{x} and \mathbf{y} are respectively the real part and imaginary part of \mathbf{z} , it holds that

$$\mathbf{A}\mathbf{z} - \mathbf{b} = \mathbf{A}_{\text{re}}\mathbf{x} - \mathbf{A}_{\text{im}}\mathbf{y} - \mathbf{b}_{\text{re}} + i[\mathbf{A}_{\text{im}}\mathbf{x} + \mathbf{A}_{\text{re}}\mathbf{y} - \mathbf{b}_{\text{im}}].$$

Thus, we can consider the following loss function for the Cauchy measurement noise:

$$\left\| \begin{bmatrix} \mathbf{A}_{\text{re}}\mathbf{x} - \mathbf{A}_{\text{im}}\mathbf{y} - \mathbf{b}_{\text{re}} \\ \mathbf{A}_{\text{im}}\mathbf{x} + \mathbf{A}_{\text{re}}\mathbf{y} - \mathbf{b}_{\text{im}} \end{bmatrix} \right\|_{LL_2, \gamma} = \|Ax - b\|_{LL_2, \gamma}, \quad \text{where } x := \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix},$$

and $\gamma > 0$ is an estimate on the standard deviation of the Cauchy distribution. Based on the above observation and motivated by the use of ℓ_{1-2} -type regularizer in [45] for inducing sparsity, we consider the following model for recovering sparse signal with entries in complex numbers from measurements contaminated with Cauchy measurement noise:

$$\begin{aligned} \min_{x \in \mathbb{R}^{2n}} \quad & \sum_{J \in \mathcal{J}} \|x_J\| - \mu\|x\| \\ \text{s.t.} \quad & \|Ax - b\|_{LL_2, \gamma} \leq \bar{\sigma}, \end{aligned} \quad (6.4)$$

where $A \in \mathbb{R}^{2p \times 2n}$ and $b \in \mathbb{R}^{2p}$ are given in (6.3), $\mu \in (0, 1)$ and $\bar{\sigma} \in (0, \|b\|_{LL_2, \gamma})$, $\mathcal{J} := \bigcup_{i=1}^n \mathcal{J}_i$ with $\mathcal{J}_i := \{i, i+n\}$ for each i , and $x_J \in \mathbb{R}^2$ is the subvector of x indexed by $J \in \mathcal{J}$. Moreover, we assume without loss of generality that A has full row rank: this can be inferred from the fact that the measurement matrix \mathbf{A} typically has full row rank.

Although the feasible region of (6.4) is unbounded and is not a special case of (4.1), one can argue as in the discussion following (6.2) that (6.4) is equivalent to the following model:

$$\begin{aligned} \min_{x \in \mathbb{R}^{2n}} \quad & \sum_{J \in \mathcal{J}} \|x_J\| - \mu \|x\| \\ \text{s.t.} \quad & \|Ax - b\|_{LL_2, \gamma} \leq \bar{\sigma}, \\ & \max_{J \in \mathcal{J}} \|x_J\| \leq M_1 \end{aligned} \tag{6.5}$$

as long as $M_1 \geq (1 - \mu)^{-1} [\sum_{J \in \mathcal{J}} \|(A^\dagger b)_J\| - \mu \|A^\dagger b\|]$. Notice that (6.5) is a special case of (4.1) with $P_1(x) = \sum_{J \in \mathcal{J}} \|x_J\|$, $P_2(x) = \mu \|x\|$, $\sigma = \bar{\sigma}$, $\ell(y) = \|y\|_{LL_2, \gamma}$, and $C = \{x \in \mathbb{R}^{2n} : \max_{J \in \mathcal{J}} \|x_J\| \leq M_1\}$. One can see from Example 4.2 that Assumption 4.1 is satisfied for (6.5), and Assumption 4.2 is satisfied for (6.5) with $x^\circ = A^\dagger b$. Hence, we can apply Algorithm 2 to (6.5). In our numerical experiments below, we will be solving random instances of (6.4) with

$$M_1 = (1 - \mu)^{-1} [\sum_{J \in \mathcal{J}} \|(A^\dagger b)_J\| - \mu \|A^\dagger b\|].$$

6.3 Numerical results

In this subsection, we perform numerical experiments to study the performance of Algorithm 1 on solving (6.2) and that of Algorithm 2 on solving (6.5). We compare our approaches with Algorithm 3 below, which is an adaptation of the ESQM with line search (ESQM_{ls}) from [7]. The ESQM_{ls} in [7] was originally designed for (1.1) with P and all the g_i 's being twice continuously differentiable on C , and subsequential convergence was established under suitable assumptions. Here, we adapt the ESQM_{ls} as Algorithm 3 for our problems. Note that the convergence analysis in [7] can be readily adapted to establish subsequential convergence of Algorithm 3 for solving (6.2). Similar to our algorithms, Algorithm 3 also involves affine approximations in each subproblem (see (6.6)). However, unlike our approaches, it does not force all iterates to be feasible.

Algorithm 3 (Adaptation of) ESQM_{ls} in [7] for (1.1)

Step 0. Choose $x^0 \in C$, $c \in (0, 1)$, $\beta_0 > 0$, $t \in (0, 1)$ and $\delta > 0$. Set $k = 0$.

Step 1. Pick any $\xi^k \in \partial P_2(x^k)$ and compute

$$\begin{aligned} (\tilde{u}, \tilde{s}) = \arg \min_{(x, s) \in \mathbb{R}^{n+1}} \quad & P_1(x) - \langle \xi^k, x - x^k \rangle + \frac{1}{\beta_k} s + \frac{1}{2\beta_k} \|x - x^k\|^2 \\ \text{s.t.} \quad & g_i(x^k) + \langle \nabla g_i(x^k), x - x^k \rangle \leq s, \quad i = 1, \dots, m, \\ & s \geq 0, \quad x \in C. \end{aligned} \tag{6.6}$$

Step 2. Let t_k be the largest element in $\{1, t, t^2, \dots\}$ so that

$$F_k(x^k + t_k(\tilde{u} - x^k)) \leq F_k(x^k) - ct_k\beta_k^{-1} \|\tilde{u} - x^k\|^2$$

where $F_k(x) := P(x) + \beta_k^{-1} \max\{\hat{g}(x), 0\}$ and $\hat{g}(x) := \max_{1 \leq i \leq m} \{g_i(x)\}$.

Step 3. If $g_i(x^k) + \langle \nabla g_i(x^k), \tilde{u} - x^k \rangle \leq 0$ for all i , then $\beta_{k+1} = \beta_k$, otherwise $\beta_{k+1}^{-1} = \beta_k^{-1} + \delta$. Set $u^k = \tilde{u}$ and $x^{k+1} = x^k + t_k(\tilde{u} - x^k)$. Update $k \leftarrow k + 1$ and go to **Step 1**.

We next present the parameters, initial points and termination criteria used in our numerical experiments. Solution approaches for the subproblems involved are outlined in the appendix.

All numerical experiments are performed in MATLAB R2016a on a 64-bit PC with an Intel(R) Core(TM) i7-10710U CPU (@1.10GHz, 1.61GHz) and 16GB of RAM.

Parameter settings: In Algorithms 1 and 2, we let $c = 10^{-4}$, $\eta = \frac{1}{2}$, $\underline{\beta} = 10^{-8}$, $\bar{\beta} = 10^8$. We set $\beta_0^0 = 1$ and, for $k \geq 1$, we choose

$$\beta_k^0 = \begin{cases} \min\{\max\{\underline{\beta}, 2\beta_{k-1}^0\}, \bar{\beta}\} & \text{if } \beta_{k-1} = \beta_{k-1}^0, \\ \min\{\max\{\underline{\beta}, \beta_{k-1}\}, \bar{\beta}\} & \text{otherwise.} \end{cases}$$

In Algorithm 3, we let $c = 10^{-4}$, $\beta_0 = 1$, $t = \frac{1}{2}$. It appears that the performance of ESQM_{ls} can be sensitive to the choice of δ . In our experiments below, we consider 3 choices of δ : 0.5, 0.1 or 0.02.

Slater point: We take the Slater points as $x^\odot = x^\odot = A^\dagger b$, and compute $A^\dagger b$ via the MATLAB commands `[Q,R]=qr(A',0); x=Q*(R'\b)`.

Initialization: For (6.2), we first solve (6.1) with $\mu = 0$ approximately using SPGL1 [10] (version 2.1), under default settings, to obtain an approximate solution x_{spgl1} . We then project x_{spgl1} onto the box $\{x : \max_{J \in \mathcal{J}} \|x_J\| \leq M\}$ to obtain \hat{x}_{spgl1} . The initial point x^0 is then generated as follows:

$$x^0 = \begin{cases} \hat{x}_{\text{spgl1}} + \tau(A^\dagger b - \hat{x}_{\text{spgl1}}) & \text{if } \|A\hat{x}_{\text{spgl1}} - b\| > \sigma, \\ \hat{x}_{\text{spgl1}} & \text{otherwise,} \end{cases} \quad (6.7)$$

where $\tau \in (0, 1)$ is a positive root of the quadratic equation $\|A(\hat{x}_{\text{spgl1}} + \tau(A^\dagger b - \hat{x}_{\text{spgl1}})) - b\|^2 = \sigma^2$.

For (6.5), we first find a positive constant $\tau_0 \in (0, 1)$ so that $\|b - A(\tau_0 A^\dagger b)\|_{LL_2, \gamma} = \bar{\sigma}$.⁶ Then we apply the SPGL1 [10] (version 2.1) with default settings to approximately solve the following convex problem to obtain an approximate solution x_{spgl1} :

$$\begin{aligned} \min_{x \in \mathbb{R}^n} \quad & \sum_{J \in \mathcal{J}} \|x_J\| \\ \text{s.t.} \quad & \ell^{\tau_0 A^\dagger b}(Ax - b) \leq \bar{\sigma}^{\tau_0 A^\dagger b}, \end{aligned} \quad (6.8)$$

where $\ell^{\tau_0 A^\dagger b}$ and $\bar{\sigma}^{\tau_0 A^\dagger b}$ are defined as in (4.4) and (4.3) respectively, with $\tau_0 A^\dagger b$ in place of y there. We then project x_{spgl1} onto the box $\{x : \|x\|_\infty \leq M_1\}$ to obtain \tilde{x}_{spgl1} . The initial point x^0 is then generated as follows:

$$x^0 = \begin{cases} \tilde{x}_{\text{spgl1}} + \tau_1(A^\dagger b - \tilde{x}_{\text{spgl1}}) & \text{if } \|A\tilde{x}_{\text{spgl1}} - b\|_{LL_2, \gamma} > \bar{\sigma}, \\ \tilde{x}_{\text{spgl1}} & \text{otherwise,} \end{cases} \quad (6.9)$$

where $\tau_1 \in (0, 1)$ is chosen so that $\|b - A[\tilde{x}_{\text{spgl1}} + \tau_1(A^\dagger b - \tilde{x}_{\text{spgl1}})]\|_{LL_2, \gamma} = \bar{\sigma}$.⁷ One can find such a constant τ_1 efficiently by first applying Newton's method to obtain a root s_* of the function $s \mapsto \|\sqrt{s}(b - A\tilde{x}_{\text{spgl1}})\|_{LL_2, \gamma} - \bar{\sigma}$ (initialized at $s = 0$) and then setting $\tau_1 = 1 - \sqrt{s_*}$.

Termination criteria. The termination criteria are based on verifying approximately conditions (i), (ii) and (iii) in Definition 2.3 by $x = u^k$ and $\lambda = \lambda_k$, where u^k comes from the solution of the subproblems (3.1), (4.6) or (6.6), and λ_k is a Lagrange multiplier corresponding to the affine inequality constraint induced from g_1 .

Specifically, the algorithms are terminated when

$$\max\{\|\xi_u^k - \xi^k\| + \mathfrak{L}_k\|u^k - x^k\|, 10^2 \cdot \max\{|\lambda_k g_1(u^k)|, g_1(u^k)\}\} \leq 10^{-4} \cdot \max\{\|u^k\|, 1\}, \quad (6.10)$$

where $\xi_u^k = \arg \min_{\xi \in \mu \partial \|u^k\|} \{\|\xi\|\}$ and

$$\mathfrak{L}_k = \begin{cases} 2\lambda_k \|A\|^2 + \beta_k^{-1} & \text{for (6.2),} \\ 2\lambda_k \|A\|^2 \gamma^{-2} + \beta_k^{-1} & \text{for (6.5).} \end{cases}$$

⁶Note that $b - A(\tau_0 A^\dagger b) = (1 - \tau_0)b$, and hence such a τ_0 can be found efficiently by first applying the Newton's method to obtain a root s_* of the function $s \mapsto \|\sqrt{s}b\|_{LL_2, \gamma} - \bar{\sigma}$ (initialized at $s = 0$) and then setting $\tau_0 = 1 - \sqrt{s_*}$.

⁷Note that $b - A[\tilde{x}_{\text{spgl1}} + \tau_1(A^\dagger b - \tilde{x}_{\text{spgl1}})] = (1 - \tau_1)(b - A\tilde{x}_{\text{spgl1}})$

Moreover, as a safeguard, we also terminate when stepsizes are *small*. Specifically, we terminate Algorithms 1 and 2 when $\beta_k \leq 10^{-10}$, and terminate Algorithm 3 when $t_k \leq 10^{-10}$. Note that when (6.10) is satisfied, we have $g_1(u^k) \leq 10^{-6} \cdot \max\{\|u^k\|, 1\}$ and $|\lambda_k g_1(u^k)| \leq 10^{-6} \cdot \max\{\|u^k\|, 1\}$, and we can deduce from the KKT conditions of subproblems (3.1), (4.6) and (6.6) that $\text{dist}(0, \partial F_k(u^k)) \leq 10^{-4} \cdot \max\{\|u^k\|, 1\}$ when $u^k \neq 0$, where $F_k(x) = P_1(x) - P_2(x) + \lambda_k g_1(x) + \delta_C(x)$.

6.3.1 Group LASSO with Gaussian noise.

We compare Algorithm 1 and Algorithm 3 on solving random instances of (6.2). Specifically, we generate an $A \in \mathbb{R}^{p \times n}$ with i.i.d. standard Gaussian entries and then normalize each column of A . Next, we use the following MATLAB command to generate the original block-sparse signal $x_{\text{orig}} \in \mathbb{R}^n$:

```
I = randperm(n/j); I = I(k+1:end); x0 = randn(j,n/j);
x0(:,I) = 0; x_orig = reshape(x0,n,1),
```

where j is the size of each block and k is the number of nonzero blocks. We let $b = Ax_{\text{orig}} + 0.005\bar{\epsilon}$, where $\bar{\epsilon}$ is a random vector which has i.i.d standard Gaussian entries. Finally, we set $\sigma = 1.2\|0.005\bar{\epsilon}\|$.

In our numerical tests, we let $\mu = 0.95$, $j = 2$, $(p, n, k) = (720i, 2560i, 120i)$ with $i \in \{2, 4, 6, 8, 10\}$. For each i , we generate 20 random data as described above. We present the computational results in Table 1, averaged over the 20 random instances. Here, Algorithm 1 is denoted by $\text{FPA}_{\text{retract}}$, Algorithm 3 with different δ is denoted by ESQM_{ls} with the δ specified, and recall that SPGL1 is the algorithm used for obtaining initial points for $\text{FPA}_{\text{retract}}$ and ESQM_{ls} . We present the time for computing the QR decomposition of A^T (denoted by QR), the time for computing $A^\dagger b$ given the QR factorization of A^T (denoted by Slater), the CPU times of the algorithms (CPU time),⁸ the number of iterations (denoted by Iter), the recovery errors $\text{RecErr} := \frac{\|x^* - x_{\text{orig}}\|}{\max\{1, \|x_{\text{orig}}\|\}}$ and the residuals $\text{Residual} := \frac{\|Ax^* - b\| - \sigma}{\sigma}$, where x^* is the approximate solution returned by the respective algorithm.

From Table 1, one can see that the recovery errors of $\text{FPA}_{\text{retract}}$ and ESQM_{ls} are comparable, and $\text{FPA}_{\text{retract}}$ is usually faster. Moreover, the performance of ESQM_{ls} appears to be quite sensitive to the choice of δ . **On passing, we also point out that the solutions returned by $\text{FPA}_{\text{retract}}$ and ESQM_{ls} tend to have smaller residuals (in absolute value) than those obtained by SPGL1.**

6.3.2 Compressed sensing with Cauchy noise

We compare Algorithm 2 and Algorithm 3 on solving random instances of (6.5). Specifically, we generate two matrices $\mathbf{A}_{\text{re}} \in \mathbb{R}^{p \times n}$ and $\mathbf{A}_{\text{im}} \in \mathbb{R}^{p \times n}$ with i.i.d standard Gaussian entries, let

$$A := \begin{bmatrix} \mathbf{A}_{\text{re}} & -\mathbf{A}_{\text{im}} \\ \mathbf{A}_{\text{im}} & \mathbf{A}_{\text{re}} \end{bmatrix}$$

and then normalize each column of A . Next, we generate two vectors $\mathbf{u} \in \mathbb{R}^k$ and $\mathbf{v} \in \mathbb{R}^k$ with i.i.d standard Gaussian entries. We choose an index set I of size k at random and define the original sparse signal $\mathbf{z}_{\text{orig}} \in \mathbb{C}^n$ by setting $\mathbf{z}_j := \mathbf{u}_j + i\mathbf{v}_j$ for all $j \in I$ and $\mathbf{z}_j = 0$ otherwise. We then let

$$x_{\text{orig}} = \begin{bmatrix} \mathbf{z}_{\text{re}} \\ \mathbf{z}_{\text{im}} \end{bmatrix} \quad \text{and} \quad b = Ax_{\text{orig}} + 0.005\hat{\epsilon},$$

where \mathbf{z}_{re} and \mathbf{z}_{im} are respectively the real and imaginary parts of \mathbf{z}_{orig} , and $\hat{\epsilon}$ is a random vector with each entry independently following the Cauchy(0, 1) distribution, i.e., $\hat{\epsilon}_j = \tan(\pi(\tilde{\epsilon}_j - \frac{1}{2}))$ with each $\tilde{\epsilon}_j \in \mathbb{R}^{2p}$ being uniformly chosen in $[0, 1]$. Finally, we set $\bar{\sigma} = 1.2\|0.005\hat{\epsilon}\|_{LL_2, \gamma}$ with $\gamma = 0.05$.

⁸The reported CPU times of $\text{FPA}_{\text{retract}}$ and ESQM_{ls} do not include the time for generating x^\odot and x^0 .

Table 1: Computational results for problem (6.2)

	Method	$i = 2$	$i = 4$	$i = 6$	$i = 8$	$i = 10$
CPU time	QR	0.55	2.64	8.27	21.07	40.90
	Slater	0.01	0.02	0.05	0.09	0.14
	SPGL1	1.37	6.52	16.54	29.97	51.87
	FPA _{retract}	2.62	11.00	31.82	57.61	82.91
	ESQM _{ls} , $\delta=0.5$	4.36	21.02	57.37	102.78	162.15
	ESQM _{ls} , $\delta=0.1$	3.99	18.85	49.01	87.80	139.50
	ESQM _{ls} , $\delta=0.02$	11.87	54.43	132.39	235.45	378.77
Iter	FPA _{retract}	342	349	449	460	421
	ESQM _{ls} , $\delta=0.5$	756	899	1094	1111	1120
	ESQM _{ls} , $\delta=0.1$	774	853	964	972	976
	ESQM _{ls} , $\delta=0.02$	2447	2536	2650	2648	2665
RecErr	SPGL1	0.054	0.044	0.053	0.054	0.045
	FPA _{retract}	0.030	0.032	0.035	0.035	0.035
	ESQM _{ls} , $\delta=0.5$	0.030	0.032	0.035	0.035	0.035
	ESQM _{ls} , $\delta=0.1$	0.030	0.032	0.035	0.035	0.035
	ESQM _{ls} , $\delta=0.02$	0.030	0.032	0.035	0.035	0.035
Residual	SPGL1	-2.17e-04	-1.90e-04	-1.68e-04	-1.47e-04	-1.27e-04
	FPA _{retract}	-1.48e-15	6.36e-16	-1.46e-16	2.41e-15	-1.04e-15
	ESQM _{ls} , $\delta=0.5$	1.13e-10	1.24e-10	1.10e-10	1.13e-10	1.13e-10
	ESQM _{ls} , $\delta=0.1$	9.71e-11	1.01e-10	9.62e-11	1.01e-10	1.02e-10
	ESQM _{ls} , $\delta=0.02$	1.06e-10	9.77e-11	9.87e-11	9.51e-11	9.51e-11

In our numerical experiments, we let $\mu = 0.95$, $j = 2$, $\gamma = 0.05$, $(p, n, k) = (360i, 1280i, 60i)$ with $i \in \{2, 4, 6, 8, 10\}$. For each i , we generate 20 random instances as described above. The computational results averaged over the 20 random instances are shown in Table 2. In the table, Algorithm 2 is denoted by FPA_{retract}, Algorithm 3 with different δ is denoted by ESQM_{ls} with the δ specified, and recall that SPGL1 is the algorithm used for obtaining initial points for FPA_{retract} and ESQM_{ls} by solving the auxiliary problem (6.8). As before, we present the time for performing QR decomposition on A^T (denoted by QR), the time for computing $A^\dagger b$ given the QR factorization of A^T (denoted by Slater), the CPU time of algorithms (CPU time),⁹ the number of iterations (denoted by Iter), the recovery errors $\text{RecErr} := \frac{\|x^* - x_{\text{orig}}\|}{\max\{1, \|x_{\text{orig}}\|\}}$ and the residuals $\text{Residual} := \frac{\|Ax^* - b\|_{LL_2, \gamma - \bar{\sigma}}}{\bar{\sigma}}$, where x^* is the approximate solution returned by the respective algorithm.

From Table 2, one can see that the recovery errors of FPA_{retract} and ESQM_{ls} are comparable, and FPA_{retract} is usually faster, except for the case when $i = 6$. Indeed, we observe that the performance of FPA_{retract} can be quite sensitive to the “quality” of the initial point (reflected partly by the RecErr of the approximate solution returned by SPGL1).

7 Conclusions

In this paper, we considered difference-of-convex (DC) programs with smooth inequality and simple geometric constraints, and proposed first-order feasible methods for solving this class of nonconvex optimization problems. Our proposed methods were based on affine approximations for the constraints, and maintained the feasibility of the iterates by using a “retraction step” inspired from the development of manifold optimization. We then established the global subsequential convergence of the sequence generated by the proposed algorithms under strict feasibility condition if the constraint functions are convex. We also showed that our methods can be extended to solve

⁹The reported CPU times of FPA_{retract} and ESQM_{ls} do not include the time for generating x^\odot and x^0 .

Table 2: Computational results for problem (6.5)

	Method	$i = 2$	$i = 4$	$i = 6$	$i = 8$	$i = 10$
CPU time	QR	0.61	2.82	8.42	21.15	40.40
	Slater	0.01	0.02	0.05	0.10	0.15
	SPGL1	0.97	12.69	68.82	22.22	98.06
	FPA _{retract}	2.58	13.57	424.98	35.03	73.58
	ESQM _{ls} , $\delta=0.5$	13.16	38.65	69.74	102.27	132.12
	ESQM _{ls} , $\delta=0.1$	14.25	46.93	104.76	177.87	189.31
	ESQM _{ls} , $\delta=0.02$	13.83	47.29	102.40	169.08	266.92
Iter	FPA _{retract}	221	368	5611	272	371
	ESQM _{ls} , $\delta=0.5$	1745	1480	1229	1072	896
	ESQM _{ls} , $\delta=0.1$	1888	1796	1862	1877	1288
	ESQM _{ls} , $\delta=0.02$	1828	1814	1824	1787	1823
RecErr	SPGL1	0.869	1.494	5.501	0.964	1.443
	FPA _{retract}	0.048	0.051	0.051	0.051	0.052
	ESQM _{ls} , $\delta=0.5$	0.048	0.051	0.051	0.051	0.052
	ESQM _{ls} , $\delta=0.1$	0.048	0.051	0.051	0.051	0.052
	ESQM _{ls} , $\delta=0.02$	0.048	0.051	0.051	0.051	0.052
Residual	SPGL1	-5.23e-01	-6.27e-01	-7.35e-01	-6.01e-01	-6.86e-01
	FPA _{retract}	-2.86e-12	-2.37e-12	-2.19e-12	-2.45e-12	-2.32e-12
	ESQM _{ls} , $\delta=0.5$	7.20e-11	1.05e-10	8.33e-11	2.08e-10	1.11e-10
	ESQM _{ls} , $\delta=0.1$	2.95e-11	3.25e-11	3.15e-11	3.67e-11	9.77e-11
	ESQM _{ls} , $\delta=0.02$	2.70e-11	2.53e-11	2.76e-11	2.58e-11	2.93e-11

DC programs with a single specially structured nonconvex constraint. Finally, we illustrated the promising numerical performance of the proposed methods via two concrete optimization models: group-structured compressed sensing problems with Gaussian measurement noise and compressed sensing problems with Cauchy measurement noise.

Our results in this paper point out several interesting research questions. For example, in the case where the constraint functions are convex, we established that the sequence generated by our method is bounded, and any cluster point of the generated sequence is a critical point of the underlying optimization problem. It would be interesting to see whether the convexity assumption of the constraint functions can be relaxed as some suitable generalized convexity conditions such as quasi-convexity. Moreover, for nonconvex constrained DC programs, we showed that our method can be extended to solve DC programs with a single specially structured nonconvex constraint. Although this nontrivial extension allows us to cover some important applications such as compressed sensing problems with Cauchy measurement noise, how to further extend this to general nonconvex problems with multiple constraints is also an interesting and challenging research question to pursue. Finally, [36] discussed the so-called d-stationary points for DC programs which is a stronger notion than the critical points, and designed algorithms with convergence guarantee to d-stationary points. Another interesting research direction is to examine how our methods can be modified to give rise to convergence to d-stationary points. These are some potential future research topics which deserve further study.

A Subproblems in FPA_{retract} and ESQM_{ls} for (6.2) and (6.5)

In this appendix, we outline how the subproblems (3.1), (4.6) and (6.6) are solved in our numerical experiments, which solve random instances of (6.2) and (6.5).

A.1 Solving the subproblem in $\text{FPA}_{\text{retract}}$ for (6.2) and (6.5)

The subproblems (3.1) (with $m = 1$) and (4.6) can be reformulated as follows:

$$\begin{aligned} \min_{x \in \mathbb{R}^n} \quad & P_1(x) + \frac{1}{2\beta} \|x - y\|^2 + \delta_C(x) \\ \text{s.t.} \quad & \langle a, x \rangle \leq r, \end{aligned} \quad (\text{A.1})$$

where C is a compact convex set, $y, a \in \mathbb{R}^n$, $r \in \mathbb{R}$ and $\beta > 0$. Moreover, the Slater condition holds thanks to Theorem 3.1(i) and Theorem 4.1(i).

By [38, Corollary 28.2.1, Theorem 28.3], x^* is optimal for (A.1) if and only if x^* is feasible for (A.1) and there exists $\lambda_* \geq 0$ such that the following conditions hold:

- (I) $0 \in \partial(P_1 + \delta_C)(x^*) + \frac{1}{\beta}(x^* - y) + \lambda_* a$;
- (II) $\lambda_*(\langle a, x^* \rangle - r) = 0$.

Note that (I) is equivalent to

$$x^* = \arg \min_{x \in C} \left\{ \beta P_1(x) + \frac{1}{2} \|y - \lambda_* \beta a - x\|^2 \right\}. \quad (\text{A.2})$$

For (6.2), since $P_1(x) = \sum_{J \in \mathcal{J}} \|x_J\|$ and $C = \{x : \max_{J \in \mathcal{J}} \|x_J\| \leq M\}$, through an argument similar to the one outlined in [34, Appendix], the relation (A.2) is equivalent to

$$x_J^* = \min \left\{ \max \left\{ 1 - \frac{\beta}{\|y_J - \lambda_* \beta a_J\|}, 0 \right\}, \frac{M}{\|y_J - \lambda_* \beta a_J\|} \right\} (y_J - \lambda_* \beta a_J), \quad (\text{A.3})$$

for each $J \in \mathcal{J}$.

If $\lambda_* = 0$, we can obtain the optimal solution x^* directly from (A.3). On the other hand, if $\lambda_* > 0$, plugging (A.3) into (II) above, we see that λ_* can be obtained as a root of the following function:

$$T(\lambda) := r - \sum_{J \in \mathcal{J}} \min \left\{ \max \left\{ 1 - \frac{\beta}{\|y_J - \lambda \beta a_J\|}, 0 \right\}, \frac{M}{\|y_J - \lambda \beta a_J\|} \right\} a_J^T (y_J - \lambda \beta a_J). \quad (\text{A.4})$$

The solution x^* can then be recovered via (A.3). In our implementation, we apply Newton's method to find a root of T . Specifically, we follow the framework of Newton's method described in [41] (see the algorithmic description in [41, Algorithm 2.1]).¹⁰ At iteration $k = 0$, we initialize the Newton's method at $\lambda = 0$, and at each subsequent iteration, we warm-start this algorithm using the λ obtained in the previous iteration; we terminate the Newton's method when the approximate solution λ_* satisfies $\|T(\lambda_*)\| \leq 10^{-10}$. Moreover, as a safeguard, we also terminate when stepsizes are small: Specifically, we terminate the algorithm when the stepsize α_k in [41, Algorithm 2.1] falls below 10^{-10} .

A.2 Solving the subproblems in ESQM_{ls} for (6.2) and (6.5)

The subproblem (6.6) (with $m = 1$) takes the following form:

$$\begin{aligned} \min_{(x,s) \in \mathbb{R}^{n+1}} \quad & P_1(x) + \frac{1}{\beta} s + \frac{1}{2\beta} \|x - y\|^2 + \delta_C(x) \\ \text{s.t.} \quad & \langle a, x \rangle - r \leq s, \\ & s \geq 0, \end{aligned} \quad (\text{A.5})$$

¹⁰Follow the notation in [41, Algorithm 2.1], we set $\beta = \frac{1}{2}$, $\lambda = 10^{-4}$, and choose G_k from the generalized Jacobian of T at λ_k , $\mu_k = 10^{-4} \|T(\lambda_k)\|^{\frac{1}{2}}$ and $\rho_k = 0$ at each iteration k .

where C is a compact convex set, $y, a \in \mathbb{R}^n$, $\beta > 0$ and $r \in \mathbb{R}$. The Slater condition holds trivially for the above problem, i.e., there exists $\hat{x} \in C$ and $\hat{s} > 0$ so that the first inequality constraint also holds strictly.

By [38, Corollary 28.2.1, Theorem 28.3], (x^*, s_*) is optimal for (A.5) if and only if (x^*, s_*) is feasible for (A.5) and there exists $\lambda_* \geq 0$ such that the following conditions hold:

- (A) $0 \in \partial(P_1 + \delta_C)(x^*) + \frac{1}{\beta}(x^* - y) + \lambda_* a$ and $0 \in \frac{1}{\beta} - \lambda_* + \mathcal{N}_{\mathbb{R}_+}(s_*)$;
- (B) $\lambda_*(\langle a, x^* \rangle - r - s_*) = 0$.

Note that the first inclusion in (A) is equivalent to (A.2). Hence, when solving (6.2), one can again rewrite x^* as (A.3). Next, from the second inclusion of (A), we have that $\lambda_* - \frac{1}{\beta} \leq 0$ and $s_*(\lambda_* - \frac{1}{\beta}) = 0$. Thus:

- If $\lambda_* = 0$ or $\lambda_* = \frac{1}{\beta}$, we can obtain the optimal solution x^* from (A.3);
- If $0 < \lambda_* < \frac{1}{\beta}$, then $s_* = 0$. In this case, (B) become $\lambda_*(\langle a, x^* \rangle - r) = 0$, which is the same as (II) in Section A.1. The λ_* is then obtained as a root of T in (A.4), and we find it using the Newton's method with the same parameter settings used in Section A.1. The x^* is then recovered via (A.3).

Funding Yongle Zhang was supported partly by the National Natural Science Foundation of China (11901414) and (11871359) and Sichuan Science and Technology Program (2018JY0201). Guoyin Li was partially supported by a Future fellowship from Australian Research Council (FT130100038) and a discovery project from Australian Research Council (DP190100555). Ting Kei Pong was supported partly by Hong Kong Research Grants Council PolyU153000/20p.

Declarations

Conflict of interest The second author is a guest editor of the topical collection "Mathematics of Computation and Optimisation" of this journal.

References

- [1] W. van Ackooij and W. de Oliveira. Convexity and optimization with copulae structured probability constraints. *Optimization*, 65, 1349–1376, 2016.
- [2] W. van Ackooij, S. Demassey, P. Javal, H. Morais, W. de Oliveira and B. Swaminathan. A bundle method for nonsmooth DC programming with application to chance-constrained problems. *Computational Optimization and Applications*, 78, 451–490, 2021.
- [3] M. Ahn, J. S. Pang and J. Xin. Difference-of-convex learning: directional stationarity, optimality, and sparsity. *SIAM Journal on Optimization* 27, 1637–1665, 2017.
- [4] H. Attouch and J. Bolte. On the convergence of the proximal algorithm for nonsmooth functions involving analytic features. *Mathematical Programming* 116, 5–16, 2009.
- [5] H. Attouch, J. Bolte, P. Redont and A. Soubeyran. Proximal alternating minimization and projection methods for nonconvex problems: an approach based on the Kurdyka-Łojasiewicz inequality. *Mathematics of Operations Research* 35, 438–457, 2010.
- [6] H. Attouch, J. Bolte and B. F. Svaiter. Convergence of descent methods for semi-algebraic and tame problems: proximal algorithms, forward-backward splitting, and regularized Gauss-Seidel methods. *Mathematical Programming* 137, 91–129, 2013.

- [7] A. Auslender. An extended sequential quadratically constrained quadratic programming algorithm for nonlinear, semidefinite, and second-order cone programming. *Journal of Optimization Theory and Applications* 156, 183–212, 2013.
- [8] A. Auslender, R. Shefi and M. Teboulle. A moving balls approximation method for a class of smooth constrained minimization problems. *SIAM Journal on Optimization* 20, 3232–3259, 2010.
- [9] H. Bauschke, J. Borwein and W. Li. Strong conical hull intersection property, bounded linear regularity, Jameson’s property (G), and error bounds in convex optimization. *Mathematical Programming* 86, 135–160, 1999.
- [10] E. van den Berg and M. P. Friedlander. Probing the Pareto frontier for basis pursuit solutions. *SIAM Journal on Scientific Computing* 31, 890–912, 2008.
- [11] D. Bertsekas. *Nonlinear Programming*. Athena Scientific, Belmont, MA, 1995.
- [12] J. Bolte and E. Pauwels. Majorization-minimization procedures and convergence of SQP methods for semi-algebraic and tame programs. *Mathematics of Operations Research* 41, 442–465, 2016.
- [13] J. Bolte, S. Sabach and M. Teboulle. Proximal alternating linearized minimization for nonconvex and nonsmooth problems. *Mathematical Programming* 146, 459–494, 2014.
- [14] E. J. Candès. The restricted isometry property and its implications for compressed sensing. *Comptes Rendus Mathématique* 346, 589–592, 2008.
- [15] E. J. Candès, J. K. Romberg and T. Tao. Stable signal recovery from incomplete and inaccurate measurements. *Communications on Pure and Applied Mathematics* 59, 1207–1223, 2006.
- [16] E. J. Candès, M. B. Wakin and S. P. Boyd. Enhancing sparsity by reweighted l_1 minimization. *Journal of Fourier Analysis and Applications* 14, 877–905, 2008.
- [17] R. E. Carrillo, K. E. Barner and T. C. Aysal. Robust sampling and reconstruction methods for sparse signals in the presence of impulsive noise. *IEEE Journal of Selected Topics in Signal Processing* 4, 392–408, 2010.
- [18] R. E. Carrillo, A. B. Ramirez, G. R. Arce, K. E. Barner and B. M. Sadler. Robust compressive sensing of sparse signals: a review. *EURASIP Journal on Advances in Signal Processing* 108, 2016.
- [19] S. Chen, Z. Deng, S. Ma and A. M.-C. So. Manifold proximal point algorithms for dual principal component pursuit and orthogonal dictionary learning. *Asilomar Conference on Signals, Systems, and Computers*, 2019.
- [20] S. S. Chen, D. L. Donoho and M. A. Saunders. Atomic decomposition by basis pursuit. *SIAM Review* 43, 129–159, 2001.
- [21] S. Chen, S. Ma, A. M.-C. So and T. Zhang. Proximal gradient method for nonsmooth optimization over the Stiefel manifold. *SIAM Journal on Optimization* 30, 210–239, 2020.
- [22] A. L. Dontchev and R. T. Rockafellar. *Implicit Functions and Solution Mappings*. Springer Monographs in Mathematics. Springer, New York, 2009.

- [23] Y. C. Eldar and M. Mishali. Robust recovery of signals from a structured union of subspaces. *IEEE Transactions on Information Theory* 55, 5302–5316, 2009.
- [24] R. Fletcher, N. I. M. Gould, S. Leyffer, P. L. Toint and A. Wächter. Global convergence of a trust-region SQP-filter algorithm for general nonlinear programming. *SIAM Journal on Optimization* 13, 635–659, 2002.
- [25] C. A. Floudas and P. M. Pardalos. *Encyclopedia of Optimization*. Springer Science & Business Media, 2008.
- [26] M. P. Friedlander, I. Macêdo and T. K. Pong. Gauge optimization and duality. *SIAM Journal on Optimization* 24, 1999–2022, 2014.
- [27] M. P. Friedlander, I. Macêdo and T. K. Pong. Polar convolution. *SIAM Journal on Optimization* 29, 1366–1391, 2019.
- [28] P. E. Gill and E. Wong. Sequential quadratic programming methods. In *Mixed Integer Nonlinear Programming*. (J. Lee and S. Leyffer eds.) Springer, New York, NY, 147–224, 2012.
- [29] J.-B. Hiriart-Urruty and C. Lemaréchal. *Fundamentals of Convex Analysis*. Springer, New York, 2001.
- [30] J. D. W. Hosmer, S. Lemeshow and R. X. Sturdivant. *Applied Logistic Regression*. John Wiley Sons, 3rd edition, 2013.
- [31] C. T. Lawrence and A. L. Tits. A computational efficient feasible sequential quadratic programming algorithm. *SIAM Journal on Optimization* 11, 1092–1118, 2001.
- [32] H. A. Le Thi and T. P. Dinh, DC programming and DCA: thirty years of developments. *Mathematical Programming Ser. B* 169, 5–68, 2018.
- [33] G. Li and T. K. Pong. Calculus of the exponent of Kurdyka-Łojasiewicz inequality and its applications to linear convergence of first-order methods. *Foundations of Computational Mathematics* 18, 1199–1232, 2018.
- [34] T. Liu and T. K. Pong. Further properties of the forward-backward envelope with applications to difference-of-convex programming. *Computational Optimization and Applications* 67, 489–520, 2017.
- [35] J. Nocedal and S. Wright. *Numerical Optimization*. 2nd edition. Springer, 2006.
- [36] J.S. Pang, M. Razaviyayn and A. Alvarado, Computing B-stationary points of nonsmooth DC programs. *Mathematics of Operation Research* 42, 95–118, 2017.
- [37] E. R. Panier and A. L. Tits. On combining feasibility, descent and superlinear convergence in inequality constrained optimization. *Mathematical Programming* 59, 261–276, 1993.
- [38] R. T. Rockafellar. *Convex Analysis*. Princeton University Press, Princeton, 1970.
- [39] R. T. Rockafellar and R. J-B. Wets. *Variational Analysis*. Springer, 1998.
- [40] M. V. Solodov. Global convergence of an SQP method without boundedness assumptions on any of the iterative sequences. *Mathematical Programming* 118, 1–12, 2009.
- [41] M. V. Solodov and B. F. Svaiter. A globally convergent inexact Newton method for systems of monotone equations. In *Reformulation: Nonsmooth, Piecewise Smooth, Semismooth and Smoothing Methods*. (M. Fukushima and L. Qi eds.), Springer, Boston, MA, 355–369, 1998.

- [42] H. Tuy. *Convex Analysis and Global Optimization*. Springer, 1998.
- [43] A. Veinott. The supporting hyperplane method for unimodal programming, *Operations Research*, 15, 147-152, 1967.
- [44] S. J. Wright and M. J. Tenny. A feasible trust-region sequential quadratic programming algorithm. *SIAM Journal on Optimization* 14, 1074–1105, 2004.
- [45] P. Yin, Y. Lou, Q. He and J. Xin. Minimization of ℓ_{1-2} for compressed sensing. *SIAM Journal on Scientific Computing* 37, A536–A563, 2015.
- [46] P. Yu, G. Li and T. K. Pong. Kurdyka-Łojasiewicz exponent via inf-projection. Preprint (2019). Available at <https://arxiv.org/abs/1902.03635>.
- [47] P. Yu and T. K. Pong. Iteratively reweighted ℓ_1 algorithms with extrapolation. *Computational Optimization and Applications* 73, 353–386, 2019.
- [48] P. Yu, T. K. Pong and Z. Lu. Convergence rate analysis of a sequential convex programming method with line search for a class of constrained difference-of-convex optimization problems. *SIAM Journal on Optimization* 31, 2024–2054, 2021.
- [49] M. Yuan and Y. Lin. Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 68, 49–67, 2006.
- [50] L. Zeng, P. Yu and T. K. Pong. Analysis and algorithms for some compressed sensing models based on L1/L2 minimization. *SIAM Journal on Optimization* 31, 1576–1603, 2021.
- [51] Z. Zhou and A. M.-C. So. A unified approach to error bounds for structured convex optimization problems. *Mathematical Programming* 165, 689–728, 2017.
- [52] G. Zou. A modified Poisson regression approach to prospective studies with binary data. *American Journal of Epidemiology* 159, 702–706, 2004.