Xue et al. *Journal of Inequalities and Applications* (2024) 2024:65
https://doi.org/10.1186/s13660-024-03141-1

Journal of Inequalities and Applications
a SpringerOpen Journal

RESEARCH **Open Access**

Sequential inertial linear ADMM algorithm for nonconvex and nonsmooth multiblock problems with nonseparable structure

Zhonghui Xue[1], Kaiyuan Yang[2], Qianfeng Ma[1] and Yazheng Dang[2*]

*Correspondence: jgdyz@163.com
[2]School of Managemant, University of Shanghai for Science and Technology, Jungong Road 516, Yangpu District, Shanghai, 200093, Shanghai, China
Full list of author information is available at the end of the article

Abstract

The alternating direction method of multipliers (ADMM) has been widely used to solve linear constrained problems in signal processing, matrix decomposition, machine learning, and many other fields. This paper introduces two linearized ADMM algorithms, namely sequential partial linear inertial ADMM (SPLI-ADMM) and sequential complete linear inertial ADMM (SCLI-ADMM), which integrate linearized ADMM approach with inertial technique in the full nonconvex framework with nonseparable structure. Iterative schemes are formulated using either partial or full linearization while also incorporating the sequential gradient of the composite term in each subproblem's update. This adaptation ensures that each iteration utilizes the latest information to improve the efficiency of the algorithms. Under some mild conditions, we prove that the sequences generated by two proposed algorithms converge to the critical points of the problem with the help of KŁ property. Finally, some numerical results are reported to show the effectiveness of the proposed algorithms.

Keywords: Inertial technique; Linear ADMM; Multiblock optimization problems; Global convergence; Kurdyka; Łojasiewicz property; Sequential gradient

1 Introduction

In this paper, we consider the following linearly constrained nonconvex optimization problem with multiple block variables:

$$\begin{aligned} &\min_{x_i,y} \sum_{i=1}^{n} f_i(x_i) + g(x_1, x_2, \dots, x_n, y), \\ &\text{s.t.} \sum_{i=1}^{n} A_i x_i + By = b, \end{aligned} \tag{1.1}$$

where $x_i \in \mathbb{R}^{p_i} (i = 1, 2, \dots n)$ and $y \in \mathbb{R}^q$ are variables, each $f_i : \mathbb{R}^{p_i} \to \mathbb{R} \cup \{+\infty\} (i = 1, 2, \dots n)$ are proper lower semicontinuous functions, which are nonconvex and (possibly) nonsmooth, $g : \mathbb{R}^{p_1} \times \mathbb{R}^{p_2} \times \cdots \times \mathbb{R}^{p_n} \times \mathbb{R}^q \to \mathbb{R}$ is continuously differentiable, and ∇g is Lips-

© The Author(s) 2024. **Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by/4.0/.

chitz continuous with modulus $l_g > 0$, $A_i \in \mathbb{R}^{m\times p_i}(i = 1, 2, \dots n)$, $B \in \mathbb{R}^{m\times q}$ are given matrices, and $b \in \mathbb{R}^m$. Denote $\mathbf{x}_{[i,j]} = (x_i, x_{i+1}, \dots, x_{j-1}, x_j)$ and $\mathbf{A}\mathbf{x}_{[j,k]} = \sum_{i=j}^{k} A_i x_i$.

The Augmented Lagrangian Function (ALF) of (1.1) is defined as

$$L(\mathbf{x}_{[1,n]}, y, \lambda) = \sum_{i=1}^{n} f_i(x_i) + g(\mathbf{x}_{[1,n]}, y) - \langle \lambda, \mathbf{A}\mathbf{x}_{[1,n]} + By - b\rangle + \frac{\beta}{2}\|\mathbf{A}\mathbf{x}_{[1,n]} + By - b\|^2,$$

where $\lambda \in \mathbb{R}^m$ is the Lagrangian dual variable, and $\beta > 0$ is a penalty parameter.

The problem (1.1) encapsulates a multitude of nonconvex optimization problems across various domains, including signal processing, image reconstruction, matrix decomposition, machine learning, etc. [1–3]. When the number of blocks n equals 2, and $g(\cdot)$ is identically zero, this problem degenerates into two-block separable problem. If the problem contains merely a mixed term, it becomes similar to the problem in [4]. On the other hand, if variable y is absent, the problem becomes the study in [5]. Hence, problem (1.1) extends the scope of the objective functions found in the literature [4–6], encompassing a broader range of scenarios with additional variables and potential mixed terms, thereby reflecting the versatility and complexity encountered in contemporary applications.

Indeed, ADMM has been established as a powerful tool for solving two-block separable convex optimization problems [7, 8]. However, its effectiveness and convergence guarantees become much more intricate when dealing with nonconvex problems, especially when the number of blocks exceeds two. Zhang et al. [9] tackled this challenge by proposing a proximal ADMM for solving three-block nonconvex optimization tasks, building upon the groundwork laid by Sun et al. [10]. Meanwhile, Wang et al. [11] proposed an inertial proximal partially symmetric ADMM, suitable for handling multiblock separable nonconvex optimization problems. Hien et al. [12] developed an inertial version of ADMM, referred to as iADMM, which integrated the majorization-minimization principle within each block update step to address a specific class of nonconvex low-rank representation problems. Chao et al. [13] contributed to this area with a linear Bregman ADMM algorithm for nonconvex multiblock optimization problems featuring nonseparable structures.

Linearized Alternating Direction Method of Multipliers (LADMM) simplifies the problem-solving process and significantly decreases the computational overhead associated with traditional ADMM. By linearizing certain components of the optimization problem at each iteration, LADMM allows for more straightforward and efficient updates. Li et al. [14] effectively utilized LADMM in the context of the least absolute shrinkage and selection operator (LASSO) problem, demonstrating that this linearized approach is simple and highly efficient. Ling et al. [15] further extended the application of LADMM by introducing a decentralized linearized ADMM algorithm, which solely linearizes the objective functions at each iterative step. This method facilitates distributed computation and can handle large-scale problems more effectively. Specifically addressing nonconvex and nonsmooth scenarios, Liu et al. [16] proposed a two-block linearized ADMM. This variant linearizes the mixed term and the quadratic penalty term in the Augmented Lagrangian Function (ALF), thereby providing a viable solution strategy for such challenging optimization problems. Chao et al. [13] presented a linear Bregman ADMM, which only linearized the mixed term for solving three-block nonseparable problems. This approach maintains the efficiency gains of LADMM while adapting it to accommodate the complexities inherent in multiblock and nonseparable optimizations.

Inertial technique, initially conceived by Polyak [17], serves as an acceleration strategy that takes into account the dynamics of the optimization process by incorporating information from the last two iterations, thereby mitigating substantial differences between consecutive points. Subsequently, Zavriv et al. [18] expanded the use of the inertial technique to tackle nonconvex optimization problems, marking a significant milestone in broadening the applicability of this methodology. Recently, the inertial technique has seen widespread adoption in conjunction with various optimization algorithms to enhance their performance in solving nonconvex optimization problems. Bot et al. [19] proposed an inertial forward-backward algorithm for the minimization of the sum of two non-convex functions. Attouch et al. [20] introduced an inertial proximal method and a proximal alternating projection method for maximal-monotone problems and minimization problems, respectively. Pock et al. [21] went on to propose a linear Inertial Proximal Alternating Minimization Algorithm (IPAMA) for a diverse range of nonconvex and nonsmooth optimization problems. Building upon these advancements, researchers have successfully integrated the inertial technique with the Alternating Direction Method of Multipliers (ADMM). Hien et al. [22] developed an Inertial Alternating Direction Method of Multipliers (iADMM) specifically designed for a class of nonconvex multiblock optimization problems with nonlinear coupling constraints. Wang et al. [11] also introduced an Inertial Proximal Partially Symmetric ADMM, tailored for nonconvex settings, further highlighting the versatility and efficacy of combining inertial techniques with ADMM in modern optimization methodologies.

Inspired by the previous works [11, 13, 16, 23], in this paper, we construct two new variant linear inertial ADMM algorithms, sequential partial linear inertial ADMM (SPLI-ADMM) and sequential complete linear inertial ADMM (SCLI-ADMM) for problem (1.1).

The novelty of this paper can be summarized as follows:

(I) The proposed algorithms combine the inertial effect with the linearization skill. The former improves the feasibility of the algorithms, while the latter contributes to fast convergence.

(II) Unlike conventional approaches such as those in [13], during the linearization phase, the gradient of the mixed term of the x_j-sub-problem is calculated as $\nabla_{x_j} g(\mathbf{x}^{k+1}_{[1,j-1]}, \mathbf{x}^{k}_{[j,k]}, y^k)$ rather than $\nabla_{x_j} g(\mathbf{x}^{k}_{[1,n]}, y^k)$. This distinctive characteristic enables us to linearize the mixed term dynamically based on the progress of the indicator sequence, meaning that each update depends on the current state of the indicators. Consequently, it is referred to as a sequential gradient iteration scheme.

The rest of this paper is organized as follows: In Sect. 2, some necessary preliminaries for further analysis are summarized. Then, we establish the convergence of the two algorithms in Sect. 3. Section 4 shows the validity of the algorithms by some numerical experiments. Finally, some conclusions are drawn in Sect. 5.

2 Preliminaries

In this section, we recall some basic notations and preliminary results, which will be used in this paper. Throughout, $\mathbb{R}^n$ denotes the n-dimensional Euclidean space, $\mathbb{R} \cup \{+\infty\}$ denotes the extended real number set, and $\mathbb{N}$ denotes the natural number set. The image space of a matrix $Q \in \mathbb{R}^{m\times n}$ is defined as $\mathrm{Im}Q := \{Qx : x \in \mathbb{R}^n\}$. If matrix $Q \neq 0$, let $\rho_{\min(Q^{\mathrm{T}}Q)}$

denote the smallest positive singular value of the matrix $Q^{\mathrm{T}}Q$. $\|\cdot\|$ represents the Euclidean norm. $\mathrm{dom} f := \{x \in \mathbb{R}^n : f(x) < +\infty\}$ is the domain of a function $f : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$, $\langle x, y\rangle = x^{\mathrm{T}}y = \sum_{i=1}^{n} x_i y_i$.

Definition 2.1 ([24]) Let $f : \mathbb{R}^n \to \mathbb{R} \bigcup \{+\infty\}$ be a proper lower semicontinuous function.

(I) The Fréchet subdifferential, or regular subdifferential, of f at $x \in \mathrm{dom} f$, written $\hat{\partial} f(x)$, is defined as

$$\hat{\partial} f(x) = \left\{ x^* \in \mathbb{R}^n : \liminf_{y \neq x \ y \neq x} \frac{f(y) - f(x) - \langle x^*, y - x\rangle}{\|y - x\|} \geq 0 \right\},$$

when $x \notin \mathrm{dom} f$, we set $\hat{\partial} f(x) = \emptyset$.

(II) The limiting-subdifferential, or simply the subdifferential, of f at $x \in \mathrm{dom} f$, written $\partial f(x)$, is defined as

$$\partial f(x) = \left\{ x^* \in \mathbb{R}^n : \exists x_k \to x, s.t. f(x_k) \to f(x), x_k^* \in \hat{\partial} f(x), x_k^* \to x^* \right\}.$$

(III) A point that satisfies

$$0 \in \partial f(x)$$

is called a critical point or a stationary point of the function f. The set of critical points of f is denoted by $\mathrm{crit} f$.

Proposition 2.1 *We collect some basic properties of the subdifferential* [24].

(I) $\hat{f}(x) \subseteq \partial f(x)$ *for each* $x \in \mathbb{R}^n$, *where the first set is closed convex, and the second set is only closed.*

(II) Let $x_k^* \in \partial f(x_k)$ *and* $\lim_{k\to\infty}(x_k, x_k^*) = (x, x^*)$, *then,* $x^* \in \partial f(x)$.

(III) If $f : \mathbb{R}^n \to \mathbb{R} \bigcup \{+\infty\}$ *is proper lower semicontinuous, and* $g : \mathbb{R}^m \to \mathbb{R}$ *is continuous differentiable, then* $\partial(f + g)(x) = \partial f(x) + \nabla g(x)$ *for any* $x \in \mathrm{dom} f$.

Definition 2.2 If $\omega^* = (x_1^*, \ldots x_n^*, y^*, \lambda^*)^T$ such that

$$\begin{cases} A_i^{\mathrm{T}}\lambda^* \in \partial f_i(x_i^*) + \nabla_{x_i} g(x_1^*, \ldots x_n^*, y^*), \quad i = 1, 2, \ldots n, \\ B^{\mathrm{T}}\lambda^* = \nabla_y g(x_1^*, \ldots x_n^*, y^*), \\ A_1 x_1^* + \cdots + A_n x_n^* + By^* = b, \end{cases} \tag{2.1}$$

then ω^* is called a critical point or stationary point of the Lagrangian function $L(x_1, \ldots, x_n, y, \lambda)$.

A very important technique to prove the convergence of ADMM for nonconvex optimization problems is the assumption that the Lagrangian function satisfies the Kurdyka-Łojasiewicz property (KŁ property) [19, 25]. For notational simplicity, we use $\Phi_\eta (\eta > 0)$ to denote the set of concave functions $\varphi : [0, \eta) \to [0, \infty)$ such that

(I) $\varphi(0) = 0$;

(II) φ is continuously differentiable on $(0, \eta)$ and continuous at 0;

(III) $\varphi'(s) > 0$ for all $s \in (0, \eta)$.

The KŁ property can be described as follows.

Definition 2.3 (see [19, 26]) (KŁ property) Let $f : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ be a proper lower semicontinuous function. If there exist $\eta \in (0, +\infty]$, a neighborhood U of x^*, and a continuous concave function $\varphi \in \Phi_\eta$ such that for all $x \in U \cap \{x \in R^m : f(x^*) < f(x) < f(x^*) + \eta\}$, it holds that

$$\varphi'\big(f(x) - f(x^*)\big)\operatorname{dist}\big(0, \partial f(x)\big) \geq 1, \tag{2.2}$$

where the distance from x to S is defined by $d(x, S) := \inf\{\|y - x\| : y \in S\}$. Then, f is said to have the KŁ property at x^*.

Lemma 2.1 (*see* [25]) (*Uniformized KŁ property*) *Suppose that $f : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ is a proper lower semicontinuous function, and Ω is a compact set. If $f(x) \equiv f^*$ for all $x \in \Omega$ and satisfies the KŁ property at each point of Ω, then there exist $\varepsilon > 0, \eta > 0$ and $\varphi \in \Phi_\eta$ such that*

$$\varphi'\big(f(x) - f^*\big)\operatorname{dist}\big(0, \partial f(x)\big) \geq 1, \tag{2.3}$$

for all $x \in \{x \in \mathbb{R}^m : \operatorname{dist}(x, \Omega) < \varepsilon\} \cap \{f^* < f(x) < f^* + \eta\}$.

Lemma 2.2 (*see* [25]) (*Descent lemma*) *Let $h : \mathbb{R}^n \to \mathbb{R}$ be a continuous differentiable function where gradient ∇h is Lipschitz continuous with the modulus $l_h > 0$, then for any $x, y \in \mathbb{R}^n$, we have*

$$\big|h(y) - h(x) - \big\langle \nabla h(x), y - x\big\rangle\big|^2 \leq \frac{l_h}{2}\|y - x\|^2. \tag{2.4}$$

Lemma 2.3 (*see* [27]) *Let $Q \in \mathbb{R}^{m\times n}$ be a nonzero matrix, and let $\rho_{\min(Q^{\mathrm{T}}Q)}$ denote the smallest positive eigenvalue of $Q^{\mathrm{T}}Q$. Then, for every $u \in \mathbb{R}^n$, it holds that*

$$\sqrt{\rho_{\min(Q^{\mathrm{T}}Q)}}\|P_Q u\| \leq \|Qu\|, \tag{2.5}$$

where P_Q denotes the Euclidean projection onto $\operatorname{Im}(Q)$.

3 Algorithms and their convergence

In this section, we propose two linear inertial ADMM algorithms, sequential partial linear inertial ADMM (SPLI-ADMM), and sequential complete linear inertial ADMM (SCLI-ADMM) and prove their convergence with some suitable conditions. Furthermore, we prove the boundedness of the sequence.

3.1 Two linear inertial algorithms

First, we present Algorithm 1 for (1.1).

In every iteration of the subproblems, our approach utilizes sequential gradient to update the variables. Specifically, for the $(k+1)$th iteration of x_i $(i = 1, \ldots, n)$, the mixed term $g(\mathbf{x}^{k+1}_{[1,i-1]}, x_i, \mathbf{x}^k_{[i+1,n]}, y^k)$ is replaced with a linearized approximation that includes an inertial proximal term: $g_{x_i}(\mathbf{x}^{k+1}_{[1,i-1]}, \mathbf{x}^k_{[i,n]}, y^k) + \langle x_i - x_i^k, \nabla g_{x_i}(\mathbf{x}^{k+1}_{[1,i-1]}, \mathbf{x}^k_{[i,n]}, y^k)\rangle + \frac{\tau}{2}\|x_i - z_i^k\|^2$. Here, the sequential gradient $\nabla g_{x_i}(\mathbf{x}^{k+1}_{[1,i-1]}, \mathbf{x}^k_{[i,n]}, y^k)$ is refreshed for each subproblem, reflecting the most recent variable updates. Note that the y-subproblem remains unlinearized, so we call it sequential partial linear inertial ADMM.

For x_j-subproblem ($i = 1, \dots, n$) and y-subproblem, respectively, we get the following auxiliary functions:

$$\begin{aligned}\hat{f}_j^k(x_j) = {} & f_j(x_j) + \langle x_j - x_j^k, \nabla_{x_j} g(\mathbf{x}_{[1,j-1]}^{k+1}, \mathbf{x}_{[j,n]}^k, y^k)\rangle \\ & + \frac{\beta}{2}\left\| \mathbf{A}\mathbf{x}_{[1,j-1]}^{k+1} + A_j x_j + \mathbf{A}\mathbf{x}_{[j+1,n]}^k + By^k - b - \frac{\lambda^k}{\beta}\right\|^2 + \frac{\tau}{2}\|x_j - z_j^k\|^2,\end{aligned} \tag{3.1}$$

$$\hat{h}^k(y) = g(\mathbf{x}_{[1,n]}^{k+1}, y) + \frac{\beta}{2}\left\| \mathbf{A}\mathbf{x}_{[1,n]}^{k+1} + By - b - \frac{\lambda^k}{\beta}\right\|^2 + \frac{\tau}{2}\|y - y^k\|^2, \tag{3.2}$$

where

$$\begin{cases} z_1^k = x_1^k + \theta_k(x_1^{k-1} - x_1^k), \\ z_2^k = x_2^k + \theta_k(x_2^{k-1} - x_2^k), \\ \vdots \\ z_n^k = x_n^k + \theta_k(x_n^{k-1} - x_n^k), \end{cases} \tag{3.3}$$

and $\theta_k \in [0, \frac{1}{2})$. Utilizing the auxiliary functions above, the update rules are summarized in Algorithm 1 as follows:

Algorithm 1 SPLI-ADMM (Sequential Partial Linear Inertial ADMM)

1: Initialize $\mathbf{x}_{[1,n]}^{-1} = \mathbf{x}_{[1,n]}^0 = 0, y^{-1} = y^0 = 0, \lambda^0, \theta_k \in [0, \frac{1}{2}), \tau, \beta$.
2: **while** $\mathbf{A}\mathbf{x}_{[1,n]}^k + By^k - b \geq \varepsilon$ **do**
3: **for** $i = 1, \dots, k$ **do**

$$x_i^{k+1} = \arg\min_{x_i} \hat{f}_i^k(x_i). \tag{3.4}$$

4: **end for**
5:

$$y^{k+1} = \arg\min_{y} \hat{h}^k(y). \tag{3.5}$$

6:

$$\lambda^{k+1} = \lambda^k + \beta(\mathbf{A}\mathbf{x}_{[1,n]}^{k+1} + By^{k+1} - b). \tag{3.6}$$

7: **end while**
8: **return** $(x_1^{k+1}, \dots, x_n^{k+1}, y^{k+1}, \lambda^{k+1})$.

Remark 1 (I) The auxiliary functions defined in (3.1) own the inertial term $\frac{\tau}{2}\|x_i - z_i^k\|^2$, $i = 1, 2, \dots, n$, respectively. The inertial schemes update the new iteration by employing the two previous iterations. By adding the inertial term to x_i subproblems, the iteration trends to the direction $x_i^k - x_i^{k-1}$.

(II) The purpose of linearizing the mixed term in x_i-subproblem is to use the properties of differentiable blocks and simplify the calculation of each iteration.

(III) The initial point $\mathbf{x}_{[1,n]}^{-1} = \mathbf{x}_{[1,n]}^{0} = 0, y^{-1} = y^0 = 0$ was designed for demonstrating the boundedness of the sequence $\{\omega^k\}$ generated by the algorithm.

The update rules of Algorithm 2 can be written as follows:

Algorithm 2 SCLI-ADMM (Sequential Complete Linear Inertial ADMM)

1: Initialize $\mathbf{x}_{[1,n]}^{-1} = \mathbf{x}_{[1,n]}^{0} = 0, y^{-1} = y^0 = 0, y^{-1} = y^0 = 0, \lambda^0, \theta_k \in [0, \frac{1}{2}), \tau, \beta$.
2: **while** $\mathbf{A}\mathbf{x}_{[1,n]}^{k} + By^k - b \geq \varepsilon$ **do**
3: **for** $i = 1, \ldots, n$ **do**

$$x_i^{k+1} = \arg\min_{x_i} \hat{f}_i^k(x_i).$$

4: **end for**
5:

$$y^{k+1} = \arg\min_{y} \bar{h}^k(y).$$

6:

$$\lambda^{k+1} = \lambda^k + \beta\left(\mathbf{A}\mathbf{x}_{[1,n]}^{k+1} + By^{k+1} - b\right). \tag{3.7}$$

7: **end while**
8: **return** $(x_1^{k+1}, \ldots, x_n^{k+1}, y^{k+1}, \lambda^{k+1})$.

Algorithm 2 is obtained by further linearization on the basis of Algorithm 1. The x_i-subproblems ($i = 1, \ldots, n$) are same to that of Algorithm 1, the iterative scheme can be written as (3.4). During the $(k+1)$th iteration for updating y, we replace the function in $g(\mathbf{x}_{[1,n]}^{k+1}, y)$ with a linearized approximation plus a regularization term $g_y(\mathbf{x}_{[1,n]}^{k+1}, y^k) + \langle y - y^k, \nabla g_y(\mathbf{x}_{[1,n]}^{k+1}, y^k)\rangle + \frac{\tau}{2}\|y - y^k\|^2$. In Algorithm 2, all the subproblems were linearized and sequential updated, hence we call it the Sequential Complete Linear Inertial ADMM.

The auxiliary function of y-subproblem is as follows

$$\bar{h}^k(y) = \left\langle y - y^k, \nabla_y g\left(\mathbf{x}_{[1,n]}^{k+1}, y^k\right)\right\rangle + \frac{\beta}{2}\left\|\mathbf{A}\mathbf{x}_{[1,n]}^{k+1} + By - b - \frac{\lambda^k}{\beta}\right\|^2 + \frac{\tau}{2}\left\|y - y^k\right\|^2. \tag{3.8}$$

3.2 A descent inequality

A crucial element in establishing the convergence of these algorithms is to verify the descent property of the regularized augmented Lagrangian function sequence. To facilitate our analysis, the following notations are introduced throughout this paper. For $k \geq 1$,

$$\Delta x_i^{k+1} = x_i^{k+1} - x_i^k, \qquad \Delta y^{k+1} = y^{k+1} - y^k, \qquad \Delta\lambda^{k+1} = \lambda^{k+1} - \lambda^k.$$

$$\Delta\mathbf{x}_{[i,j]}^{k+1} = \left(\Delta x_i^{k+1}, \ldots, \Delta x_j^{k+1}\right), \qquad \theta\left\|\Delta\mathbf{x}_{[i,j]}^{k+1}\right\| = \sum_{s=i}^{j}\theta\left\|\Delta x_s^{k+1}\right\|.$$

The convergence analysis relies on the following assumptions:

Assumption A (I) g is l_g-Lipschitz differentiable, and g is bounded from below. ∇g is l_g-Lipschitz continuous, i.e., $\|\nabla g(u) - \nabla g(v)\| \le l_g\|u - v\|$ for all $u, v \in \mathbb{R}^{p_1} \times \mathbb{R}^{p_2} \times \cdots \times \mathbb{R}^{p_n} \times \mathbb{R}^q$;

(II) f_i, $i = 1, \ldots, n$ are proper lower semicontinuous, and f_i are bounded from below;

(III) The linear operator B is surjective, i.e., $B \neq 0$ and $\{b\} \bigcup \{\bigcup_{i=1}^n \operatorname{Im} A_i\} \subset \operatorname{Im} B$;

(IV) For Algorithm 1 and Algorithm 2, $\theta_k \in [0, \frac{1}{2})$, $\tau > 0$ and β is large enough such that $\tau > \frac{2+l_g}{1-2\theta_k}$, $\beta > \max\{\frac{3(l_g^2+\tau^2)}{\rho_{\min(B^{\mathrm{T}}B)}}, \frac{6(\tau^2+l_g^2)}{\tau\theta_k\rho_{\min(B^{\mathrm{T}}B)}}\}$;

(V) Let $X := \mathbb{R}^{p_1} \times \cdots \times \mathbb{R}^{p_n} \times \mathbb{R}^q \times \mathbb{R}^m$. The set $\{\omega \in X : L_\beta(\omega) \le L_\beta(\omega^0)\}$ is bounded.

For showing the descent property, the following lemmas are necessary.

Lemma 3.1 *For Algorithm 1, for each $k \in N$, we have*

$$\|\Delta\lambda^{k+1}\|^2 \le \frac{3l_g^2}{\rho_{\min(B^{\mathrm{T}}B)}}\|\Delta\mathbf{x}_{[1,n]}^{k+1}\|^2 + \frac{3(l_g^2+\tau^2)}{\rho_{\min(B^{\mathrm{T}}B)}}\|\Delta y^{k+1}\|^2 + \frac{3\tau^2}{\rho_{\min(B^{\mathrm{T}}B)}}\|\Delta y^k\|^2. \tag{3.9}$$

For Algorithm 2, for each $k \in N$, we have

$$\|\Delta\lambda^{k+1}\|^2 \le \frac{3l_g^2}{\rho_{\min(B^{\mathrm{T}}B)}}\|\Delta\mathbf{x}_{[1,n]}^{k+1}\|^2 + \frac{3\tau^2}{\rho_{\min(B^{\mathrm{T}}B)}}\|\Delta y^{k+1}\|^2 + \frac{3(l_g^2+\tau^2)}{\rho_{\min(B^{\mathrm{T}}B)}}\|\Delta y^k\|^2. \tag{3.10}$$

Proof Using Assumption A(III) and Lemma 2.3, we have

$$\|\Delta\lambda^{k+1}\| \le \frac{1}{\sqrt{\rho_{\min(B^{\mathrm{T}}B)}}}\|B^{\mathrm{T}}\Delta\lambda^{k+1}\|. \tag{3.11}$$

For Algorithm 1, the optimal condition of y-subproblem in (3.2) yields

$$0 = \nabla_y g\big(\mathbf{x}_{[1,n]}^{k+1}, y^{k+1}\big) - B^{\mathrm{T}}\lambda^k + \beta B^{\mathrm{T}}\big(\mathbf{A}\mathbf{x}_{[1,n]}^{k+1} + By^{k+1} - b\big) + \tau\big(\Delta y^{k+1}\big).$$

Since $\lambda^{k+1} = \lambda^k - \beta(\mathbf{A}\mathbf{x}_{[1,n]}^{k+1} + By^{k+1} - b)$, we have

$$B^{\mathrm{T}}\lambda^{k+1} = \nabla_y g\big(\mathbf{x}_{[1,n]}^{k+1}, y^{k+1}\big) + \tau\big(\Delta y^{k+1}\big). \tag{3.12}$$

Let $u^k = (\mathbf{x}_{[1,n]}^k, y^k)$. Using Assumption A (I) and (3.12), we have

$$\begin{aligned}
&\|B^{\mathrm{T}}\lambda^{k+1} - B^{\mathrm{T}}\lambda^k\|^2 \\
&\quad= \|\nabla_y g\big(u^{k+1}\big) - \nabla_y g\big(u^k\big) + \tau\Delta y^{k+1} - \tau\Delta y^k\|^2 \\
&\quad= \|\nabla_y g\big(u^{k+1}\big) - \nabla_y g\big(u^k\big)\|^2 + \|\tau\Delta y^{k+1}\|^2 + \|\tau\Delta y^k\|^2 - 2\big\langle\tau\Delta y^{k+1}, \tau\Delta y^k\big\rangle \\
&\qquad - 2\big\langle\nabla_y g\big(u^{k+1}\big) - \nabla_y g\big(u^k\big), \tau\Delta y^k\big\rangle + 2\big\langle\nabla_y g\big(u^{k+1}\big) - \nabla_y g\big(u^k\big), \tau\Delta y^{k+1}\big\rangle \\
&\quad\le 3l_g^2\|\Delta u^{k+1}\|^2 + 3\tau^2\|\Delta y^{k+1}\|^2 + 3\tau^2\|\Delta y^k\|^2 \\
&\quad\le 3l_g^2\|\Delta\mathbf{x}_{[1,n]}^{k+1}\|^2 + 3\big(l_g^2+\tau^2\big)\|\Delta y^{k+1}\|^2 + 3\tau^2\|\Delta y^k\|^2.
\end{aligned} \tag{3.13}$$

It follows from the above mentioned formula and (3.11) that

$$\|\Delta\lambda^{k+1}\|^2 \le \frac{3l_g^2}{\rho_{\min(B^{\mathrm{T}}B)}}\|\Delta\mathbf{x}_{[1,n]}^{k+1}\|^2 + \frac{3(l_g^2+\tau^2)}{\rho_{\min(B^{\mathrm{T}}B)}}\|\Delta y^{k+1}\|^2 + \frac{3\tau^2}{\rho_{\min(B^{\mathrm{T}}B)}}\|\Delta y^k\|^2.$$

For Algorithm 2, similarly, we get

$$\|\Delta\lambda^{k+1}\|^2 \le \frac{3l_g^2}{\rho_{\min(B^{\mathrm{T}}B)}}\|\Delta\mathbf{x}_{[1,n]}^{k+1}\|^2 + \frac{3\tau^2}{\rho_{\min(B^{\mathrm{T}}B)}}\|\Delta y^{k+1}\|^2 + \frac{3(l_g^2+\tau^2)}{\rho_{\min(B^{\mathrm{T}}B)}}\|\Delta y^k\|^2.$$

The proof is completed. □

To brief the analysis, some notations are given below. Let $w^k = (\mathbf{x}_{[1,n]}^k, y^k, \lambda^k)$, $u^k = (\mathbf{x}_{[1,n]}^k, y^k)$, $r_k = \mathbf{A}\mathbf{x}_{[1,n]}^k + By^k - b$. The following lemma is important to prove the monotonicity of the sequence $\{\hat{L}_\beta(\hat{w}^{k+1})\}$ defined as (3.20).

Lemma 3.2 *For Algorithm 1 and Algorithm 2, select $\theta_k \in [0, \frac{1}{2})$ and τ, β large enough to assure $\tau > \frac{2+l_g}{1-2\theta_k}$, $\beta > \max\{\frac{3(l_g^2+\tau^2)}{\rho_{\min(B^{\mathrm{T}}B)}}, \frac{6(\tau^2+l_g^2)}{\tau\theta_k\rho_{\min(B^{\mathrm{T}}B)}}\}$.*

Then, for each $k \in \mathrm{N}$, we have

$$L_\beta(w^{k+1}) + \delta_2\left(\|\Delta\mathbf{x}_{[1,n]}^{k+1}\|^2 + \|\Delta y^{k+1}\|^2\right) \le L_\beta(w^k) + \delta_1\left(\|\Delta\mathbf{x}_{[1,n]}^k\|^2 + \|\Delta y^k\|^2\right), \tag{3.14}$$

where $\delta_2 > \delta_1 > 0$.

Proof We first give the proof of Algorithm 1.

From (3.1) and (3.4), for $j = 1, \dots, n$, we have

$$\begin{aligned}
&f_j(x_j^{k+1}) + \langle \Delta x_j^{k+1}, \nabla_{x_j} g(\mathbf{x}_{[1,j-1]}^{k+1}, \mathbf{x}_{[j,n]}^k, y^k)\rangle \\
&\quad - \langle \lambda^k, \mathbf{A}\mathbf{x}_{[1,j]}^{k+1} + \mathbf{A}\mathbf{x}_{[j+1,n]}^k + By^k - b\rangle + \frac{\beta}{2}\|\mathbf{A}\mathbf{x}_{[1,j-1]}^{k+1} + \mathbf{A}\mathbf{x}_{[j,n]}^k + By^k - b\|^2 \\
&\le f_j(x_j^k) - \langle \lambda^k, \mathbf{A}\mathbf{x}_{[1,j-1]}^{k+1} + \mathbf{A}\mathbf{x}_{[j,n]}^k + By^k - b\rangle \\
&\quad + \frac{\beta}{2}\|\mathbf{A}\mathbf{x}_{[1,j-1]}^{k+1} + \mathbf{A}\mathbf{x}_{[j,n]}^k + By^k - b\|^2 + \frac{\tau}{2}\|x_j^k - z_j^k\|^2 - \frac{\tau}{2}\|x_j^{k+1} - z_j^k\|^2.
\end{aligned}$$

From (3.2) and (3.5), we have

$$\begin{aligned}
&g(u^{k+1}) - \langle \lambda^k, r_{k+1}\rangle + \frac{\beta}{2}\|r_{k+1}\|^2 \\
&\quad \le g(\mathbf{x}_{[1,n]}^{k+1}, y^k) - \langle \lambda^k, \mathbf{A}\mathbf{x}_{[1,n]}^{k+1} + By^k - b\rangle + \frac{\beta}{2}\|\mathbf{A}\mathbf{x}_{[1,n]}^{k+1} + By^k - b\|^2 - \frac{\tau}{2}\|\Delta y^{k+1}\|^2.
\end{aligned}$$

Adding up the above mentioned formulas from $j = 1, \dots, n$, we have

$$\begin{aligned}
&\sum_{i=1}^n f_i(x_i^{k+1}) + g(u^{k+1}) - \langle \lambda^k, r_{k+1}\rangle + \frac{\beta}{2}\|r_{k+1}\|^2 \\
&\quad \le \sum_{i=1}^n f_i(x_i^k) + g(\mathbf{x}_{[1,n]}^{k+1}, y^k) - \langle \lambda^k, r_k\rangle + \frac{\beta}{2}\|r_k\|^2
\end{aligned}$$

$$
\begin{aligned}
&-\sum_{i=1}^{n}\left\langle \Delta x_i^{k+1}, \nabla_{x_i} g\left(\mathbf{x}_{[1,i-1]}^{k+1}, \mathbf{x}_{[i,n]}^{k}, y^k\right)\right\rangle + \frac{\tau}{2}\sum_{i=1}^{n}\left\|x_i^k - z_i^k\right\|^2 \\
&-\frac{\tau}{2}\sum_{i=1}^{n}\left\|x_i^{k+1} - z_i^k\right\|^2 - \frac{\tau}{2}\left\|y^{k+1} - y^k\right\|^2,
\end{aligned}
$$

hence

$$
\begin{aligned}
&\sum_{i=1}^{n} f_i\left(x_i^{k+1}\right) + g\left(u^{k+1}\right) - \left\langle \lambda^k, r_{k+1}\right\rangle + \frac{\beta}{2}\|r_{k+1}\|^2 \\
&\quad \le \sum_{i=1}^{n} f_i\left(x_i^k\right) + g\left(u^k\right) - \left\langle \lambda^k, r_k\right\rangle + \frac{\beta}{2}\|r_k\|^2 \\
&\qquad \underbrace{+ g\left(\mathbf{x}_{[1,n]}^{k+1}, y^k\right) - g\left(u^k\right) - \sum_{i=1}^{n}\left\langle \Delta x_i^{k+1}, \nabla_{x_i} g\left(\mathbf{x}_{[1,i-1]}^{k+1}, \mathbf{x}_{[i,n]}^{k}, y^k\right)\right\rangle}_{\mathcal{A}} \\
&\qquad \underbrace{+\frac{\tau}{2}\sum_{i=1}^{n}\left\|x_i^k - z_i^k\right\|^2 - \frac{\tau}{2}\sum_{i=1}^{n}\left\|x_i^{k+1} - z_i^k\right\|^2}_{\mathcal{B}} - \frac{\tau}{2}\left\|\Delta y^{k+1}\right\|^2.
\end{aligned}
$$

One the one hand, from Lemma 2.2, part $\mathcal{A}$ can be written as

$$
\begin{aligned}
&g\left(\mathbf{x}_{[1,n]}^{k+1}, y^k\right) - g\left(u^k\right) - \sum_{i=1}^{n}\left\langle \Delta x_i^{k+1}, \nabla_{x_i} g\left(\mathbf{x}_{[1,i-1]}^{k+1}, \mathbf{x}_{[i,n]}^{k}, y^k\right)\right\rangle \\
&\quad = \sum_{i=1}^{n}\left\{g\left(\mathbf{x}_{[1,i]}^{k+1}, \mathbf{x}_{[i+1,n]}^{k}, y^k\right)\right. \\
&\qquad \left. - g\left(\mathbf{x}_{[1,i-1]}^{k+1}, \mathbf{x}_{[i,n]}^{k}, y^k\right) - \left\langle \Delta x_i^{k+1}, \nabla_{x_i} g\left(\mathbf{x}_{[1,i-1]}^{k+1}, \mathbf{x}_{[i,n]}^{k}, y^k\right)\right\rangle\right\} \\
&\quad \le \frac{l_g}{2}\left\|\Delta \mathbf{x}_{[1,n]}^{k+1}\right\|^2.
\end{aligned} \tag{3.15}
$$

On the other hand, by the definitions of $z_i^k, i = 1, 2, \ldots, n$, we have

$$
\begin{aligned}
&\left\|x_i^k - z_i^k\right\|^2 - \left\|x_i^{k+1} - z_i^k\right\|^2 \\
&\quad = \theta_k^2\left\|x_i^{k-1} - x_i^k\right\|^2 - \left\|x_i^{k+1} - x_i^k + \theta_k\left(x_i^k - x_i^{k-1}\right)\right\|^2 \\
&\quad = -\left\|x_i^{k+1} - x_i^k\right\|^2 - 2\theta_k\left\langle x_i^k - x_i^{k+1}, x_i^k - x_i^{k-1}\right\rangle \\
&\quad \le -\left\|x_i^{k+1} - x_i^k\right\|^2 + \theta_k\left\|x_i^{k+1} - x_i^k\right\|^2 + \theta_k\left\|x_i^k - x_i^{k-1}\right\|^2 \\
&\quad = -(1-\theta_k)\left\|x_i^{k+1} - x_i^k\right\|^2 + \theta_k\left\|x_i^k - x_i^{k-1}\right\|^2.
\end{aligned}
$$

Thus, it can be inferred from part $\mathcal{B}$ that

$$
\sum_{i=1}^{n}\left\|x_i^k - z_i^k\right\|^2 - \sum_{i=1}^{n}\left\|x_i^{k+1} - z_i^k\right\|^2 \le -(1-\theta_k)\left\|\Delta \mathbf{x}_{[1,n]}^{k+1}\right\|^2 + \theta_k\left\|\Delta \mathbf{x}_{[1,n]}^{k}\right\|^2. \tag{3.16}
$$

From Lemma 2.2, (3.15) and (3.16), we obtain

$$\begin{aligned} L_\beta\left(\mathbf{x}_{[1,n]}^{k+1}, y^{k+1}, \lambda^k\right) \le L_\beta\left(w^k\right) &+ \frac{l_g}{2}\left\|\Delta\mathbf{x}_{[1,n]}^{k+1}\right\|^2 - \frac{\tau(1-\theta_k)}{2}\left\|\Delta\mathbf{x}_{[1,n]}^{k+1}\right\|^2 \\ &- \frac{\tau}{2}\left\|\Delta y^{k+1}\right\|^2 + \frac{\tau\theta_k}{2}\left\|\Delta\mathbf{x}_{[1,n]}^{k+1}\right\|^2. \end{aligned} \tag{3.17}$$

Recall that

$$\begin{aligned} L_\beta\left(w^{k+1}\right) &= L_\beta\left(\mathbf{x}_{[1,n]}^{k+1}, y^{k+1}, \lambda^k\right) + \left\langle\Delta\lambda^{k+1}, r_{k+1}\right\rangle \\ &= L_\beta\left(\mathbf{x}_{[1,n]}^{k+1}, y^{k+1}, \lambda^k\right) + \frac{1}{\beta}\left\langle\Delta\lambda^{k+1}, \Delta\lambda^{k+1}\right\rangle \\ &\le L_\beta\left(\mathbf{x}_{[1,n]}^{k+1}, y^{k+1}, \lambda^k\right) + \frac{1}{\beta}\left\|\Delta\lambda^{k+1}\right\|^2. \end{aligned} \tag{3.18}$$

Submitting (3.9) and (3.17) into (3.18), we have

$$\begin{aligned} &L_\beta\left(w^{k+1}\right) \\ &\quad\le L_\beta\left(w^k\right) + \frac{l_g}{2}\left\|\Delta\mathbf{x}_{[1,n]}^{k+1}\right\|^2 - \frac{\tau(1-\theta_k)}{2}\left\|\Delta\mathbf{x}_{[1,n]}^{k+1}\right\|^2 - \frac{\tau}{2}\left\|\Delta y^{k+1}\right\|^2 + \frac{\tau}{2}\theta_k\left\|\Delta\mathbf{x}_{[1,n]}^{k}\right\|^2 \\ &\qquad + \frac{3l_g^2}{\beta\rho_{\min(B^{\mathrm{T}}B)}}\left\|\Delta\mathbf{x}_{[1,n]}^{k}\right\|^2 + \frac{3(l_g^2+\tau^2)}{\beta\rho_{\min(B^{\mathrm{T}}B)}}\left\|\Delta y^{k+1}\right\|^2 + \frac{3\tau^2}{\beta\rho_{\min(B^{\mathrm{T}}B)}}\left\|\Delta y^{k}\right\|^2 \\ &\quad= L_\beta\left(w^k\right) - \left(\frac{\tau(1-\theta_k)}{2} - \frac{l_g}{2} - \frac{3l_g^2}{\beta\rho_{\min(B^{\mathrm{T}}B)}}\right)\left\|\Delta\mathbf{x}_{[1,n]}^{k+1}\right\|^2 \\ &\qquad - \left(\frac{\tau}{2} - \frac{3(\tau^2+l_g^2)}{\beta\rho_{\min(B^{\mathrm{T}}B)}}\right)\left\|\Delta y^{k+1}\right\|^2 + \frac{\tau\theta_k}{2}\left\|\Delta\mathbf{x}_{[1,n]}^{k}\right\|^2 + \frac{3\tau^2}{\beta\rho_{\min(B^{\mathrm{T}}B)}}\left\|\Delta y^{k}\right\|^2. \end{aligned}$$

Hence,

$$\begin{aligned} &L_\beta\left(w^{k+1}\right) + \left(\frac{\tau(1-\theta_k)}{2} - \frac{l_g}{2} - \frac{3l_g^2}{\beta\rho_{\min(B^{\mathrm{T}}B)}}\right)\left\|\Delta\mathbf{x}_{[1,n]}^{k+1}\right\|^2 + \left(\frac{\tau}{2} - \frac{3(\tau^2+l_g^2)}{\beta\rho_{\min(B^{\mathrm{T}}B)}}\right)\left\|\Delta y^{k+1}\right\|^2 \\ &\quad\le L_\beta\left(w^k\right) + \frac{\tau\theta_k}{2}\left\|\Delta\mathbf{x}_{[1,n]}^{k}\right\|^2 + \frac{3\tau^2}{\beta\rho_{\min(B^{\mathrm{T}}B)}}\left\|\Delta y^{k}\right\|^2 \\ &\quad\le L_\beta\left(w^k\right) + \frac{\tau\theta_k}{2}\left\|\Delta\mathbf{x}_{[1,n]}^{k}\right\|^2 + \frac{3(l_g^2+\tau^2)}{\beta\rho_{\min(B^{\mathrm{T}}B)}}\left\|\Delta y^{k}\right\|^2. \end{aligned}$$

Since $\beta > \max\{\frac{3(l_g^2+\tau^2)}{\rho_{\min(B^{\mathrm{T}}B)}}, \frac{3(\tau^2+l_g^2)}{\tau\theta_k\rho_{\min(B^{\mathrm{T}}B)}}\}$, which further implies $\frac{6(l_g^2+\tau^2)}{\beta\rho_{\min(B^{\mathrm{T}}B)}} < 1$ and $\frac{\tau\theta_k}{2} > \frac{3(\tau^2+l_g^2)}{\beta\rho_{\min(B^{\mathrm{T}}B)}}$, then have

$$\begin{aligned} &L_\beta\left(w^{k+1}\right) + \left(\frac{\tau(1-\theta_k)}{2} - \frac{l_g}{2} - 1\right)\left\|\Delta\mathbf{x}_{[1,n]}^{k+1}\right\|^2 + \left(\frac{\tau}{2} - 1\right)\left\|\Delta y^{k+1}\right\|^2 \\ &\quad\le L_\beta\left(w^k\right) + \frac{\tau\theta_k}{2}\left\|\Delta\mathbf{x}_{[1,n]}^{k}\right\|^2 + \frac{3(\tau^2+l_g^2)}{\beta\rho_{\min(B^{\mathrm{T}}B)}}\left\|\Delta y^{k}\right\|^2 \\ &\quad\le L_\beta\left(w^k\right) + \frac{\tau\theta_k}{2}\left\|\Delta\mathbf{x}_{[1,n]}^{k}\right\|^2 + \frac{\tau\theta_k}{2}\left\|\Delta y^{k}\right\|^2. \end{aligned}$$

Let $\delta_2 = \frac{\tau(1-\theta_k)}{2} - \frac{l_g}{2} - 1, \delta_1 = \frac{\tau}{2}\theta_k$. We get

$$L_\beta\left(w^{k+1}\right) + \delta_2\left(\left\|\Delta \mathbf{x}_{[1,n]}^{k+1}\right\|^2 + \left\|\Delta y^{k+1}\right\|^2\right) \le L_\beta\left(w^k\right) + \delta_1\left(\left\|\Delta \mathbf{x}_{[1,n]}^{k}\right\|^2 + \left\|y^k\right\|^2\right). \tag{3.19}$$

Since $\tau > \frac{2+l_g}{1-2\theta_k}$, which further implies that $\frac{\tau(1-\theta_k)}{2} - \frac{l_g}{2} - 1 > \frac{\tau\theta_k}{2}$, we obtain $\delta_2 > \delta_1 > 0$. That is, (3.14) holds.

Similarly, for Algorithm 2, we obtain

$$\begin{aligned} &L_\beta\left(w^{k+1}\right) + \left(\frac{\tau(1-\theta_k)}{2} - \frac{l_g}{2} - \frac{3l_g^2}{\beta\rho_{\min(B^{\mathrm{T}}B)}}\right)\left\|\Delta \mathbf{x}_{[1,n]}^{k+1}\right\|^2 \\ &\quad + \left(\frac{\tau}{2} - \frac{l_g}{2} - \frac{3\tau^2}{\beta\rho_{\min(B^{\mathrm{T}}B)}}\right)\left\|\Delta y^{k+1}\right\|^2 \\ &\le L_\beta\left(w^k\right) + \frac{\tau\theta_k}{2}\left\|\Delta \mathbf{x}_{[1,n]}^{k}\right\|^2 + \frac{3(l_g^2+\tau^2)}{\beta\rho_{\min(B^{\mathrm{T}}B)}}\left\|\Delta y^k\right\|^2. \end{aligned}$$

Since $\beta > \max\{\frac{3(l_g^2+\tau^2)}{\rho_{\min(B^{\mathrm{T}}B)}}, \frac{6(\tau^2+l_g^2)}{\tau\theta_k\rho_{\min(B^{\mathrm{T}}B)}}\}$, which further implies $\frac{3(l_g^2+\tau^2)}{\beta\rho_{\min(B^{\mathrm{T}}B)}} < 1$ and $\frac{\tau\theta_k}{2} > \frac{3(\tau^2+l_g^2)}{\beta\rho_{\min(B^{\mathrm{T}}B)}}$, it follows that

$$\begin{aligned} &L_\beta\left(w^{k+1}\right) + \left(\frac{\tau(1-\theta_k)}{2} - \frac{l_g}{2} - 1\right)\left\|\Delta \mathbf{x}_{[1,n]}^{k+1}\right\|^2 + \left(\frac{\tau(1-\theta_k)}{2} - \frac{l_g}{2} - 1\right)\left\|\Delta y^{k+1}\right\|^2 \\ &\le L_\beta\left(w^k\right) + \frac{\tau\theta_k}{2}\left\|\Delta \mathbf{x}_{[1,n]}^{k}\right\|^2 + \frac{\tau\theta_k}{2}\left\|\Delta y^k\right\|^2. \end{aligned}$$

Let $\delta_2 = \frac{\tau(1-\theta_k)}{2} - \frac{l_g}{2} - 1, \delta_1 = \frac{\tau}{2}\theta_k$. We have

$$L_\beta\left(w^{k+1}\right) + \delta_2\left(\left\|\Delta \mathbf{x}_{[1,n]}^{k+1}\right\|^2 + \left\|\Delta y^{k+1}\right\|^2\right) \le L_\beta\left(w^k\right) + \delta_1\left(\left\|\Delta \mathbf{x}_{[1,n]}^{k}\right\|^2 + \left\|\Delta y^k\right\|^2\right).$$

Since $\tau > \frac{2+l_g}{1-2\theta_k}$, which further implies that $\frac{\tau(1-\theta_k)}{2} - \frac{l_g}{2} - 1 > \frac{\tau\theta_k}{2}$, then we get $\delta_2' > \delta_1' > 0$. That is, (3.14) holds. The lemma is proved. □

Remark 2 Based on Lemma 3.2, we can define the following function

$$\hat{L}_\beta(\hat{w}) = \hat{L}_\beta(u,\lambda,v) = L_\beta(u,\lambda) + \delta_1\|u-v\|^2, \tag{3.20}$$

where

$$u = (\mathbf{x}_{[1,n]}, y), v = (\tilde{\mathbf{x}}_{[1,n]}, \tilde{y}), \hat{w} = (u,\lambda,v) = (\mathbf{x}_{[1,n]}, y, \lambda, \tilde{\mathbf{x}}_{[1,n]}, \tilde{y})$$

and

$$\|u-v\|^2 = \|\mathbf{x}_{[1,n]} - \tilde{\mathbf{x}}_{[1,n]}\|^2 + \|y-\tilde{y}\|^2.$$

Set $\hat{\omega}^{k+1} = (\mathbf{x}_{[1,n]}^{k+1}, y^{k+1}, \lambda^{k+1}, \mathbf{x}_{[1,n]}^{k}, y^k), u^{k+1} = (\mathbf{x}_{[1,n]}^{k+1}, y^{k+1})$. Thus,

$$\hat{L}_\beta\left(\hat{\omega}^{k+1}\right) = \hat{L}_\beta\left(u^{k+1}, \lambda^{k+1}, u^k\right) = L_\beta\left(u^{k+1}, \lambda^{k+1}\right) + \delta_1\left(\left\|\Delta u^{k+1}\right\|^2\right). \tag{3.21}$$

The following lemma implies that the sequence $\hat{L}_\beta(u^k, \lambda^k, u^{k-1})$ is decreasing monotonically.

Lemma 3.3 *Suppose $\hat{L}_\beta(\hat{\omega}^{k+1})$ is defined as (3.20). Then, under Assumption A, for Algorithm 1 and Algorithm 2, we have*:

$$\hat{L}_\beta(\hat{\omega}^{k+1}) + \delta(\|\Delta u^{k+1}\|^2) \le \hat{L}_\beta(\hat{w}^k). \tag{3.22}$$

That is, the sequence $\{\hat{L}_\beta(\hat{\omega}^{k+1})\}$ is decreasing.

Proof Set $\delta = \delta_2 - \delta_1 > 0$. Then the result follows directly from Lemma 3.2. □

3.3 The cluster points of $\{\omega_k\}$ are contained in *critL*

In this subsection, together with the closeness of the limiting subdifferential mentioned above, we prove the subsequential convergence of the sequence $\{\omega^k\}$. The proof of Algorithm 2 is similar to that of Algorithm 1, so we omit the proof of Algorithm 2 here.

Lemma 3.4 *Suppose $\{\omega^k\}$ is the sequence generated by Algorithm 1. If Assumption A holds, then the following statements are true*:

(*I*) *The sequence $\{\omega^k\}$ is bounded.* (*II*) *$\hat{L}_\beta(\hat{\omega}^k)$ is bounded from below and convergent, additionally,*

$$\sum_{k\ge 0} \|\omega^{k+1} - \omega^k\|^2 < +\infty. \tag{3.23}$$

(*III*) *The sequences $\hat{L}_\beta(\hat{\omega}^k)$ and $L_\beta(\omega^k)$ have the same limit $\hat{L}_*$.*

Proof (I) Because of the decreasing property of $\{\hat{L}_\beta(\hat{\omega}^k)\}$, we get

$$L_\beta(\omega^k) \le \hat{L}_\beta(\hat{\omega}^k) \le \hat{L}_\beta(\hat{\omega}^0) = L_\beta(\omega^0) + \delta(\|u^0 - u^{-1}\|^2) = L_\beta(\omega^0),$$

where $\|u^0 - u^{-1}\|^2$ is due to the Initialization parameters $x_i^0 = x_i^{-1}, i = 1, \dots, n$ and $y^0 = y^{-1}$ in Algorithm 1. Hence, $\{\omega^k\} \subseteq \{\omega^k \in X : L_\beta(\omega) \le L_\beta(\omega^0)\}$. By Assumption A(V), the sequence $\{\omega^k\}$ is bounded.

(II) Since $\{\omega^k\}$ is bounded, $\{\hat{\omega}^k\}$ is also bounded, and it has at least one cluster point. Let $\hat{\omega}^*$ be a cluster point of $\{\hat{\omega}^k\}$, and $\lim_{j\to+\infty} \hat{\omega}^{k_j} = \hat{\omega}^*$. Because of the fact that $f_i (i = 1, 2, \dots, n)$ are proper lower semicontinuous, and g is continuously differentiable, then $\hat{L}_\beta(\cdot)$ is proper lower semicontinuous. Hence, we have

$$\lim_{j\to+\infty} \inf \hat{L}_\beta(\hat{\omega}^{k_j}) \ge \hat{L}_\beta(\hat{\omega}^*).$$

According to the boundedness of f_i, g, $\{\omega^k\}_{k\ge 0}$ and the definition of $\hat{L}_\beta(\hat{\omega}^k)$, we have $\hat{L}_\beta(\omega^k)$ is bounded from below. Thus, $\hat{L}_\beta(\hat{\omega}^{k_j})$ is also bounded from below. From Lemma 3.3, $\hat{L}_\beta(\hat{\omega}^k)$ is monotonically decreasing, and we obtain that $\hat{L}_\beta(\hat{\omega}^{k_j})$ is convergent. Since $\hat{L}_\beta(\hat{\omega}^k)$ is monotonically decreasing, $\hat{L}_\beta(\hat{\omega}^k)$ is also convergent and $\hat{L}_\beta(\hat{\omega}^*) \le \hat{L}_\beta(\hat{\omega}^k)$. It follows from (3.22) that

$$\delta(\|\Delta u^{k+1}\|^2) \le \hat{L}_\beta(\hat{w}^k) - \hat{L}_\beta(\hat{w}^{k+1}).$$

Summing up the above inequality for $k = 0, \dots, N$ and letting $N \to \infty$, we have

$$\delta \sum_{k=1}^{+\infty} \left(\left\| \Delta \mathbf{x}_{[1,n]}^{k+1} \right\|^2 + \left\| \Delta y^{k+1} \right\|^2 \right) \le \hat{L}_\beta(\hat{w}^0) - \hat{L}_\beta(\hat{w}^*) < +\infty.$$

Since $\delta > 0$, it follows that

$$\sum_{k=1}^{+\infty} \left\| \Delta \mathbf{x}_{[1,n]}^{k+1} \right\|^2 < +\infty, \qquad \sum_{k=1}^{+\infty} \left\| \Delta y^{k+1} \right\|^2 < +\infty. \tag{3.24}$$

Consequently, due to (3.9), we have

$$\sum_{k=1}^{+\infty} \left\| \lambda^{k+1} - \lambda^k \right\|^2 < +\infty. \tag{3.25}$$

Then, $\sum_{k=1}^{\infty} \|\omega^{k+1} - \omega^k\|^2 < +\infty$.

(III) From (3.24), we have $\|\Delta \mathbf{x}_{[1,n]}^{k+1}\|^2 \to 0$ and $\|\Delta y^{k+1}\|^2 \to 0$. Combining with the definition of $\hat{L}_\beta(\hat{w}^k)$ in (3.21) yields $\hat{L}_* = \lim_{k\to+\infty} \hat{L}_\beta(\hat{\omega}^k) = \lim_{k\to+\infty} L_\beta(\omega^k)$. The lemma is proved. □

The following lemma provides upper estimates for the limiting subgradients of $\hat{L}_\beta(\cdot)$, which is important for the convergence analysis of the sequence generated by Algorithm 1 and Algorithm 2.

Lemma 3.5 *Let $\{\omega^k\}$ be a sequence generated by Algorithm 1. Then, there exists $C > 0$ such that*

$$d\left(0, \partial L_\beta\left(\omega^{k+1}\right)\right) \le C\left(\sum_{i=1}^{n} \left\| \Delta x_i^{k+1} \right\| + \left\| \Delta y^{k+1} \right\| + \sum_{i=1}^{n} \left\| \Delta x_i^k \right\| + \left\| \Delta y^k \right\| \right). \tag{3.26}$$

Proof By the definition of the augmented Lagrangian function $L_\beta(\cdot)$, we have

$$\begin{cases} \partial_{x_j} L_\beta(u^{k+1}, \lambda^{k+1}) = \partial f_j(x_j^{k+1}) + \nabla_{x_j} g(\mathbf{x}_{[1,n]}^{k+1}, y^{k+1}) - A_j^T(\lambda^{k+1} - \beta r^{k+1}), \\ \partial_y L_\beta(u^{k+1}, \lambda^{k+1}) = \nabla_y g(\mathbf{x}_{[1,n]}^{k+1}, y^{k+1}) - B^T \lambda^{k+1} + \beta B^T r^{k+1}, \\ \partial_\lambda L_\beta(u^{k+1}, \lambda^{k+1}) = \frac{1}{\beta}(\lambda^k - \lambda^{k+1}). \end{cases} \tag{3.27}$$

From the optimality conditions of (3.1)–(3.2), we have

$$\begin{cases} -\nabla_{x_j} g(\mathbf{x}_{[1,j-1]}^{k+1}, \mathbf{x}_{[j,n]}^k, y^k) + A_j^T \lambda^{k+1} - \beta A_j^T \Delta \mathbf{A}\mathbf{x}_{[j+1,n]}^{k+1} - \beta A_j^T B(y^k - y^{k+1}) \\ \quad - \tau(x_j^{k+1} - z_j^k) \in \partial f_j(x_j^{k+1}), \\ B^{\mathrm{T}} \lambda^{k+1} - \tau(y^{k+1} - y^k) = \nabla_y g(u^{k+1}), \\ \lambda^{k+1} = \lambda^k - \beta(\mathbf{A}\mathbf{x}_{[1,n]}^{k+1} + By^{k+1} - b), \end{cases} \tag{3.28}$$

where $\Delta \mathbf{A}\mathbf{x}_{[j+1,n]}^{k+1} = \mathbf{A}\mathbf{x}_{[j+1,n]}^{k+1} - \mathbf{A}\mathbf{x}_{[j+1,n]}^k$. Putting (3.28) into (3.27), we have

$$\left(\rho_1^{k+1}, \rho_2^{k+1}, \dots, \rho_n^{k+1}, \rho_{n+1}^{k+1}, \rho_{n+2}^{k+1}\right)^T \in \partial L_\beta\left(x_1^{k+1}, x_2^{k+1}, \dots, x_n^{k+1}, y^{k+1}, \lambda^{k+1}\right),$$

where

$$\begin{cases} \rho_j^{k+1} = \nabla_{x_j} g(\mathbf{x}_{[1,n]}^{k+1}, y^{k+1}) - \nabla_{x_j} g(\mathbf{x}_{[1,j-1]}^{k+1}, \mathbf{x}_{[j,n]}^{k}, y^k) + A_j^T(\lambda^k - \lambda^{k+1}) \\ \qquad + \beta A_j^T \Delta \mathbf{A}\mathbf{x}_{[j+1,n]}^{k+1} + \beta A_j^T B(y^k - y^{k+1}) - \tau(x_j^{k+1} - z_j^k), (j = 1, \dots, n), \\ \rho_{n+1}^{k+1} = \beta B^{\mathrm{T}}(\lambda^k - \lambda^{k+1}) - \tau(y^{k+1} - y^k), \\ \rho_{n+2}^{k+1} = \frac{1}{\beta}(\lambda^k - \lambda^{k+1}). \end{cases} \tag{3.29}$$

Since ∇g is Lipschitz continuous on bounded subsets and $\{\omega^k\}$ is bounded, by (III) of Assumption A, combining (3.14), there exists $C > 0$ such that

$$d\big(0, \partial L_\beta\big(\omega^{k+1}\big)\big) \le C\left(\sum_{i=1}^n \big\|\Delta x_i^{k+1}\big\| + \big\|\Delta y^{k+1}\big\| + \sum_{i=1}^n \big\|\Delta x_i^k\big\| + \big\|\Delta y^k\big\|\right).$$

Similarly, we can derive the same conclusion for Algorithm 2. We omit the proof here. □

Theorem 3.1 *Denote the set of the cluster points of the sequence* $\{\omega^k\}$ *and* $\{\hat{\omega}^k\}$ *by* Ω *and* $\hat{\Omega}$, *respectively. We have that*:

(*I*) *If* ω^* *is a cluster of* $\{\omega^k\}$, *then it has a convergent subsequence* $\{\omega^{k_j}\}_{j\ge 0}$ *such that* $\lim_{j\to+\infty} w^{k_j} = w^*$, *then*

$$\lim_{j\to\infty} L_\beta\big(\omega^{k_j}\big) = L_\beta\big(\omega^*\big).$$

(*II*) $\Omega \subseteq crit L_\beta$.

(*III*) $\lim_{k\to+\infty} d(\omega^k, \Omega)$.

(*IV*) $\{\omega^k\}$ *is non-empty compact and connected sets.*

Proof (I) Since $x_i^{k_j+1}$ is the minimizer of x_i-subproblem, we have

$$\begin{aligned} & f_i\big(x_i^{k_j+1}\big) + \big\langle x_i^{k_j+1} - x_i^{k_j}, \nabla_{x_i} g\big(\mathbf{x}_{[1,i-1]}^{k_j+1}, \mathbf{x}_{[i,n]}^{k_j}, y^{k_j}\big)\big\rangle - \big\langle \lambda^{k_j}, \mathbf{A}\mathbf{x}_{[1,i]}^{k_j+1} + \mathbf{A}\mathbf{x}_{[i+1,n]}^{k_j} + By^{k_j} - b\big\rangle \\ & \quad + \frac{\beta}{2}\big\|\mathbf{A}\mathbf{x}_{[1,i]}^{k_j+1} + \mathbf{A}\mathbf{x}_{[i+1,n]}^{k_j} + By^{k_j} - b\big\|^2 \\ & \le f_i\big(x_i^*\big) + \big\langle x_i^* - x_i^{k_j}, \nabla_{x_i} g\big(\mathbf{x}_{[1,i-1]}^{k_j+1}, \mathbf{x}_{[i,n]}^{k_j}, y^{k_j}\big)\big\rangle \\ & \quad - \big\langle \lambda^{k_j}, \mathbf{A}\mathbf{x}_{[1,i-1]}^{k_j+1} + A_i x_i^* + \mathbf{A}\mathbf{x}_{[i+1,n]}^{k_j} + By^{k_j} - b\big\rangle \\ & \quad + \frac{\beta}{2}\big\|\mathbf{A}\mathbf{x}_{[1,i-1]}^{k_j+1} + A_i x_i^* + \mathbf{A}\mathbf{x}_{[i+1,n]}^{k_j} + By^{k_j} - b\big\|^2 + \frac{\tau}{2}\big\|x_i^* - z_i^{k_j}\big\|^2 - \frac{\tau}{2}\big\|x_i^{k_j+1} - z_i^{k_j}\big\|^2. \end{aligned}$$

Combing the inequality above with $\lim_{j\to\infty} \omega^{k_j+1} = \omega^*$, we have

$$\limsup_{j\to\infty} f_i\big(x_i^{k_j+1}\big) \le f_i\big(x^*\big).$$

Since $f_i (i = 1, \dots, n)$ is lower semicontinous, $f_i(x_i^*) \le \liminf_{j\to\infty} f_i(x_i^{k_j+1})$. It follows that

$$\lim_{j\to\infty} f_i\big(x_i^{k_j+1}\big) = f_i\big(x^*\big).$$

Since g is continuous, we further obtain

$$\begin{aligned}
&\lim_{j\to+\infty} L_\beta\left(\omega^{k_j}\right)\\
&\quad= \lim_{j\to+\infty}\left(\sum_{i=1}^{n} f_i\left(x_i^{k_j}\right) + g\left(\mathbf{x}_{[1,n]}^{k_j}, y^{k_j}\right) - \left\langle \lambda^{k_j}, \mathbf{A}\mathbf{x}_{[1,n]}^{k_j} + By^{k_j} - b\right\rangle\right.\\
&\qquad\left. + \frac{\beta}{2}\left\|\mathbf{A}\mathbf{x}_{[1,n]}^{k_j} + By^{k_j} - b\right\|^2\right)\\
&\quad= \sum_{i=1}^{n} f_i\left(x_i^*\right) + g\left(\mathbf{x}_{[1,n]}^*, y^*\right) - \left\langle \lambda^*, \mathbf{A}\mathbf{x}_{[1,n]}^* + By^* - b\right\rangle + \frac{\beta}{2}\left\|\mathbf{A}\mathbf{x}_{[1,n]}^* + By^* - b\right\|^2\\
&\quad= L_\beta\left(\omega^*\right).
\end{aligned}$$

(II) From Lemma 3.4, we have that $x_i^{k+1} - \imath^k \to 0, y^{k+1} - y^k \to 0$ and $\lambda^{k+1} - \lambda^k \to 0$. Thus, according to Lemma 3.5, it follows that $\partial L_(\omega^{k_j}) \to 0$ as $j \to \infty$, while $\omega^{k_j} \to \omega^*$ and $L_\beta(\omega^{k_j}) \to L_\beta(\omega^*)$ as $j \to \infty$. Because of the closeness of ∂f_i, the continuity of ∇g and the relation above, we take limit $k = k_j \to \infty$ in (3.28), and then we have

$$\begin{cases}
-\nabla_{x_j} g(\mathbf{x}_{[1,n]}^*, y^*) + A_j^{\mathrm{T}}\lambda^* \in \partial f_j(x_j^*), \quad j = 1, \ldots, n,\\
\nabla_y g(\mathbf{x}_{[1,n]}^*, y^*) = B^{\mathrm{T}}\lambda^*,\\
\mathbf{A}\mathbf{x}_{[1,n]}^* + By^* - b = 0,
\end{cases}$$

which implies that ω^* is a critial point of $L_\beta(\cdot)$. According to (3.23), $\{\omega^k\}$ is convergent. Thus, ω^* is a cluster point of $\{\omega^k\}$, i.e., $\Omega \subseteq critL_\beta$.

(III), (IV) The proof follows a similar approach to that of [Theorems 5(ii) and (iii) in Bolte et al. [19]], while incorporating the insights from Remark 5 within the same reference. This remark establishes that the properties detailed in (III) and (IV) are inherent to sequences satisfying the convergence condition $w^{k+1} - w^k \to 0$ as $k \to +\infty$. Such generic nature is indeed applicable in our context, as demonstrated by (3.23). □

3.4 Global convergence under the Kurdyka–Łojasiewicz property

In this subsection, we prove the global convergence of $\{(\mathbf{x}_{[1,n]}, y^k, \lambda^k)\}$ generated by Algorithm 1 and Algorithm 2 with the help of the Kurdyka–Łojasiewicz property. Since the proofs of two algorithms are identical, in this subsection, we only prove the global convergence of Algorithm 1.

Theorem 3.2 (Global convergence) *Suppose that Assumption A holds, and $\hat{L}(\hat{\omega})$ satisfies the KŁ property at each point of $\hat{\Omega}$, then*

(I) $\sum_{k=1}^{\infty} \|\omega^k - \omega^{k-1}\| < \infty$.

(II) $\{\omega^k\}$ *converges to a critical point of* $L(\cdot)$.

Proof From Theorem 3.1, we have $\lim_{k\to+\infty} \hat{L}(\hat{\omega}^k) = \hat{L}(\hat{\omega}^*)$ for all $\hat{\omega}^* \in \hat{\Omega}$. We consider two cases.

(i) If there exists an integer k_0 such that $\hat{L}_\beta(\hat{\omega}^{k_0}) = \hat{L}_\beta(\hat{\omega}^*)$. From Lemma 3.3, for all $k > k_0$, we have

$$\delta\left(\|\Delta \mathbf{x}_{[1,n]}\|^2 + \left\|\Delta y^{k+1}\right\|^2\right) \le \hat{L}_\beta\left(\hat{\omega}^k\right) - \hat{L}_\beta\left(\hat{\omega}^{k+1}\right) \le \hat{L}_\beta\left(\hat{\omega}^{k_0}\right) - \hat{L}_\beta\left(\hat{\omega}^*\right) = 0. \tag{3.30}$$

Thus, for any $k > k_0$, we have $x_i^{k+1} = x_i^k, i = 1, 2, \ldots, n, y^{k+1} = y^k$. Hence, for any $k > k_0 + 1$, one has $\hat{\omega}^{k+1} = \hat{\omega}^k$, and the assertion holds.

(ii) Since $\{\hat{L}_\beta(\hat{\omega}^k)\}$ is nonincreasing, it holds that $\hat{L}_\beta(\hat{\omega}^k) > \hat{L}_\beta(\hat{\omega}^*)$ for all $k > 1$. Since $\lim_{k\to+\infty} d(\hat{\omega}^k, \hat{\Omega}) = 0$, for any given $\varepsilon > 0$, there exists $k_1 > 0$, such that for any $k > k_1$, $d(\hat{\omega}^k, \hat{\Omega}) < \varepsilon$. Since $\lim_{k_j\to+\infty} \hat{L}_\beta(\hat{\omega}^k) = \hat{L}_\beta(\hat{\omega}^*)$, for any given $\eta > 0$, there exists $k_2 > 0, \hat{L}_\beta(\hat{\omega}^k) < \hat{L}_\beta(\hat{\omega}^*) + \eta$, for all $k > k_2$. Consequently, when $k > \tilde{k} := \max\{k_1, k_2\}$,

$$d\left(\hat{\omega}^k, \hat{\Omega}\right) < \varepsilon, \hat{L}_\beta\left(\hat{\omega}^*\right) < \hat{L}_\beta\left(\hat{\omega}^k\right) < \hat{L}_\beta\left(\hat{\omega}^*\right) + \eta. \tag{3.31}$$

Since $\{\hat{\omega}^k\}$ is non-empty compact set, and $\hat{L}_\beta(\cdot)$ is constant on $\hat{\Omega}$, applying Lemma 2.1, we have

$$\varphi'\left(\hat{L}_\beta\left(\hat{\omega}^k\right) - \hat{L}_\beta\hat{\omega}^*\right) d\left(0, \partial\hat{L}_\beta\left(\hat{\omega}^k\right)\right) \ge 1, \quad \forall k > \tilde{k}. \tag{3.32}$$

Let $a_k := \sum_{i=1}^n \|\Delta x_i^k\| + \|\Delta y^k\|$. $\forall k > \tilde{k}$. From Lemma (3.5), one has

$$\frac{1}{\varphi'(\hat{L}_\beta(\hat{\omega}^k) - \hat{L}_\beta(\hat{\omega}^*))} \le d\left(0, \partial\hat{L}_\beta\left(\hat{\omega}^k\right)\right) \le C_2(a_k + a_{k+1}). \tag{3.33}$$

From the concavity of φ, we have

$$\begin{aligned} &\varphi\left(\hat{L}_\beta\left(\hat{\omega}^k\right) - \hat{L}_\beta\left(\hat{\omega}^*\right)\right) - \varphi\left(\hat{L}_\beta\left(\hat{\omega}^{k+1}\right) - \hat{L}_\beta\left(\hat{\omega}^*\right)\right) \\ &\quad \ge \varphi'\left(\hat{L}_\beta\left(\hat{\omega}^k\right) - \hat{L}_\beta\left(\hat{\omega}^*\right)\right)\left(\hat{L}_\beta\left(\hat{\omega}^k\right) - \hat{L}_\beta\left(\hat{\omega}^{k+1}\right)\right) \\ &\quad \ge \frac{\hat{L}_\beta(\hat{\omega}^k) - \hat{L}_\beta(\hat{\omega}^{k+1})}{d(0, \partial\hat{L}_\beta(\hat{\omega}^k))} \ge \frac{\hat{L}_\beta(\hat{\omega}^k) - \hat{L}_\beta(\hat{\omega}^{k+1})}{C(a_k + a_{k+1})}. \end{aligned} \tag{3.34}$$

From Lemma 3.3, we have

$$\begin{aligned} &\delta\left(\left\|\Delta \mathbf{x}_{[1,n]}^{k+1}\right\|^2 + \left\|\Delta y^{k+1}\right\|^2\right) \\ &\quad \le \left(\varphi\left(\hat{L}_\beta\left(\hat{\omega}^k\right) - \hat{L}_\beta\left(\hat{\omega}^*\right)\right) - \varphi\left(\hat{L}_\beta\left(\hat{\omega}^{k+1}\right) - \hat{L}_\beta\left(\hat{\omega}^*\right)\right)\right) C(a_k + a_{k+1}). \end{aligned}$$

From the inequality $\sum_{i=1}^n a_i \le \sqrt{n\sum_{i=1}^n a_i^2}$ and $\sqrt{ab} \le a + \frac{1}{4}b$, we obtain

$$\begin{aligned} a_{k+1} &\le \left((n+1)\left\|\Delta \mathbf{x}_{[1,n]}^{k+1}\right\|^2 + (n+1)\left\|\Delta y^{k+1}\right\|^2\right)^{\frac{1}{2}} \\ &\le \sqrt{\frac{C(n+1)}{\delta}\left(\varphi\left(\hat{L}_\beta\left(\hat{\omega}^k\right) - \hat{L}_\beta\left(\hat{\omega}^*\right)\right) - \varphi\left(\hat{L}_\beta\left(\hat{\omega}^{k+1}\right) - \hat{L}_\beta\left(\hat{\omega}^*\right)\right)\right)(a_k + a_{k+1})} \\ &\le \underbrace{\sqrt{\frac{C(n+1)}{\delta}}\left(\varphi\left(\hat{L}_\beta\left(\hat{\omega}^k\right) - \hat{L}_\beta\left(\hat{\omega}^*\right)\right) - \varphi\left(\hat{L}_\beta\left(\hat{\omega}^{k+1}\right) - \hat{L}_\beta\left(\hat{\omega}^*\right)\right)\right)}_{a} + \frac{1}{4}\underbrace{(a_k + a_{k+1})}_{b}. \end{aligned}$$

Summing up the above inequality from $k = k' + 2, \ldots, M$ yields

$$\sum_{k=k'+2}^{M} a_{k+1} \leq \sqrt{\frac{C(n+1)}{\delta}}\left(\varphi\left(\hat{L}_\beta\left(\hat{\omega}^{k'+2}\right) - \hat{L}_\beta\left(\hat{\omega}^*\right)\right) - \varphi\left(\hat{L}_\beta\left(\hat{\omega}^{M}\right) - \hat{L}_\beta\left(\hat{\omega}^*\right)\right)\right) + \frac{1}{4}\sum_{k=k'+1}^{M}(a_k + a_{k+1}).$$

Letting $M \to \infty$, we get

$$\sum_{k=k'+2}^{\infty} a_{k+1} \leq 2\sqrt{\frac{C(n+1)}{\delta}}\left(\varphi\left(\hat{L}_\beta\left(\hat{\omega}^{k'+2}\right) - \hat{L}_\beta\left(\hat{\omega}^*\right)\right)\right) - \frac{1}{2}a_{k'+1}.$$

Since $\delta, C > 0$ and $a_{k'+1}$ is a constant, $\sum_{k=k'+2}^{\infty} a_{k+1} < \infty$. Therefore, $\sum_{k=1}^{\infty} \|\omega^{k+1} - \omega^k\| < \infty$. (I) is proved.

(II) $\{\omega^k\}$ is a Cauchy sequence, and thus it is convergent. Combining (I) with Theorem 3.1, we obtain that $\{\omega^k\}$ converges to a critical point of $L_\beta(\cdot)$. □

4 Numerical experiments

This section presents the numerical experiment outcomes of applying Algorithm 1 and Algorithm 2 to $l_{\frac{1}{2}}$-regularization problem and matrix decomposition problem. All experimental computations were executed using Matlab 2020b running on a Windows 11 system-equipped laptop with an AMD Ryzen 5 3550H CPU operating at 3.5 GHz and backed by 16 GB of RAM.

4.1 $l_{\frac{1}{2}}$-regularization problem

In compressed sensing, we consider the following optimization problem

$$\min_x \|Mx - b\|^2 + \varphi\|x\|_0, \tag{4.1}$$

where $M \in \mathbb{R}^{m\times n}$ is the measuring matrix, $b \in \mathbb{R}^n$ is the observation vector, φ is the regular parameter. $\|x\|_0$ denotes the number of nonzero components of x. However, the problem (4.1) is NP-hard, some scholars relax l_0 norm to $l_{\frac{1}{2}}$ norm in practical applications [28], then the problem is exported to the following nonconvex problem:

$$\begin{aligned} &\min \varphi\|x\|_{(1/2)}^{(1/2)} + \frac{1}{2}\|y\|^2 \\ &\text{s.t. } Mx - y = b, \end{aligned} \tag{4.2}$$

where $\|x\|_{\frac{1}{2}} = (\sum_{i=1}^{n} \|x_i\|^{\frac{1}{2}})^2$.

Based on (4.2), we construct the following problem:

$$\begin{aligned} &\min_{x_1,x_2,y} c\|x_1\|_{(1/2)}^{(1/2)} + \frac{1}{2}\|x_2\|^2 + \frac{1}{2}\|B_1x_1 + B_2x_2 + y\|^2 \\ &\text{s.t.} A_1x_1 + A_2x_2 + y = b. \end{aligned} \tag{4.3}$$

To verify the validity of Algorithm 1 and Algorithm 2, we test them and compare them with LADMM.[1]

Applying Algorithm 1 to problem (4.3) yields

$$\begin{cases} {x_1}^{k+1} = H\left(\frac{1}{\mu_1}\left[\tau z_1^k - {B_1}^T\left(B_1x^k + B_2x_2^k + y^k\right)\right.\right. \\ \qquad \left.\left. - \beta {A_1}^T\left(A_2{x_2}^k + y^k - b - \frac{\lambda^k}{\beta}\right)\right], \frac{2c}{\mu_1}\right), \\ {x_2}^{k+1} = \frac{1}{\mu_2}\left[\tau z_2^k - B_2^T\left(B_1x_1^k + B_2x_2^k + y^k\right) - \beta A_2^T\left(A_1x_1^{k+1} + y^k - b - \frac{\lambda_k}{\beta}\right)\right], \\ y^{k+1} = \frac{1}{\mu_3}\left[\tau y^k - \left(B_1x^{k+1} + B_2{x_2}^{k+1}\right) - \beta\left(A_1{x_1}^{k+1} + A_2x_2^{k+1} - b - \frac{\lambda^k}{\beta}\right)\right], \\ \lambda^{k+1} = \lambda^k - \beta\left(Ax^{k+1} + By^{k+1} - b\right), \end{cases}$$

where $\mu_1 = \tau + \beta\rho_{\max(A_1^{\mathrm{T}}A_1)}, \mu_2 = 1 + \tau + \beta\rho_{\max(A_2^{\mathrm{T}}A_2)}, \mu_3 = 1 + \tau + \beta$, and $H(\cdot,\cdot)$ is the half shrinkage operator [29] defined as $H(x,\alpha) = \{h_\alpha^1, h_\alpha^2, \dots h_\alpha^n\}$ with

$$x_1(i) = \begin{cases} \frac{2x_i}{3}(1 + \cos(\frac{2}{3}(\pi - \phi(|h_\alpha^i|)))) & |h_\alpha^i| > \frac{\sqrt[3]{54}}{4}\alpha^{2/3}; \\ 0 & \text{otherwise}; \end{cases} \tag{4.4}$$

where

$$\phi\left(\left|h_\alpha^i\right|\right) = \arccos\left(\frac{\alpha}{8}\left(\frac{|h_\alpha^i|}{3}\right)^{-(3/2)}\right).$$

Applying Algorithm 2 to problem (4.3) yields

$$\begin{cases} {x_1}^{k+1} = H(\frac{1}{\mu_1}[\tau z_1^k - {B_1}^T(B_1x^k + B_2x_2^k + y^k) - \beta {A_1}^T(A_2{x_2}^k - b - \frac{\lambda^k}{\beta})], \frac{2c}{\mu_1}), \\ {x_2}^{k+1} = \frac{1}{\mu_2}[\tau z_2^k - B_2^T(B_1x_1^k + B_2x_2^k + y^k) - \beta A_2^T(A_1x_1^{k+1} + By^k - b - \frac{\lambda_k}{\beta})], \\ y^{k+1} = \frac{1}{\mu_4}[\tau y^k - (B_1x^{k+1} + B_2{x_2}^{k+1} + y^k) - \beta(A_1{x_1}^{k+1} + A_2x_2^{k+1} - b - \frac{\lambda^k}{\beta})], \\ \lambda^{k+1} = \lambda^k - \beta(Ax^{k+1} + By^{k+1} - b), \end{cases}$$

where $\mu_4 = \tau + \beta$. Applying LADMM to problem (4.3), we obtain

$$\begin{cases} {x_1}^{k+1} = H(\frac{1}{\mu_1}[\tau x_1^k - {B_1}^T(B_1x^k + B_2x_2^k + y^k) - \beta {A_1}^T(A_2{x_2}^k - b - \frac{\lambda^k}{\beta})], \frac{2c}{\mu_1}), \\ {x_2}^{k+1} = \frac{1}{\mu_2}[\tau x_2^k - B_2^T(B_1x_1^k + B_2x_2^k + y^k) - \beta A_2^T(A_1x_1^{k+1} + By^k - b - \frac{\lambda_k}{\beta})], \\ y^{k+1} = \frac{1}{\mu_3}[\tau y^k - (B_1x^{k+1} + B_2{x_2}^{k+1}) - \beta(A_1{x_1}^{k+1} + A_2x_2^{k+1} - b - \frac{\lambda^k}{\beta})], \\ \lambda^{k+1} = \lambda^k - \beta(Ax^{k+1} + By^{k+1} - b). \end{cases}$$

In experiment, we configure the parameter as follows: the dimensions are set to $m = 5000, n = 1000$, the regularization parameter is chosen as $\beta = 1000$. $b = 0$, $c = 1$, and the inertial parameter is fixed at $\theta = 0.15$. The initial points are selected as $x_1^{-1} = x_1^0 = 0$, $x_2^{-1} = x_2^0 = 0$, $y^0 = 0$, and $\lambda^0 = 0$. A_1, A_2, B_1, B_2 are random matrices. The stopping criterion of all

[1]LADMM is a special case of SPLI-ADMM that the inertial parameter $\theta_k = 0$.

Table 1 Numerical results under different τ

	SPLI-ADMM			SCLI-ADMM			LADMM		
	times(s)	Iter	log(Crit)	times(s)	Iter	log(Crit)	times(s)	Iter	log(Crit)
$\tau = 30$	14.84	222	−8.99	14.37	229	−9.60	16.03	235	−9.67
$\tau = 35$	15.57	211	−9.26	16.49	220	−9.97	17.10	239	−9.84
$\tau = 40$	14.86	228	−9.57	15.09	218	−9.24	17.00	256	−9.97
$\tau = 45$	15.24	217	−9.72	16.27	220	−9.62	17.38	235	−9.97

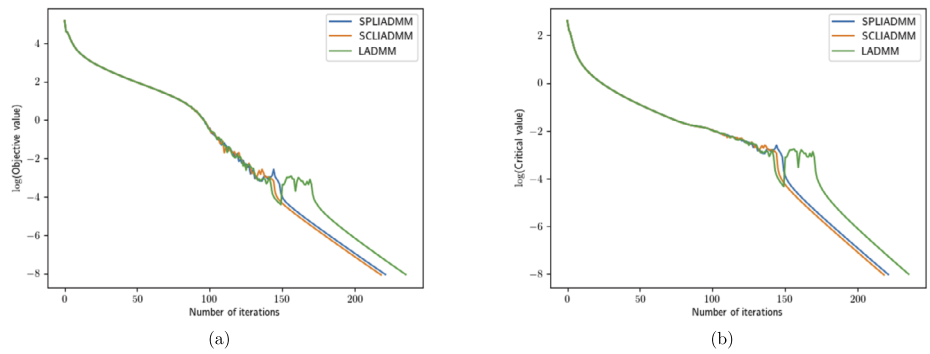

Figure 1 $m = 5000, n = 1000, \tau = 45, \beta = 1000$, the convergence results for LADMM ($\theta = 0$), SPLI-ADMM, and SCLI-ADMM ($\theta = 0.2$)

these methods are defined as

$$\|r_k\| = \left\|A_1x_1^k + A_2x_2^k + y - b\right\| \leq 10^{-8}.$$

Throughout the testing phase, we conduct experiments with four cases $\tau = 30, \tau = 35, \tau = 40$ and $\tau = 45$, respectively. The numerical results of the three algorithms are reported in Table 1. We report the number of iterations required to satisfy the stopping criterion ("Iter"), the total computing time in seconds ("times"), and the value of the stopping criterion ("log(Crit)"). Moreover, to visually illustrate the convergence behavior, the curves of the objective value and $\log(\|r_k\|)$ at $\tau = 45\cdot$ are presented in Fig. 1.

From Table 1, we can see that the two proposed algorithms have higher time efficiency and fewer iterations in comparison with LADMM. Figure 1(a) illustrates the trends of the objective value under the same iterations, clearly indicating that SPLIADMM and SCLI-ADMM have better performance of convergence than LADMM. Figure 1(b) again demonstrates the high time efficiency of our two algorithms, especially when "log(Crit)" is less than −4.

4.2 Matrix decomposition

Now, we consider the matrix decomposition problem, which has the following form:

$$\min \|L\|_* + \alpha\|S\|_1 + \frac{\omega}{2}\|T - M\|^2 \quad \text{s.t. } L + S = T, \tag{4.5}$$

where $M \in \mathbb{R}^{p\times n}$ is the observed matrix, and $L, S, T \in \mathbb{R}^{p\times n}$ are the decision variables. The nuclear norm $\|L\|_* := \sum_{i=1}^{\min(p,n)} |\sigma_i(L)|^{\frac{1}{2}}$, the spares term $\|S\|_1 := \sum_{i=1}^{n}\sum_{i=1}^{p} |S_{ij}|$, ω is the penalty factor, and α is the trade-off parameter between the nuclear norm $\|L\|_*$ and the

l_1-norm $\|S\|_1$. The ALF of problem (4.5) is defined as

$$L_\beta(L,S,T,\lambda) = \|L\|_* + \alpha\|S\|_1 + \frac{\omega}{2}\|T-M\|^2 - \langle\lambda, L+S-T\rangle + \frac{\beta}{2}\|L+S-T\|^2,$$

where λ is the Lagrange multiplier.

Applying SPLI-ADMM to problem (4.5), we get the closed-form iterative formulas:

$$\begin{cases} z_L^k = L^k + \theta(L^k - L^{k-1}), z_S^k = S^k + \theta(S^k - S^{k-1}), \\ L^{k+1} = V(\frac{\beta(T^k - S^k) + \lambda^k + \tau z_L^k}{\beta+\tau}, \frac{1}{\beta+\tau}), \\ S^{k+1} = S(\frac{\beta(T^k - L^{k+1}) + \lambda^k + \tau z_S^k}{\beta+\tau}, \frac{\alpha}{\beta+\tau}), \\ T^{k+1} = \frac{\tau T^k + \beta(L^{k+1} + S^{k+1}) + \omega M - \lambda^k}{\beta+\omega+\tau}, \\ \lambda^{k+1} = \lambda^k - \beta(L^{k+1} + S^{k+1} - T^{k+1}), \end{cases}$$

where $V(\cdot,\mu)$ is the singular value thresholding operator [30], $S(\cdot,\mu)$ is the softshrinkage operator [31]. Applying SCLI-ADMM to problem (4.5), we get

$$\begin{cases} z_L^k = L^k + \theta(L^k - L^{k-1}), z_S^k = S^k + \theta(S^k - S^{k-1}), \\ L^{k+1} = V(\frac{\beta(T^k - S^k) + \lambda^k + \tau z_L^k}{\beta+\tau}, \frac{1}{\beta+\tau}), \\ S^{k+1} = S(\frac{\beta(T^k - L^{k+1}) + \lambda^k + \tau z_S^k}{\beta+\tau}, \frac{\alpha}{\beta+\tau}), \\ T^{k+1} = \frac{\tau T^k + \beta(L^{k+1} + S^{k+1}) + \omega(M - T^k) - \lambda^k}{\beta+\tau}, \\ \lambda^{k+1} = \lambda^k - \beta(L^{k+1} + S^{k+1} - T^{k+1}), \end{cases}$$

Applying LADMM to problem (4.5), we have

$$\begin{cases} L^{k+1} = V(\frac{\beta(T^k - S^k) + \lambda^k + \tau L^k}{\beta+\tau}, \frac{1}{\beta+\tau}), \\ S^{k+1} = S(\frac{\beta(T^k - L^{k+1}) + \lambda^k + \tau S^k}{\beta+\tau}, \frac{\alpha}{\beta+\tau}), \\ T^{k+1} = \frac{\beta(L^{k+1} + S^{k+1}) + \omega M - \lambda^k}{\beta+\omega}, \\ \lambda^{k+1} = \lambda^k - \beta(L^{k+1} + S^{k+1} - T^{k+1}). \end{cases}$$

We set $p = n = 100$, and take 8 different $(r., spr.)$. Besides, we choose $\alpha = \frac{0.2}{\sqrt{m}}, \theta = 0.3, \omega = 1000$, the matrix L, S and T are initialized to be zero. We take $\beta = 5, \tau = 1, M$ was generated in MATLAB randomly. The stopping criterion is defined as

$$\text{RelChg} := \frac{\|(L^{k+1}, S^{k+1}, T^{k+1}) - (L^k, S^k, T^k)\|_F}{\|(L^k, S^k, T^k)\|_F + 1} \le 10^{-8} \quad \text{or} \quad k > 3000.$$

Let $\hat{S}$ and $\hat{T}$ be a numerical solution of problem (4.5). We measure the quality of the recovery by the relative error, which is defined by

$$\text{RelErr} := \frac{\|(\hat{L}, \hat{S}, \hat{T}) - (L^*, S^*, T^*)\|_F}{\|(L^*, S^*, T^*)\|_F + 1}.$$

Table 2 illustrates the comparison between different $(r., spr.)$, where "$r.$" represents the rank of matrix L, "$spr.$" represents the sparsity of the sparse matrix S, "Iter" represents

Table 2 Summary of three algorithms for eight different ($r.$, spr)

($r.$, $spr.$)	alg	Iter	Times	RelErr	$r.(L)$	$r.(\hat{L})$	$\|S\|_0$	$\|\hat{S}\|_0$	$spr.(\hat{S})$
(5,0.05)	SPLIADMM	358	1.213	3.34e-06	5	5	500	499	0.0499
	SCLIADMM	358	0.049	3.34e-06	5	5	500	499	0.0499
	LADMM	509	0.051	3.17e-06	5	5	500	501	0.0501
(5,0.1)	SPLIADMM	305	1.38	3.80e-06	5	5	1000	1000	0.1
	SCLIADMM	358	1.213	3.34e-06	5	5	500	999	0.0999
	LADMM	434	2.15	3.68e-06	5	5	1000	999	0.0999
(10,0.05)	SPLIADMM	538	2.56	2.24e-06	10	10	500	500	0.05
	SCLIADMM	548	2.49	2.18e-06	10	10	500	499	0.0499
	LADMM	603	5.00	2.21e-06	10	10	500	500	0.05
(10,0.1)	SPLIADMM	671	3.08	2.74e-06	10	10	1000	1000	0.1
	SPLIADMM	680	3.01	2.72e-06	10	10	1000	1000	0.1
	LADMM	760	6.32	2.71e-06	10	10	1000	1000	0.1
(15,0.05)	SPLIADMM	552	2.49	2.03e-6	15	15	500	500	0.05
	SCLIADMM	531	2.38	2.11e-6	15	15	500	500	0.05
	LADMM	642	5.06	2.00e-6	15	15	500	501	0.0501
(15,0.1)	SPLIADMM	884	4.04	2.43e-06	15	15	1000	1000	0.1
	SCLIADMM	776	3.34	2.51e-06	15	15	1000	1000	0.1
	LADMM	902	7.23	2.47e-06	15	15	1000	1000	0.1
(20,0.05)	SPLIADMM	640	2.93	1.91e-06	20	20	500	499	0.0499
	SCLIADMM	669	3.38	1.83e-06	20	20	500	499	0.0499
	LADMM	742	6.74	1.87e-06	20	20	500	498	0.0498
(20,0.1)	SPLIADMM	836	3.74	2.52e-06	20	20	1000	1000	0.1
	SCLIADMM	886	4.14	2.49e-06	20	20	1000	1001	0.1001
	LADMM	993	8.38	2.49e-06	20	20	1000	999	0.0999

the number of iterations. $\|S\|_0$ denotes the number of nonzero elements of S. Besides, the iterative curves of the stopping criterion and relative error of the three algorithms are plotted in Fig. 2, respectively.

Table 2 shows that SPLIADMM and SCLIADMM take less time and fewer iterations under the same condition, which demonstrates that our proposed two algorithms are more efficient than LADMM for different rank and sparse ratios. In Fig. 2, the curves of stopping criterion (see Fig. 2(a) and (c)) in two trials demonstrate that SPLI-ADMM and SCLIADMM converge faster than LADMM. Figure 2(b) and (d) indicate clearly that the matrices L and S are better recovered by SPLI-ADMM and SCLI-ADMM because "RelErr" of LADMM is greater than that of SPLI-ADMM for the same "Iter".

5 Conclusion

This paper made some extensions in the field of nonconvex optimization through the development and convergence analysis of two linearized ADMM algorithms, SPLI-ADMM and SCLI-ADMM. By integrating inertial strategy within a linearized framework, these algorithms improve the efficacy for solving linear constrained problems with nonseparable structure. A key novelty lies in the utilization of sequential gradients of the mixed term, which is not typically found in conventional ADMM approaches, enabling the proposed algorithms to use the latest information to update each variable. The KŁ property has been used to guarantee the convergence of the generated sequences. Finally, the results of numerical experiments show that the proposed algorithms exhibit superior time efficiency and validity.

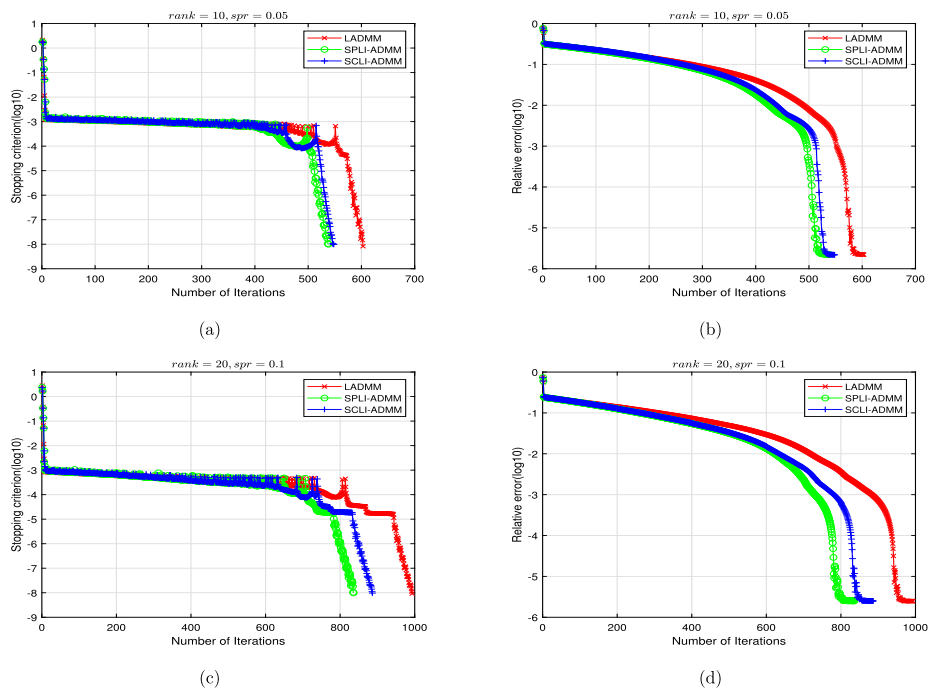

Figure 2 The performance comparison between LADMM ($\theta = 0$) , SPLI-ADMM ($\theta = 0.3$), and SCLI-ADMM ($\theta = 0.3$) with different (*r*., *spr*.)

Author contributions
Z. write the introduction. Z. and K. finsih the theoretical framework and establish Convergence analysis for the entire study. K. and Q. Assisted in Numerical experiment and preparing the figures and tables. Y. finalized the manuscript content and structure, ensuring consistency and coherence. Z. and Y. acquired of the financial support for the project leading to this publication All authors reviewed the manuscript.

Funding
This work is supported by National Natural Science Foundation of China under grants 72071130, 71901145 and 12371308; The Program for Professor of Special Appointment (Eastern Scholar) at Shanghai Institutions of Higher Learning(No.TP2022126); Key Lab of Intelligent and Green Flexographic Printing(No.KLIGFP-01)

Data Availability
No datasets were generated or analysed during the current study.

Declarations

Ethics approval and consent to participate
Not applicable.

Competing interests
The authors declare no competing interests.

Author details
[1]Shanghai Publishing and Printing College, Shuifeng Road 100, Yangpu District, Shanghai, 200093, Shanghai, China.
[2]School of Managemant, University of Shanghai for Science and Technology, Jungong Road 516, Yangpu District, Shanghai, 200093, Shanghai, China.

Received: 16 January 2024 Accepted: 24 April 2024 Published online: 08 May 2024

References
1. Yang, J., Zhang, Y.: Alternating direction algorithms for ℓ_1-problems in compressive sensing. SIAM J. Sci. Comput. **33**(1), 250–278 (2011)
2. Chambolle, A., Pock, T.: A first-order primal-dual algorithm for convex problems with applications to imaging. J. Math. Imaging Vis. **40**(1), 120–145 (2011)
3. Ding, J., Zhang, X., Chen, M., Xue, K., Zhang, C., Pan, M.: Differentially private robust admm for distributed machine learning. In: 2019 IEEE International Conference on Big Data (Big Data), pp. 1302–1311. IEEE, Los Angeles (2019)

4. Wang, Y., Yin, W., Zeng, J.: Global convergence of admm in nonconvex nonsmooth optimization. J. Sci. Comput. **78**, 29–63 (2019)
5. Jiang, B., Lin, T., Ma, S., Zhang, S.: Structured nonconvex and nonsmooth optimization: algorithms and iteration complexity analysis. Comput. Optim. Appl. **72**(1), 115–157 (2019)
6. Peng, Z., Xu, Y., Yan, M., Arock, W.Y.: An algorithmic framework for asynchronous parallel coordinate updates. SIAM J. Sci. Comput. **38**(5), A2851–A2879 (2016)
7. Chen, L., Sun, D., Toh, K.-C.: A note on the convergence of admm for linearly constrained convex optimization problems. Comput. Optim. Appl. **66**(2), 327–343 (2017)
8. Chang, X., Liu, S., Zhao, P., Song, D.: A generalization of linearized alternating direction method of multipliers for solving two-block separable convex programming. J. Comput. Appl. Math. **357**, 251–272 (2019)
9. Zhang, C., Song, Y., Cai, X., Han, D.: An extended proximal admm algorithm for three-block nonconvex optimization problems. J. Comput. Appl. Math. **398**, 113681 (2021)
10. Sun, D., Toh, K.-C., Yang, L.: A convergent 3-block semiproximal alternating direction method of multipliers for conic programming with 4-type constraints. SIAM J. Optim. **25**(2), 882–915 (2015)
11. Wang, X., Shao, H., Liu, P., Wu, T.: An inertial proximal partially symmetric admm-based algorithm for linearly constrained multi-block nonconvex optimization problems with applications. J. Comput. Appl. Math. **420**, 114821 (2023)
12. Hien, L.T.K., Phan, D.N., Gillis, N.: Inertial alternating direction method of multipliers for non-convex non-smooth optimization. Comput. Optim. Appl. **83**(1), 247–285 (2022)
13. Chao, M., Deng, Z., Jian, J.: Convergence of linear Bregman admm for nonconvex and nonsmooth problems with nonseparable structure. Complexity **2020**, 1–14 (2020)
14. Li, X., Mo, L., Yuan, X., Zhang, J.: Linearized alternating direction method of multipliers for sparse group and fused lasso models. Comput. Stat. Data Anal. **79**, 203–221 (2014)
15. Ling, Q., Shi, W., Wu, G., Dlm, A.R.: Decentralized linearized alternating direction method of multipliers. IEEE Trans. Signal Process. **63**(15), 4051–4064 (2015)
16. Liu, Q., Shen, X., Gu, Y.: Linearized admm for nonconvex nonsmooth optimization with convergence analysis. IEEE Access **7**, 76131–76144 (2019)
17. Polyak, B.T.: Some methods of speeding up the convergence of iteration methods. USSR Comput. Math. Math. Phys. **4**(5), 1–17 (1964)
18. Zavriev, S.K., Kostyuk, F.V.: Heavy-ball method in nonconvex optimization problems. Comput. Math. Model. **4**(4), 336–341 (1993)
19. Bolte, J., Sabach, S., Teboulle, M.: Proximal alternating linearized minimization for nonconvex and nonsmooth problems. Math. Program. **146**(1–2), 459–494 (2014)
20. Alvarez, F., Attouch, H.: An inertial proximal method for maximal monotone operators via discretization of a nonlinear oscillator with damping. Set-Valued Anal. **9**(1), 3–11 (2001)
21. Pock, T., Sabach, S.: Inertial proximal alternating linearized minimization (ipalm) for nonconvex and nonsmooth problems. SIAM J. Imaging Sci. **9**(4), 1756–1787 (2016)
22. Hien, L.T.K., Papadimitriou, D.: An inertial admm for a class of nonconvex composite optimization with nonlinear coupling constraints (2022). arXiv preprint. arXiv:2212.11336
23. Boţ, R.I., Nguyen, D.-K.: The proximal alternating direction method of multipliers in the nonconvex setting: convergence analysis and rates. Math. Oper. Res. **45**(2), 682–712 (2020)
24. Rockafellar, R.T., Wets, R.J.B.: Variational Analysis. Grundlehren der Mathematischen Wissenschaften, vol. 317. Springer, Berlin (1998)
25. Attouch, H., Bolte, J., Svaiter, B.F.: Convergence of descent methods for semi-algebraic and tame problems: proximal algorithms, forward–backward splitting, and regularized Gauss–Seidel methods. Math. Program. **137**(1–2), 91–129 (2013)
26. Boţ, R.I., Csetnek, E.R., László, S.C.: An inertial forward–backward algorithm for the minimization of the sum of two nonconvex functions. EURO J. Comput. Optim. **4**(1), 3–25 (2016)
27. Goncalves, M.L.N., Melo, J.G., Monteiro, R.D.C.: Convergence rate bounds for a proximal admm with over-relaxation stepsize parameter for solving nonconvex linearly constrained problems (2017). arXiv preprint. arXiv:1702.01850
28. Zeng, J., Lin, S., Wang, Y., Xu, Z.: $l_{1/2}$ regularization: convergence of iterative half thresholding algorithm. IEEE Trans. Signal Process. **62**(9), 2317–2329 (2014)
29. Xu, Z., Chang, X., Xu, F., Zhang, H.: $l_{1/2}$ regularization: a thresholding representation theory and a fast solver. IEEE Trans. Neural Netw. Learn. Syst. **23**(7), 1013–1027 (2012)
30. Cai, J.-F., Candès, E.J., Shen, Z.: A singular value thresholding algorithm for matrix completion. SIAM J. Optim. **20**(4), 1956–1982 (2010)
31. Bonesky, T., Maass, P.: Iterated soft shrinkage with adaptive operator evaluations. J. Inverse Ill-Posed Probl. **17**(4), 337–358 (2009)

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.