A faster proximal-indefinite augmented Lagrangian method with $\mathcal{O}(1/k^2)$ convergence rate *

Jianchao Bai† Yang Chen ‡ Yu-Hong Dai§ Yong-Jin Liu¶

Abstract

The Augmented Lagrangian Method (ALM), firstly proposed in 1969, remains a vital framework in large-scale constrained optimization. This paper addresses a linearly constrained composite convex minimization problem and presents a general proximal ALM that incorporates both Nesterov acceleration and relaxed acceleration, while enjoying a proximal-indefinite term. Under mild assumptions without requiring prior knowledge of the objective function's strong convexity modulus, we establish the global convergence of the proposed method and derive an $\mathcal{O}(1/k^2)$ nonergodic convergence rate for the Lagrangian residual, the objective gap, and the constraint violation, where k denotes the iteration number. Numerical experiments on testing large-scale sparse signal reconstruction tasks demonstrate the method's superior performance against several well-established methods.

Keywords: Augmented Lagrangian method, Proximal-indefinite term, Accelerated convergence rate, Signal reconstruction

Mathematics Subject Classification(2010): 65Y20, 90C25

1 Introduction

Let $\mathbb{R}^{m\times n}$ and $\mathbb{R}^n$ denote the set of $m \times n$ dimensional matrix space and the set of n dimensional vector space, respectively. Both spaces are endowed with the Euclidean norm $\|\cdot\| = \sqrt{\langle \cdot, \cdot \rangle}$, where $\langle \cdot, \cdot \rangle$ is the inner product. The objective of this paper is to develop an accelerated first-order method for solving the following linearly constrained composite programming problem

$$\min_{x\in\mathbb{R}^m} \theta(x) := f(x) + p(x) \qquad \text{s.t. } Ax = b, \tag{1.1}$$

*This work was supported by the National Natural Science Foundation of China (12471298, 12271097), the Shaanxi Fundamental Science Research Project for Mathematics and Physics (23JSQ031), the Open Project Program of Center for Applied Mathematics of Fujian Province (S20250605), the Key Program of National Science Foundation of Fujian Province of China (2023J02007), the Central Guidance on Local Science and Technology Development Fund of Fujian Province (2023L3003), and the Fujian Alliance of Mathematics (2023SXLMMS01).

†Corresponding author. School of Mathematics and Statistics, Northwestern Polytechnical University, Xi'an 710129, China; Center for Applied Mathematics of Fujian Province, Fuzhou University, Fuzhou 350116, China (jianchaobai@nwpu.edu.cn).

‡Department of Applied Mathematics and Physics, Graduate School of Informatics, Kyoto University, Yoshida-Honmachi, Sakyo-ku, Kyoto 606-8501, Japan (yang.chen@amp.i.kyoto-u.ac.jp).

§Corresponding author. LSEC, ICMSEC, AMSS, Chinese Academy of Sciences, Beijing 100190, China; School of Mathematics Sciences, University of Chinese Academy of Sciences, Beijing 100049, China (dyh@lsec.cc.ac.cn).

¶Center for Applied Mathematics of Fujian Province & School of Mathematics and Statistics, Fuzhou University, Fuzhou 350108, China (yjliu@fzu.edu.cn).

where $A \in \mathbb{R}^{n\times m}$ and $b \in \mathbb{R}^n$ are given, $f : \mathbb{R}^m \to \mathbb{R}$ is a proper, lower semicontinuous convex function (not necessarily smooth), and $p : \mathbb{R}^m \to \mathbb{R}$ is a smooth convex function whose gradient is Lipschitz continuous with a constant L:

$$\big\|\nabla p(x_1) - \nabla p(x_2)\big\| \le L\big\|x_1 - x_2\big\|, \quad \forall x_1, x_2 \in \mathbb{R}^m. \tag{1.2}$$

Throughout this paper, the solution set of (1.1) is assumed to be nonempty. Constraints of the form $x \in \mathcal{X}$, where $\mathcal{X}$ is a simple closed convex set, can be incorporated into the objective function by introducing the corresponding indicator function. Due to this transformation, Problem (1.1) covers numerous applications, including signal recovery [7, 21], federated learning [29], image processing [8], statistical learning [13] and so on.

1.1 Mathematical notation

The bold symbols $\mathbf{I}$ and $\mathbf{0}$ denote identity matrix and zero matrix/vector with proper dimensions, respectively. For any symmetric matrix G, we adopt the notation $\|x\|_G^2 = x^\top Gx$. Specifically, when G is positive semidefinite, $\|x\|_G = \sqrt{x^\top Gx}$ denotes a weighted norm where the superscript $^\top$ represents the transpose operator. The subdifferential of a convex function f is denoted by $\partial f(\cdot)$, and it reduces to $\nabla f(\cdot)$ when f is differentiable. The notation $\tilde{\nabla} f(x)$ denotes the subgradient of f at x, which satisfies the inequality $f(y) - f(x) \ge \big\langle \tilde{\nabla} f(x), y - x\big\rangle$ for all $x, y \in \mathrm{dom}(f)$. The proximal operator of f with given parameter $\tau > 0$ is defined as

$$\mathbf{prox}_{\tau f}(\cdot) := \arg\min_{x\in\mathbb{R}^m} \Big\{ f(x) + \frac{1}{2\tau}\|x - \cdot\|^2 \Big\}.$$

1.2 Related work

A benchmark approach for solving (1.1) is the Augmented Lagrangian Method (ALM, [9]) which proceeds recursively through the following iterative steps:

$$\begin{cases} x_{k+1} = \arg\min_{x\in\mathbb{R}^m} \mathcal{L}_\beta(x, \lambda_k) := \mathcal{L}(x, \lambda_k) + \frac{\beta}{2}\big\|Ax - b\big\|^2, \\ \lambda_{k+1} = \lambda_k + \beta\big(Ax_{k+1} - b\big), \end{cases}$$

where $\mathcal{L}(x, \lambda) = \theta(x) + \lambda^\top(Ax - b)$ is the associated Lagrange function, λ denotes the Lagrange multiplier, and $\beta > 0$ represents the penalty parameter for the equality constraint. The standard ALM described above, along with its variants, has garnered significant attention from multiple perspectives, including the acceleration of the convergence rate, the simplification of solving the subproblem, and the exploration of model applications. Relevant studies can be found in [4, 10, 12, 14, 18, 19, 26, 28]. Most of ALM-type methods are developed based on the principle of iteratively minimizing approximations to the nonsmooth objective function of the core subproblem, followed by the update of the dual variable. However, a crucial factor that determines the whole performance of ALM is how to efficiently handle the subproblem involved. A widely adopted and effective technique is to incorporate a quadratic proximal term in the form of $\frac{1}{2}\|x - x_k\|_{\mathcal{D}}^2$, where $\mathcal{D}$ represents a symmetric matrix that may be indefinite. Leveraging this technique, we can reformulate the above subproblem as

$$\min_{x\in\mathbb{R}^m} \Big\{ \theta(x) + \frac{\beta}{2}\big\|Ax - b + \lambda_k/\beta\big\|^2 + \frac{1}{2}\big\|x - x_k\big\|_{\mathcal{D}}^2 \Big\}.$$

Simple algebra shows that it is equivalent to $\mathbf{prox}_{\frac{1}{r}\theta}\big(x_k - \frac{1}{r}A^\top[\lambda_k + \beta(Ax_k - b)]\big)$ by choosing $\mathcal{D} = r\mathbf{I} - \beta A^\top A$. Consequently, the task of solving such a subproblem becomes relatively easier than the original, provided that the proximity operator of θ can be readily obtained. Otherwise, efficient approaches are imperative to derive an inexact solution. For example, the Hamilton-Jacobi-based proximal operator proposed by Osher, et al. [23] offers a viable approach in such situations.

Recently, the aforementioned kind of proximal matrices has been extended to be positive indefinite, as studied in [12]. This extension has given rise to a globally convergent yet optimal proximal ALM:

$$\begin{cases} x_{k+1} = \arg\min_{x\in\mathbb{R}^m} \left\{ \mathcal{L}_\beta(x,\lambda_k) + \frac{1}{2}\left\|x - x_k\right\|_{\mathcal{D}_0}^2 \right\}, \\ \lambda_{k+1} = \lambda_k + \gamma\beta\big(Ax_{k+1} - b\big). \end{cases} \tag{1.3}$$

Here, $\gamma \in (0,2)$ denotes the stepsize parameter for the dual update, $\mathcal{D}_0 = \mathcal{D} - (1-\tau)\beta A^\top A$ and $\mathcal{D} \in \mathbb{R}^{m\times m}$ is an arbitrarily chosen positive definite matrix. The global sublinear convergence of the algorithm specified in (1.3) has been detailedly proven for any $\tau > \frac{2+\gamma}{4}$. Note that, here the proximal matrix $\mathcal{D}_0$ is not necessarily positive definite due to the region of τ. Although the value of γ can not reach 2 in the deterministic ALM-type methods, it can be 2 as elaborated in the stochastic ALM [3, Section A.2].

Another pivotal issue lies in the establishment of accelerated convergence rates for ALM-type methods. Earlier accelerated method can be traced back to the accelerated gradient method [22], which was specifically designed for unconstrained smooth optimization problems. Since that seminal work, an increasing number of researchers have been eager to enhance the performance of the standard ALM and its associated splitting variants by applying or extending the renowned Nesterov acceleration technique. For example, He, et al. [10] proposed an accelerated ALM based on positive definite proximal matrices. They further demonstrated that the associated Lagrange residual of their method exhibited the $\mathcal{O}(1/k^2)$ convergence rate. For other ALM variants achieving the same convergence rate, we refer to [2, 17, 18, 19]. More recently, a novel accelerated ALM, as presented in [16], was proposed for solving (1.1), that is,

$$\begin{cases} \bar{x}_k = x_k + \frac{t_k - 1}{t_{k+1}}\big(x_k - x_{k-1}\big), \\ u_{k+1} = \arg\min_{u\in\mathbb{R}^m} \left\{ \begin{array}{c} f(u) + \langle A^\top\lambda_k + \nabla p(\bar{x}_k), u\rangle \\ + \frac{\beta t_{k+1}}{2}\left\|Au - b\right\|^2 + \frac{1}{2\sigma t_{k+1}}\left\|u - u_k\right\|^2 \end{array} \right\}, \\ x_{k+1} = \frac{1}{t_{k+1}}u_{k+1} + \frac{t_{k+1}-1}{t_{k+1}}x_k, \\ \lambda_{k+1} = \lambda_k + \beta t_{k+1}(Au_{k+1} - b), \end{cases} \tag{1.4}$$

where $\sigma > 0$ and $\{t_{k+1}\}$ is a positive sequence satisfying $t_{k+1}^2 \le t_k^2 + t_{k+1}$. When the function f exhibits strong convexity, additional conditions must be imposed to guarantee convergence of (1.4). It is worth noting that the subproblem of (1.4) may be as challenging to solve as the original problem, since it does not make full use of the proximity operator of f. Consequently, to streamline the subproblem and enhance the flexibility of the algorithm (1.4), a pertinent and natural question arises: *Can we show similar accelerated results for a more practical version of (1.4)? This variant would incorporate a potentially proximal-indefinite term and a larger dual stepsize, while ensuring that the subproblem allows for the effective utilization of its proximity operator.*

1.3 Methodology and contribution

Before addressing the above motivating question, we first introduce a double-accelerated proximal-indefinite ALM (see Alg. 1.1), where $\{\tau_k\}$ is a nonincreasing sequence confined to the following specified region:

$$1 + \frac{2\alpha L}{rt_k^2} \ge \tau_k > \max\left\{ \frac{2\alpha L/r + \gamma\alpha t_{k-1}^2/2 + t_k^2}{t_k^2 + t_{k-1}^2}, \frac{\alpha L}{rt_k^2} \right\}.$$

Note that the sequence $\{\tau_k\}$ is well-defined, since $1 + \frac{2\alpha L}{rt_k^2} > \frac{\alpha L}{rt_k^2}$ and it holds

$$1 + \frac{2\alpha L}{rt_k^2} > \frac{2\alpha L/r + \gamma\alpha t_{k-1}^2/2 + t_k^2}{t_k^2 + t_{k-1}^2} \Longleftrightarrow \frac{2\alpha L t_{k-1}^2}{rt_k^2} + \frac{2-\gamma\alpha}{2} t_{k-1}^2 > 0$$

for any $\gamma \in (0, \frac{2}{\alpha})$ and $\alpha \in (0, 2)$. Moreover, the nondecreasing property of $\{t_k\}$ implies

$$\frac{2\alpha L/r + \gamma\alpha t_{k-1}^2/2 + t_k^2}{t_k^2 + t_{k-1}^2} > \frac{\alpha L}{rt_k^2} \Longleftrightarrow \alpha L \frac{t_k^2 - t_{k-1}^2}{t_k^2 t_{k-1}^2} + \frac{rt_k^2}{t_{k-1}^2} + \frac{r\gamma\alpha^2}{2} > 0.$$

As a result, the region of τ_k reduces to

$$1 + \frac{2\alpha L}{rt_k^2} \geq \tau_k > \frac{2\alpha L/r + \gamma\alpha t_{k-1}^2/2 + t_k^2}{t_k^2 + t_{k-1}^2}. \tag{1.5}$$

Algorithm 1.1 *Accelerated Proximal-indefinite ALM (AP-ALM) for solving (1.1).*

Parameters: $\beta > 0,\ \alpha \in (0, 2),\ \gamma \in (0, \frac{2}{\alpha}),\ \mathcal{D}_k = \tau_k r\mathbf{I} - \beta A^\top A$ *with* $r > \beta\|A^\top A\|$ *and* τ_k *satisfying* (1.5).

Initialization: $(x_1, \lambda_1) \in \mathbb{R}^m \times \mathbb{R}^n,\ u_1 = x_1,\ t_0 = \alpha.$

for $k = 1, 2, \cdots,$ **do**

1. *Determine a positive nondecreasing sequence* $\{t_k\} \geq \alpha$ *such that*

$$t_k^2 \leq t_{k-1}^2 + \alpha t_k. \tag{1.6}$$

2. $\bar{x}_k = \frac{\alpha}{t_k} u_k + \frac{t_k - \alpha}{t_k} x_k.$
3. $u_{k+1} = \arg\min_{u \in \mathbb{R}^m} \Big\{ f(u) + \langle A^\top \lambda_k + \nabla p(\bar{x}_k), u \rangle + \frac{\beta t_k}{2} \|Au - b\|^2 + \frac{t_k}{2} \|u - u_k\|_{\mathcal{D}_k}^2 \Big\}.$
4. $\hat{x}_{k+1} = \frac{1}{t_k} u_{k+1} + \frac{t_k - 1}{t_k} x_k.$
5. $\hat{\lambda}_{k+1} = \lambda_k + \gamma\beta t_k (Au_{k+1} - b).$
6. *Relaxation step:* $\begin{pmatrix} x_{k+1} \\ \lambda_{k+1} \end{pmatrix} = \begin{pmatrix} x_k \\ \lambda_k \end{pmatrix} + \alpha \begin{pmatrix} \hat{x}_{k+1} - x_k \\ \hat{\lambda}_{k+1} - \lambda_k \end{pmatrix}.$

end

The first-order optimality condition of u_{k+1}-subproblem is

$$\mathcal{D}_k(u_{k+1} - u_k) = -\frac{1}{t_k}\big[\tilde{\nabla} f(u_{k+1}) + \nabla p(\bar{x}_k) + A^\top \tilde{\lambda}_{k+1}\big], \tag{1.7}$$

where $\tilde{\nabla} f(u_{k+1}) \in \partial f(u_{k+1})$ and

$$\tilde{\lambda}_{k+1} = \lambda_k + \beta t_k \big(Au_{k+1} - b\big). \tag{1.8}$$

By the choice of $\mathcal{D}_k$ as in Algorithm 1.1, the subproblem in the third step amounts to

$$u_{k+1} = \mathbf{prox}_{\frac{1}{r\tau_k t_k} f}\Big(u_k - \frac{1}{r\tau_k t_k}\big[\nabla p(\bar{x}_k) + A^\top \lambda_k + \beta t_k A^\top \big(Au_k - b\big)\big]\Big).$$

Contributions of this paper are summarized as the following aspects:

- **Generality of algorithmic parameters.** Compared with the existing dual updates presented in [2, 6, 14], our dual variable in the fifth step exhibits a more flexible parameter $\gamma \in (0, 2/\alpha)$, along with a dynamic, nondecreasing sequence $\{t_k\}$. In this context, t_k serves the function of Nesterov acceleration. To be more precise, the sequence $\{t_k\}$ satisfying (1.6) is called the general Nesterov acceleration technique. In the specific case where the relaxation parameter $\alpha = 1$, one can choose $t_k = \frac{1+\sqrt{1+4t_{k-1}^2}}{2}$, which reduces to the classical Nesterov acceleration. We further provide nice properties of this general Nesterov acceleration in Lemma 3.3 as well as feasible updating rules that are suitable for experimental implementation.

- **Flexibility of the proximal subproblem.** The definition of $\mathcal{D}_k$ and the range of $\{\tau_k\}$ imply that this dynamic proximal matrix $\mathcal{D}_k$ may be positive indefinite. If $\{t_k\} \geq \alpha$ satisfies (1.6) and f is only convex, then a nonpositive definite proximal matrix can be employed according to Lemma 3.1. When $(A, b) = (\mathbf{0}, \mathbf{0})$, problem (1.1) degenerates into an unconstrained composite convex programming problem. In this case, by selecting $\tau_k = 1$, $\mathcal{D}_k$ will reduce to $\mathcal{D}_k = r\mathbf{I}$ with $r > 0$, the subproblem takes the following form

$$u_{k+1} = \arg\min_u \left\{ f(u) + \frac{rt_k}{2} \left\| u - u_k + \frac{1}{rt_k} \nabla p(\bar{x}_k) \right\|^2 \right\}.$$

  This is a variant of proximal gradient method. For this particular case, when $\alpha = 1$, Algorithm 1.1 reduces to an extension of the method in [25] for minimizing $f + p$ and a variant of the accelerated proximal point method [11] for minimizing f.

- **Accelerated convergence rate and high performance.** In contrast to the $\mathcal{O}(1/k)$ convergence rate exhibited by certain existing ALM-type methods, such as those presented in [4, 12], we have derived an $\mathcal{O}(1/k^2)$ accelerated convergence rate, where k denotes the iteration number. This remarkable result is established using a potential energy function that incorporates the Lagrange residual, the primal residual and the dual residual. Notably, this accelerated convergence rate is achieved without requiring the objective function to be strongly convex or assuming that the constraint matrix A has full row rank. To the best of our knowledge, this is the first time to establish the accelerated convergence rate for a proximal-indefinite ALM-type method, while abandoning the strongly convexity of the objective function. Furthermore, we have proven the global convergence of our method, a topic that was not explored in previous references [6, 10, 16, 18, 19]. Comparative experiments on testing large-scale sparse signal reconstruction problems demonstrate that the proposed algorithm not only exhibits accelerated convergence behavior but also outperforms several existing first-order methods.

2 Technical preliminaries

This section is dedicated to presenting necessary technical preliminaries that will streamline the convergence analysis of our AP-ALM. Leveraging the Taylor expansion, we can infer from (1.2) that

$$p(x_2) - p(x_1) - \langle \nabla p(x_1), x_2 - x_1 \rangle \leq \frac{L}{2} \|x_2 - x_1\|^2, \quad \forall x_1, x_2 \in \mathbb{R}^m.$$

Combine it with the convexity of function p to obtain

$$\begin{aligned} \langle \nabla p(\bar{x}), x_2 - x_1 \rangle &= \langle \nabla p(\bar{x}), x_2 - \bar{x} \rangle + \langle \nabla p(\bar{x}), \bar{x} - x_1 \rangle \\ &\geq p(x_2) - p(\bar{x}) - \frac{L}{2} \|x_2 - \bar{x}\|^2 + p(\bar{x}) - p(x_1) \\ &= p(x_2) - p(x_1) - \frac{L}{2} \|x_2 - \bar{x}\|^2. \end{aligned} \tag{2.1}$$

The property in (2.1), along with the upcoming lemmas, will serve as crucial tools for conducting a comprehensive analysis of both the convergence and the iteration complexity of the proposed method.

Lemma 2.1 *[14, Lemma 4] Let* $\{h_k\}_{k=1}^{+\infty}$ *be a sequence in* $\mathbb{R}^n$ *and* $\{a_k\}$ *be a sequence in* $[0, 1)$*. Assume* $\left\|h_{k+1} + \sum_{i=1}^{k} a_i h_i\right\| \le c$ *for all integer* $k \ge 1$*. Then,*

$$\sup_{k\ge1} \|h_k\| \le \|h_1\| + 2c.$$

Lemma 2.2 *[24, Theorem 1] Let* $\{a_k\}_{k=0}^{+\infty}$*,* $\{b_k\}_{k=0}^{+\infty}$*,* $\{c_k\}_{k=0}^{+\infty}$*, and* $\{d_k\}_{k=0}^{+\infty}$ *be nonnegative sequences in* $\mathbb{R}$ *satisfying*

$$a_{k+1} \le a_k(1 + b_k) + c_k - d_k, \ \forall k \ge 0.$$

If $\sum\limits_{k=0}^{\infty} b_k < +\infty$ *and* $\sum\limits_{k=0}^{\infty} c_k < +\infty$*, then* $\lim\limits_{k\to\infty} a_k$ *exists and* $\sum\limits_{k=0}^{\infty} d_k < +\infty$*.*

Let (x^*, λ^*) be the saddle-point of $\mathcal{L}(x, \lambda)$. Then, it is also the primal-dual solution to the problem (1.1) and satisfies the following saddle-point inequality

$$\mathcal{L}(x^*, \lambda) \le \mathcal{L}(x^*, \lambda^*) \le \mathcal{L}(x, \lambda^*), \quad \forall (x, \lambda) \in \mathbb{R}^m \times \mathbb{R}^n. \tag{2.2}$$

Alternatively, it satisfies the following Karush-Kuhn-Tucker (KKT) system

$$-A^\top \lambda^* \in \partial f(x^*) + \nabla p(x^*) \quad \text{and} \quad Ax^* = b, \tag{2.3}$$

where λ^* is the solution to the corresponding dual problem

$$\min_\lambda \{\theta^*(-A^\top \lambda) + b^\top \lambda\}. \tag{2.4}$$

Here, θ^* denotes the Fenchel conjugate of the convex function θ, defined as $\theta^*(y) = \sup_{x\in\mathrm{dom}(\theta)} \left\{y^\top x - \theta(x)\right\}$, where $\mathrm{dom}(\theta)$ represents the domain of θ.

3 Main results

By constructing an energy sequence that incorporates the Lagrange residual, as well as the iterative residuals associated with the primal and dual variables, we will prove the global convergence of the proposed algorithm. Additionally, we will establish its $\mathcal{O}(1/k^2)$ accelerated convergence rate, where k denotes the iteration number.

3.1 Energy sequence and its property

Define the following potential energy sequence

$$\mathbf{E}_k = \mathbf{E}_k^{(1)} + \mathbf{E}_k^{(2)} + \mathbf{E}_k^{(3)}, \tag{3.1}$$

where

$$\begin{cases} \mathbf{E}_k^{(1)} = t_{k-1}^2 \left[\mathcal{L}(x_k, \lambda^*) - \mathcal{L}(x^*, \lambda^*)\right], \\ \mathbf{E}_k^{(2)} = \dfrac{\alpha t_k^2}{2} \left\|u_k - x^*\right\|_{\tau_k \mathcal{D}}^2, \ \mathbf{E}_k^{(3)} = \dfrac{1}{2\gamma\beta} \left\|\lambda_k - \lambda^*\right\|^2, \end{cases}$$

and $\mathcal{D} = r\mathbf{I} - \beta A^\top A$. Clearly, $\mathbf{E}_k \ge 0$ for all $k \ge 1$ according to (1.5) and (2.2). In order to investigate the properties of $\mathbf{E}_k$, we recall the following identity

$$\|x\|_G^2 - \|y\|_G^2 = 2\langle x, G(x-y)\rangle - \left\|x - y\right\|_G^2 \tag{3.2}$$

for any $x, y \in \mathbb{R}^m$ and symmetric matrix $G \in \mathbb{R}^{m\times m}$.

Lemma 3.1 *Let $\{\mathbf{E}_k\}$ be defined in (3.1), $\{\tau_k\}$ and $\{t_k\}$ satisfy (1.5) and (1.6), respectively. If $(t_k^2 - t_{k-1}^2)\mathcal{D}_k \preceq \mu_f t_{k-1}\mathbf{I}$, where $\mu_f \geq 0$ is the convex modulus of f, then the iterates generated by Algorithm 1.1 satisfy*

$$\begin{aligned}&\left(\mathbf{E}_{k+1} - \frac{(1-\tau_{k+1})\alpha\beta t_{k+1}^2}{2}\big\|Au_{k+1} - b\big\|^2\right)\\ &\quad\leq \left(\mathbf{E}_k - \frac{(1-\tau_k)\alpha\beta t_k^2}{2}\big\|Au_k - b\big\|^2\right) - \frac{\alpha}{2}\big\|v_{k+1} - v_k\big\|_{G_k}^2,\end{aligned} \tag{3.3}$$

where

$$v_k = \begin{pmatrix} u_k \\ \lambda_k \end{pmatrix} \quad \text{and} \quad G_k = \begin{bmatrix} (\tau_k t_k^2 r - \alpha L)\mathbf{I} - t_k^2\beta A^\top A & \mathbf{0} \\ \mathbf{0} & \frac{2-\gamma\alpha}{\gamma^2\alpha^2\beta}\mathbf{I} \end{bmatrix}.$$

Proof. Firstly, combine the fourth step with sixth step of Algorithm 1.1 to obtain

$$u_{k+1} = x_k + t_k\big(\hat{x}_{k+1} - x_k\big) = x_k + \frac{t_k}{\alpha}\big(x_{k+1} - x_k\big) = x_{k+1} + \frac{t_k - \alpha}{\alpha}\big(x_{k+1} - x_k\big), \tag{3.4}$$

equivalently,

$$x_{k+1} = \frac{\alpha}{t_k}u_{k+1} + \frac{t_k - \alpha}{t_k}x_k. \tag{3.5}$$

Since $t_k \geq \alpha$, we have by the convexity of f that

$$\begin{aligned}f(x_{k+1}) + \big\langle\lambda^*, Ax_{k+1} - b\big\rangle \leq& \frac{\alpha}{t_k}\Big(f(u_{k+1}) + \big\langle\lambda^*, Au_{k+1} - b\big\rangle\Big)\\ &+ \frac{t_k - \alpha}{t_k}\Big(f(x_k) + \big\langle\lambda^*, Ax_k - b\big\rangle\Big).\end{aligned}$$

Consequently, combining the last inequality and (1.6) leads to

$$\begin{aligned}\mathbf{E}_{k+1}^{(1)} - \mathbf{E}_k^{(1)} &= \big[t_k(t_k - \alpha) - t_{k-1}^2\big]\big[\mathcal{L}(x_k, \lambda^*) - \mathcal{L}(x^*, \lambda^*)\big]\\ &\quad + t_k^2\mathcal{L}(x_{k+1}, \lambda^*) - \alpha t_k\mathcal{L}(x^*, \lambda^*) - t_k(t_k - \alpha)\mathcal{L}(x_k, \lambda^*)\\ &\leq t_k^2\mathcal{L}(x_{k+1}, \lambda^*) - \alpha t_k\mathcal{L}(x^*, \lambda^*) - t_k(t_k - \alpha)\mathcal{L}(x_k, \lambda^*)\\ &= t_k^2\Big[f(x_{k+1}) + \big\langle\lambda^*, Ax_{k+1} - b\big\rangle + p(x_{k+1})\Big] - \alpha t_k\Big[f(x^*) + p(x^*)\Big]\\ &\quad - t_k(t_k - \alpha)\Big[f(x_k) + \big\langle\lambda^*, Ax_k - b\big\rangle + p(x_k)\Big]\\ &\leq \alpha t_k\big[f(u_{k+1}) - f(x^*) + \langle\lambda^*, Au_{k+1} - b\rangle\big]\\ &\quad + \alpha t_k\big[p(x_{k+1}) - p(x^*)\big] + t_k(t_k - \alpha)\big[p(x_{k+1}) - p(x_k)\big].\end{aligned} \tag{3.6}$$

Secondly, we turn to estimate an upper bound of the term $\mathbf{E}_{k+1}^{(2)} - \mathbf{E}_k^{(2)}$. The second step of Algorithm 1.1 and (3.5) indicate

$$x_{k+1} - \bar{x}_k = \frac{\alpha}{t_k}\big(u_{k+1} - u_k\big). \tag{3.7}$$

Combine (3.7) with (3.4) and (2.1) to obtain

$$\begin{aligned}&\big\langle u_{k+1} - x^*, \nabla p(\bar{x}_k)\big\rangle\\ =\;& \big\langle x_{k+1} - x^*, \nabla p(\bar{x}_k)\big\rangle + \frac{t_k - \alpha}{\alpha}\big\langle x_{k+1} - x_k, \nabla p(\bar{x}_k)\big\rangle\\ \geq\;& p(x_{k+1}) - p(x^*) + \frac{t_k - \alpha}{\alpha}\big[p(x_{k+1}) - p(x_k)\big] - \frac{Lt_k}{2\alpha}\big\|x_{k+1} - \bar{x}_k\big\|^2\\ =\;& p(x_{k+1}) - p(x^*) + \frac{t_k - \alpha}{\alpha}\big[p(x_{k+1}) - p(x_k)\big] - \frac{\alpha L}{2t_k}\big\|u_{k+1} - u_k\big\|^2.\end{aligned} \tag{3.8}$$

By the known relationships $\mathcal{D}_k = \tau_k r\mathbf{I} - \beta A^\top A = \tau_k \mathcal{D} - (1-\tau_k)\beta A^\top A$ with $\mathcal{D} = r\mathbf{I} - \beta A^\top A$ and $\mathcal{D}_{k+1} \preceq \mathcal{D}_k$(as we choose $\{\tau_k\}$ as a nonincreasing sequence satisfying (1.5)), it holds that

$$\begin{aligned}
& \mathbf{E}_{k+1}^{(2)} - \mathbf{E}_k^{(2)} - \frac{(1-\tau_{k+1})\alpha\beta t_{k+1}^2}{2}\big\|A(u_{k+1}-x^*)\big\|^2 + \frac{(1-\tau_k)\alpha\beta t_k^2}{2}\big\|A(u_k-x^*)\big\|^2 \\
= \; & \frac{\alpha t_{k+1}^2}{2}\|u_{k+1}-x^*\|_{\mathcal{D}_{k+1}}^2 - \frac{\alpha t_k^2}{2}\|u_k - x^*\|_{\mathcal{D}_k}^2 \\
= \; & \frac{\alpha t_k^2}{2}\big(\|u_{k+1}-x^*\|_{\mathcal{D}_k}^2 - \|u_k - x^*\|_{\mathcal{D}_k}^2\big) \\
& + \frac{\alpha t_k^2}{2}\|u_{k+1}-x^*\|_{\mathcal{D}_{k+1}-\mathcal{D}_k}^2 + \frac{\alpha(t_{k+1}^2 - t_k^2)}{2}\|u_{k+1}-x^*\|_{\mathcal{D}_{k+1}}^2 \\
\leq \; & \frac{\alpha t_k^2}{2}\big(\|u_{k+1}-x^*\|_{\mathcal{D}_k}^2 - \|u_k - x^*\|_{\mathcal{D}_k}^2\big) + \frac{\alpha(t_{k+1}^2 - t_k^2)}{2}\|u_{k+1}-x^*\|_{\mathcal{D}_{k+1}}^2.
\end{aligned}$$

For the proximal matrix $\mathcal{D}_k$ satisfying $(t_k^2 - t_{k-1}^2)\mathcal{D}_k \preceq \mu_f t_{k-1}\mathbf{I}$, combine these conditions with (3.2) and (1.7) to obtain

$$\begin{aligned}
& \mathbf{E}_{k+1}^{(2)} - \mathbf{E}_k^{(2)} - \frac{(1-\tau_{k+1})\alpha\beta t_{k+1}^2}{2}\big\|A(u_{k+1}-x^*)\big\|^2 + \frac{(1-\tau_k)\alpha\beta t_k^2}{2}\big\|A(u_k-x^*)\big\|^2 \\
\leq \; & \alpha t_k^2\Big(\big\langle u_{k+1}-x^*, \mathcal{D}_k(u_{k+1}-u_k)\big\rangle - \frac{1}{2}\|u_{k+1}-u_k\|_{\mathcal{D}_k}^2\Big) \\
& + \frac{\alpha(t_{k+1}^2 - t_k^2)}{2}\|u_{k+1}-x^*\|_{\mathcal{D}_{k+1}}^2 \\
= \; & -\alpha t_k\big\langle u_{k+1}-x^*, \nabla p(\overline{x}_k) + \widetilde{\nabla} f(u_{k+1}) + A^\top\lambda^*\big\rangle - \frac{\alpha t_k^2}{2}\|u_{k+1}-u_k\|_{\mathcal{D}_k}^2 \\
& -\alpha t_k\big\langle u_{k+1}-x^*, A^\top(\tilde{\lambda}_{k+1}-\lambda^*)\big\rangle + \frac{\alpha(t_{k+1}^2 - t_k^2)}{2}\|u_{k+1}-x^*\|_{\mathcal{D}_{k+1}}^2 \\
\leq \; & -\alpha t_k\Big(f(u_{k+1}) - f(x^*) + \big\langle\lambda^*, A(u_{k+1}-x^*)\big\rangle\Big) + \frac{\alpha}{2}\|u_{k+1}-x^*\|_{\bar{\mathcal{D}}_{k+1}}^2 \\
& -\alpha t_k\big[p(x_{k+1}) - p(x^*)\big] - (t_k - \alpha)t_k\big[p(x_{k+1}) - p(x_k)\big] \\
& -\alpha t_k\langle A(u_{k+1}-x^*), \tilde{\lambda}_{k+1} - \lambda^*\rangle - \frac{\alpha}{2}\|u_{k+1}-u_k\|_{t_k^2\mathcal{D}_k - \alpha L\mathbf{I}}^2 \\
\leq \; & -\alpha t_k\Big(f(u_{k+1}) - f(x^*) + \big\langle\lambda^*, A(u_{k+1}-x^*)\big\rangle\Big) \\
& -\alpha t_k\big[p(x_{k+1}) - p(x^*)\big] - (t_k - \alpha)t_k\big[p(x_{k+1}) - p(x_k)\big] \\
& -\alpha t_k\langle A(u_{k+1}-x^*), \tilde{\lambda}_{k+1} - \lambda^*\rangle - \frac{\alpha}{2}\|u_{k+1}-u_k\|_{t_k^2\mathcal{D}_k - \alpha L\mathbf{I}}^2.
\end{aligned} \tag{3.9}$$

where the last inequality uses (3.8) and the notation

$$\bar{\mathcal{D}}_{k+1} := (t_{k+1}^2 - t_k^2)\mathcal{D}_{k+1} - \mu_f t_k \mathbf{I} \preceq \mathbf{0}.$$

Note that by the last two steps of Algorithm 1.1 and $Ax^* = b$, we have

$$\lambda_{k+1} - \lambda_k = \alpha(\hat{\lambda}_{k+1} - \lambda_k) = \gamma\alpha\beta t_k A\big(u_{k+1} - x^*\big). \tag{3.10}$$

Combining (3.10) with (3.2) results in

$$\begin{aligned}
\mathbf{E}_{k+1}^{(3)} - \mathbf{E}_k^{(3)} &= \frac{1}{2\gamma\beta}\big(\|\lambda_{k+1}-\lambda^*\|^2 - \|\lambda_k - \lambda^*\|^2\big) \\
&= \frac{1}{\gamma\beta}\big\langle\lambda_{k+1}-\lambda^*, \lambda_{k+1}-\lambda_k\big\rangle - \frac{1}{2\gamma\beta}\big\|\lambda_{k+1}-\lambda_k\big\|^2 \\
&= \alpha t_k\big\langle\lambda_{k+1}-\lambda^*, A(u_{k+1}-x^*)\big\rangle - \frac{1}{2\gamma\beta}\big\|\lambda_{k+1}-\lambda_k\big\|^2.
\end{aligned} \tag{3.11}$$

Finally, summing up the above inequalities (3.6), (3.9) and (3.11) together with the second inequality in (1.6), we obtain

$$\begin{aligned}\mathbf{E}_{k+1}-\mathbf{E}_k \quad \le \quad & -\frac{\alpha}{2}\big\|u_{k+1}-u_k\big\|^2_{t_k^2\mathcal{D}-\alpha L\mathbf{I}}-\frac{1}{2\gamma\beta}\big\|\lambda_{k+1}-\lambda_k\big\|^2\\ &+\frac{(1-\tau_{k+1})\alpha\beta t_{k+1}^2}{2}\big\|A(u_{k+1}-x^*)\big\|^2-\frac{(1-\tau_k)\alpha\beta t_k^2}{2}\big\|A(u_k-x^*)\big\|^2\\ &+\alpha t_k\big\langle A(u_{k+1}-x^*),\lambda_{k+1}-\tilde{\lambda}_{k+1}\big\rangle. \end{aligned}\tag{3.12}$$

In addition, it follows from (3.10) that

$$t_kA\big(u_{k+1}-x^*\big)=\frac{1}{\gamma\alpha\beta}\big(\lambda_{k+1}-\lambda_k\big),$$

while the fifth step of Algorithm 1.1 and the definition of $\tilde{\lambda}_{k+1}$ indicate

$$\lambda_{k+1}-\tilde{\lambda}_{k+1}=\lambda_{k+1}-\lambda_k-\beta t_kA(u_{k+1}-x^*)=\Big(1-\frac{1}{\gamma\alpha}\Big)\big(\lambda_{k+1}-\lambda_k\big).$$

Substituting the last two relationships into (3.12) gives

$$\begin{aligned}&\mathbf{E}_{k+1}-\mathbf{E}_k\le-\frac{\alpha}{2}\big\|u_{k+1}-u_k\big\|^2_{t_k^2\mathcal{D}_k-\alpha L\mathbf{I}}-\frac{1}{2\gamma\beta}\big\|\lambda_{k+1}-\lambda_k\big\|^2\\ &\quad+\frac{(1-\tau_{k+1})\alpha\beta t_{k+1}^2}{2}\big\|A(u_{k+1}-x^*)\big\|^2-\frac{(1-\tau_k)\alpha\beta t_k^2}{2}\big\|A(u_k-x^*)\big\|^2\\ &\quad+\frac{1}{\gamma\beta}\Big\langle\lambda_{k+1}-\lambda_k,\big(1-\frac{1}{\gamma\alpha}\big)\big(\lambda_{k+1}-\lambda_k\big)\Big\rangle\\ &=-\frac{\alpha}{2}\big\|u_{k+1}-u_k\big\|^2_{t_k^2\mathcal{D}_k-\alpha LI}-\frac{2-\gamma\alpha}{2\gamma^2\alpha\beta}\big\|\lambda_{k+1}-\lambda_k\big\|^2\\ &\quad+\frac{(1-\tau_{k+1})\alpha\beta t_{k+1}^2}{2}\big\|A(u_{k+1}-x^*)\big\|^2-\frac{(1-\tau_k)\alpha\beta t_k^2}{2}\big\|A(u_k-x^*)\big\|^2\\ &=-\frac{\alpha}{2}\big\|v_{k+1}-v_k\big\|^2_G+\frac{(1-\tau_{k+1})\alpha\beta t_{k+1}^2}{2}\big\|A(u_{k+1}-x^*)\big\|^2-\frac{(1-\tau_k)\alpha\beta t_k^2}{2}\big\|A(u_k-x^*)\big\|^2.\end{aligned}$$

So, rearranging it with $Ax^*=b$ completes the proof. ■

Remark 3.1 *The condition $(t_k^2-t_{k-1}^2)\mathcal{D}_k\preceq\mu_f t_{k-1}\mathbf{I}$ can be removed for the case $\mu_f=0$ (that is, f is just a convex function). In this case, we have from the nondecreasing property of $\{t_k\}$ that the matrix $\mathcal{D}_k$ can be negative indefinite.*

Lemma 3.1 does not guarantee the monotonicity of the sequence $\{\mathbf{E}_k\}$. Nevertheless, by combining (1.5) with an estimate for the lower bound of the term $\frac{\alpha}{2}\big\|v_{k+1}-v_k\big\|^2_G$, the monotonicity of certain variants of this sequence can be established, as stated in the following corollary.

Corollary 3.1 *Let $\{\mathbf{E}_k\}$ be defined in (3.1) and $\mathcal{D}=r\mathbf{I}-\beta A^\top A$. Then, under the conditions in Lemma 3.1, the iterates generated by Algorithm 1.1 satisfy*

$$\begin{aligned}&\Big(\mathbf{E}_{k+1}+\frac{(1-\tau_{k+1})t_{k+1}^2+2\alpha L/r}{2}\alpha\beta\|Au_{k+1}-b\|^2\Big)\\ &-\Big(\mathbf{E}_k+\frac{(1-\tau_k)t_k^2+2\alpha L/r}{2}\alpha\beta(\|Au_k-b\|^2\Big)\\ \le\quad&-\frac{2[(\tau_k-1)t_k^2+(\tau_{k+1}-1)t_{k+1}^2-2\alpha L/r]+(2-\gamma\alpha)t_k^2}{2}\alpha\beta\big\|Au_{k+1}-b\big\|^2\\ &-\frac{\alpha(\tau_kt_k^2-\alpha L/r)}{2}\big\|u_{k+1}-u_k\big\|^2_{\mathcal{D}}.\end{aligned}\tag{3.13}$$

Proof. By the structure of the matrix G_k and (3.10), we deduce

$$\begin{aligned} & -\frac{\alpha}{2}\big\|v_{k+1}-v_k\big\|_{G_k}^2+\frac{\alpha(\tau_k t_k^2-\alpha L/r)}{2}\big\|u_{k+1}-u_k\big\|_{\mathcal{D}}^2 \\ = & -\frac{(2-\gamma\alpha)\alpha\beta t_k^2}{2}\big\|Au_{k+1}-b\big\|^2+\frac{(1-\tau_k)t_k^2+\alpha L/r}{2}\alpha\beta\big\|A(u_{k+1}-u_k)\big\|^2 \\ \leq & -\frac{2\big[(\tau_k-1)t_k^2-\alpha L/r\big]+(2-\gamma\alpha)t_k^2}{2}\alpha\beta\big\|Au_{k+1}-b\big\|^2 \\ & +\big[(1-\tau_k)t_k^2+\alpha L/r\big]\alpha\beta\big\|Au_k-b\big\|^2 \\ = & -\frac{2[(\tau_k-1)t_k^2+(\tau_{k+1}-1)t_{k+1}^2-2\alpha L/r\big]+(2-\gamma\alpha)t_k^2}{2}\alpha\beta\big\|Au_{k+1}-b\big\|^2 \\ & -\Big[(1-\tau_{k+1})t_{k+1}^2+\alpha L/r\Big]\alpha\beta\big\|Au_{k+1}-b\big\|^2+\big[(1-\tau_k)t_k^2+\alpha L/r\big]\alpha\beta\big\|Au_k-b\big\|^2, \end{aligned} \tag{3.14}$$

where the inequality uses the property

$$\|\xi-\eta\|^2\leq 2\|\xi\|^2+2\|\eta\|^2 \quad \text{with} \quad (\xi,\eta)=(Au_{k+1}-b,Au_k-b).$$

Finally, plug (3.14) into the right-hand side of (3.3) to end the proof. ■

Now, based on the above corollary and the notation $\mathbf{E}_1$ derived from (3.1):

$$\mathbf{E}_1=t_0^2\big[\mathcal{L}(x_1,\lambda^*)-\mathcal{L}(x^*,\lambda^*)\big]+\frac{\alpha\tau_1 t_1^2}{2}\big\|x_1-x^*\big\|_{\mathcal{D}}^2+\frac{1}{2\gamma\beta}\big\|\lambda_1-\lambda^*\big\|^2, \tag{3.15}$$

we show a preliminary result for the convergence of the proposed algorithm.

Lemma 3.2 *Under the conditions in Lemma 3.1, we have*

$$\big\|u_k-x^*\big\|^2\leq\frac{2}{\alpha\tau_k t_k^2\lambda_{\min}(\mathcal{D})}\Big\{\mathbf{E}_1+\frac{(1-\tau_1)t_1^2+2\alpha L/r}{2}\alpha\beta\big\|Ax_1-b\big\|^2\Big\}, \tag{3.16}$$

where $\lambda_{\min}(\mathcal{D})$ *denotes the minimal eigenvalue of* $\mathcal{D}$ *and* $\mathbf{E}_1$ *is given by* (3.15).

Proof. For the sake of conciseness, denote

$$\bar{\mathbf{E}}_k=\mathbf{E}_k+\frac{(1-\tau_k)t_k^2+2\alpha L/r}{2}\alpha\beta\big\|Au_k-b\big\|^2.$$

It follows from (3.13) and (1.5) that the sequence $\{\bar{\mathbf{E}}_k\}$ is nonincreasing.

According to the nonincreasing property of $\{\bar{\mathbf{E}}_k\}$, (1.5), the relationships $b=Ax^*$ and $u_1=x_1$, we have

$$\mathbf{E}_k\leq\bar{\mathbf{E}}_k\leq\bar{\mathbf{E}}_1=\mathbf{E}_1+\frac{(1-\tau_1)t_1^2+2\alpha L/r}{2}\alpha\beta\big\|Ax_1-b\big\|^2, \tag{3.17}$$

where $\mathbf{E}_1$ defined by (3.15) is a nonnegative constant. Besides, it follows from the definition of $\mathbf{E}_k$ and the positive definiteness of matrix $\mathcal{D}$ that

$$\mathbf{E}_k\geq\frac{\alpha\tau_k t_k^2}{2}\big\|u_k-x^*\big\|_{\mathcal{D}}^2\geq\frac{\alpha\tau_k t_k^2\lambda_{\min}(\mathcal{D})}{2}\big\|u_k-x^*\big\|^2.$$

Combine it with (3.17) to confirm the conclusion. ■

3.2 Convergence analysis

In this section, we analyze the convergence of the proposed algorithm as well as its accelerated convergence rate.

Theorem 3.1 *Under the conditions in Lemma 3.1, we have*
(i) $\lim_{k\to\infty} \|(x_{k+1}, \lambda_{k+1}) - (x_k, \lambda_k)\| = 0$ *and* $\lim_{k\to\infty} \|Ax_{k+1} - b\| = 0$*;*
(ii) Any limit point of $\{\lambda_k\}$ *is the solution to the dual problem (2.4), and any limit point of* $\{u_k\}$ *is the solution to the primal problem (1.1).*

Proof. Sum the inequality (3.13) over $k = 1, 2, \dots, \infty$ together with (3.16) to have

$$\sum_{k=1}^{\infty} \frac{2[(\tau_k - 1)t_k^2 + (\tau_{k+1} - 1)t_{k+1}^2 - 2\alpha L/r] + (2 - \gamma\alpha)t_k^2}{2} \alpha\beta \big\|Au_{k+1} - b\big\|^2$$
$$+ \sum_{k=1}^{\infty} \frac{\alpha(\tau_k t_k^2 - \alpha L/r)}{2} \big\|u_{k+1} - u_k\big\|_{\mathcal{D}}^2 \le \mathbf{E}_1 + \frac{(1 - \tau_1)t_1^2 + 2\alpha L/r}{2} \alpha\beta \big\|Ax_1 - b\big\|^2 < +\infty,$$

which, by (1.5) and the positive definiteness of $\mathcal{D}$, implies

$$\lim_{k\to\infty} \big\|u_{k+1} - u_k\big\| = 0 \quad \text{and} \quad \lim_{k\to\infty} \big\|t_k(Au_{k+1} - b)\big\| = 0. \tag{3.18}$$

Combine the last equality in (3.18) with the fifth and sixth steps of Algorithm 1.1 to get

$$\lim_{k\to\infty} \|\lambda_{k+1} - \lambda_k\| = 0.$$

Recalling (3.4), it holds that

$$u_{k+1} - u_k = \frac{t_k}{\alpha}(x_{k+1} - x_k) - \frac{t_{k-1} - \alpha}{\alpha}(x_k - x_{k-1}).$$

So,

$$\begin{aligned} \frac{t_k}{\alpha}\|x_{k+1} - x_k\| \le& \frac{t_{k-1} - \alpha}{\alpha}\|x_k - x_{k-1}\| + \|u_{k+1} - u_k\| \\ =& \frac{t_{k-1}}{\alpha}\|x_k - x_{k-1}\| + \|u_{k+1} - u_k\| - \|x_k - x_{k-1}\|, \end{aligned}$$

which, by Lemma 2.2 with

$$a_k := \frac{t_k}{\alpha}\|x_{k+1} - x_k\|,\ b_k := 0,\ c_k := \|u_{k+1} - u_k\|,\ d_k := \|x_k - x_{k-1}\|$$

and (3.18), implies $\lim_{k\to\infty} d_k = 0$ and $\lim_{k\to\infty} a_k$ exists. So, the first part of the assertion (i) is proved. By (3.4) again, we have

$$Ax_{k+1} - b = \big(Au_{k+1} - b\big) - \frac{t_k - \alpha}{\alpha} A\big(x_{k+1} - x_k\big), \tag{3.19}$$

which ensures the second part of the assertion (i) by (3.18) and $\lim_{k\to\infty} \|x_{k+1} - x_k\| = 0$.

Next, we prove the second assertion. Combining (3.17), the definition of $\mathbf{E}_k$ and the positive definiteness of $\mathcal{D}$, we know both $\{u_k\}$ and $\{\lambda_k\}$ are bounded. Let λ_∞ be a limit point of $\{\lambda_k\}$ and assume the subsequence $\{\lambda_{k_j}\}_{k_j \in K}$ converges to it. Then, combine (1.2) with (3.4) and (3.7) to have

$$\|\nabla p(u_{k+1}) - \nabla p(\bar{x}_k)\| \le L\|u_{k+1} - \bar{x}_k\| = L\bigg\|\frac{t_k - \alpha}{\alpha}\big(x_{k+1} - x_k\big) + \frac{\alpha}{t_k}(u_{k+1} - u_k)\bigg\|,$$

and moreover $\lim_{k\to\infty} \|\nabla p(u_{k+1}) - \nabla p(\bar{x}_k)\| = 0$. Based on these results and a reformulation of (1.7):

$$\begin{aligned}\delta_{k+1} :=& -A^\top \lambda_{k+1} + A^\top(\lambda_{k+1} - \lambda_k) - \beta t_k A^\top(Au_k - b) - \tau_k t_k r(u_{k+1} - u_k)\\ &+ \nabla p(u_{k+1}) - \nabla p(\bar{x}_k)\\ \in& \partial f(u_{k+1}) + \nabla p(u_{k+1}),\end{aligned}$$

we conclude that $u_{k+1} \in \partial\theta^*(\delta_{k+1})$ and $\lim_{k\to\infty} \delta_{k+1} = -A^\top\lambda_\infty$.

Now, let u_∞ be the limit point of $\{u_k\}$ accompanied with $(u_\infty, \lambda_\infty)$. Then, since θ^* is a proper closed convex function, it holds

$$\theta^*(-A^\top\lambda^*) \geq \theta^*(\delta_{k+1}) + \langle u_{k+1}, -A^\top\lambda^* - \delta_{k+1}\rangle.$$

Take limit to both sides of the above inequality with $k \to \infty$ and $Au_\infty = b$ to obtain

$$\theta^*(-A^\top\lambda^*) + b^\top\lambda^* \geq \theta^*(-A^\top\lambda^\infty) + b^\top\lambda_\infty.$$

Note that λ^* is a solution to the dual problem (2.4). So, it follows from the above inequality that λ_∞ is also a solution to (2.4).

The equality $Au_\infty = b$ implies that u_∞ is a feasible point of the primal problem (1.1). Since $\{\lambda_k\}$ is bounded, there exists a subset of indices $K_1 \subseteq K$ such that $\lim_{k\to\infty}\lambda_k = \lambda_\infty$, where λ_∞ is a dual solution to (2.4) and $k \in K_1$. So,

$$\lim_{K_1 \ni k\to\infty} (u_k, \lambda_k) = (u_\infty, \lambda_\infty),$$

which, together with the relation $-A^\top\lambda_\infty \in \partial\theta(u_\infty)$, ensures that $(u_\infty, \lambda_\infty)$ is a solution to the KKT system (2.3). As a result, x_∞ is a primal solution to (1.1). ■

From (3.4) and Theorem 3.1 (i), we have $\lim_{k\to\infty} x_{k+1} = u_\infty$. Hence, any limit point of the sequence $\{x_k\}$ is the solution to the primal problem (1.1). In what follows, we will establish the accelerated convergence rate for the proposed method, as presented in Theorem 3.2. The accelerated results obtained here are better than the ergodic iteration-complexity bounds reported in [27].

Theorem 3.2 *Let (x^*, λ^*) be the primal-dual solution to (1.1). Then, under the conditions in Lemma 3.1, there exist constants $c_1, c_2 > 0$ such that*

$$\begin{cases} \mathcal{L}(x_k, \lambda^*) - \mathcal{L}(x^*, \lambda^*) \leq \frac{c_1}{t_{k-1}^2},\\ \left|\theta(x_k) - \theta(x^*)\right| \leq \frac{c_2 + c_1\|\lambda^*\|}{t_{k-1}^2},\\ \left\|Ax_k - b\right\| \leq \frac{c_1}{t_{k-1}^2}. \end{cases} \tag{3.20}$$

Proof. Denote

$$c_1 = \mathbf{E}_1 + \frac{(1-\tau_1)t_1^2 + 2\alpha L/r}{2}\alpha\beta\left\|Ax_1 - b\right\|^2,$$

where $\mathbf{E}_1$ defined by (3.15) is a nonnegative constant. According to (3.17) and the definition of $\mathbf{E}_k$ in (3.1), we have

$$\mathcal{L}(x_k, \lambda^*) - \mathcal{L}(x^*, \lambda^*) \leq \frac{c_1}{t_{k-1}^2}$$

and

$$\left\|\lambda_k - \lambda^*\right\| \leq \sqrt{2\gamma\beta c_1}. \tag{3.21}$$

The equality in (3.4) can be rewritten as $u_{k+1} = \frac{1}{\alpha}\big[t_k x_{k+1} - (t_k - \alpha)x_k\big]$, which by using the auxiliary notations

$$h_k := t_{k-1}^2(Ax_k - b) \quad \text{and} \quad a_k := \frac{t_{k-1}^2 - t_k(t_k-\alpha)}{t_{k-1}^2} \tag{3.22}$$

gives

$$\begin{aligned}
\lambda_{k+1} - \lambda_1 &= \sum_{i=1}^{k}(\lambda_{i+1} - \lambda_i) = \alpha\gamma\beta\sum_{i=1}^{k} t_i\big(Au_{i+1} - b\big) \\
&= \gamma\beta\sum_{i=1}^{k}\big[t_i^2\big(Ax_{i+1} - b\big) - t_i\big(t_i - \alpha\big)\big(Ax_i - b\big)\big] \\
&= \gamma\beta\sum_{i=1}^{k}\Big[h_{i+1} - h_i + \frac{t_{i-1}^2 - t_i(t_i-\alpha)}{t_{i-1}^2}h_i\Big] \\
&= \gamma\beta\Big[h_{k+1} - h_1 + \sum_{i=1}^{k} a_i h_i\Big].
\end{aligned}$$

Combine this relationship with (3.21) to obtain

$$\Big\|h_{k+1} + \sum_{i=1}^{k} a_i h_i\Big\| \le \|h_1\| + \frac{\|\lambda_{k+1} - \lambda^* + \lambda^* - \lambda_1\|}{\gamma\beta} \le \|h_1\| + \frac{\|\lambda_1 - \lambda^*\| + \sqrt{2\gamma\beta c_1}}{\gamma\beta}.$$

By (1.6), we know a_k defined in (3.22) belongs to $[0,1)$ for all $k \ge 1$. Hence, it follows from Lemma 2.1 that

$$\|h_k\| \le c_2 := 3\|h_1\| + \frac{2\|\lambda_1 - \lambda^*\| + 2\sqrt{2\gamma\beta c_1}}{\gamma\beta}, \quad \forall k \ge 1.$$

Combining it with the definition of h_k in (3.22) leads to

$$\|Ax_k - b\| = \frac{\|h_k\|}{t_{k-1}^2} \le \frac{c_2}{t_{k-1}^2}, \quad \forall k \ge 1.$$

Besides, it follows from the definition of $\mathcal{L}(x,\lambda)$ and the first inequality in (3.20) that

$$\big|\theta(x_k) - \theta(x^*)\big| \le \mathcal{L}(x_k,\lambda^*) - \mathcal{L}(x^*,\lambda^*) + \big\|\lambda^*\big\|\big\|Ax_k - b\big\| \le \frac{c_1 + c_2\big\|\lambda^*\big\|}{t_{k-1}^2}.$$

The proof is completed. ■

Finally, combining the definition of c_1 in Theorem 3.2 and Lemma 3.2, we conclude that there exists a constant $c_3 = \frac{2c_1}{\underline{\tau}\lambda_{\min}(\mathcal{D})} > 0$ with $\underline{\tau}$ being the lower bound of $\{\tau_k\}$ such that $\big\|u_k - x^*\big\|^2 \le \frac{c_3}{\alpha t_k^2}$. Note that this result, including both relaxation factor and Nesterov parameter, implies that $\{u_k\}$ converges to x^* in the worst-case $\mathcal{O}(\frac{1}{\alpha t_k^2})$ accelerated rate.

3.3 Adaptive parameters

Since the selection of $\{t_k\}$ will affect the convergence rate of our algorithm as shown in (3.20) and the subsequent experiments, next we analyze some properties of this sequence and provide feasible updating rules.

Lemma 3.3 *For any $\alpha \in (0,2)$ and every $k \geq 1$, we have*
(i) The sequence $\{t_k\}$ satisfying (1.6) yields

$$t_k - t_{k-1} \leq \frac{\sqrt{5}-1}{2}\alpha, \quad \text{and} \quad t_k \leq \frac{\sqrt{5}+1}{2}\alpha k;$$

(ii) If we choose

$$t_k = \frac{\alpha + \sqrt{\alpha^2 + 4t_{k-1}^2}}{2}, \tag{3.23}$$

then (1.6) holds and moreover $t_k \geq \frac{k+1}{2}\alpha$.

Proof. The way of updating t_k in (1.6) implies $\alpha \leq t_k \leq \frac{\alpha+\sqrt{\alpha^2+4t_{k-1}^2}}{2}$ and hence

$$t_k - t_{k-1} \leq \frac{\alpha + \sqrt{\alpha^2 + 4t_k^2}}{2} - t_{k-1}.$$

Let $\phi(y) = \frac{\alpha+\sqrt{\alpha^2+4y^2}}{2} - y$. Then, the fact that $\phi'(y) = \frac{2y}{\sqrt{\alpha^2+4y^2}} - 1 < 0$ shows the nonincreasing property of $\phi(y)$. So, we deduce

$$\begin{cases} t_k - t_{k-1} \leq \phi(t_0) = \phi(\alpha) = \frac{\sqrt{5}-1}{2}\alpha, \\ t_k \leq t_0 + k\phi(\alpha) \leq (\alpha + \phi(\alpha))k = \frac{\sqrt{5}+1}{2}\alpha k. \end{cases}$$

The first conclusion in (ii) can be proved by introduction. ■

Remark 3.2 *Lemma 3.3 distinguishes itself from [6, Lemma 3.5] due to the fact that $\alpha \in (0,2)$. The rules in (ii) offer viable options for achieving an $\mathcal{O}(1/k^2)$ convergence rate. In fact, the rule specified in (3.23) identical to those in [1, 2, 6], yet the range of α differs from the existing literature. More precisely, in [1, 6], α is an artificial parameter constrained to be less than or equal to 1. When $\alpha = 1$, the rule in (3.23) reduces to the classical Nesterov rule [22]. However, when $\alpha \neq 1$, (3.23) diverges from the strategy outlined in [20, Lemma 2.1], which is given by $t_k = \frac{p+\sqrt{q+4t_{k-1}^2}}{2}$ with $p \in (0,1]$ and $q > 0$.*

4 Numerical experiments

In this section, we carry out a series of experiments on large-scale sparse signal reconstruction problems with the aim of assessing the performance of our AP-ALM (i.e., Algorithm 1.1). All experiments are implemented using MATLAB R2019b (64-bit) and performed on a PC with Windows 10 operating system, equipping with an Intel i7-8565U CPU and 16GB RAM.

Recalling the following linearly constrained ℓ_1-ℓ_2 minimization problem:

$$\min_{x\in\mathbb{R}^m} \theta(x) = \|x\|_1 + \frac{\mu}{2}\|x\|^2, \qquad \text{s.t. } Ax = b, \tag{4.1}$$

where A is an $n \times m$ measurement matrix, $b \in \mathbb{R}^n$ is a response vector and $\mu > 0$ is the regularization parameter. The goal of model (4.1) is to reconstruct a signal x that closely approximates the original sparse signal x^*, utilizing the given data A and b. Notably, when $\mu = 0$, the above problem reduces to the popular basis pursuit problem in compressive sensing. In the following experiments, the matrix A is configured as a Gaussian measurement matrix whose elements are randomly drawn from standard

Gaussian distribution $\mathcal{N}(0,1)$. The original true signal x^* contains a specific number of non-zero elements (which we fix at $m/50$), and each of these non-zero elements is sampled from a Gaussian distribution $\mathcal{N}(0,2)$. The response vector b is generated by $b = Ax^* + \epsilon$, where the noise ϵ is first generated by standard Gaussian distribution m $\mathcal{N}(0,1)$ and subsequently normalized with the norm 10^{-5}.

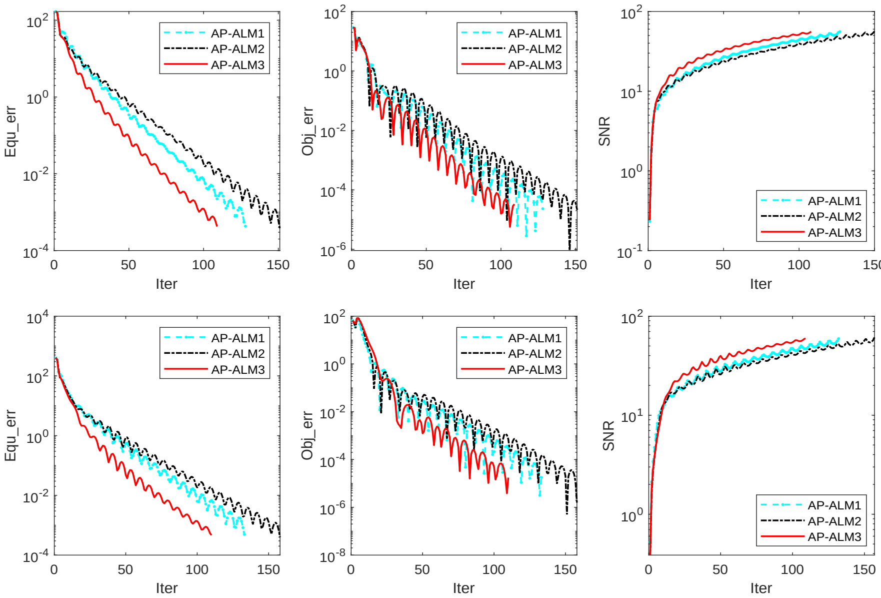

Figure 1: Comparison of three types of AP-ALM for solving Problem (4.1): the top with $(n,m) = (500,1000)$ and the bottom with $(n,m) = (1000,2000)$.

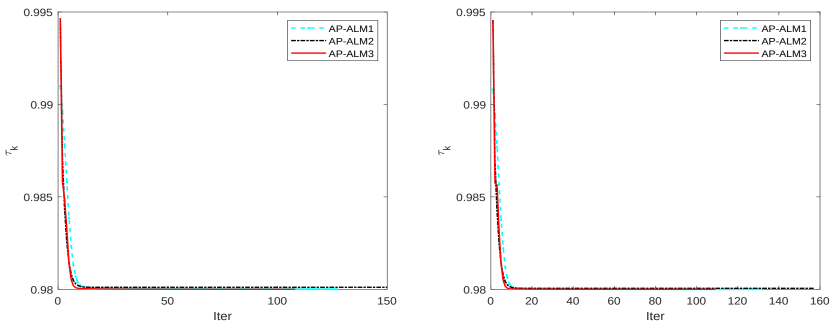

Figure 2: Variation tendency of τ_k by three types of AP-ALM for solving Problem (4.1): the left with $(n,m) = (500,1000)$ and the right with $(n,m) = (1000,2000)$.

4.1 Performance of AP-ALM with different parameter and data

In this subsection, we initiate the evaluation of the performance of our AP-ALM under different strategies: (s1) $t_k = \frac{p+\sqrt{q+4t_{k-1}^2}}{2}$ with $(p,q) = (1/20, 1/2)$; (s2) $t_k = \frac{\alpha+\sqrt{\alpha^2+4t_{k-1}^2}}{2}$; (s3) $t_k = \alpha + \frac{k}{c-1}$ with $c = 7$, simply denoted by AP-ALM1, AP-ALM2 and AP-ALM3, respectively. The initial points are set as $(x_1, \lambda_1) = (\mathbf{0}, \mathbf{0})$ and the regularization parameter is fixed as $\mu = 0.001$. For the parameter τ_k, it is selected as the average of the upper and lower bounds specified (1.5). The penalty parameter and relaxation stepsize use the tuned values $(\beta, \alpha) = (0.001, 1.2)$. Then, we use tuned values $(r, \gamma) = (\beta\|A^\top A\| + 0.001, 1.6)$. According to Theorem 3.2, the following qualities will be employed to examine the performance of our proposed methods:

- Constraint violation $\text{Equ\_err}(k) = \|Ax_k - b\|$;
- Objective error: $\text{Obj\_err}(k) = |\theta(x_k) - \theta(x^*)|$;
- Signal-to-noise ratio $\text{SNR}(k) = 10\log_{10}\frac{\|x^* - \text{mean}(x^*)\|_2}{\|x_{k+1} - x^*\|_2}$.

By applying AP-ALM to the problem (4.1) with $(n, m) \in \{(500, 1000), (1000, 2000)\}$ under the stopping criterion $\|Ax_k - b\| \leq 5 \times 10^{-4}$ (as also referenced in [18, 19]), Figure 1 illustrates the convergence behaviours of the aforementioned qualities. Figure 2 implies that a proximal-indefinite term is exploited in experiments as τ_k is nonincreasing and smaller than 1. From Figure 1, it is evident that AP-ALM3 performs significantly better than AP-ALM1 and AP-ALM2. Consequently, in the subsequent experiments, we will adopt AP-ALM3 as the default implementation of Algorithm 1.1.

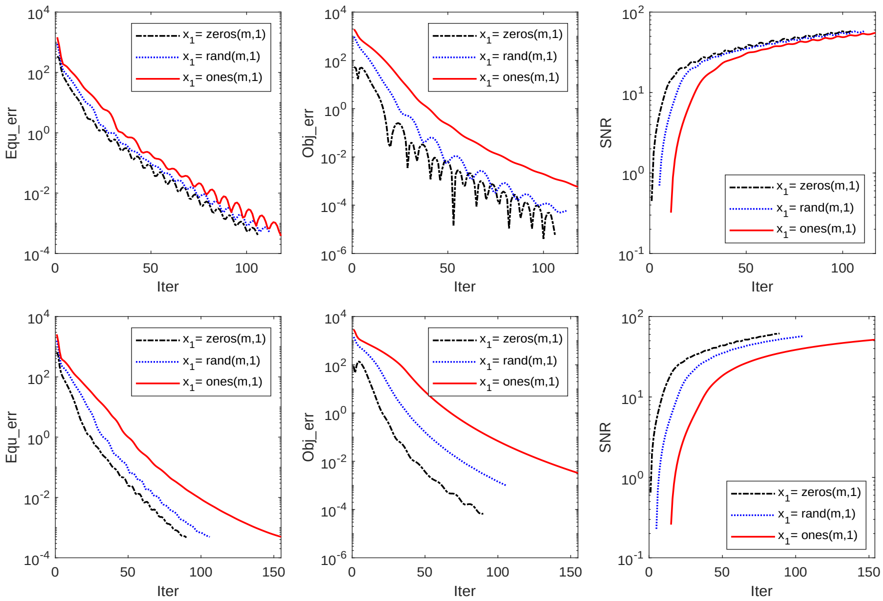

Figure 3: Comparison of AP-ALM3 for solving Problem (4.1) with different initial point: the top with $(n, m) = (1000, 2000)$ and the bottom with $(n, m) = (2000, 3000)$.

Figure 3 presents a comparison of AP-ALM3 using different initial points x_1. It is evident that AP-ALM3 exhibits relatively superior performance when x_1 is set as a zero vector. As a result, we will adopt the zero vector as the default initial point for our method. Finally, we conduct an investigation on the performance of AP-ALM3 for solving the problem (4.1), considering various methods of generating a square matrix A: (1) Gaussian measurement matrix generated in the same manner as previously described; (2) Skew symmetric matrix generated using the formula $A = B + B^\top$, where $B = \text{randn}(n)$; (3) Tridiagonal matrix constructed by $A = \text{diag}(\text{ones}(1, n) * 4) + \text{diag}(-\text{ones}(1, n-1), 1) + \text{diag}(-\text{ones}(1, n-1), -1)$. Figure 4 provides the convergence curves of AP-ALM3 for the problem (4.1) with $n \in \{2000, 4000\}$. As illustrated in Figure 4, the proposed method is demonstrated to be remarkable valid in handling different structures of A.

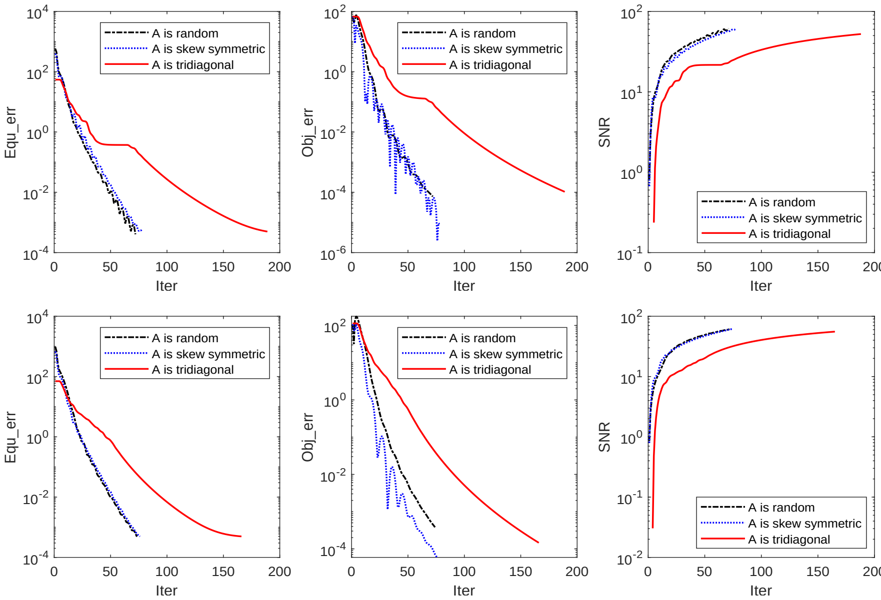

Figure 4: Comparison of AP-ALM3 for solving Problem (4.1) with different constrained matrix A: the top with $n = 2000$ and the bottom with $n = 4000$.

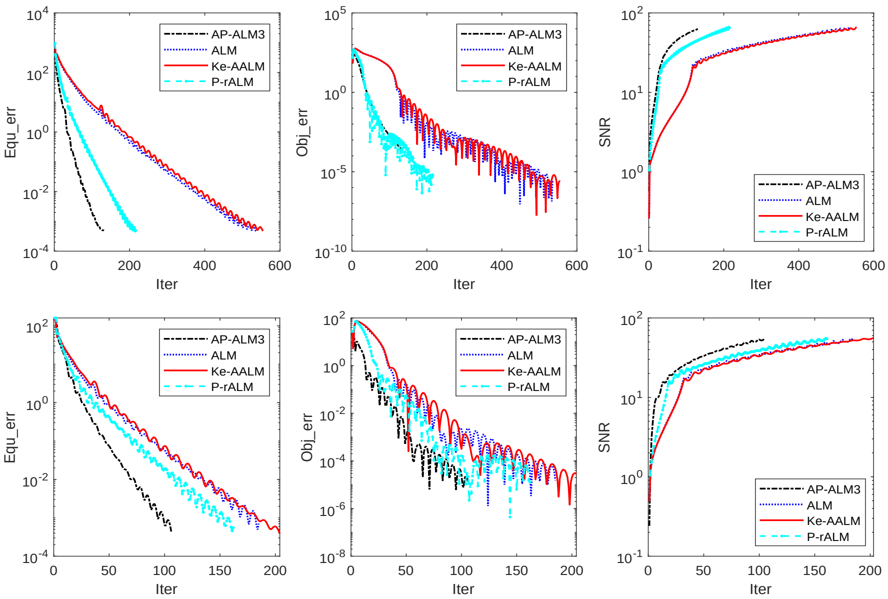

Figure 5: Comparison of different ALM-type methods for solving Problem (4.1): the top with $(n, m) = (500, 1000)$ and the bottom with $(n, m) = (2000, 4000)$.

4.2 Comparison of AP-ALM with other existing methods

In this subsection, we use AP-ALM3 as the default algorithm and compare it with several existing accelerated ALM-type methods:

- the accelerated ALM (Ke-ALM, [19]) with tuned penalty value $\beta = m/(10\|A^\top b\|)$ which performs better than the original setting in [19];
- The well-known ALM (ALM, that is (1.3) with $\mathcal{D}_0 = \mathbf{0}$) with tuned dual stepsize $\gamma = 1.8$ and $\beta = m/(10\|A^\top b\|)$;
- the double-penalty ALM (P-rALM, [4]) with the involved relaxation factor $\gamma = 1.4$, $r = m/(10\|A^\top b\|)$, and the proximal matrix $Q = \tau \mathbf{I} - rA^\top A$ with $\tau = 1.1r\|A^\top A\|$.

The problem data are generated in the same way as that in the first part of Section 4.1, but the penalty parameter μ is selected as 0.01 hereafter. Figure 5 depicts the convergence curves of different ALM-type methods described above. Additionally, Figure 6 shows the CPU time comparisons, and Figure 7 presents the comparison between the original signal and the signal reconstructed by AP-ALM3 when the dimensions are $(n, m) = (2000, 4000)$, where the bottom subfigure shows the minimum energy reconstruction signal $A^\dagger b$ (the point satisfying $A^\top A = A^\top b$) versus the original signal. After identifying the nonzero positions in the reconstructed signal shown in Figure 7, it always has the correct number of spikes for the case with dimensions $(n, m) = (2000, 4000)$ and is closer to the original signal. In addition, we can observe that the proposed AP-ALM3 converges significantly faster than other existing methods in terms of the measurement metrics. Although Ke-ALM shares the same $\mathcal{O}(1/k^2)$ accelerated convergence rate as our AP-ALM3, it performs less favorably compared to P-rALM that has a $\mathcal{O}(1/k)$ convergence rate, where k denotes the number of iterations. These comparison results validate the numerical acceleration of our proposed method and its theoretical acceleration as sated in Theorem 3.2.

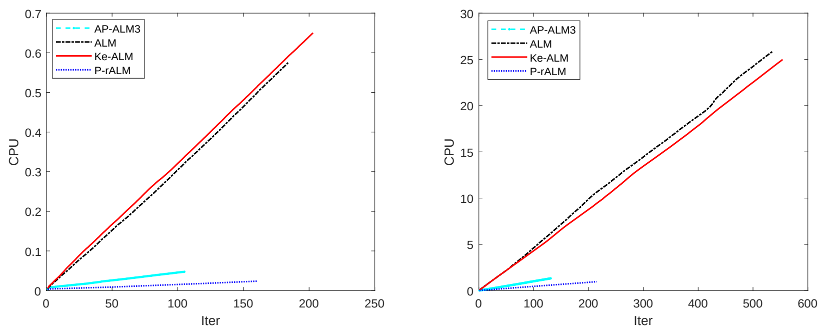

Figure 6: Comparison of CPU time by different ALM-type methods for solving Problem (4.1): the left with $(n, m) = (500, 1000)$ and the right with $(n, m) = (2000, 4000)$.

5 Concluding remarks

The so-called Nesterov acceleration technique has been extensively investigated in some first-order methods to improve the theoretical convergence rate, however, the resulting subproblem often remains as challenging to solve as the original problem. To address this limitation, this paper introduces an accelerated proximal-indefinite augmented Lagrangian method (AP-ALM) for linearly constrained composite convex programs. Our

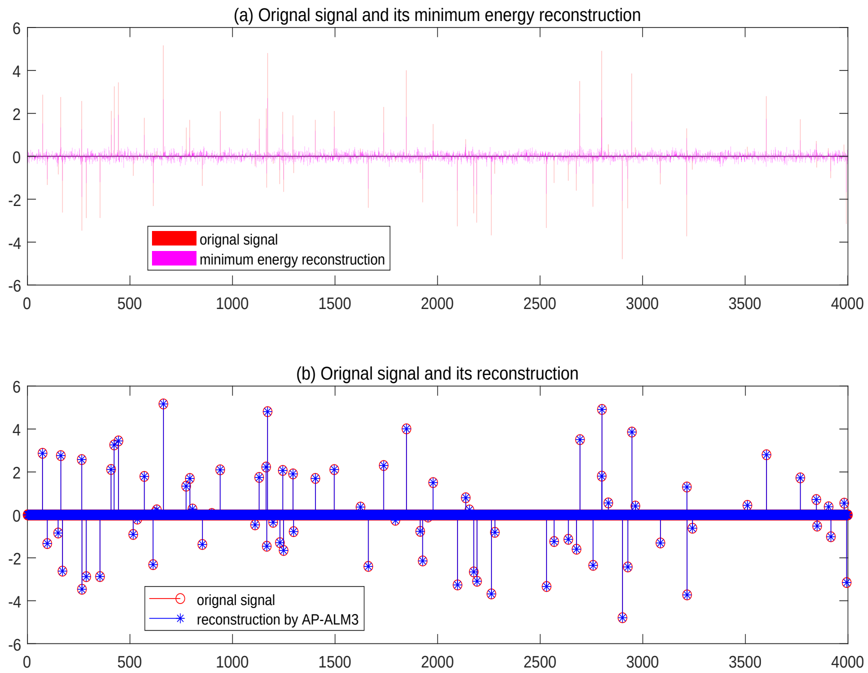

Figure 7: Original signal and reconstructed signal by AP-ALM3 for the case with $(n, m) = (2000, 4000)$.

proposed method not only incorporates a general Nesterov acceleration technique, but also admits a much easier proximal subproblem by exploiting a widely-used proximal term. Additionally, the so-called relaxation step is implemented to numerically accelerate the method. We have conducted a comprehensive analysis of the global convergence of AP-ALM and its accelerated convergence rates. Furthermore, the numerical performance of AP-ALM has been validated through comparisons with several existing methods in the context of large-scale sparse signal reconstruction problems. In the future, our research will focus on developing a stochastic version of this AP-ALM but with the general Bregman distance [5] as the proximal term for some nonconvex composite programming problems arising from machine learning.

Acknowledgements

The authors would like to thank the editor and anonymous referees for their valuable comments and suggestions, which have significantly improved the quality of the paper.

References

[1] H. Attouch, A. Cabot, Convergence rates of inertial forward-backward algorithms, SIAM J. Optim., 28, 849-874 (2018)

[2] J. Bai, J. Li, F. Xu, P. Dai, A novel method for a class of structured low rank minimization with equality constraint, J. Comput. Appl. Math., 330, 475-487 (2018)

[3] J. Bai, D. Han, H. Sun, H. Zhang, Convergence on a symmetric accelerated stochastic ADMM with larger stepsizes, CSIAM Trans. Appl. Math., 3, 448-479 (2022)

[4] J. Bai, L. Jia, Z. Peng, A new insight on augmented Lagrangian method with applications in machine learning, J. Sci. Comput., 99, 53, (2024)

[5] J. Bai, X. Cui, Z. Wu, A proximal-perturbed Bregman ADMM for solving nonsmooth and nonconvex composite optimization, Numer. Math. Theor. Meth. Appl., To Appear, (2026)

[6] R. Bot, E. Csetnek, D. Nguyen, Fast augmented Lagrangian method in the convex regime with convergence guarantees for the iterates, Math. Program., 200, 147-197 (2023)

[7] Y. Ding, H. Zhang, P. Li, Y. Xiao, An efficient semismooth Newton method for adaptive sparse signal recovery problems, Optim. Methods Soft., 38, 262-288 (2023)

[8] D. Han, X. Yuan, W. Zhang, An augmented-Lagrangian-based parallel splitting method for separable convex minimization with applications to image processing, Math. Comput., 83, 2263-2291 (2014)

[9] M. Hestenes, Multiplier and gradient methods, J. Optim. Theory Appl., 4, 303-320 (1969)

[10] B. He, X. Yuan, On the acceleration of augmented Lagrangian method for linearly constrained optimization, Optimization Online, (2010) https://optimization-online.org/2010/10/2760

[11] B. He, X. Yuan, An accelerated inexact proximal point algorithm for convex minimization, J. Optim. Theory Appl., 154, 536-548 (2012)

[12] B. He, F. Ma, X. Yuan, Optimal proximal augmented Lagrangian method and its application to full Jacobian splitting for multi-block separable convex minimization problems, IMA J. Numer. Anal., 40, 1188-1216 (2020)

[13] H. He, X. Cai, D. Han, A fast splitting method tailored for Dantzig selector, Comput. Optim. Appl., 62, 347-372 (2015)

[14] X. He, R. Hu, Y. Fang, Fast primal-dual algorithm via dynamical system for a linearly constrained convex optimization problem, Automatica, 146, 110547 (2022)

[15] X. He, R. Hu, Y. Fang, Inertial accelerated augmented Lagrangian algorithms with scaling coefficients to solve exactly and inexactly linearly constrained convex optimization problems, J. Comput. Appl. Math., 460, 116425 (2025)

[16] X. He, N. Huang, Y. Fang, Accelerated linearized alternating direction method of multipliers with Nesterov extrapolation, arXiv:2310.16404, (2023)

[17] B. Huang, S. Ma, D. Goldfarb, Accelerated linearized Bregman method, J. Sci. Comput., 54, 428-453 (2013)

[18] M. Kang, M. Kang, M. Jung, Inexact accelerated augmented Lagrangian methods, Comput. Optim. Appl., 62, 373-404 (2015)

[19] Y. Ke, C. Ma, An accelerated augmented Lagrangian method for linearly constrained convex programming with the rate of convergence $\mathcal{O}(1/k^2)$, Appl. Math. (A Journal of Chinese Universities), 32, 117-126 (2017)

[20] J. Liang, T. Luo, C. Schonlieb, Improving "fast iterative shrinkage-thresholding algorithm": faster, smarter, and greedier, SIAM J. Sci. Comput., 44, A1069-A1091 (2022)

[21] X. Long, J. Nie, Z. Gou, X. Sun, G. Li, A accelerated preconditioned primal-dual gradient algorithm for structured nonconvex optimization problems, Commun. Nonlinear Sci. Numer. Simulat., 153, 109480 (2026)

[22] Y. Nesterov, A method for unconstrained convex minimization problem with the rate of convergence $\mathcal{O}(1/k^2)$, Doklady AN USSR, 269, 543-547 (1983)

[23] S. Osher, H. Heaton, S. Fung, A Hamilton-Jacobi-based proximal operator, PNAS, 120, e2220469120 (2023)

[24] H. Robbins, D. Siegmund, A convergence theorem for non negative almost supermartingales and some applications, Optimizing methods in statistics, Academic Press, 233-257 (1971)

[25] M. Schmidt, N. Roux, F. Bach, Convergence rates of inexact proximal-gradient methods for convex optimization, arXiv:1109.2415v2 (2011)

[26] S. Sabach, M. Teboulle, Faster Lagrangian-based methods in convex optimization, SIAM J. Optim., 32, 204-227 (2022)

[27] Z. Wu, Y. Song, F. Jiang, Inexact generalized ADMM with relative error criteria for linearly constrained convex optimization problems, Optim. Lett., 18, 447-470 (2024)

[28] Y. Xu, Accelerated first-order primal-dual proximal methods for linearly constrained composite convex programming, SIAM J. Optim., 27, 1459-1484 (2017)

[29] X. Yu, Z. Liu, Y. Sun, W. Wang, Clustered federated learning for heterogeneous data, AAAI, 37, 16378-16379 (2023)