

Accelerated Symmetric ADMM and Its Applications in Signal Processing ^{*}

Jianchao Bai [†] Junli Liang^{* ‡} Ke Guo [§] Yang Jing [¶]

June 14th, 2019

Abstract

The alternating direction method of multipliers (ADMM) were extensively investigated in the past decades for solving separable convex optimization problems. Fewer researchers focused on exploring its convergence properties for the nonconvex case although it performed surprisingly efficient. In this paper, we propose a symmetric ADMM based on different acceleration techniques for a family of potentially nonsmooth nonconvex programming problems with equality constraints, where the dual variables are updated twice with different stepsizes. Under proper assumptions instead of using the so-called Kurdyka-Lojasiewicz inequality, convergence of the proposed algorithm as well as its pointwise iteration-complexity are analyzed in terms of the corresponding augmented Lagrangian function and the primal-dual residuals, respectively. Performance of our algorithm is verified by some preliminary numerical examples on applications in sparse nonconvex/convex regularized minimization signal processing problems.

Keywords: Nonconvex optimization, symmetric ADMM, acceleration technique, complexity, signal processing

Mathematics Subject Classification(2010): 47A30; 65Y20; 90C26; 90C90

1 Introduction

We consider a potentially nonsmooth and nonconvex separable optimization problem subject to linear equality constraints:

$$\min \{f(\mathbf{x}) + g(\mathbf{y}) \mid \text{s.t. } \mathbf{Ax} + \mathbf{By} = \mathbf{b}, \mathbf{x} \in \mathcal{R}^m, \mathbf{y} \in \mathcal{R}^n\}, \quad (1)$$

^{*}This research was partially supported by the National Statistical Science Research Project of China (Grant No. 2018LZ23), the Natural Science Foundation of China (Grant No. 11801455, 11571178), Fundamental Research Funds of China West Normal University (Grant No. 17E084, 18B031).

[†]Department of Applied Mathematics, Northwestern Polytechnical University, Xi'an 710129, China (jianchaobai@nwpu.edu.cn).

[‡](Corresponding author) School of Electronics and Information, Northwestern Polytechnical University, Xi'an 710129, China (liangjunli@nwpu.edu.cn).

[§]School of Mathematics and Information, China West Normal University, Nanchong, Sichuan 637002, China (keguo2014@126.com).

[¶]School of Electronics and Information, Northwestern Polytechnical University, Xi'an 710129, China (jingyang@mail.nwpu.edu.cn).

where $f : \mathcal{R}^m \rightarrow (-\infty, +\infty]$ is a proper lower semicontinuous function, $g : \mathcal{R}^n \rightarrow (-\infty, +\infty)$ is a continuous differentiable function with its gradient ∇g being L_g -Lipschitz continuous, $A \in \mathcal{R}^{l \times m}$, $B \in \mathcal{R}^{l \times n}$, $b \in \mathcal{R}^l$ are respectively given matrices and vector. Minimization problem in the form of (1) covers many important applications in science and engineering. For example, the following l_1 -regularized least square problem arising in signal processing/statistical learning [2, 3, 25]:

$$\min_{x \in \mathcal{R}^m} \frac{1}{2} \|A\mathbf{x} - c\|^2 + \mu \|\mathbf{x}\|_1, \quad (2)$$

where $c \in \mathcal{R}^l$ is the vector of observations, $A \in \mathcal{R}^{l \times m}$ is the data matrix and $\mu > 0$ denotes the regularization parameter and is often set as $\mu = 0.1\mu_{\max}$ where $\mu_{\max} = \|A^\top c\|_\infty$ (see e.g. [11, 25]). Due to the convexity of the problem (2), it can be handled by a number of standard methods, to list a few, including the alternating direction method of multipliers (ADMM, [10, 13, 14]), proximal point algorithm [3, 10], interior point method [25] and primal-dual hybrid gradient method [5, 39]. However, in many cases the l_1 -regularization has been shown to be sub-optimal. For instance, it can not recover a signal with the fewest measurements when being applied in compressed sensing techniques [6]. Therefore, an acceptable improvement is to adopt the $l_{1/2}$ -regularization term, which results in the following form

$$\min_{x \in \mathcal{R}^m} \frac{1}{2} \|A\mathbf{x} - c\|^2 + \mu \|\mathbf{x}\|_{1/2}^{1/2}.$$

Here, $\|\mathbf{x}\|_{1/2} = (\sum_{i=1}^n |\mathbf{x}_i|^{1/2})^2$ is a nonconvex function characterizing sparsity of the variable, and it has been verified [37] practically to be better than l_1 -norm. Clearly, by introducing an auxiliary variable, the problem can be converted to a special case of (1), i.e.,

$$\min \left\{ \mu \|\mathbf{x}\|_{1/2}^{1/2} + \frac{1}{2} \|\mathbf{y} - c\|^2 \mid \text{s.t. } A\mathbf{x} - \mathbf{y} = 0 \right\}. \quad (3)$$

Another interesting example is the regularized empirical risk minimization arising from big data applications, such as many kinds of classification and regression models in machine learning [34, 36]. And the $l_{1/2}$ -regularized reformulation case is of the form:

$$\min \left\{ \mu \|\mathbf{x}\|_{1/2}^{1/2} + \frac{1}{N} \sum_{j=1}^N g_j(\mathbf{y}) \mid \text{s.t. } \mathbf{x} - \mathbf{y} = 0 \right\}, \quad (4)$$

where N is a large number, and $g_j(\mathbf{y}) = \log(1 + \exp(-b_j a_j^\top \mathbf{y}))$ denotes the logistic loss function on the feature-label pair (a_j, b_j) with $a_j \in \mathcal{R}^l$ and $b_j \in \{-1, 1\}$.

In the literature, the most standard method for solving the equality constrained problem (1) is the augmented Lagrangian method (ALM) which firstly solves a joint minimization problem

$$\min_{\mathbf{x}, \mathbf{y}} \mathcal{L}_\beta(\mathbf{x}, \mathbf{y}, \lambda) := f(\mathbf{x}) + g(\mathbf{y}) - \langle \lambda, A\mathbf{x} + B\mathbf{y} - b \rangle + \frac{\beta}{2} \|A\mathbf{x} + B\mathbf{y} - b\|^2, \quad (5)$$

and then updates the Lagrange multiplier λ by using the newest iteration of other variables. The penalty factor $\beta > 0$, in each iterative loop, can be set as a tuned reasonable value or updated adaptively according to the ratio of the primal residual to the dual residual of the problem. However, ALM does not make full use of the separable structure of the objective function of (1) and hence, could not take advantage of the special properties of each component objective

function. This would make it very expensive even infeasible for application problems involving big-data and nonconvex objectives. By contrast, a powerful first-order method, that is ADMM, aims to split the joint core problem (5) into some relatively simple and smaller-dimensional subproblems so that variables can be updated separately to make full use of special properties of each component. Another obvious feature of ADMM is that the resultant subproblems could admit explicit solution form in special applications, or in a linearized update for the differentiable objective/quadratic penalty term. We refer to, e.g., [2, 3, 12, 15, 23, 22, 35] for some reviews on ADMM.

Interestingly, under the existence assumption of a solution to the Karush-Kuhn Tucker condition of the two-block separable convex optimization problem, it was explained [13] that the original ADMM amounts to the Douglas-Rachford splitting method (DRSM, [9, 26]) when it was applied to a stationary system to the dual of the problem. Moreover, as elaborated in [13], if applying the classic Peaceman-Rachford splitting method (PRSM, [26, 32]) to the dual of the problem, we obtain the following iterative scheme

$$\begin{cases} \mathbf{x}_{k+1} &= \arg \min_{\mathbf{x}} \mathcal{L}_{\beta}(\mathbf{x}, \mathbf{y}_k, \lambda_k), \\ \lambda_{k+\frac{1}{2}} &= \lambda_k - \beta (A\mathbf{x}_{k+1} + B\mathbf{y}_k - b), \\ \mathbf{y}_{k+1} &= \arg \min_{\mathbf{y}} \mathcal{L}_{\beta}(\mathbf{x}_{k+1}, \mathbf{y}, \lambda_{k+\frac{1}{2}}), \\ \lambda_{k+1} &= \lambda_{k+\frac{1}{2}} - \beta (A\mathbf{x}_{k+1} + B\mathbf{y}_{k+1} - b). \end{cases} \quad (6)$$

Unfortunately, scheme (6) is not convergent under the standard convexity assumptions as ADMM [8]. However, it was verified [16] that scheme (6) could perform faster than the ADMM when its global convergent was ensured. In view of this, He et al. in [21] proposed and studied the convergence of a strictly contractive Peaceman-Rachford splitting method (also called the symmetric version of ADMM)

$$\begin{cases} \mathbf{x}_{k+1} &= \arg \min_{\mathbf{x}} \mathcal{L}_{\beta}(\mathbf{x}, \mathbf{y}_k, \lambda_k), \\ \lambda_{k+\frac{1}{2}} &= \lambda_k - \alpha\beta (A\mathbf{x}_{k+1} + B\mathbf{y}_k - b), \\ \mathbf{y}_{k+1} &= \arg \min_{\mathbf{y}} \mathcal{L}_{\beta}(\mathbf{x}_{k+1}, \mathbf{y}, \lambda_{k+\frac{1}{2}}), \\ \lambda_{k+1} &= \lambda_{k+\frac{1}{2}} - \alpha\beta (A\mathbf{x}_{k+1} + B\mathbf{y}_{k+1} - b), \end{cases} \quad (7)$$

where $\alpha \in (0, 1)$ is the relaxation parameter. Later, He et al. [23] improved the scheme (7) to the case with larger range of relaxation parameters, which was generalized by Bai et al. [4] to the multi-block separable convex programming. Besides, Chang, et. al.[7] also shown a generalization of linearized ADMM for two-block separable convex minimization model by adding a proper proximal term to each core subproblem.

If the convexity is lose, then the convergence analysis for ADMM (or its variant) is much more challenging. However, for some special nonconvex optimization problems, one can establish convergence of ADMM by making full use of special structures of the problems, see e.g. [24] for the consensus and sharing problems. Another widely used technique to prove convergence of ADMM for nonconvex optimization problems relies on the assumption that the objective function of (1) satisfies the so-called Kurdyka-Lojasiewicz (KL) inequality [1], since many important classes of functions satisfy the KL inequality, see [17, 18, 19, 27, 35, 36, 38]. Without assuming the KL property and convexity of the objective function, recently, Goncalves et al. in [20] established convergence rate bounds of the classical ADMM with proximal terms for solving nonconvex linearly constrained optimization problem (1). In addition, by linearizing the smooth

part in the objective and quadratic penalty term, Liu, et al. [28] proposed a two-block linearized ADMM for the problem (1) with $b = 0$ and extended the method to a multi-block version, but convergence of their extended method holds with an extra hypothesis on the full column rank of the matrix B compared to (A1) (see Section 3).

Motivated by the above mentioned work [20, 28] and the empirical validity of the symmetric ADMM, we would present a Two-stage Accelerated Symmetric ADMM (abbreviated as ‘‘TAS-ADM’’) for solving the problem (1), whose framework reads Algorithm 1.1. Our algorithm combines both the so-called Nesterov’s acceleration technique in (32) and the relaxation scheme in e.g., [10, 12]. By adding a proper proximal term for the first \mathbf{x} -subproblem, this possibly nonsmooth nonconvex subproblem will turn to a proximal mapping shown in (16), which admits closed solution form if f is easy. Step 7 actually uses the idea of convex combination for fast convergence.

We should emphasize that the recent work [36] also considered a symmetric ADMM for solving the problem (1). The method in [36] actually can be treated as our proposed Algorithm 1.1 barring the acceleration techniques and proximal regularization terms, while convergence of Algorithm 1.1 is analyzed in a different way. More precisely, their analyses are based on the Kurdyka-Lojasiewicz property of the augmented Lagrangian function for problem (1) and other proper assumptions on both the penalty parameter and the objective function. Under Assumptions (A1-A3) (see Section 3), we show in the sequel section that any accumulation point of $\{w_k := (\mathbf{x}_k, \mathbf{y}_k, \lambda_k)\}$ is the stationary point of $\{L_\beta(w_k)\}$, and we also establish the worst-case $\mathcal{O}(1/k)$ convergence rate of the algorithm in terms of the primal-dual residuals. Although we consider problem (1) with vector variables, the subsequent convergence results of our proposed algorithm are applicable for the general case with matrix variables, because matrix can be vectorized as vector.

Algorithm 1.1 [TAS-ADM for Solving Problem (1)]

- 1 Initialize $(\mathbf{x}_0, \mathbf{y}_0, \lambda_0) \in \mathcal{R}^m \times \mathcal{R}^n \times \mathcal{R}^l$ and set $(\mathbf{x}_{-1}, \mathbf{y}_{-1}) = (\mathbf{x}_0, \mathbf{y}_0)$.
- 2 Choose parameters $\beta > 0, \gamma_k \in [0, \frac{1}{2}), G \succeq 0$ and

$$(\tau, \alpha) \in \mathcal{D} := \{(\tau, \alpha) \mid 0 < \tau + \alpha < 1\}. \quad (8)$$

- 3 for $k = 0, 1, \dots$, do
 - 4 $\mathbf{x}_k^{md} = \mathbf{x}_k + \gamma_k(\mathbf{x}_k - \mathbf{x}_{k-1})$.
 - 5 $\mathbf{x}_{k+1} = \arg \min \left\{ \mathcal{L}_\beta(\mathbf{x}, \mathbf{y}_k, \lambda_k) + \frac{1}{2} \|\mathbf{x} - \mathbf{x}_k^{md}\|_G^2 \right\}$.
 - 6 $\lambda_{k+\frac{1}{2}} = \lambda_k - \tau\beta(\mathbf{A}\mathbf{x}_{k+1} + \mathbf{B}\mathbf{y}_k - b)$.
 - 7 $\mathbf{x}_{k+1}^{ad} = \alpha\mathbf{A}\mathbf{x}_{k+1} + (1 - \alpha)(b - \mathbf{B}\mathbf{y}_k)$.
 - 8 $\mathbf{y}_{k+1} = \arg \min \left\{ g(\mathbf{y}) - \left\langle \lambda_{k+\frac{1}{2}}, \mathbf{B}\mathbf{y} \right\rangle + \frac{\beta}{2} \|\mathbf{x}_{k+1}^{ad} + \mathbf{B}\mathbf{y} - b\|^2 \right\}$.
 - 9 $\lambda_{k+1} = \lambda_{k+\frac{1}{2}} - \beta(\mathbf{x}_{k+1}^{ad} + \mathbf{B}\mathbf{y}_{k+1} - b)$.
 - 10 end
 - 11 Output $(\mathbf{x}_{k+1}, \mathbf{y}_{k+1})$.
-

The remaining parts of this paper are organized as follows. In Section 2, some preliminaries are prepared to analyze convergence of Algorithm 1.1. In Section 3, we show its convergence properties and its pointwise iteration complexity based on the analysis for the augmented Lagrangian sequence $\{L_\beta(w_k)\}$. Section 4 tests some examples about the popular sparse signal

recovery problem with different regularization terms and compared with the popular CVX toolbox, which aims to investigate numerical performance of our algorithm. Finally, we conclude the paper in Section 5.

2 Preliminaries

Throughout this paper, let $\mathcal{R}, \mathcal{R}^n, \mathcal{R}^{m \times n}$ be the sets of real numbers, n dimensional real column vectors and $m \times n$ dimensional real matrices, respectively. The symbol I denotes the identity matrix with proper dimension and σ_B denotes the smallest positive eigenvalue of the matrix BB^\top . For any symmetric matrices A and B whose dimensions are the same, $A \succ B$ ($A \succeq B$) means $A - B$ is a positive definite (semidefinite) matrix. We slightly denote $\|x\|_G^2 = x^\top Gx$ for any symmetric matrix G , and let $\|x\|_G = \sqrt{x^\top Gx}$ when G is positive semidefinite, where the superscript \top denotes the transpose of a matrix or vector. We simply use $\|\cdot\|$ to represent the standard Euclidean norm equipped with inner product $\langle \cdot, \cdot \rangle$. The image space of a matrix $A \in \mathcal{R}^{m \times n}$ is defined as $\text{Im}(A) := \{Az \mid z \in \mathcal{R}^n\}$ and a function $f : \mathcal{S} \rightarrow \mathcal{R}$ is lower semicontinuous at $\bar{x} \in \mathcal{S}$ if and only if $\liminf_{x \rightarrow \bar{x}} f(x) = f(\bar{x})$. The distance from any point z to the set $\mathcal{S} \subseteq \mathcal{R}^n$ is defined as $d(z, \mathcal{S}) := \inf\{\|z - y\| \mid y \in \mathcal{S}\}$.

Definition 2.1 [30, 33] *Let $f : \mathcal{R}^m \rightarrow \mathcal{R}$ be a proper lower semicontinuous function.*

- (a) *For a given $x \in \text{dom}(f)$, the Frechet subdifferential of f at x , written by $\hat{\partial}f(x)$, is the set of all vectors $s \in \mathcal{R}^m$ which satisfy*

$$\liminf_{y \neq x, y \rightarrow x} \frac{f(y) - f(x) - \langle s, y - x \rangle}{\|y - x\|} \geq 0,$$

and we let $\hat{\partial}f(x) = \emptyset$ when $x \notin \text{dom}(f)$.

- (b) *The limiting subdifferential, or the subdifferential of f at $x \in \mathcal{R}^m$, written by $\partial f(x)$, is defined by $\partial f(x) = \left\{ s \in \mathcal{R}^m \mid \exists x_k \rightarrow x, f(x_k) \rightarrow f(x), \hat{\partial}f(x_k) \ni s_k \rightarrow s \text{ as } k \rightarrow \infty \right\}$.*
- (c) *A point x_* is called critical point or stationary point of $f(x)$ if it satisfies $0 \in \partial f(x_*)$.*

Definition 2.2 *A triple $w_* := (\mathbf{x}_*, \mathbf{y}_*, \lambda_*) \in \mathcal{R}^m \times \mathcal{R}^n \times \mathcal{R}^l$ is a stationary point of (1) if*

$$A^\top \lambda_* \in \partial f(\mathbf{x}_*), \quad B^\top \lambda_* = \nabla g(\mathbf{y}_*) \quad \text{and} \quad A\mathbf{x}_* + B\mathbf{y}_* - b = 0.$$

The following lemmas are provided to simplify convergence analysis in the sequel sections.

Lemma 2.1 [20, Lemma A.2] *Let $A \in \mathcal{R}^{m \times n}$ be a nonzero matrix and \mathcal{P}_A be the Euclidean projection onto $\text{Im}(A)$. Then, for any $u \in \mathcal{R}^n$ we have*

$$\|\mathcal{P}_A(u)\| \leq \frac{1}{\sqrt{\sigma_A}} \|A^\top u\|. \quad (9)$$

Lemma 2.2 *For any vectors $a, b, c \in \mathcal{R}^n$ and symmetric matrix $0 \preceq M \in \mathcal{R}^{n \times n}$, it holds*

$$\langle a - b, M(a - c) \rangle = \frac{1}{2} \left\{ \|c - a\|_M^2 - \|c - b\|_M^2 + \|a - b\|_M^2 \right\}. \quad (10)$$

3 Theoretical Results

In this section, by making use of the following primal-dual residuals

$$\Delta \mathbf{x}_k = \mathbf{x}_k - \mathbf{x}_{k-1}, \quad \Delta \mathbf{y}_k = \mathbf{y}_k - \mathbf{y}_{k-1} \quad \text{and} \quad \Delta \lambda_k = \lambda_k - \lambda_{k-1}, \quad (11)$$

the proposed algorithm will be demonstrated to be convergent according to a quasi-monotonically nonincreasing property of the sequence $\{\mathcal{L}_\beta(w_k)\}$, and its pointwise iteration-complexity will be established in detail. Next, we make some assumptions.

- **(A1)** $B \neq 0$, $\text{Im}(B) \supset b \cup \text{Im}(A)$;
- **(A2)** The penalty parameter β satisfies

$$\beta > \frac{L_g}{\sqrt{1 - \tau - \alpha\sigma_B}}, \quad (\tau, \alpha) \in \mathcal{D} \text{ with } \mathcal{D} \text{ given in (8);}$$

- **(A3)** $\underline{L} = \inf_{(\mathbf{x}, \mathbf{y})} \left\{ f(\mathbf{x}) + g(\mathbf{y}) - \frac{1}{2L_g} \|\nabla g(\mathbf{y})\|^2 \right\} > -\infty$.

Indeed, we can check that the aforementioned Assumptions (A1)-(A3) hold for the two examples mentioned in the introduction. Here and hereafter, we denote $w_k = (\mathbf{x}_k, \mathbf{y}_k, \lambda_k)$ and $w = (\mathbf{x}, \mathbf{y}, \lambda)$.

Lemma 3.1 *Let $\{w_k\}$ be generated by Algorithm 1.1. Then, under (A2) we have*

$$\|B^\top \Delta \lambda_{k+1}\| \leq L_g \|\Delta \mathbf{y}_{k+1}\|. \quad (12)$$

Proof According to the optimality condition of \mathbf{y} -subproblem, it holds

$$\nabla g(\mathbf{y}_{k+1}) - B^\top \lambda_{k+\frac{1}{2}} + \beta B^\top (\mathbf{x}_{k+1}^{ad} + B\mathbf{y}_{k+1} - b) = 0. \quad (13)$$

So, we have by the update of λ_{k+1} that

$$B^\top \lambda_{k+1} = B^\top \left[\lambda_{k+\frac{1}{2}} - \beta (\mathbf{x}_{k+1}^{ad} + B\mathbf{y}_{k+1} - b) \right] = \nabla g(\mathbf{y}_{k+1}), \quad (14)$$

which further gives

$$B^\top \lambda_k = \nabla g(\mathbf{y}_k). \quad (15)$$

Subtracting (15) from (14) and taking norm on both sides, we can obtain by (A2) that

$$\|B^\top \Delta \lambda_{k+1}\| \leq L_g \|\Delta \mathbf{y}_{k+1}\|. \quad \blacksquare$$

Note that optimality condition of the following problem is the same as (13):

$$\min \left\{ g(\mathbf{y}) + \frac{\beta}{2} \|B\mathbf{y} - c_y\|^2 \right\},$$

where $c_y = b + \frac{\lambda_{k+\frac{1}{2}}}{\beta} - \mathbf{x}_{k+1}^{ad}$. So, this problem is equivalent to the \mathbf{y} -subproblem in Algorithm 1.1. Under the case that g is linearized or B has full column rank, the above problem could have

closed solution form. In addition, by choosing $G = \sigma I - \beta A^\top A$ with $\sigma \geq \beta \|A^\top A\|$, the quadratic term $\|A\mathbf{x}\|^2$ will be cancelled in the iteration. As a result, the \mathbf{x} -subproblem in Algorithm 1.1 is converted to a proximal mapping as the following

$$\text{Prox}_{f,\sigma}(c_x) := \text{Arg min} \left\{ f(\mathbf{x}) + \frac{\sigma}{2} \|\mathbf{x} - c_x\|^2 \right\}, \quad (16)$$

where $c_x = \mathbf{x}_k^{md} - \frac{\beta A^\top (A\mathbf{x}_k^{md} + B\mathbf{y}_k - b) - A^\top \lambda_k}{\sigma}$. Since f is a proper lower semicontinuous function and bounded from below (in view of Assumptions (A3)), by the proximal behavior in [33] the set $\text{Prox}_{f,\sigma}(c_x)$ is nonempty and compact.

Now, adding the update of $\lambda_{k+\frac{1}{2}}$ to the update of λ_{k+1} , we have

$$\begin{aligned} \frac{1}{\beta} \Delta \lambda_{k+1} &= -\tau(A\mathbf{x}_{k+1} + B\mathbf{y}_k - b) - [\alpha A\mathbf{x}_{k+1} + (1 - \alpha)(b - B\mathbf{y}_k) + B\mathbf{y}_{k+1} - b] \\ &= -(\tau + \alpha)(A\mathbf{x}_{k+1} + B\mathbf{y}_k - b) - B\Delta \mathbf{y}_{k+1}, \end{aligned}$$

which by $\tau + \alpha > 0$ gives the following lemma immediately.

Lemma 3.2 *Assume $\tau + \alpha > 0$, then the sequence $\{w_k\}$ generated by Algorithm 1.1 satisfies*

$$A\mathbf{x}_{k+1} + B\mathbf{y}_k - b = -\frac{1}{\tau + \alpha} \left(\frac{1}{\beta} \Delta \lambda_{k+1} + B\Delta \mathbf{y}_{k+1} \right). \quad (17)$$

Next, we present a fundamental lemma that plays a key role in analyzing convergence and convergence rate bound of Algorithm 1.1.

Lemma 3.3 *Under Assumptions (A1) and (A2), there exist three constants $\zeta_0 \geq 0$ and $\zeta_1, \zeta_2 > 0$ such that*

$$\tilde{L}_\beta(w_k) - \tilde{L}_\beta(w_{k+1}) \geq \zeta_1 \|\Delta \mathbf{x}_{k+1}\|_G^2 + \zeta_2 \|\Delta \mathbf{y}_{k+1}\|^2, \quad (18)$$

where $\tilde{L}_\beta(w_k) := \mathcal{L}_\beta(w_k) + \zeta_0 \|\Delta \mathbf{x}_k\|_G^2$.

Proof The inequality (18) can be proved by the following four steps.

(Step 1) By the update of \mathbf{x} -subproblem together with the way of generating \mathbf{x}_k^{md} , we have

$$\begin{aligned} &\mathcal{L}_\beta(\mathbf{x}_k, \mathbf{y}_k, \lambda_k) - \mathcal{L}_\beta(\mathbf{x}_{k+1}, \mathbf{y}_k, \lambda_k) \\ &\geq \frac{1}{2} \left[\|\mathbf{x}_{k+1} - \mathbf{x}_k^{md}\|_G^2 - \|\mathbf{x}_k - \mathbf{x}_k^{md}\|_G^2 \right] \\ &= \frac{1}{2} \left[\|\mathbf{x}_{k+1} - \mathbf{x}_k\|_G^2 + 2 \left\langle \mathbf{x}_{k+1} - \mathbf{x}_k, G(\mathbf{x}_k - \mathbf{x}_k^{md}) \right\rangle \right] \\ &= \frac{1}{2} \left[\|\Delta \mathbf{x}_{k+1}\|_G^2 - 2\gamma_k \langle \Delta \mathbf{x}_{k+1}, G\Delta \mathbf{x}_k \rangle \right] \\ &\geq \frac{1}{2} \left[\|\Delta \mathbf{x}_{k+1}\|_G^2 - \gamma_k (\|\Delta \mathbf{x}_{k+1}\|_G^2 + \|\Delta \mathbf{x}_k\|_G^2) \right] \\ &= \zeta_0 \left[\|\Delta \mathbf{x}_{k+1}\|_G^2 - \|\Delta \mathbf{x}_k\|_G^2 \right] + \zeta_1 \|\Delta \mathbf{x}_{k+1}\|_G^2, \end{aligned} \quad (19)$$

where

$$\zeta_0 = \frac{\gamma_k}{2} \geq 0, \quad \text{and} \quad \zeta_1 = \frac{1 - 2\gamma_k}{2} > 0. \quad (20)$$

(Step 2) By the update of \mathbf{y} -subproblem we obtain

$$g(\mathbf{y}_k) - \langle \lambda_{k+\frac{1}{2}}, B\mathbf{y}_k \rangle + \frac{\beta}{2} \|\mathbf{x}_{k+1}^{ad} + B\mathbf{y}_k - b\|^2 \geq g(\mathbf{y}_{k+1}) - \langle \lambda_{k+\frac{1}{2}}, B\mathbf{y}_{k+1} \rangle + \frac{\beta}{2} \|\mathbf{x}_{k+1}^{ad} + B\mathbf{y}_{k+1} - b\|^2,$$

which, by Lemma 2.2, is equivalently expressed as

$$g(\mathbf{y}_k) - g(\mathbf{y}_{k+1}) + \langle \lambda_{k+\frac{1}{2}}, B\Delta\mathbf{y}_{k+1} \rangle + \frac{\beta}{2} \|B\Delta\mathbf{y}_{k+1}\|^2 \geq \beta \langle B\Delta\mathbf{y}_{k+1}, \mathbf{x}_{k+1}^{ad} + B\mathbf{y}_{k+1} - b \rangle. \quad (21)$$

Therefore, it can be deduced that

$$\begin{aligned} & \mathcal{L}_\beta(\mathbf{x}_{k+1}, \mathbf{y}_k, \lambda_{k+\frac{1}{2}}) - \mathcal{L}_\beta(\mathbf{x}_{k+1}, \mathbf{y}_{k+1}, \lambda_{k+\frac{1}{2}}) \\ &= g(\mathbf{y}_k) - g(\mathbf{y}_{k+1}) + \langle \lambda_{k+\frac{1}{2}}, B\Delta\mathbf{y}_{k+1} \rangle + \frac{\beta}{2} (\|\mathbf{A}\mathbf{x}_{k+1} + B\mathbf{y}_k - b\|^2 - \|\mathbf{A}\mathbf{x}_{k+1} + B\mathbf{y}_{k+1} - b\|^2) \\ &= g(\mathbf{y}_k) - g(\mathbf{y}_{k+1}) + \langle \lambda_{k+\frac{1}{2}}, B\Delta\mathbf{y}_{k+1} \rangle - \beta \langle B\Delta\mathbf{y}_{k+1}, \mathbf{A}\mathbf{x}_{k+1} + B\mathbf{y}_{k+1} - b \rangle + \frac{\beta}{2} \|B\Delta\mathbf{y}_{k+1}\|^2 \\ &\geq \beta \langle B\Delta\mathbf{y}_{k+1}, \mathbf{x}_{k+1}^{ad} + B\mathbf{y}_{k+1} - b \rangle - \beta \langle B\Delta\mathbf{y}_{k+1}, \mathbf{A}\mathbf{x}_{k+1} + B\mathbf{y}_{k+1} - b \rangle \\ &= \beta(\alpha - 1) \langle B\Delta\mathbf{y}_{k+1}, \mathbf{A}\mathbf{x}_{k+1} + B\mathbf{y}_k - b \rangle \\ &= \frac{1 - \alpha}{\tau + \alpha} \left[\beta \|B\Delta\mathbf{y}_{k+1}\|^2 + \langle \Delta\mathbf{y}_{k+1}, B^\top \Delta\lambda_{k+1} \rangle \right], \end{aligned} \quad (22)$$

where the second equality follows Lemma 2.2, the first inequality uses (21), the third equality uses the update of \mathbf{x}_{k+1}^{ad} and the final equality uses (17).

(Step 3) Note that

$$\begin{aligned} & \mathcal{L}_\beta(\mathbf{x}_{k+1}, \mathbf{y}_k, \lambda_k) - \mathcal{L}_\beta(\mathbf{x}_{k+1}, \mathbf{y}_k, \lambda_{k+\frac{1}{2}}) + \mathcal{L}_\beta(\mathbf{x}_{k+1}, \mathbf{y}_{k+1}, \lambda_{k+\frac{1}{2}}) - \mathcal{L}_\beta(\mathbf{x}_{k+1}, \mathbf{y}_{k+1}, \lambda_{k+1}) \\ &= \langle \lambda_{k+\frac{1}{2}} - \lambda_k, \mathbf{A}\mathbf{x}_{k+1} + B\mathbf{y}_k - b \rangle - \langle \lambda_{k+\frac{1}{2}} - \lambda_{k+1}, \mathbf{A}\mathbf{x}_{k+1} + B\mathbf{y}_{k+1} - b \rangle \\ &= \langle \lambda_{k+\frac{1}{2}} - \lambda_k, \mathbf{A}\mathbf{x}_{k+1} + B\mathbf{y}_k - b \rangle - \langle \lambda_{k+\frac{1}{2}} - \underbrace{\lambda_k + \lambda_k - \lambda_{k+1}}_{\Delta\lambda_{k+1}}, \mathbf{A}\mathbf{x}_{k+1} + B\mathbf{y}_{k+1} - b \rangle \\ &= \langle \lambda_{k+\frac{1}{2}} - \lambda_k, -B\Delta\mathbf{y}_{k+1} \rangle + \langle \Delta\lambda_{k+1}, \mathbf{A}\mathbf{x}_{k+1} + B\mathbf{y}_{k+1} - b \rangle \\ &= \tau\beta \langle \mathbf{A}\mathbf{x}_{k+1} + B\mathbf{y}_k - b, B\Delta\mathbf{y}_{k+1} \rangle + \langle \Delta\lambda_{k+1}, \mathbf{A}\mathbf{x}_{k+1} + B\mathbf{y}_{k+1} + \underbrace{B\mathbf{y}_k - B\mathbf{y}_{k+1}}_{-B\Delta\mathbf{y}_{k+1}} - b \rangle \\ &= \langle \mathbf{A}\mathbf{x}_{k+1} + B\mathbf{y}_k - b, \Delta\lambda_{k+1} + \tau\beta B\Delta\mathbf{y}_{k+1} \rangle + \langle \Delta\lambda_{k+1}, B\Delta\mathbf{y}_{k+1} \rangle \\ &= -\frac{1}{\tau + \alpha} \left\langle \frac{1}{\beta} \Delta\lambda_{k+1} + B\Delta\mathbf{y}_{k+1}, \Delta\lambda_{k+1} + \tau\beta B\Delta\mathbf{y}_{k+1} \right\rangle + \langle \Delta\lambda_{k+1}, B\Delta\mathbf{y}_{k+1} \rangle \\ &= -\frac{\tau\beta}{\tau + \alpha} \|B\Delta\mathbf{y}_{k+1}\|^2 - \frac{1}{(\tau + \alpha)\beta} \|\Delta\lambda_{k+1}\|^2 - \frac{1 - \alpha}{\tau + \alpha} \langle \Delta\lambda_{k+1}, B\Delta\mathbf{y}_{k+1} \rangle. \end{aligned} \quad (23)$$

(Step 4) Summing the above inequalities (19), (22) and the equality (23), we get

$$\begin{aligned} & \mathcal{L}_\beta(\mathbf{x}_k, \mathbf{y}_k, \lambda_k) - \mathcal{L}_\beta(\mathbf{x}_{k+1}, \mathbf{y}_{k+1}, \lambda_{k+1}) \\ &\geq \frac{\gamma_k}{2} [\|\Delta\mathbf{x}_{k+1}\|_G^2 - \|\Delta\mathbf{x}_k\|_G^2] + \frac{1 - 2\gamma_k}{2} \|\Delta\mathbf{x}_{k+1}\|_G^2 + R_\Delta, \end{aligned}$$

where

$$\begin{aligned}
R_\Delta &= \left(\frac{1}{\tau + \alpha} - 1\right) \beta \|B\Delta \mathbf{y}_{k+1}\|^2 - \frac{1}{(\tau + \alpha)\beta} \|\Delta \lambda_{k+1}\|^2 \\
&\geq \left(\frac{1}{\tau + \alpha} - 1\right) \beta \|B\Delta \mathbf{y}_{k+1}\|^2 - \frac{1}{(\tau + \alpha)\beta\sigma_B} \left\|B^\top \Delta \lambda_{k+1}\right\|^2 \\
&\geq \left(\frac{1}{\tau + \alpha} - 1\right) \beta \|B\Delta \mathbf{y}_{k+1}\|^2 - \frac{L_g^2}{(\tau + \alpha)\beta\sigma_B} \|\Delta \mathbf{y}_{k+1}\|^2 \\
&\geq \left(\frac{1}{\tau + \alpha} - 1\right) \beta\sigma_B \|\Delta \mathbf{y}_{k+1}\|^2 - \frac{L_g^2}{(\tau + \alpha)\beta\sigma_B} \|\Delta \mathbf{y}_{k+1}\|^2 \\
&= \zeta_2 \|\Delta \mathbf{y}_{k+1}\|^2
\end{aligned}$$

with

$$\zeta_2 = \frac{(1 - \tau - \alpha)\beta^2\sigma_B^2 - L_g^2}{(\tau + \alpha)\beta\sigma_B} > 0. \quad [\text{due to (A2)}]$$

Actually, in the first inequality of R_Δ , we use the fact that $\Delta \lambda_{k+1} \in \text{Im}(B)$ because of Assumption (A1). So, the whole proof is completed by the notation $\tilde{L}_\beta(w_k)$. \blacksquare

Theorem 3.1 *Let $\{w_k\}$ be generated by Algorithm 1.1. Then, under (A1)-(A3) we have*

- *The sequence $\{L_\beta(w_k)\}$ is convergent;*
- *The residuals $\|\Delta \mathbf{x}_{k+1}\|_G$, $\|\Delta \mathbf{y}_{k+1}\|$ and $\|\Delta \lambda_{k+1}\|$ converge to zero as k goes to infinity.*

Proof To demonstrate convergence of $\{L_\beta(w_k)\}$, we need to make ensure that the sequence $\{w_k\}$ is bounded at first. By Assumption (A2), it holds

$$L_g < \sqrt{1 - \tau - \alpha}\beta\sigma_B < \beta\sigma_B.$$

Combining the above inequality and Lemma 3.3, we achieve

$$\begin{aligned}
L_\beta(\mathbf{x}_0, \mathbf{y}_0, \lambda_0) &= L_\beta(\mathbf{x}_0, \mathbf{y}_0, \lambda_0) + \zeta_1 \|\Delta \mathbf{x}_0\|_G^2 \\
&\geq L_\beta(\mathbf{x}_{k+1}, \mathbf{y}_{k+1}, \lambda_{k+1}) + \zeta_1 \|\Delta \mathbf{x}_{k+1}\|_G^2 \geq L_\beta(\mathbf{x}_{k+1}, \mathbf{y}_{k+1}, \lambda_{k+1}) \\
&= f(\mathbf{x}_{k+1}) + g(\mathbf{y}_{k+1}) - \frac{1}{2\beta} \|\lambda_{k+1}\|^2 + \frac{\beta}{2} \left\|A\mathbf{x}_{k+1} + B\mathbf{y}_{k+1} - b - \frac{\lambda_{k+1}}{\beta}\right\|^2 \\
&\geq f(\mathbf{x}_{k+1}) + g(\mathbf{y}_{k+1}) - \frac{1}{2\beta\sigma_B} \|B^\top \lambda_{k+1}\|^2 + \frac{\beta}{2} \left\|A\mathbf{x}_{k+1} + B\mathbf{y}_{k+1} - b - \frac{\lambda_{k+1}}{\beta}\right\|^2 \\
&= \left(f(\mathbf{x}_{k+1}) + g(\mathbf{y}_{k+1}) - \frac{1}{2L_g} \|\nabla g(\mathbf{y}_{k+1})\|^2\right) + \left(\frac{1}{2L_g} - \frac{1}{2\beta\sigma_B}\right) \|B^\top \lambda_{k+1}\|^2 \\
&\quad + \frac{\beta}{2} \left\|A\mathbf{x}_{k+1} + B\mathbf{y}_{k+1} - b - \frac{\lambda_{k+1}}{\beta}\right\|^2 \\
&\geq \underline{L} + \left(\frac{1}{2L_g} - \frac{1}{2\beta\sigma_B}\right) \|B^\top \lambda_{k+1}\|^2 + \frac{\beta}{2} \left\|A\mathbf{x}_{k+1} + B\mathbf{y}_{k+1} - b - \frac{\lambda_{k+1}}{\beta}\right\|^2, \tag{24}
\end{aligned}$$

which implies that the sequences $\{\lambda_k\}$, $\{\frac{\beta}{2} \|A\mathbf{x}_{k+1} + B\mathbf{y}_{k+1} - b - \lambda_{k+1}/\beta\|^2\}$ are bounded, and furthermore both $\{\mathbf{x}_k\}$ and $\{\mathbf{y}_k\}$ are bounded. So, the sequence $\{w_k\}$ is bounded.

Since $\{w_k\}$ is bounded, $\{\tilde{L}_\beta(w_k)\}$ is also bounded from below and there exists at least one limit point. Without loss of generality, let w_* be the limit point of $\{w_k\}$ whose subsequence is $\{w_{k_j}\}$. Then, the lower semicontinuity of $\{\tilde{L}_\beta(w)\}$ indicates

$$\tilde{L}_\beta(w_*) \leq \liminf_{j \rightarrow +\infty} \tilde{L}_\beta(w_{k_j}).$$

That is, $\{\tilde{L}_\beta(w_{k_j})\}$ is bounded from below, which further implies convergence of $\{\tilde{L}_\beta(w_k)\}$ based on Lemma 3.3.

Now, summing the inequality (18) over $k = 0, 1, \dots, \infty$, we have by the convergence of $\{\tilde{L}_\beta(w_k)\}$ that

$$\zeta_1 \sum_{k=0}^{\infty} \|\Delta \mathbf{x}_{k+1}\|_G^2 + \zeta_2 \sum_{k=0}^{\infty} \|\Delta \mathbf{y}_{k+1}\|^2 \leq \mathcal{L}_\beta(w_0) - \tilde{\mathcal{L}}_\beta(w_{k+1}) < \infty,$$

which suggests $\|\Delta \mathbf{x}_{k+1}\|_G \rightarrow 0$ and $\|\Delta \mathbf{y}_{k+1}\| \rightarrow 0$. So, using Lemma 2.1 and Lemma 3.1 the following holds clearly

$$\|\Delta \lambda_{k+1}\| \leq \frac{1}{\sqrt{\sigma_B}} \|B^\top \Delta \lambda_{k+1}\| \leq \frac{Lg}{\sqrt{\sigma_B}} \|\Delta \mathbf{y}_{k+1}\| \rightarrow 0. \quad (25)$$

This completes the proof. \blacksquare

Theorem 3.1 illustrates that the augmented Lagrange function of the problem (1) is convergent, and the primal and dual residuals converge to zero. In what follows, we would present a key theorem about pointwise iteration-complexity of the proposed algorithm w.r.t. the primal-dual residuals. Actually, the following first assertion implies that any accumulation point of $\{w_k\}$ is a stationary point of $\{L_\beta(w_k)\}$ compared to Definition 2.2.

Theorem 3.2 *Let $\{w_k\}$ be generated by Algorithm 1.1. Then, under Assumptions (A1)-(A3)*

- *It holds*

$$\lim_{k \rightarrow \infty} d(0, \partial L_\beta(w^{k+1})) = 0. \quad (26)$$

- *The sequence $\{f(\mathbf{x}_{k+1}) + g(\mathbf{y}_{k+1})\}$ is convergent.*
- *Let $C_0 := \mathcal{L}_\beta(w_0) - \underline{L}$. Then, for any integer $k \geq 1$, there exists $j \leq k$ and $\zeta_i > 0$ ($i = 1, 2, 3$) such that*

$$\|\Delta \mathbf{x}_j\|_G^2 \leq \frac{C_0}{\zeta_1(k+1)}, \quad \|\Delta \mathbf{y}_j\|^2 \leq \frac{C_0}{\zeta_2(k+1)}, \quad \|\Delta \lambda_j\|^2 \leq \frac{C_0}{\zeta_3(k+1)}. \quad (27)$$

Proof Using (17) again, we have

$$\mathbf{A}\mathbf{x}_{k+1} + \mathbf{B}\mathbf{y}_{k+1} - b = -\frac{1}{\tau + \alpha} \left(\frac{1}{\beta} \Delta \lambda_{k+1} + \mathbf{B} \Delta \mathbf{y}_{k+1} \right) + \mathbf{B} \Delta \mathbf{y}_{k+1},$$

which by the third result of Theorem 3.1 suggests

$$\lim_{k \rightarrow \infty} \mathbf{A}\mathbf{x}_{k+1} + \mathbf{B}\mathbf{y}_{k+1} - b = 0. \quad (28)$$

Therefore,

$$\lim_{k \rightarrow \infty} \nabla_{\lambda} L_{\beta}(w_{k+1}) = \lim_{k \rightarrow \infty} -(\mathbf{A}\mathbf{x}_{k+1} + \mathbf{B}\mathbf{y}_{k+1} - b) = 0. \quad (29)$$

By the first-order optimality condition of \mathbf{y} -subproblem, it holds

$$\begin{aligned} 0 &= \nabla g(\mathbf{y}_{k+1}) - B^{\top} \lambda_{k+\frac{1}{2}} + \beta B^{\top} (\mathbf{x}_{k+1}^{ad} + \mathbf{B}\mathbf{y}_{k+1} - b) \\ &= \nabla g(\mathbf{y}_{k+1}) - B^{\top} \lambda_{k+1} + \beta B^{\top} (\mathbf{A}\mathbf{x}_{k+1} + \mathbf{B}\mathbf{y}_{k+1} - b) \\ &\quad + B^{\top} (\lambda_{k+1} - \lambda_{k+\frac{1}{2}}) + \beta B^{\top} (\mathbf{x}_{k+1}^{ad} - \mathbf{A}\mathbf{x}_{k+1}) \\ &= \nabla g(\mathbf{y}_{k+1}) - B^{\top} \lambda_{k+1} + \beta B^{\top} (\mathbf{A}\mathbf{x}_{k+1} + \mathbf{B}\mathbf{y}_{k+1} - b) \\ &\quad - \beta B^{\top} (\mathbf{A}\mathbf{x}_{k+1} + \mathbf{B}\mathbf{y}_{k+1} - b), \end{aligned}$$

which gives

$$\lim_{k \rightarrow \infty} \nabla_{\mathbf{y}} L_{\beta}(w_{k+1}) = \lim_{k \rightarrow \infty} \beta B^{\top} (\mathbf{A}\mathbf{x}_{k+1} + \mathbf{B}\mathbf{y}_{k+1} - b) = 0. \quad (30)$$

Analogously, by the update of \mathbf{x} -subproblem, there exists $d_{k+1} \in \partial f(\mathbf{x}_{k+1})$ such that

$$\begin{aligned} 0 &= d_{k+1} - A^{\top} \lambda_{k+1} + \beta A^{\top} (\mathbf{A}\mathbf{x}_{k+1} + \mathbf{B}\mathbf{y}_k - b) + G(\mathbf{x}_{k+1} - \mathbf{x}_k^{md}) \\ &= d_{k+1} - A^{\top} \lambda_{k+1} + \beta A^{\top} (\mathbf{A}\mathbf{x}_{k+1} + \mathbf{B}\mathbf{y}_{k+1} - b) \\ &\quad + \beta A^{\top} B(\mathbf{y}_k - \mathbf{y}_{k+1}) + G(\mathbf{x}_{k+1} - \mathbf{x}_k - \gamma_k \Delta \mathbf{x}_k) \\ &= d_{k+1} - A^{\top} \lambda_{k+1} + \beta A^{\top} (\mathbf{A}\mathbf{x}_{k+1} + \mathbf{B}\mathbf{y}_{k+1} - b) \\ &\quad - \beta A^{\top} B \Delta \mathbf{y}_{k+1} - G(\gamma_k \Delta \mathbf{x}_k - \Delta \mathbf{x}_{k+1}). \end{aligned}$$

By defining

$$\bar{d}_{k+1} := d_{k+1} - A^{\top} \lambda_{k+1} + \beta A^{\top} (\mathbf{A}\mathbf{x}_{k+1} + \mathbf{B}\mathbf{y}_{k+1} - b),$$

we have $\bar{d}_{k+1} \in \partial_{\mathbf{x}} L_{\beta}(w_{k+1})$ and furthermore

$$\lim_{k \rightarrow \infty} \bar{d}_{k+1} = \lim_{k \rightarrow \infty} \left[\beta A^{\top} B \Delta \mathbf{y}_{k+1} + G(\gamma_k \Delta \mathbf{x}_k - \Delta \mathbf{x}_{k+1}) \right] = 0. \quad (31)$$

Thus, it follows from (29), (30) and (31) that (26) holds.

For the second assertion, it holds by (28) that

$$f(\mathbf{x}_{k+1}) + g(\mathbf{y}_{k+1}) = L_{\beta}(w_{k+1}) + \langle \lambda, \mathbf{A}\mathbf{x}_{k+1} + \mathbf{B}\mathbf{y}_{k+1} - b \rangle + \frac{\beta}{2} \|\mathbf{A}\mathbf{x}_{k+1} + \mathbf{B}\mathbf{y}_{k+1} - b\|^2 \rightarrow L_{\beta}(w_{k+1}).$$

So, the sequence $\{f(\mathbf{x}_{k+1}) + g(\mathbf{y}_{k+1})\}$ is convergent by the first conclusion of Theorem 3.1.

We finally prove the pointwise iteration complexity in (27). Using (24) again, we have

$$-\tilde{L}_{\beta}(w_{k+1}) \leq -\underline{L} - \left(\frac{1}{2L_g} - \frac{1}{2\beta\sigma_B} \right) \|B^{\top} \lambda_{k+1}\|^2 - \frac{\beta}{2} \left\| \mathbf{A}\mathbf{x}_{k+1} + \mathbf{B}\mathbf{y}_{k+1} - b - \frac{\lambda_{k+1}}{\beta} \right\|^2 \leq -\underline{L}.$$

So, for any $k \geq 0$, it follows from Lemma 3.3 that

$$\sum_{j=0}^k (\zeta_1 \|\Delta \mathbf{x}_j\|_G^2 + \zeta_2 \|\Delta \mathbf{y}_j\|^2) \leq \mathcal{L}_{\beta}(w_0) + \zeta_0 \|\Delta \mathbf{x}_0\|_G^2 - \underline{L} = C_0,$$

which shows

$$\|\Delta \mathbf{x}_j\|_G^2 \leq \frac{C_0}{\zeta_1(k+1)} \quad \text{and} \quad \|\Delta \mathbf{y}_j\|^2 \leq \frac{C_0}{\zeta_2(k+1)}.$$

The final convergence rate bound in (27) can be also verified by (25) with $\zeta_3 = L_g^2/\zeta_2$. ■

In order to reduce error bounds of the primal-dual residuals, the following remark provides an adaptive way to update the parameter γ_k related to ζ_1 by making use of the so-called Nesterov's acceleration (proposed originally in [31]), and it also suggests how to choose reasonable values of the parameters τ and α .

Remark 3.1 *By the above convergence analysis, if $G \succ 0$, then convergence of Algorithm 1.1 can be guaranteed by $\gamma_k \in [0, 1/2)$. In such case we can update γ_k adaptively by the following*

$$\gamma_k = \frac{\theta_{k-1} - 1}{2\theta_k}, \quad \text{where } \theta_k = \frac{1 + \sqrt{1 + 4\theta_{k-1}^2}}{2} \quad \text{with } \theta_{-1} := 1. \quad (32)$$

Note that $\zeta_2 = -\beta\sigma_B + \frac{1}{\tau+\alpha}[\beta\sigma_B - \frac{L_g^2}{\beta\sigma_B}]$ is inversely proportional to $(\tau + \alpha)$ since $L_g < \beta\sigma_B$. This together with the connection $\zeta_3 = L_g^2/\zeta_2$ imply that we could choose $(\tau + \alpha) \rightarrow 1$ to get smaller error bound of $\|\Delta \lambda_j\|^2$ in (27). In the next section, related numerical experiments will show how to determine reasonable values of τ and α in detail.

4 Numerical Experiments

In this section, we apply the proposed algorithm to solve a class of practical examples from signal processing to investigate its numerical performance. All experiments are performed by using Windows 10 system and MATLAB R2018a (64-bit) with an Intel Core i7-8700K CPU (3.70 GHz) and 16GB memory.

Applying Algorithm 1.1 to solve (3), we have by (16) that

$$\mathbf{x}_{k+1} = \text{Prox}_{\|\mathbf{x}\|_{1/2, \sigma/\mu}^{1/2}} \left(\mathbf{x}_k^{md} - \frac{\beta A^\top (A \mathbf{x}_k^{md} - \mathbf{y}_k - b) - A^\top \lambda_k}{\sigma} \right),$$

which is the half shrinkage operator [37] defined as $\text{Prox}_{\|\mathbf{x}\|_{1/2, \nu}^{1/2}}(\mathbf{x}) = (l_\nu(\mathbf{x}_1), l_\nu(\mathbf{x}_2), \dots, l_\nu(\mathbf{x}_m))^\top$

where

$$l_\nu(\mathbf{x}_i) = \begin{cases} \frac{2\mathbf{x}_i}{3} [1 + \cos \frac{2}{3}(\pi - \phi(\mathbf{x}_i))] , & \text{if } |\mathbf{x}_i| > \frac{3\sqrt[3]{2}}{4}\nu^{2/3}, \\ 0, & \text{otherwise,} \end{cases}$$

and $\phi(\mathbf{x}_i) = \arccos(\frac{\nu}{8}(\frac{|\mathbf{x}_i|}{3})^{-3/2})$. Besides, it is easy to obtain $\mathbf{y}_{k+1} = (c + \beta \mathbf{x}_{k+1}^{ad} - \lambda_{k+\frac{1}{2}})/(1 + \beta)$.

With the purpose of fast convergence and making performance of Algorithm 1.1 less independent on an initial guess of the penalty parameter β , as suggested by He et al.[22] we would adopt the following technique to update it adaptively:

$$\beta_{k+1} = \begin{cases} \eta^{\text{incr}} \beta_k & \text{if } \|r_k\|_2 > \nu \|s_k\|_2, \\ \beta_k / \eta^{\text{decr}} & \text{if } \|s_k\|_2 > \nu \|r_k\|_2, \\ \beta_k & \text{otherwise,} \end{cases} \quad (33)$$

where ν, η^{incr} and η^{decr} are three positive parameters with suggested values larger than 1, for instance, $\nu = 10, \eta^{\text{incr}} = \eta^{\text{decr}} = 2$. For Algorithm 1.1 to solve (1) we have

$$\|r_k\| = \|A\mathbf{x}_{k+1} + B\mathbf{y}_{k+1} - b\| \quad (34)$$

and

$$\|s_k\| = \left\| A^\top \Delta \lambda_{k+1} + \beta A^\top (A\mathbf{x}_{k+1} + B\mathbf{y}_k - b) + G(\Delta \mathbf{x}_{k+1} - \gamma_k \Delta \mathbf{x}_k) \right\|,$$

which represent the equality constrained error and the optimality error, respectively. Here, it is easy to check that $0 \in \partial f(\mathbf{x}_{k+1}) - A^\top \lambda_{k+1} + s_k$. In order to satisfy Assumption (A2), we need to update $\beta = \min \left\{ \beta_{k+1}, \frac{1.01L_g}{\sqrt{1-\tau-\alpha\sigma_B}} \right\}$ at each iteration. As for the problem (3), we have $L_g = 1$ and $\sigma_B = 1$. If not specified, the initial penalty parameter β_0 is chosen as 0.04, the starting points $(\mathbf{x}_0, \mathbf{y}_0)$ and λ_0 are respectively set as zero and ones vector with proper dimensions, and the matrix $G = \sigma I - \beta A^\top A$ with $\sigma = 1.01\beta \|A^\top A\|$. The parameter γ_k is updated adaptively according to (32). Throughout we use the following stopping criterion as mentioned in [27] to terminate Algorithm 1.1:

$$\text{IRE}(k) := \frac{\max \{ \|\mathbf{x}_k - \mathbf{x}_{k-1}\|, \|\mathbf{y}_k - \mathbf{y}_{k-1}\|, \|\lambda_k - \lambda_{k-1}\| \}}{\max \{ \|\mathbf{x}_{k-1}\|, \|\mathbf{y}_{k-1}\|, \|\lambda_{k-1}\|, 1 \}} < \epsilon, \quad (35)$$

where ϵ is a given tolerance error. Note that this stopping criterion corresponds to the pointwise iteration complexity shown in (27), so such stopping criterion is well defined.

As the first experiment, we consider the reformulated sparse signal recovery problem (3) with an original signal $x \in \mathcal{R}^{3072}$ containing 160 spikes with amplitude ± 1 . The measurement matrix $A \in \mathcal{R}^{1024 \times 3072}$ is drawn firstly from the standard norm distribution $\mathcal{N}(0, 1)$ and then each of its column is normalized. Specifically, we use the following MATLAB codes to generate the original signal x_{orig} , the data A, c and μ :

```

randn('state', 0); rand('state', 0);
l = 1024; m = 3072;
T = 160; % number of spikes
x_orig = zeros(m,1); q = randperm(m);
x_orig(q(1:T)) = sign(randn(T,1)); % original signal
A = randn(l,m);
A = A*spdiags(1./sqrt(sum(A.^2))', 0,m,m); % normalize columns
sig = 0.01; % noise standard deviation
c = A*x_orig + sig*randn(l,1); % noisy observations
mu_max = norm(A'*c, 'inf');
mu = 0.1*mu_max; % regularization parameter

```

Under tolerance $\epsilon = 10^{-15}$, we test the effect of parameters (τ, α) restricted in (8) on the numerical performance of Algorithm 1.1 (In fact, we choose parameter values around $(\tau, \alpha) = (0.3, 0.32)$, because we find it performs slightly better than some pairs after running a lot of values restricted in (8) by the aid of two level for loops in MATLAB). We also randomly choose four pairs of (τ, α) to carry out related experiments.

Table 1 reports some computational results of several quality measurements, including “IT”, “CPU”, “IRE”, “EQU” which denote respectively the iteration number, the CPU time in seconds, the final relative iterative error $\text{IRE}(k)$ defined in (35) and the final feasibility error $\|r_k\|$

defined in (34). We use l_2 -error (defined as $\frac{\|x_k - x_{\text{orig}}\|}{\|x_{\text{orig}}\|}$) to represent the relative error to measure recovery quality of a signal. As shown in Table 1, setting $(\tau, \alpha) = (0.65, 0.32)$ would be a reasonable choice for Algorithm 1.1 to solve the problem (3), because in such a choice the iteration number and the CPU time are relatively smaller while reported results in each of the last three columns are nearly the same when the stopping criterion is satisfied. Hence, in the following experiments, we use Algorithm 1.1 with default parameters $(\tau, \alpha) = (0.65, 0.32)$.

(τ, α)	IT	CPU	IRE	EQU	l_2 -error
(0.3, 0.10)	472	22.67	9.47e-16	6.10e-14	6.79e-2
(0.3, 0.15)	443	21.12	9.23e-16	5.18e-14	6.79e-2
(0.3, 0.20)	500	24.11	9.77e-16	4.82e-14	6.79e-2
(0.3, 0.25)	379	18.23	9.79e-16	6.33e-14	6.79e-2
(0.3, 0.30)	350	16.79	9.78e-16	5.88e-14	6.79e-2
(0.3, 0.32)	344	16.66	9.45e-16	6.33e-14	6.79e-2
(0.3, 0.35)	544	26.17	9.79e-16	3.28e-14	6.79e-2
(0.3, 0.40)	510	24.30	8.87e-16	5.11e-14	6.79e-2
(0.3, 0.45)	485	23.28	9.93e-16	3.34e-14	6.79e-2
(0.3, 0.50)	458	21.95	9.34e-16	3.24e-14	6.79e-2
(0.3, 0.55)	433	20.86	9.58e-16	3.29e-14	6.79e-2
(0.3, 0.60)	413	20.30	9.45e-16	3.24e-14	6.79e-2
(0.3, 0.65)	396	19.24	9.33e-16	3.26e-14	6.79e-2
(0.3, 0.68)	387	18.66	8.91e-16	3.37e-14	6.79e-2
(-0.3, 0.32)	—	—	—	—	—
(-0.2, 0.32)	—	—	—	—	—
(-0.1, 0.32)	695	33.51	9.95e-16	5.41e-14	6.79e-2
(0, 0.32)	498	23.77	9.99e-16	8.38e-14	6.79e-2
(0.1, 0.32)	697	33.49	9.33e-16	5.68e-14	6.79e-2
(0.2, 0.32)	391	18.93	9.54e-16	7.85e-14	6.79e-2
(0.3, 0.32)	344	16.62	9.45e-16	6.33e-14	6.79e-2
(0.4, 0.32)	502	24.16	8.85e-16	4.01e-14	6.79e-2
(0.5, 0.32)	451	21.61	9.49e-16	2.97e-14	6.79e-2
(0.6, 0.32)	404	19.52	9.64e-16	3.74e-14	6.79e-2
(0.62, 0.32)	397	19.20	9.71e-16	4.34e-14	6.79e-2
(0.65, 0.32)	279	13.52	9.98e-16	3.19e-14	6.79e-2
(0.67, 0.32)	318	15.59	8.43e-16	3.11e-14	6.79e-2
(0.90, 0.05)	396	19.06	9.26e-16	3.27e-14	6.79e-2
(0.80, 0.15)	396	19.02	9.33e-16	3.28e-14	6.79e-2
(0.01, 0.90)	411	19.78	9.63e-16	3.10e-14	6.79e-2
(0.05, 0.70)	485	24.62	9.62e-16	3.10e-14	6.79e-2

Table 1: Results¹ of Algorithm 1.1 with different (τ, α) for solving problem (3).

Next, we use the aforementioned codes to investigate the effect of regularization parameter μ on Algorithm 1.1 for solving the problem (3) with a large data $A \in \mathcal{R}^{2048 \times 5000}$ and the same spikes, but the tolerance is set as $\epsilon = 10^{-12}$. Fig. 1 depicts convergence behaviors of the equality constraint error $\|r_k\|$, the iterative error IRE(k) and the recovery signal quality $e_k := \log_{10} \frac{\|x_k - x_{\text{orig}}\|}{\|x_{\text{orig}}\|}$ along the iteration process after applying Algorithm 1.1 with $\mu = 0.1\mu_{\text{max}}, 0.05\mu_{\text{max}}, 0.01\mu_{\text{max}}$, respectively. Fig. 2 also presents the results to visualize

¹“—” means that the stopping criterion is not satisfied after 800 iterations, and the bold number in that row indicate the best results obtained by changing (τ, α) belong to (0, 1).

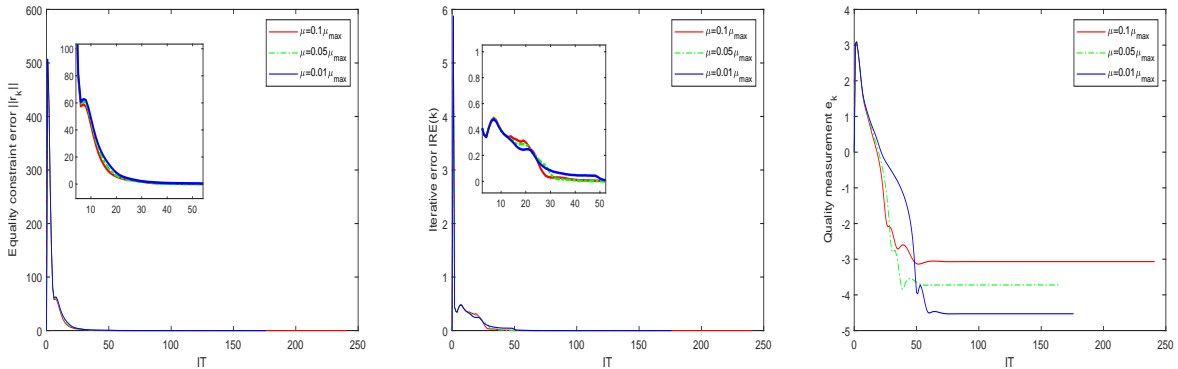


Fig. 1: Convergence tendency of the equality constraint error $\|r_k\|$ (left), the iterative error $\text{IRE}(k)$ (middle) and the recovery signal quality e_k (right) by Algorithm 1.1 for solving the problem (3) with $(l, m) = (2048, 5000)$ but with different regularization factors.

the recovery quality of the signal versus the original signal, where the upper-left plot shows the minimum energy reconstruction signal $A^\dagger c$ (which is the point satisfying $A^\top A \mathbf{x} = A^\top c$) versus the original signal. An outstanding observation from Fig. 1 is that the smaller the value of μ is, the smaller the iteration number is (and the better the recovery quality of the signal is). After identifying the nonzero positions in the reconstructed signal, it always has the correct number of spikes for the case with $\mu = 0.01\mu_{\max}$ and is closer to the original noiseless signal.

(l, m)	$l_{1/2}$ regularizer				l_1 regularizer			
	IT	CPU	EQU	l_2 error	IT	CPU	EQU	l_2 error
(1024, 3000)	358	16.37	4.60e-14	1.20e-2	501	22.60	1.93e-14	3.70e-2
(1024, 4000)	367	25.93	4.18e-14	1.28e-2	507	35.611	4.78e-14	4.26e-2
(2048, 5000)	215	30.65	2.84e-14	1.08e-2	250	36.01	3.27e-14	2.66e-2
(2048, 6000)	222	41.37	3.98e-14	1.20e-2	266	49.59	3.47e-14	3.07e-2
(3000, 7000)	201	58.08	2.71e-14	1.17e-2	231	66.91	2.93e-14	2.60e-2
(3000, 8000)	205	71.36	3.77e-14	1.10e-2	230	79.55	3.62e-14	2.58e-2
(4000, 9000)	199	97.52	3.29e-14	1.11e-2	231	112.94	2.87e-14	2.69e-2
(4000, 10000)	202	118.62	2.37e-14	1.03e-2	231	135.72	2.91e-14	2.51e-2

Table 2: Results of Algorithm 1.1 for (3) with different regularization terms and dimensions.

In the following, we use the proposed algorithm to solve two different cases of the sparse signal recovery problem to investigate which regularization term performs better: **Case (i)** the convex problem (2)² with l_1 regularization term; **Case (ii)** the nonconvex problem (3) with $l_{1/2}$ regularization term. Table 2 reports some numerical results, where the problem dimension comes from 3000 to 10000 w.r.t the dimension of the signal, the regularization parameter is fixed as $\mu = 0.01\mu_{\max}$ and Algorithm 1.1 is terminated under tolerance $\epsilon = 10^{-15}$ with maximal iteration numbers 1000. Fig. 3 depicts comparison results between the original signal and the reconstructed signal for the signal dimension $m = 10000$. First of all, it can be seen from

²Note that this is also a special case of (1) with $f(\mathbf{x}) = \mu\|\mathbf{x}\|_1, g(\mathbf{y}) = \frac{1}{2}\|\mathbf{y} - c\|^2, B = -I$ and $b = 0$.

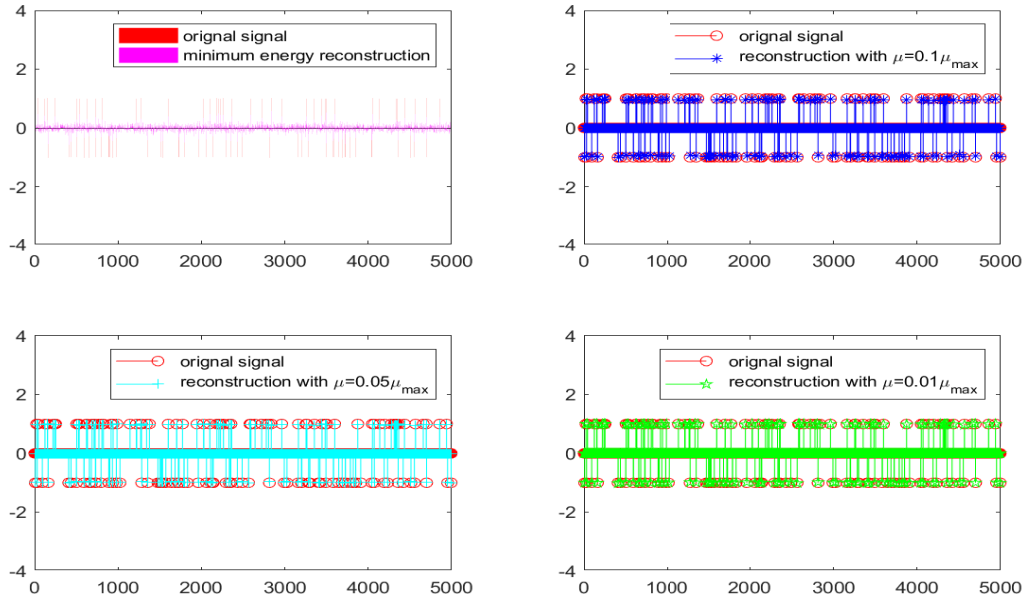


Fig. 2: Comparison between the original signal and reconstructed signal by Algorithm 1.1 for solving problem (3) with $(l, m) = (2048, 5000)$ but with different regularization factor μ .

results in Table 2 that the proposed algorithm is feasible for solving both the nonconvex and convex sparse signal recovery problem, especially for the large-scale problem. Besides, an obvious observation from Table 2 is that using $l_{1/2}$ regularizer is significantly better than l_1 regularizer to recover a signal, which could be checked from reported results of the iteration number, the CPU time and the recovery quality (i.e., l_2 error).

Finally, we would apply the proposed algorithm to solve the direction-of-arrival (DOA) estimation problem [29] with a single snapshot. Here we consider a uniformly linear array of $M = 100$ sensors with half-wavelength elements spacing. Let $\boldsymbol{\theta} = [\theta_1, \dots, \theta_L]^T$ denote the L angles of interest in $[-\frac{\pi}{2}, \frac{\pi}{2}]$. Denote $\mathbf{x} = [x_1, \dots, x_L]^T$ as the amplitudes of the potential signals from the L incoming angles. Thus, the received signal at the sensor array is given by: $\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{n}$, where $\mathbf{y} = [y_1, \dots, y_M]^T$, $\mathbf{n} = [n_1, \dots, n_M]^T$, the steering matrix $\mathbf{A} = [\mathbf{a}(\theta_1), \dots, \mathbf{a}(\theta_L)]$ and $\mathbf{a}(\theta_i) = [1, \exp(-j\pi \sin(\theta_i)), \dots, \exp(-j\pi(M-1) \sin(\theta_i))]^T$.

We consider the narrowband scenario with $K = 2$ uncorrelated far-field source signals with normalized DOA parameters $-\frac{\pi}{6}$ and $\frac{\pi}{4}$. To run the proposed method, we divide the potential angle region $[-\frac{\pi}{2}, \frac{\pi}{2}]$ into $L = 180$ uniformly discrete grid points, i.e., $\boldsymbol{\theta} = \frac{\pi}{180}[-90, -89, \dots, 89, 90]^T$. When the signal-to-noise-ratio (SNR) varies from -5dB to 20dB , i.e., $\{-5, 0, 5, 10, 15, 20\}\text{dB}$, we implement the proposed method and the well-known CVX³ toolbox for 100 Monte Carlo runs, and compute their root mean square errors and running time, as plotted in Fig. 4. For visible comparison, we plot the result from one Monte Carlo in the case of 5dB , as shown in Fig. 5. From Figs. 4-5, we can see that:

- The accuracy of the two methods increases with the increase of SNR;

³Available at: <http://cvxr.com/cvx/>.

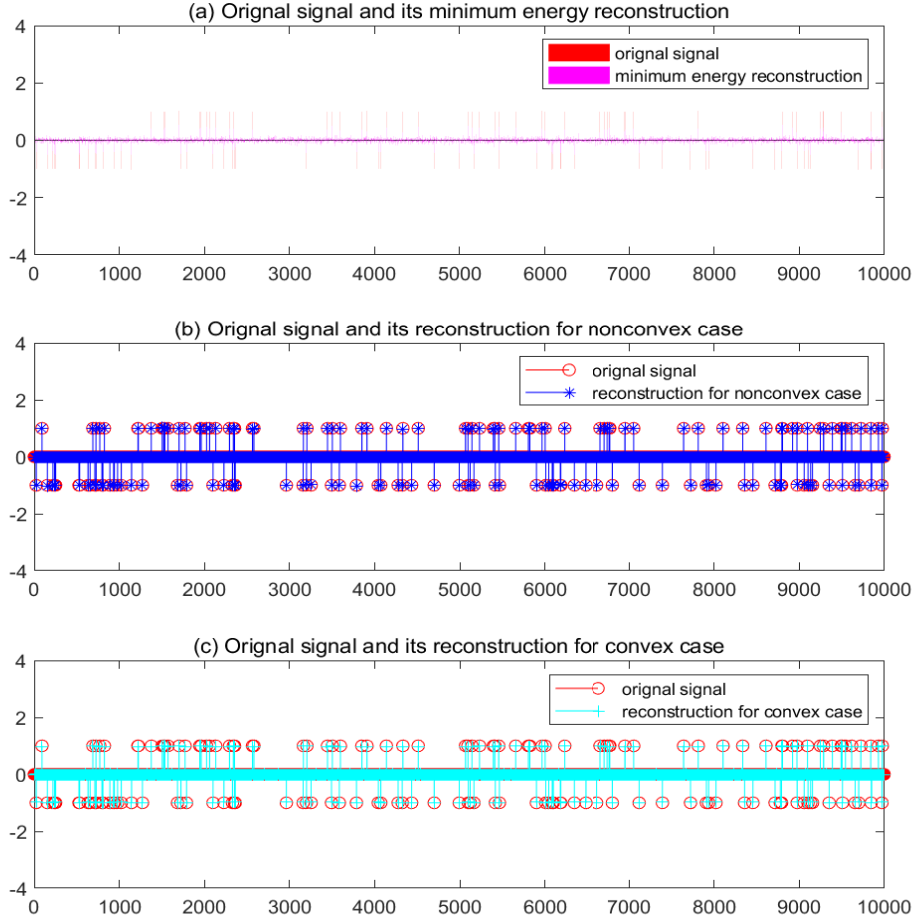


Fig. 3: Original signal and reconstructed signal by Algorithm 1.1 for solving the sparse signal recovery problem with $(l, m) = (4000, 10000)$ but with different regularization terms.

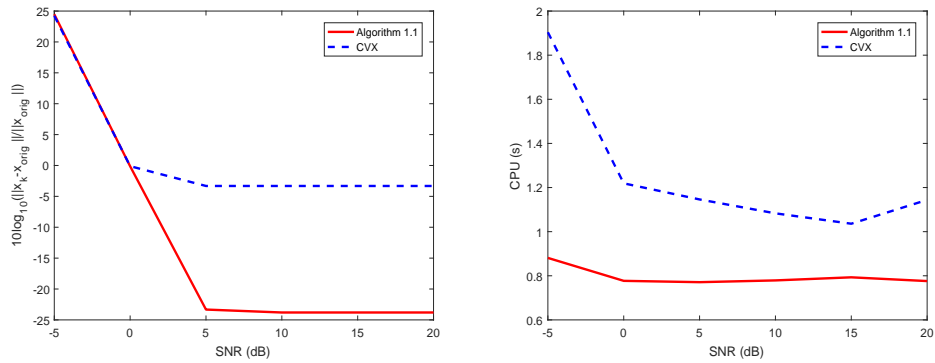


Fig. 4: The left and right are the errors and average run time versus different SNRs, respectively.

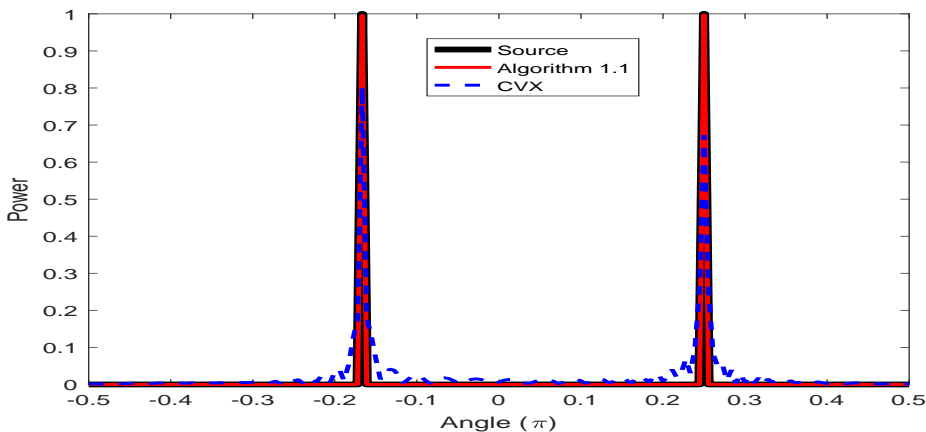


Fig. 5: DOAs at SNR= 5dB.

- The implementation of the proposed method is faster than that of the CVX method.
- In terms of DOA resolution and the estimation accuracy of the incoming signal power, the proposed method is better than that of CVX.

5 Conclusion remarks

In this paper, we construct a symmetric alternating direction method of multipliers for solving a family of possibly nonconvex nonsmooth optimization problems. Two different acceleration techniques are designed for fast convergence. Under proper assumptions, convergence of the proposed algorithm as well as its pointwise iteration complexity are analyzed in detail. By testing the so-called sparse signal recovery problem in signal processing with nonconvex/convex regularization terms and by using adaptively updating strategy for the penalty parameter, a number of numerical results demonstrate the feasibility and efficiency of the new algorithm and further show that the $l_{1/2}$ regularization term is better than the l_1 regularization term in terms of CPU time, iteration number and recovery error. Our future work will focus on solving stochastic nonconvex optimization problems by using a similar first-order algorithm to ADMM.

References

- [1] H. Attouch, J. Bolte, P. Redont, A. Soubeyran, Proximal alternating minimization and projection methods for nonconvex problems: An approach based on the Kurdyka-Lojasiewicz inequality, *Mathematics of Operations Research*, 35 (2010) 438-457.
- [2] S. Boyd, N. Parikh, E. Chu, B. Peleato, J. Eckstein, Distributed optimization and statistical learning via the alternating direction method of multipliers, *Foundations and Trends in Machine Learning*, 3 (2010) 1-122.
- [3] J. Bai, H. Zhang, J. Li, A parameterized proximal point algorithm for separable convex optimization, *Optimization Letters*, 12 (2018) 1589-1608.
- [4] J. Bai, J. Li, F. Xu, H. Zhang, Generalized symmetric ADMM for separable convex optimization, *Computational Optimization and Applications*, 70 (2018) 129-170.

- [5] J. Bai, J. Li, Z. Wu, Several variants of the primal-dual hybrid gradient algorithm with applications, *Numerical Mathematics: Theory, Methods and Applications*, 12 (2019) 1-24.
- [6] R. Chartrand, V. Staneva, Restricted isometry properties and nonconvex compressive sensing, *Inverse Problems*, 24 (2008) 20-35.
- [7] X. Chang, S. Liu, P. Zhao, D. Song, A generalization of linearized alternating direction method of multipliers for solving two-block separable convex programming, *Journal of Computational and Applied Mathematics*, 357 (2019) 251-272.
- [8] E. Corman, X. Yuan, A generalized proximal point algorithm and its convergence rate, *SIAM Journal on Optimization*, 24 (2014) 1614-1638.
- [9] J. Douglas, H. Rachford, On the numerical solution of heat conduction problems in two and three space variables, *Transactions of the American Mathematical Society*, 82 (1956) 421-439.
- [10] J. Eckstein, D. Bertsekas, On the Douglas-Rachford splitting method and the proximal point algorithm for maximal monotone operators, *Mathematical Programming*, 55 (1992) 293-318.
- [11] M. Figueiredo, R. Nowak, S. Wright, Gradient projection for sparse reconstruction: application to compressed sensing and other inverse problems, *IEEE Journal of Selected Topics in Signal Processing*, 1 (2007) 586-597.
- [12] F. Fang, B. He, H. Liu, X. Yuan, Generalized alternating direction method of multipliers: new theoretical insights and applications, *Mathematical Programming Computation*, 7 (2015) 149-187.
- [13] D. Gabay, Applications of the method of multipliers to variational inequalities, *Studies in Mathematics and Its Applications*, 15 (1983) 299-331.
- [14] R. Glowinski, Lectures on numerical methods for non-linear variational problems, Published for the Tata Institute of Fundamental Research, Bombay [by] Springer-Verlag, 1980.
- [15] R. Glowinski, On alternating direction methods of multipliers: A historical perspective, In *Modeling, Simulation and Optimization for Science and Technology*, W. Fitzgibbon, Y.A. Kuznetsov, P. Neittaanmaki, O. Pironneau, eds., *Computational Methods in Applied Sciences*, Vol. 34, Springer, Dordrecht, (2014), 59-82.
- [16] R. Glowinski, T. Karkkainen, K. Majava, On the convergence of operator-splitting methods, in *Numerical Methods for Scientific Computing, Variational Problems and Applications*, E. Heikkola, Y. Kuznetsov, P. Neittaanmaki, and O. Pironneau, eds., CIMNE, Barcelona, 2003, pp. 67-79.
- [17] K. Guo, D. Han, T. Wu, Convergence of alternating direction method for minimizing sum of two nonconvex functions with linear constraints, *International Journal of Computer Mathematics*, 94 (2017) 1653-1669.
- [18] K. Guo, D. Han, D. Wang, T. Wu, Convergence of ADMM for multi-block nonconvex separable optimization models, *Frontier of Mathematics in China*, 12 (2017) 1139-1162.
- [19] K. Guo, D. Han, T. Wu, Convergence of ADMM for optimization problems with nonseparable nonconvex objective and linear constraints, *Pacific Journal of Optimization*, 14 (2018) 489-506.
- [20] M. Goncalves, J. Melo, R. Monteiro, Convergence rate bounds for a proximal ADMM with over-relaxation stepsize parameter for solving nonconvex linearly constrained problems, *arXiv:1702.01850v2* (2017).
- [21] B. He, H. Liu, Z. Wang, X. Yuan, A strictly contractive Peaceman-Rachford splitting method for convex programming, *SIAM Journal on Optimization*, 24 (2014) 1011-1040.
- [22] B. He, H. Yang, S. Wang, Alternating direction method with self-adaptive penalty parameters for monotone variational inequalities, *Journal of Optimization Theory and Applications*, 106 (2000) 337-356.
- [23] B. He, F. Ma, X. Yuan, Convergence study on the symmetric version of ADMM with larger step sizes, *SIAM Journal on Imaging Science*, 9 (2016) 1467-1501.

- [24] M. Hong, Z. Luo, M. Razaviyay, Convergence analysis of alternating direction method of multipliers for a family of nonconvex problems, *SIAM Journal on Optimization*, 26 (2016) 337-364.
- [25] S. Kim, K. Koh, M. Lustig, S. Boyd, D. Gorinvesky, An interior-point method for large-scale l_1 -regularized least squares, *IEEE Journal of Selected Topics in Signal Processing*, 1 (2007) 606-617.
- [26] P. Lions, B. Mercier, Splitting algorithms for the sum of two nonlinear operators, *SIAM Journal on Numerical Analysis*, 16 (1979) 964-979.
- [27] G. Li, T. Pong, Global convergence of splitting methods for nonconvex composite optimization, *SIAM Journal on Optimization*, 25 (2015) 2434-2460.
- [28] Q. Liu, X. Shen, Y. Gu, Linearized ADMM for nonconvex non-smooth optimization with convergence analysis, *IEEE Access*, (2019) doi:10.1109/ACCESS.2019.2914461.
- [29] D. Malioutov, M. Cetin, A. Willsky, A sparse signal reconstruction perspective for source localization with sensor arrays, *IEEE Transactions on Signal Processing*, 53 (2005) 3010-3022.
- [30] B. Mordukhovich, *Variational Analysis and Generalized Differentiation I: Basic Theory*, Grundlehren der mathematischen Wissenschaften, Springer-Verlag Berlin Heidelberg, 2006.
- [31] Y. Nesterov, A method of solving a convex programming problem with convergence rate $\mathcal{O}(1/k^2)$, *Soviet Mathematics Doklady*, 27 (1983) 372-376.
- [32] D. Peaceman, H. Rachford, The numerical solution of parabolic elliptic differential equations, *SIAM Journal on Applied Mathematics*, 3 (1955) 28-41.
- [33] R. Rockafellar, R. Wets, *Variational Analysis*, Springer-Verlag Berlin Heidelberg, 1998.
- [34] T. Sun, C. Zhang, Sparse matrix inversion with scaled Lasso, *Journal of Machine Learning Research*, 14 (2013) 3385-3418.
- [35] W. Wang, Y. Yin, J. Zeng, Global convergence of ADMM in nonconvex nonsmooth optimization, *Journal of Scientific Computing*, 78 (2019) 29-63
- [36] Z. Wu, M. Li, D. Wang, D. Han, A symmetric alternating direction method of multipliers for separable nonconvex minimization problems, *Asia-Pacific Journal of Operational Research*, 34 (2017) 1750030, 27 pages.
- [37] Z. Xu, X. Chang, F. Xu, H. Zhang, $L_{1/2}$ regularization: a thresholding representation theory and a fast solver, *IEEE Transactions on Neural Networks and Learning Systems*, 23 (2012) 1013-1027.
- [38] L. Yang, T. K. Pong, X. Chen, Alternating direction method of multipliers for a class of nonconvex and nonsmooth problems with applications to background/foreground extraction, *SIAM Journal on Imaging Sciences*, 10 (2017) 74-110.
- [39] M. Zhu, T. Chan, An efficient primal-dual hybrid gradient algorithm for total variation image restoration, CAM Report 08-34, UCLA, Los Angeles, CA, 2008.