Optimal linearized symmetric ADMM for multi-block separable convex programming *

Xiaokai Chang[1,2]†, Jianchao Bai[3,4] and Sanyang Liu[1]

[1]School of Mathematics and Statistics, Xidian University, Xi'an 710071, P.R. China
[2]College of Science, Lanzhou University of Technology, Lanzhou 730050, P.R. China
[3]School of Mathematics and Statistics, Xi'an Jiaotong University, Xi'an 710049, P.R. China
[4]Department of Mathematics, Louisiana State University, Baton Rouge, LA 70803-4918, USA

April 11, 2018

Abstract

Due to its wide applications and simple implementations, the Alternating Direction Method of Multipliers (ADMM) has been extensively investigated by researchers from different areas. In this paper, we focus on a linearized symmetric ADMM (LSADMM) for solving the multi-block separable convex minimization model. This LSADMM partitions the data into two group variables and updates the Lagrange multiplier twice in different forms and with suitable stepsizes, where two grouped variables are updated in a Gauss-Seidel scheme while variables within each group are updated in a Jacobi scheme. For the second group variables, however, linearized and relaxation techniques are used to deal with the quadratic term of subproblems and to accelerate the algorithm, respectively. Since the proximal operator is designed uncertainly (positive-definite or positive-indefinite), the region of the proximal parameter, involved in the second group subproblems, is partitioned into the union of three different sets. We show the global convergence and the sublinear ergodic convergence rate of LSADMM for the two cases, and give a counter-example to illustrate that the convergence of LSADMM for the remaining case can not be guaranteed. Theoretically, we obtain the optimal lower bound of the proximal parameter. Furthermore, a linearized full Jacobian splitting version of augmented Lagrangian method can be deduced from the proposed method. Besides, numerical experiments on the Latent Variable Gaussian Graphical Model Selection (LVGGMS) problems are presented to demonstrate the efficiency of the proposed algorithm and the significant advantage of the optimal lower bound of the proximal parameter.

Key words. Alternating direction method of multipliers; Convex programming; Prediction-correction; Linearized technique; Indefinite proximal term; Global convergence; Complexity

---

*This work was supported by the National Natural Science Foundation of China under Grant 61373174.
†E-mail addresses: xkchang@lut.cn(X. Chang); bjc1987@163.com(J. Bai); liusanyang@126.com(S. Liu).

1 Introduction

This paper aims to develop a linearized version of the algorithm in [1] for the following multi-block separable convex minimization model:

$$
\begin{array}{ll}
\min & \sum_{i=1}^{p} f_i(x_i) + \sum_{j=1}^{q} g_j(y_j) \\
\text{s.t.} & \sum_{i=1}^{p} A_i x_i + \sum_{j=1}^{q} B_j y_j = c, \\
& x_i \in \mathcal{X}_i, y_j \in \mathcal{Y}_j, \quad i = 1, \cdots, p, j = 1, \cdots, q,
\end{array}
\tag{1}
$$

where $f_i(x_i) : \mathcal{X}_i \to \mathbb{R}$, $g_j(y_j) : \mathcal{Y}_j \to \mathbb{R}$ are closed and proper convex functions (possibly nonsmooth); $A_i : \mathcal{X}_i \to \mathbb{R}^m$ and $B_j : \mathcal{Y}_j \to \mathbb{R}^m$ are linear operators; $c \in \mathbb{R}^m$ is a given vector; $\mathcal{X}_i$ and $\mathcal{Y}_j$ are closed convex sets; and $p, q \geq 1$ are positive integers. The model (1) has wide applications in e.g. statistical learning [2], optimal control [3], image/signal processing [4, 5] and so forth. Throughout, we assume that the solution set of problem (1) is nonempty and all linear operators A_i $(i = 1, \cdots, p)$ and B_j $(j = 1, \cdots, q)$ have full column rank.

For any $\sigma > 0$, the augmented Lagrangian function of problem (1) is given by

$$
\mathcal{L}_\sigma(\mathbf{x}, \mathbf{y}, \lambda) = \theta(u) - \langle \lambda, \mathcal{A}\mathbf{x} + \mathcal{B}\mathbf{y} - c \rangle + \frac{\sigma}{2} \left\| \mathcal{A}\mathbf{x} + \mathcal{B}\mathbf{y} - c \right\|^2, \tag{2}
$$

where $\lambda \in \mathbb{R}^m$ denotes the Lagrange multiplier and

$$
\left.
\begin{array}{l}
\mathbf{x} = (x_1^\mathsf{T}, \cdots, x_p^\mathsf{T})^\mathsf{T}, \quad \mathbf{y} = (y_1^\mathsf{T}, \cdots, y_q^\mathsf{T})^\mathsf{T}, \\
\mathcal{A} = (A_1, \cdots, A_p), \quad \mathcal{B} = (B_1, \cdots, B_q), \\
u = (\mathbf{x}^\mathsf{T}, \mathbf{y}^\mathsf{T})^\mathsf{T}, \quad \theta(u) = \sum_{i=1}^{p} f_i(x_i) + \sum_{j=1}^{q} g_j(y_j).
\end{array}
\right\}
\tag{3}
$$

As a splitting version of the augmented Lagrangian method (ALM), the Alternating Direction Method of Multipliers (ADMM) proposed originally in [6] is regarded as a benchmark method for solving the two-block case of (1) and performs efficient in practical computation. One obvious advantage of ADMM is that it can deal with the variables separably and make full use of the separable structure of the objective functions. The direct extension of ADMM (ADMMe) for the multi-block problem (1) reads the following iterations:

$$
\left.
\begin{array}{l}
\text{For} \;\; i = 1, \cdots, p, \\
\quad x_i^{k+1} = \arg\min_{x_i \in \mathcal{X}_i} \mathcal{L}_\sigma(\mathbf{x}_{(1:i-1)}^{k+1}, x_i, \mathbf{x}_{(i+1:p)}^{k}, \; \mathbf{y}^k, \; \lambda^k), \\
\text{For} \;\; j = 1, \cdots, q, \\
\quad y_j^{k+1} = \arg\min_{y_j \in \mathcal{Y}_j} \mathcal{L}_\sigma(\mathbf{x}^{k+1}, \; \mathbf{y}_{(1:j-1)}^{k+1}, y_j, \mathbf{y}_{(j+1:q)}^{k}, \; \lambda^k), \\
\quad \lambda^{k+1} = \lambda^k - \beta\sigma \left(\mathcal{A}\mathbf{x}^{k+1} + \mathcal{B}\mathbf{y}^{k+1} - c \right),
\end{array}
\right\}
$$

where $\beta > 0$, e.g., $\beta \in \left(0, \frac{1+\sqrt{5}}{2}\right)$, is the stepsize of the dual variable; $\mathbf{x}_{(h:k)}$ denotes $\{x_h, x_{h+1}, \cdots, x_k\}$ for given positive integers h and $k (k \geq h)$, with the convention that $\mathbf{x}_{(h:h-1)} = \emptyset$. However, the convergence of ADMMe can not be guaranteed without strong assumptions on the coefficient operators [7]. With the purpose of preserving the numerical advantages of ADMMe, a large number of

ADMM's variants had been developed by adding proximal terms (or correction step) to ensure the global convergence, or by adopting linearized and generalized technique to accelerate the convergence speed. Next, we review some researches that are closely related to our proposed algorithm.

On the one hand, for the ADMM involving proximal terms or correction step, He et al. proposed an ADMM with a Gaussian back substitution to correct the iterations generated by ADMMe [8] for solving problem (1) with case $q = 0$, whose convergence was analyzed via the analytic framework of contractive-type methods based on a unified prediction-correction interpretation. Besides, He et al.[9] also introduced an ADMM-type splitting method that by adding certain proximal terms, allowed some of the subproblems to be tackled in parallel. With two additional proximal penalty terms for the single x and y updates, Gu, et al.[10] studied a semi-proximal-based strictly contractive Peaceman-Rachford splitting method, which performed well for the two-block case problem but extremely limited the applications for the problems in big-data setting. To overcome the limitation of [10] and to select more flexible stepsizes of the dual variables, Bai et al. [1] recently developed a generalized symmetric ADMM to solve problem (1), where proximal terms were designed for each subproblem and the Lagrange multiplier was updated twice at each iteration. More recently, a hybrid middle proximal ADMM was proposed for dealing with problem (1) with case $q = 0$, where larger region of the proximal parameter was specified and linear convergence rate was established [11]. In general, adding a proper proximal regularization to the ADMM's subproblems is a useful technology, since it can make the algorithm easily implementable by simplifying the objective functions. For more researches about the proximal ADMM, we refer to e.g. [12, 13, 14, 15]. Although there are many proximal ADMM-type algorithms, linearized technique still needs to be adopted when the proximal subproblem has no closed solution form, which is one of our motivations. In addition, the correction strategy was often used to ensure the convergence of ADMM-type algorithms, see e.g. [1, 16, 10, 17, 8, 18] and the references therein.

On the other hand, for the ADMM with generalized or linearized techniques, Eckstein and Bertsekas [19] proposed a generalized ADMM scheme by introducing a relaxation factor $\rho \in (0, 2)$, which can numerically accelerate the corresponding 2-block ADMM, e.g., $\rho \in (1, 2)$. For empirical studies of the acceleration performance of the method in [19], we refer to [20] for more details. It is well-known that the performance of ADMM-type algorithms depends highly on how to solve its subproblems efficiently, and linearized technique performs very useful to simplify subproblems, see e.g. [21, 22, 23]. Recently, He et al. [24] introduced the following linearized ADMM with positive-indefinite proximal regularization for problem (1) with $p = q = 1$:

$$\left.\begin{array}{l} x_1^{k+1} = \arg\min\limits_{x_1 \in \mathcal{X}_1} \mathcal{L}_\sigma(x_1, y_1^k, \lambda^k), \\ y_1^{k+1} = \arg\min\limits_{y_1 \in \mathcal{Y}_1} \mathcal{L}_\sigma(x_1^{k+1}, y_1, \lambda^k) + \frac{1}{2}\|y_1 - y_1^k\|_{D_1}^2, \\ \lambda^{k+1} = \lambda^k - \sigma\left(A_1 x_1^{k+1} + B_1 y_1^{k+1} - c\right), \end{array}\right\} \tag{4}$$

where $D_1 = \tau r_1 \mathcal{I} - \sigma B_1^* B_1$ with $r_1 > \sigma\|B_1^* B_1\|$, and they shown that any value of the proximal parameter $\tau < 0.75$ can yield divergence of the linearized ADMM. Inspired by the work on the indefinite proximal ADMM [22, 25, 21], Gao et al. [17] considered the following the linearized symmetric ADMM with positive-indefinite proximal regularization:

$$\left.\begin{array}{l} x_1^{k+1} = \arg\min_{x_1\in\mathcal{X}_1} \mathcal{L}_\sigma(x_1, y_1^k, \lambda^k), \\ \lambda^{k+\frac{1}{2}} = \lambda^k - \alpha\sigma\left(A_1x_1^{k+1} + B_1y_1^k - c\right), \\ y_1^{k+1} = \arg\min_{y_1\in\mathcal{Y}_1} \mathcal{L}_\sigma(x_1^{k+1}, y_1, \lambda^k) + \frac{1}{2}\|y_1 - y_1^k\|_{D_1}^2, \\ \lambda^{k+1} = \lambda^{k+\frac{1}{2}} - \sigma\left(A_1x_1^{k+1} + B_1y_1^{k+1} - c\right), \end{array}\right\} \tag{5}$$

where $D_1 = \tau r_1\mathcal{I} - \sigma B_1^*B_1$ with $r_1 > \sigma\|B_1^*B_1\|$, $\tau \in [\frac{\alpha^2-\alpha+4}{\alpha^2-2\alpha+5}, 1)$ and $\alpha \in (-1, 1)$. For some applications, the scheme in (5) can alleviate its subproblem as easy as estimating the proximity operator of a function in the objective. More recently, based on schemes in (4)-(5) a generalized ADMM was constructed by Jiang et al. [26], where the optimal lower bound of the proximal parameter τ was specified to $\frac{3+\alpha}{4}$ to improve the convergence.

Algorithm 1 (LSADMM for solving problem (1))

Step 0. Initialize the parameters $\sigma > 0, \rho > 0$ and starting point $w^0 \in \mathcal{W}$. Set $k = 0$.

Step 1. Solve the x_i-subproblems in parallel, and then update the Lagrange multipliers:

$$\left\{\begin{array}{l} \text{For} \quad i = 1, \cdots, p, \\ \quad x_i^{k+1} = \arg\min_{x_i\in\mathcal{X}_i} \left\{ f_i(x_i) - \langle\lambda^k, A_ix_i\rangle + \frac{\sigma}{2}\left\| A_ix_i + \sum_{l=1,l\neq i}^{p} A_lx_l^k + \mathcal{B}\mathbf{y}^k - c\right\|^2 + \mathcal{P}_i^k(x_i)\right\}, \\ \qquad\quad \text{where } \mathcal{P}_i^k(x_i) = \frac{\rho\sigma}{2}\|A_i(x_i - x_i^k)\|^2, \\ \lambda^{k+\frac{1}{2}} = \lambda^k - \alpha\sigma\left(\mathcal{A}\mathbf{x}^{k+1} + \mathcal{B}\mathbf{y}^k - c\right), \end{array}\right. \tag{6}$$

where α is a step size.

Step 2. Solve y_j-subproblems in parallel, and then update the Lagrange multipliers:

$$\left\{\begin{array}{l} \text{For} \quad j = 1, \cdots, q, \\ \quad y_j^k = \arg\min_{y_j\in\mathcal{Y}_j} \left\{ g_j(y_j) - \langle\lambda^{k+\frac{1}{2}}, B_jy_j\rangle \right. \\ \qquad\qquad \left. + \frac{\sigma}{2}\left\|\beta(\mathcal{A}\mathbf{x}^{k+1} + \sum_{l=1,l\neq j}^{q} B_ly_l^k - c) - (1-\beta)B_jy_j^k + B_jy_j\right\|^2 + \mathcal{Q}_j^k(y_j)\right\}, \\ \qquad\quad \text{where } \mathcal{Q}_j^k(y_j) = \frac{1}{2}\|y_j - y_j^k\|_{D_j}^2, \\ \lambda^{k+1} = \lambda^{k+\frac{1}{2}} - \sigma\left(\beta\mathcal{A}\mathbf{x}^{k+1} + (1-\beta)(c - \mathcal{B}\mathbf{y}^k) + \mathcal{B}\mathbf{y}^{k+1} - c\right), \end{array}\right. \tag{7}$$

where β is a relaxation factor and

$$D_j = \tau r_j\mathcal{I} - \sigma B_j^*B_j, \quad \text{with} \quad r_j > \sigma\|B_j^*B_j\|, \quad \tau \in (0, +\infty). \tag{8}$$

Step 3. Replace k by $k + 1$, and turn to Step 1.

There are three major contributions of this paper. Firstly, we partition the data of problem (1) into two group variables, where the two grouped variables are updated in a Gauss-Seidel scheme while the variables within each group are updated in a Jacobi scheme so that parallel computing can be implemented for the problems involving big data. In order to accelerate the convergence speed, the relaxation scheme proposed in [19] as well as a linearized technique are applied to the second group variables. Especially, if we treat all the variables as a whole ($p = 0$ or $q = 0$), the proposed method can result in two types of full Jacobian splitting versions of ALM. Secondly, sufficient conditions are derived to theoretically guarantee the global convergence of LSADMM with the worst-case $\mathcal{O}(1/t)$ convergence rate in the ergodic sense. We also obtain the optimal lower bound of the proximal parameter τ, since as said in [9] that slow convergence would occur in terms of solving the proximal subproblem if the proximal parameter was set too large, which may significantly affect the whole computational efficiency of the algorithm. Thirdly, numerical experiments on solving a sparse matrix problem in statistical learning validate the significant improvement of our proposed LSADMM by comparing with some state-of-the-art algorithms developed recently.

The rest of this paper is organized as follows. In order to analyze the convergence of proposed LSADMM conveniently, Section 2 interprets it as a prediction-correction procedure. Section 3 is devoted to analyzing the convergence conditions for LSADMM and obtaining an optimal lower bound for the involved proximal parameter τ. As a counter-example a simple linear programming example is also presented to show that the convergence of LSADMM can not be guaranteed for a special region of τ. Section 4 is devoted to a linearized full Jacobian splitting version of ALM, for solving the problem with $p = 0$. In Section 5, we test a series of numerical examples and analyze the reported results. Finally, the paper is summarized in Section 6.

2 A prediction-correction interpretation

Before interpreting the LSADMM as a prediction-correction procedure (see e.g. [1, 17, 26] for similar approaches), in this section we first would characterize the saddle point of the problem by the aid of the following notations:

$$\begin{aligned}
&\gamma = \alpha + \beta,\\
&\widetilde{\lambda}^k = \lambda^k - \sigma\left(\mathcal{A}\mathbf{x}^{k+1} + \mathcal{B}\mathbf{y}^k - c\right),\\
&u^k = \begin{pmatrix} \mathbf{x}^k \\ \mathbf{y}^k \end{pmatrix}, \quad w^k = \begin{pmatrix} u^k \\ \lambda^k \end{pmatrix}, \quad v^k = \begin{pmatrix} \mathbf{y}^k \\ \lambda^k \end{pmatrix},\\
&\widetilde{u}^k = \begin{pmatrix} \mathbf{x}^{k+1} \\ \mathbf{y}^{k+1} \end{pmatrix}, \quad \widetilde{w}^k = \begin{pmatrix} \widetilde{u}^k \\ \widetilde{\lambda}^k \end{pmatrix}, \quad \widetilde{v}^k = \begin{pmatrix} \mathbf{y}^k \\ \widetilde{\lambda}^k \end{pmatrix},
\end{aligned} \tag{9}$$

and we also define three operators that will be used in the sequel:

$$\mathcal{D} = \begin{bmatrix} D_1 & & \\ & \ddots & \\ & & D_q \end{bmatrix}, \mathcal{D}_0 = \begin{bmatrix} r_1\mathcal{I} - \sigma B_1^* B_1 & & \\ & \ddots & \\ & & r_q\mathcal{I} - \sigma B_q^* B_q \end{bmatrix}, \widetilde{B} = \begin{bmatrix} B_1^* B_1 & & \\ & \ddots & \\ & & B_q^* B_q \end{bmatrix}. \tag{10}$$

It is well-known from optimization that solving problem (1) is equivalent to finding a saddle point of the associated Lagrangian function:

$$L(u, \lambda) = \theta(u) - \langle \lambda, \mathcal{A}\mathbf{x} + \mathcal{B}\mathbf{y} - c \rangle, \tag{11}$$

where $\theta(u)$ is the objective functions defined in (3). From convex analysis [27, Theorem 28.3], a point $w^* = (u^*, \lambda^*)$ is called the saddle point of $L(u, \lambda)$ if and only if $L(u^*, \lambda) \le L(u^*, \lambda^*) \le L(u, \lambda^*)$ which can be expressed as a compact variational inequality:

$$\mathrm{VI}(\mathcal{W}, \mathcal{F}, \theta): \quad \theta(u) - \theta(u^*) + \langle w - w^*, \mathcal{F}(w^*) \rangle \ge 0, \quad \forall w \in \mathcal{W}, \tag{12}$$

where $\mathcal{X} = \mathcal{X}_1 \times \cdots \times \mathcal{X}_p,\ \mathcal{Y} = \mathcal{Y}_1 \times \cdots \times \mathcal{Y}_q$ and

$$\mathcal{F}(w) = \begin{pmatrix} -\mathcal{A}^*\lambda \\ -\mathcal{B}^*\lambda \\ \mathcal{A}\mathbf{x} + \mathcal{B}\mathbf{y} - c \end{pmatrix}, \quad \mathcal{W} = \mathcal{X} \times \mathcal{Y} \times \mathbb{R}^m. \tag{13}$$

Since the solution set of problem (1) is assumed to be nonempty, the solution set $\mathcal{W}^*$ of $\mathrm{VI}(\mathcal{W}, \mathcal{F}, \theta)$ is also nonempty and convex. If w^* is a saddle point of the Lagrangian function associated to problem (1), then u^* is a solution of problem (1). Compared (12) and the following (14), the global convergence of LSADMM can be proved if the extra term $(w^k - \widetilde{w}^k)$ converges to zero.

Lemma 1 *(Prediction step) Let $\{\widetilde{w}^k\}$ be generated by LSADMM from given $\{w^k\}$. Then, we have*

$$\theta(u) - \theta(\widetilde{u}^k) + \left\langle w - \widetilde{w}^k, \mathcal{F}(\widetilde{w}^k) \right\rangle \ge \left\langle w - \widetilde{w}^k, \mathcal{Q}(w^k - \widetilde{w}^k) \right\rangle, \quad \forall w \in \mathcal{W}, \tag{14}$$

where

$$\mathcal{Q} = \begin{bmatrix} \mathcal{Q}_1 & 0 \\ 0 & \mathcal{Q}_2 \end{bmatrix}, \tag{15}$$

with

$$\mathcal{Q}_1 = \sigma \begin{bmatrix} \rho A_1^* A_1 & -A_1^* A_2 & \cdots & -A_1^* A_p \\ -A_2^* A_1 & \rho A_2^* A_2 & \cdots & -A_2^* A_p \\ \vdots & \vdots & \ddots & \vdots \\ -A_p^* A_1 & -A_p^* A_2 & \cdots & \rho A_p^* A_p \end{bmatrix}, \tag{16}$$

$$\mathcal{Q}_2 = \left[\begin{array}{cccc:c} \tau r_1 \mathcal{I} & 0 & \cdots & 0 & (1-\gamma) B_1^* \\ 0 & \tau r_2 \mathcal{I} & \cdots & 0 & (1-\gamma) B_2^* \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & \tau r_q \mathcal{I} & (1-\gamma) B_q^* \\ \hdashline -B_1 & -B_2 & \cdots & -B_q & \frac{1}{\sigma}\mathcal{I} \end{array} \right].$$

Proof. It is not hard by the first-order optimality conditions of the subproblems in LSADMM that for $i = 1, \cdots, p$, we have $x_i^{k+1} \in \mathcal{X}_i$ and

$$f_i(x_i) - f_i(x_i^{k+1}) + \left\langle x_i - x_i^{k+1}, -A_i^*[\lambda^k - \sigma(A_i x_i^{k+1} + c_{x,i}) - \rho\sigma A_i(x_i^{k+1} - x_i^k)] \right\rangle \ge 0,\ \forall x_i \in \mathcal{X}_i, \tag{17}$$

where $c_{x,i} = \sum\limits_{l=1, l \ne i}^{p} A_l x_l^k + \mathcal{B}\mathbf{y}^k - c$. For $j = 1, \cdots, q$, we have $y_j^{k+1} \in \mathcal{Y}_j$ and

$$g_j(y_j) - g_j(y_j^{k+1}) + \left\langle y_j - y_j^{k+1}, -B_j^*[\lambda^{k+\frac{1}{2}} - \sigma(\beta c_{y,j} - B_j y_j^k + B_j y_j^{k+1})] \right\rangle$$
$$+ \left\langle y_j - y_j^{k+1}, D_j(y_j^{k+1} - y_j^k) \right\rangle \ge 0,\ \forall y_j \in \mathcal{Y}_j,$$

where $c_{y,j} = \mathcal{A}\mathbf{x}^{k+1} + \mathcal{B}\mathbf{y}^k - c$. Note by the definition of D_j in (8) that

$$g_j(y_j) - g_j(y_j^{k+1}) + \left\langle y_j - y_j^{k+1}, -B_j^*[\lambda^{k+\frac{1}{2}} - \sigma\beta c_{y,j}] + \tau r_j(y_j^{k+1} - y_j^k) \right\rangle \geq 0. \tag{18}$$

Because of the relationship

$$\lambda^{k+\frac{1}{2}} = \widetilde{\lambda}^k + (\alpha - 1)(\widetilde{\lambda}^k - \lambda^k) = \lambda^k - \alpha(\lambda^k - \widetilde{\lambda}^k), \tag{19}$$

the above inequalities (17) and (18) can be immediately rewritten as

$$\begin{cases} f_i(x_i) - f_i(x_i^{k+1}) + \left\langle x_i - x_i^{k+1}, -A_i^*[\widetilde{\lambda}^k + \sigma \sum\limits_{l=1,l\neq i}^{n} A_l(x_l^{k+1} - x_l^k) - \rho\sigma \sum\limits_{l=1}^{n} A_l(x_l^{k+1} - x_l^k)] \right\rangle \geq 0, \\ g_j(y_j) - g_j(y_j^{k+1}) + \left\langle y_j - y_j^{k+1}, -B_j^*[\widetilde{\lambda}^k + (\alpha + \beta - 1)(\widetilde{\lambda}^k - \lambda^k)] + \tau r_j(y_j^{k+1} - y_j^k) \right\rangle \geq 0. \end{cases} \tag{20}$$

Besides, we have from the definition of $\widetilde{\lambda}^k$ in (9) that

$$\mathcal{A}\mathbf{x}^{k+1} + \mathcal{B}\mathbf{y}^{k+1} - c - \mathcal{B}(\mathbf{y}^{k+1} - \mathbf{y}^k) + \frac{1}{\sigma}(\widetilde{\lambda}^k - \lambda^k) = 0,$$

namely,

$$\left\langle \lambda - \widetilde{\lambda}^k, \ \ \mathcal{A}\mathbf{x}^{k+1} + \mathcal{B}\mathbf{y}^{k+1} - c - \mathcal{B}(\mathbf{y}^{k+1} - \mathbf{y}^k) + \frac{1}{\sigma}(\widetilde{\lambda}^k - \lambda^k) \right\rangle = 0, \quad \forall \lambda \in \mathbb{R}^m.$$

which completes the whole proof by (20) together with the notations in (9). □

Lemma 2 *(Correction step) The sequences $\{w^k\}$ and $\{\widetilde{w}^k\}$ defined in (9) satisfy*

$$w^{k+1} = w^k - \mathcal{M}(w^k - \widetilde{w}^k), \tag{21}$$

where

$$\mathcal{M} = \left[\begin{array}{c|cccc|c} \mathcal{I} & & & & & \\ \hline & \mathcal{I} & & & & \\ & & \ddots & & & \\ & & & & \mathcal{I} & \\ \hline & -\sigma B_1 & \cdots & & -\sigma B_q & \gamma\mathcal{I} \end{array}\right]. \tag{22}$$

Proof. By the notation $\widetilde{\lambda}^k$, the updating of λ^{k+1} can be rewritten as

$$\begin{aligned} \lambda^{k+1} &= \lambda^{k+\frac{1}{2}} - \sigma\left(\beta\mathcal{A}\mathbf{x}^{k+1} + (1-\beta)(c - \mathcal{B}\mathbf{y}^k) + \mathcal{B}\mathbf{y}^{k+1} - c\right) \\ &= \lambda^{k+\frac{1}{2}} - \beta\sigma(\mathcal{A}\mathbf{x}^{k+1} + \mathcal{B}\mathbf{y}^k - c) + \sigma\mathcal{B}(\mathbf{y}^k - \mathbf{y}^{k+1}) \\ &= \lambda^k - \alpha(\lambda^k - \widetilde{\lambda}^k) - \beta(\lambda^k - \widetilde{\lambda}^k) + \sigma\mathcal{B}(\mathbf{y}^k - \mathbf{y}^{k+1}) \\ &= \lambda^k - (\alpha + \beta)(\lambda^k - \widetilde{\lambda}^k) + \sigma\mathcal{B}(\mathbf{y}^k - \mathbf{y}^{k+1}). \end{aligned} \tag{23}$$

The above equation as well as the notations $\widetilde{u}^k = u^{k+1}$ in (9) show (21) holds. □

Bases on Lemmas 1-2, LSADMM can be interpreted as a prediction-correction procedure, where the prediction step (14) is to compute $\widetilde{w}^k$, and the correction step (21) is to obtain w^{k+1}. In the next section, we analyze the convergence of LSADMM with the use of these two lemmas.

3 Convergence analysis

In what follows, we first investigate some properties of the sequence $\{w^k - w^*\}$. Then, the global convergence of LSADMM and its sublinear ergodic convergence rate are discussed in detail for different cases of the proximal parameter τ. Throughout we slightly abuse the notation $\|v\|_{\mathcal{M}}^2 := \langle v, \mathcal{M}v\rangle$ when $\mathcal{M}$ is not positive definite.

3.1 Basic Properties

The following lemma gives a sufficient condition to ensure the positive definiteness of the operator $\mathcal{H}$, which plays an important role in proving the convergence of ADMM-type algorithm.

Lemma 3 *The operator $\mathcal{H} := \mathcal{Q}\mathcal{M}^{-1}$ is symmetric positive definite if*

$$\rho > p-1, \quad \tau > \frac{q(2+\gamma)}{4} \quad \textit{with} \quad \gamma \in (0,2), \tag{24}$$

and all the operators $A_i, B_j (i = 1, \cdots, p; j = 1, \cdots, q)$ have full column rank.

Proof. After simple calculations, we can obtain

$$\mathcal{H} = \begin{bmatrix} \mathcal{Q}_1 & 0 \\ 0 & \mathcal{H}_2 \end{bmatrix}, \tag{25}$$

where $\mathcal{Q}_1$ is defined in (16) and

$$\mathcal{H}_2 = \frac{1-\gamma}{\gamma}\left[\begin{array}{cccc|c} \frac{\gamma\tau r_1}{1-\gamma}\mathcal{I} + \sigma B_1^*B_1 & \sigma B_1^*B_2 & \cdots & \sigma B_1^*B_q & B_1^* \\ \sigma B_2^*B_1 & \frac{\gamma\tau r_2}{1-\gamma}\mathcal{I} + \sigma B_2^*B_2 & \cdots & \sigma B_2^*B_q & B_2^* \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \sigma B_q^*B_1 & \sigma B_q^*B_2 & \cdots & \frac{\gamma\tau r_q}{1-\gamma}\mathcal{I} + \sigma B_q^*B_q & B_q^* \\ \hline B_1 & B_2 & \cdots & B_q & \frac{1}{\sigma(1-\gamma)}\mathcal{I} \end{array}\right].$$

Clearly, each block of $\mathcal{H}$ is symmetric, and the upper-left operator $\mathcal{Q}_1$ is positive definite for any $\rho > p-1$ and the full column rank assumption on A_i. So in the following we only need to show that the lower-right operator $\mathcal{H}_2$ is positive definite.

Notice that by $r_j > \sigma\|B_j^*B_j\|$ $(j = 1, \cdots, q)$ we have

$$\mathcal{H}_2 \succ \widetilde{\mathcal{H}_2} := \frac{1-\gamma}{\gamma}\left[\begin{array}{cccc|c} \frac{\gamma\tau+1-\gamma}{1-\gamma}\sigma B_1^*B_1 & \sigma B_1^*B_2 & \cdots & \sigma B_1^*B_q & B_1^* \\ \sigma B_2^*B_1 & \frac{\gamma\tau+1-\gamma}{1-\gamma}\sigma B_2^*B_2 & \cdots & \sigma B_2^*B_q & B_2^* \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \sigma B_q^*B_1 & \sigma B_q^*B_2 & \cdots & \frac{\gamma\tau+1-\gamma}{1-\gamma}\sigma B_q^*B_q & B_q^* \\ \hline B_1 & B_2 & \cdots & B_q & \frac{1}{\sigma(1-\gamma)}\mathcal{I} \end{array}\right].$$

Hence $\mathcal{H}_2$ is positive definite if $\widetilde{\mathcal{H}_2}$ is positive definite. Noting that the operator $\widetilde{\mathcal{H}_2}$ can be decomposed as $\widetilde{\mathcal{H}_2} = \widetilde{\mathcal{D}}^*\mathcal{H}_2^0\widetilde{\mathcal{D}}$, where

$$\widetilde{\mathcal{D}} = \left[\begin{array}{cccc|c} \sigma^{\frac{1}{2}}B_1 & & & & \\ & \sigma^{\frac{1}{2}}B_2 & & & \\ & & \ddots & & \\ & & & \sigma^{\frac{1}{2}}B_q & \\ \hline & & & & \sigma^{-\frac{1}{2}}\mathcal{I} \end{array}\right],$$

and

$$\mathcal{H}_2^0 = \frac{1-\gamma}{\gamma}\left[\begin{array}{cccc:c} \frac{\gamma\tau+1-\gamma}{1-\gamma}\mathcal{I} & \mathcal{I} & \cdots & \mathcal{I} & \mathcal{I} \\ \mathcal{I} & \frac{\gamma\tau+1-\gamma}{1-\gamma}\mathcal{I} & \cdots & \mathcal{I} & \mathcal{I} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \mathcal{I} & \mathcal{I} & \cdots & \frac{\gamma\tau+1-\gamma}{1-\gamma}\mathcal{I} & \mathcal{I} \\ \hdashline \mathcal{I} & \mathcal{I} & \cdots & \mathcal{I} & \frac{1}{1-\gamma}\mathcal{I} \end{array}\right].$$

According to the fact that

$$\left[\begin{array}{ccc:c} \mathcal{I} & & & (\gamma-1)\mathcal{I} \\ & \ddots & & \vdots \\ & & \mathcal{I} & (\gamma-1)\mathcal{I} \\ \hdashline & & & \mathcal{I} \end{array}\right]\mathcal{H}_2^0\left[\begin{array}{ccc:c} \mathcal{I} & & & (\gamma-1)\mathcal{I} \\ & \ddots & & \vdots \\ & & \mathcal{I} & (\gamma-1)\mathcal{I} \\ \hdashline & & & \mathcal{I} \end{array}\right]^* = \left[\begin{array}{c:c} \tau\mathcal{I}+(1-\gamma)\mathcal{E}\mathcal{E}^* & 0 \\ \hdashline 0 & \frac{1}{\gamma}\mathcal{I} \end{array}\right],$$

where $\mathcal{E}^* = (\mathcal{I}, \cdots, \mathcal{I})$, we can achieve that $\widetilde{\mathcal{H}_2}$ is positive definite if each operator B_i has full column rank and

$$\tau > q(\gamma-1), \quad \gamma \in (0,+\infty) \supset (0,2).$$

Clearly, the inequality $\frac{q(2+\gamma)}{4} > q(\gamma-1)$ is equivalent to $\gamma < 2$, so the operator $\mathcal{H}$ also is positive definite for any (ρ,τ,γ) satisfying (24). □

Now, we define

$$\mathcal{G} := \mathcal{Q} + \mathcal{Q}^* - \mathcal{M}^*\mathcal{H}\mathcal{M}, \tag{26}$$

which can be explicitly expressed as

$$\mathcal{G} = \begin{bmatrix} \mathcal{Q}_1 & 0 \\ 0 & \widetilde{\mathcal{G}} \end{bmatrix}, \tag{27}$$

with $\mathcal{Q}_1$ being defined in (16) and

$$\begin{aligned} \widetilde{\mathcal{G}} &= \left[\begin{array}{c:c} \tau\mathcal{D}_0 - \sigma\mathcal{B}^*\mathcal{B} + \tau\sigma\widetilde{B} & 0 \\ \hdashline 0 & \frac{2-\gamma}{\sigma}\mathcal{I} \end{array}\right] \\ &= \left[\begin{array}{cccc:c} \tau r_1\mathcal{I} - \sigma B_1^*B_1 & -\sigma B_1^*B_2 & \cdots & -\sigma B_1^*B_q & 0 \\ -\sigma B_2^*B_1 & \tau r_2\mathcal{I} - \sigma B_2^*B_2 & \cdots & -\sigma B_2^*B_q & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ -\sigma B_q^*B_1 & -\sigma B_q^*B_2 & \cdots & \tau r_q\mathcal{I} - \sigma B_q^*B_q & 0 \\ \hdashline 0 & 0 & \cdots & 0 & \frac{2-\gamma}{\sigma}\mathcal{I} \end{array}\right]. \end{aligned} \tag{28}$$

Lemma 4 *Let $\mathcal{H}$ and $\mathcal{G}$ be defined in Lemma 3 and (26), respectively. Then, the sequence $\{\widetilde{w}^k\}$ generated by LSADMM from given $\{w^k\}$ satisfies*

$$\theta(u) - \theta(\widetilde{u}^k) + \left\langle w - \widetilde{w}^k, \mathcal{F}(\widetilde{w}^k)\right\rangle \geq \frac{1}{2}\left(\|w - w^{k+1}\|_{\mathcal{H}}^2 - \|w - w^k\|_{\mathcal{H}}^2\right) + \frac{1}{2}\|w^k - \widetilde{w}^k\|_{\mathcal{G}}^2, \quad \forall w \in \mathcal{W}.$$

Proof. Combining Lemmas 1-2 and the relationship $\mathcal{Q} = \mathcal{H}\mathcal{M}$, for any $w \in \mathcal{W}$ we have

$$\begin{aligned} \theta(u) - \theta(\widetilde{u}^k) + \left\langle w - \widetilde{w}^k, \mathcal{F}(\widetilde{w}^k) \right\rangle &\geq \left\langle w - \widetilde{w}^k, \mathcal{H}\mathcal{M}(w^k - \widetilde{w}^k) \right\rangle \\ &= \left\langle w - \widetilde{w}^k, \mathcal{H}(w^k - w^{k+1}) \right\rangle. \end{aligned} \tag{29}$$

Since $\mathcal{H}$ is symmetric, we obtain

$$\begin{aligned} \left\langle w - \widetilde{w}^k, \mathcal{H}(w^k - w^{k+1}) \right\rangle &= \frac{1}{2}\left(\|w - w^{k+1}\|_{\mathcal{H}}^2 - \|w - w^k\|_{\mathcal{H}}^2 \right) \\ &\quad + \frac{1}{2}\left(\|w^k - \widetilde{w}^k\|_{\mathcal{H}}^2 - \|w^{k+1} - \widetilde{w}^k\|_{\mathcal{H}}^2 \right), \end{aligned} \tag{30}$$

where the last two terms can be rewritten as

$$\frac{1}{2}\left(\|w^k - \widetilde{w}^k\|_{\mathcal{H}}^2 - \|w^{k+1} - \widetilde{w}^k\|_{\mathcal{H}}^2 \right) = \frac{1}{2}\|w^k - \widetilde{w}^k\|_{\mathcal{G}}^2. \tag{31}$$

Then, the proof is completed by substituting (30)-(31) into (29). □

Lemma 5 *For the sequence $\{\widetilde{w}^k\}$ generated by LSADMM from given $\{w^k\}$, we have*

$$\|w^{k+1} - w^*\|_{\mathcal{H}}^2 \leq \|w^k - w^*\|_{\mathcal{H}}^2 - \|w^k - \widetilde{w}^k\|_{\mathcal{G}}^2, \quad \forall w^* \in \mathcal{W}^*.$$

Proof. Letting $w = w^*$ in lemma 4 and using (12), the result holds. □

Noticing that the sub-block operator $\widetilde{\mathcal{G}}$ in (28) is indefinite, thus $\mathcal{G}$ is not necessarily positive definite and Lemma 5 does not imply the contractiveness of $\{w^k - w^*\}$ directly. Therefore, we need to investigate the lower bound of $\|w^k - \widetilde{w}^k\|_{\mathcal{G}}^2$ (or equivalently $\|v^k - \widetilde{v}^k\|_{\widetilde{\mathcal{G}}}^2$) to establish the convergence of LSADMM . For this purpose, the region of τ in (8) is partitioned as follows:

$$\tau \in (0, +\infty) = \left(0, \frac{q(2+\gamma)}{4}\right] \cup \left(\frac{q(2+\gamma)}{4}, q\right) \cup [q, +\infty).$$

Below the convergence of LSADMM are respectively discussed for the above three different regions.

3.2 Convergence can be guaranteed for $\tau \in [q, +\infty)$

In this section, we use a standard way to analyze the global convergence of LSADMM and its sublinear convergence rate for the proximal parameter $\tau \in [q, +\infty)$.

For any $\tau \in [q, +\infty)$, we deduce from $r_j > \sigma\|B_j^* B_j\|$ that, there exists $\xi_j > 0$ such that

$$\tau r_j = \tau\sigma\|B_j^* B_j\| + \xi_j\|B_j^* B_j\|,$$

which implies

$$\|v^k - \widetilde{v}^k\|_{\widetilde{\mathcal{G}}}^2 \geq \|v^k - \widetilde{v}^k\|_{\widehat{\mathcal{G}}}^2$$

with $\widetilde{\mathcal{G}}$ being defined in (28) and

$$\widehat{\mathcal{G}} = \left[\begin{array}{cccc:c} (\tau-1)\sigma B_1^* B_1 & -\sigma B_1^* B_2 & \cdots & -\sigma B_1^* B_q & \\ -\sigma B_2^* B_1 & (\tau-1)\sigma B_2^* B_2 & \cdots & -\sigma B_2^* B_q & \\ \vdots & \vdots & \ddots & \vdots & \\ -\sigma B_q^* B_1 & -\sigma B_q^* B_2 & \cdots & (\tau-1)\sigma B_q^* B_q & \\ \hdashline & & & & 0 \end{array}\right] + \left[\begin{array}{ccc:c} \xi_1 B_1^* B_1 & & & \\ & \ddots & & \\ & & \xi_q B_q^* B_q & \\ \hdashline & & & \frac{2-\gamma}{\sigma}\mathcal{I} \end{array}\right], \tag{32}$$

Similar to the proof of Lemma 3, we have the following result.

Lemma 6 *The operator $\widehat{\mathcal{G}}$ defined in (32) is symmetric positive definite if $\tau \in [q, +\infty), \gamma \in (0, 2)$ and all the operators $B_j (j = 1, 2, \cdots, q)$ have full column rank.*

We have by Lemmas 5-6 that

$$\begin{aligned} \|w^{k+1} - w^*\|_{\mathcal{H}}^2 &\leq \|w^k - w^*\|_{\mathcal{H}}^2 - \|w^k - \widetilde{w}^k\|_{\mathcal{G}}^2 \\ &= \|w^k - w^*\|_{\mathcal{H}}^2 - \|\mathbf{x}^k - \mathbf{x}^{k+1}\|_{\mathcal{Q}_1}^2 - \|v^k - \widetilde{v}^k\|_{\widetilde{\mathcal{G}}}^2 \\ &\leq \|w^k - w^*\|_{\mathcal{H}}^2 - \|\mathbf{x}^k - \mathbf{x}^{k+1}\|_{\mathcal{Q}_1}^2 - \|v^k - \widetilde{v}^k\|_{\widehat{\mathcal{G}}}^2 \\ &= \|w^k - w^*\|_{\mathcal{H}}^2 - \|w^k - \widetilde{w}^k\|_{\overline{\mathcal{G}}}^2, \end{aligned} \tag{33}$$

where

$$\overline{\mathcal{G}} = \begin{pmatrix} \mathcal{Q}_1 & 0 \\ 0 & \widehat{\mathcal{G}} \end{pmatrix}$$

is clearly symmetric positive definite. Hence, the sequence $\{w^k - w^*\}$ is strictly contractive implying the global convergence of LSADMM.

Theorem 1 *Let the sequence $\{\widetilde{w}^k\}$ be generated by LSADMM. Then, for any*

$$\rho > p - 1, \ \tau \in [q, +\infty) \text{ and } \gamma \in (0, 2), \tag{34}$$

there exists a point $w^\infty \in \mathcal{W}^*$ such that $\lim\limits_{k\to\infty} \widetilde{w}^k = w^\infty$.

Proof. Since the operators $\mathcal{H}$ and $\overline{\mathcal{G}}$ are both positive definite, by (33) the sequence $\{\widetilde{w}^k\}$ is uniformly bounded and

$$\lim_{k\to\infty} w^k - \widetilde{w}^k = 0. \tag{35}$$

So, there exists a subsequence $\{\tilde{w}^{k_l}\}$ converging to a point $w^\infty \in \mathcal{W}$. For any fixed $w \in \mathcal{W}$, taking $\{\tilde{w}^{k_l}\}$ in (14) and letting l go to ∞, we have

$$\theta(u) - \theta(u^\infty) + \langle w - w^\infty, \mathcal{F}(w^\infty)\rangle \geq 0,$$

which shows that $w^\infty \in \mathcal{W}^*$ is a solution of VI$(\mathcal{W}, \mathcal{F}, \theta)$ defined in (12). By (33) again, we have

$$\|w^k - w^\infty\|_{\mathcal{H}}^2 \leq \|w^j - w^\infty\|_{\mathcal{H}}^2 \quad \text{for all } k \geq j.$$

Then, it follows from w^∞ being an accumulation point that $\lim\limits_{k\to\infty} \widetilde{w}^k = w^\infty$. □

Next, we establish the worst-case $\mathcal{O}(1/t)$ ergodic convergence rate of LSADMM for the average iterates

$$\mathbf{w}_t := \frac{1}{1+t}\sum_{k=0}^{t} \widetilde{w}^k \quad \text{and} \quad \mathbf{u}_t := \frac{1}{1+t}\sum_{k=0}^{t} \widetilde{u}^k. \tag{36}$$

Theorem 2 *Let $\boldsymbol{w}_t$ and $\boldsymbol{u}_t$ be defined in (36). Then, for any $w \in \mathcal{W}$ we have*

$$\theta(\boldsymbol{u}_t) - \theta(u) + \langle \boldsymbol{w}_t - w, \mathcal{F}(\boldsymbol{w}_t)\rangle \leq \frac{1}{2(1+t)} \left\|w - w^0\right\|_{\mathcal{H}}^2. \tag{37}$$

Proof. By Lemma 6 and the positive definiteness of $\mathcal{Q}_1$, we know $\|w^k - \widetilde{w}^k\|_{\mathcal{G}}^2 > 0$. Hence, it follows from Lemma 4 that

$$\theta(u) - \theta(\widetilde{u}^k) + \left\langle w - \widetilde{w}^k, \mathcal{F}(\widetilde{w}^k) \right\rangle \geq \frac{1}{2}\left(\|w - w^{k+1}\|_{\mathcal{H}}^2 - \|w - w^k\|_{\mathcal{H}}^2 \right), \quad \forall w \in \mathcal{W},$$

which, by summing it over $k = 0, 1, 2, \cdots, t$, implies

$$\frac{1}{1+t}\sum_{k=0}^{t} \theta(\widetilde{u}^k) - \theta(u) + \langle \mathbf{w}_t - w, \mathcal{F}(\mathbf{w}_t) \rangle \leq \frac{1}{2(1+t)} \left\| w - w^0 \right\|_{\mathcal{H}}^2. \tag{38}$$

Since the function $\theta(u)$ is convex, we have

$$\theta(\mathbf{u}_t) \leq \frac{1}{1+t}\sum_{k=0}^{t} \theta(\widetilde{u}^k).$$

Therefore, (37) is true by substituting the above inequality into (38). □

3.3 Convergence can be guaranteed for $\tau \in \left(\frac{q(2+\gamma)}{4}, q\right)$

In this section, we first analyze the convergence of LSADMM for the problem with $q = 1$ by estimating the lower bound of $\|w^k - \widetilde{w}^k\|_{\mathcal{G}}^2$. Then, we show the convergence of LSADMM for the problem with $q \geq 2$ by equivalently estimating the lower bound of $\|v^k - \widetilde{v}^k\|_{\widetilde{\mathcal{G}}}^2$. Although there are small differences during the analysis, the convergence results are the same, and the final optimal lower bounds of τ have the same form.

3.3.1 A single y block ($q = 1$)

For problem (1) with $q = 1$, we have $\mathbf{y} = y_1$, $\mathcal{B} = B_1$, $\mathcal{D}_0 = r_1\mathcal{I} - \sigma B_1^* B_1$ and

$$\widetilde{\mathcal{G}} = \left[\begin{array}{cc} D_1 & 0 \\ 0 & \frac{2-\gamma}{\sigma}\mathcal{I} \end{array} \right] \quad \text{with} \quad D_1 = \tau r_1 \mathcal{I} - \sigma B_1^* B_1.$$

Since the operator $\widetilde{\mathcal{G}}$ is positive-indefinite for $\tau \in (0, 1)$, we try to estimate the lower bound of $\|w^k - \widetilde{w}^k\|_{\mathcal{G}}^2$ for proving the convergence of LSADMM.

Lemma 7 *For problem (1) with $q = 1$, let the sequence $\{\widetilde{w}^k = (\mathbf{x}^{k+1}, y_1^{k+1}, \widetilde{\lambda}^k)\}$ be generated by LSADMM with given $\{w^k = (\mathbf{x}^k, y_1^k, \lambda^k)\}$. Then, for any*

$$\rho > p - 1, \;\; \tau \in \left(\frac{2+\gamma}{4}, 1\right) \quad \textit{with} \quad \gamma \in (0, 2), \tag{39}$$

we have

$$\begin{aligned} \|w^k - \widetilde{w}^k\|_{\mathcal{G}}^2 \quad \geq \quad & \|\mathbf{x}^k - \mathbf{x}^{k+1}\|_{\mathcal{Q}_1}^2 + \frac{2-\gamma}{2\gamma}\left[\tau\|y_1^k - y_1^{k+1}\|_{\mathcal{D}_0}^2 + (1-\tau)\sigma\|B_1(y_1^k - y_1^{k+1})\|^2 \right] \\ & - \frac{2-\gamma}{2\gamma}\left[\tau\|y_1^{k-1} - y_1^k\|_{\mathcal{D}_0}^2 + (1-\tau)\sigma\|B_1(y_1^{k-1} - y_1^k)\|^2 \right] + \tau\|y_1^k - y_1^{k+1}\|_{\mathcal{D}_0}^2 \\ & + (2-\gamma)\delta_0\sigma\left[\|B_1(y_1^k - y_1^{k+1})\|^2 + \|\mathcal{A}\mathbf{x}^{k+1} + B_1 y_1^{k+1} - c\|^2 \right], \end{aligned} \tag{40}$$

where $\delta_0 = \frac{(4-\gamma)(4\tau-\gamma-2)}{2(4+\gamma)}$.

Proof. By the fact

$$\|w^k - \widetilde{w}^k\|_{\mathcal{G}}^2 = \|\mathbf{x}^k - \mathbf{x}^{k+1}\|_{\mathcal{Q}_1}^2 + \|v^k - \widetilde{v}^k\|_{\widetilde{\mathcal{G}}}^2$$

and the similar results as [26, Lemma 3.9] with setting $r = \gamma - 1$ we have

$$\begin{aligned}\|v^k - \widetilde{v}^k\|_{\widetilde{\mathcal{G}}}^2 \;\geq\; & \frac{2-\gamma}{2\gamma}\left[\tau\|y_1^k - y_1^{k+1}\|_{\mathcal{D}_0}^2 + (1-\tau)\sigma\|B_1(y_1^k - y_1^{k+1})\|^2\right] \\ & -\frac{2-\gamma}{2\gamma}\left[\tau\|y_1^{k-1} - y_1^{k}\|_{\mathcal{D}_0}^2 + (1-\tau)\sigma\|B_1(y_1^{k-1} - y_1^{k})\|^2\right] \\ & +\tau\|y_1^k - y_1^{k+1}\|_{\mathcal{D}_0}^2 + (2-\gamma)\delta_0\sigma\left[\|B_1(y_1^k - y_1^{k+1})\|^2 + \|\mathcal{A}\mathbf{x}^{k+1} + B_1 y_1^{k+1} - c\|^2\right],\end{aligned}$$

which implies the result. □

Clearly, we have by (39) that $\delta_0 > 0$, $\mathcal{D}_0$ and $\mathcal{Q}_1$ are positive definite, so the convergence of LSADMM can be established as follows.

Theorem 3 *For problem (1) with $q = 1$, the sequence $\{\widetilde{w}^k\}$ generated by LSADMM converges to a point $w^\infty \in \mathcal{W}^*$ for any (τ, γ, ρ) satisfying (39).*

Proof. First of all, it holds by Lemmas 5 and 7 that

$$\begin{aligned} & \|\mathbf{x}^k - \mathbf{x}^{k+1}\|_{\mathcal{Q}_1}^2 + \tau\|y_1^k - y_1^{k+1}\|_{\mathcal{D}_0}^2 + (2-\gamma)\delta_0\sigma\left[\|B_1(y_1^k - y_1^{k+1})\|^2 + \|\mathcal{A}\mathbf{x}^{k+1} + B_1 y_1^{k+1} - c\|^2\right] \\ \leq\; & \|w^k - w^*\|_{\mathcal{H}}^2 + \frac{2-\gamma}{2\gamma}\left[\tau\|y_1^{k-1} - y_1^{k}\|_{\mathcal{D}_0}^2 + (1-\tau)\sigma\|B_1(y_1^{k-1} - y_1^{k})\|^2\right] \\ & -\|w^{k+1} - w^*\|_{\mathcal{H}}^2 - \frac{2-\gamma}{2\gamma}\left[\tau\|y_1^k - y_1^{k+1}\|_{\mathcal{D}_0}^2 + (1-\tau)\sigma\|B_1(y_1^k - y_1^{k+1})\|^2\right].\end{aligned}$$

Summing it from $k = 1$ to ∞ gives

$$\begin{aligned} & \sum_{k=1}^{\infty}\left(\|\mathbf{x}^k - \mathbf{x}^{k+1}\|_{\mathcal{Q}_1}^2 + \tau\|y_1^k - y_1^{k+1}\|_{\mathcal{D}_0}^2 + (2-\gamma)\delta_0\sigma\left[\|B_1(y_1^k - y_1^{k+1})\|^2 + \|\mathcal{A}\mathbf{x}^{k+1} + B_1 y_1^{k+1} - c\|^2\right]\right) \\ \leq\; & \|w^1 - w^*\|_{\mathcal{H}}^2 + \tfrac{2-\gamma}{2\gamma}\left[\tau\|y_1^0 - y_1^1\|_{\mathcal{D}_0}^2 + (1-\tau)\sigma\|B_1(y_1^0 - y_1^1)\|^2\right],\end{aligned}$$

which, by the positive definiteness of $\mathcal{D}_0, \mathcal{Q}_1$ and the full column assumption on B_1, shows

$$\lim_{k\to\infty} \mathbf{x}^k - \mathbf{x}^{k+1} = 0, \quad \lim_{k\to\infty} y_1^k - y_1^{k+1} = 0 \quad \text{and} \tag{41}$$

$$\lim_{k\to\infty} \mathcal{A}\mathbf{x}^{k+1} + B_1 y_1^{k+1} - c = 0 \Leftrightarrow \lim_{k\to\infty} \lambda^k - \tilde{\lambda}^k = 0. \tag{42}$$

Equivalently,

$$\lim_{k\to\infty} w^k - \widetilde{w}^k = 0. \tag{43}$$

Since $\theta(u)$ is proper and $\mathcal{F}(w)$ is continuous, for any fixed $w \in \mathcal{W}$, by taking $\widetilde{w}^{k_j}$ in (14) and letting j go to ∞ we have

$$\theta(u) - \theta(u^\infty) + \langle w - w^\infty, \mathcal{F}(w^\infty)\rangle \geq 0.$$

Thus, $w^\infty \in \mathcal{W}^*$ is a solution point of VI$(\mathcal{W}, \mathcal{F}, \theta)$ defined in (12).

Noticing that the following inequality

$$
\begin{aligned}
&\|w^{k+1}-w^*\|_{\mathcal{H}}^2\\
&\le \|w^k-w^*\|_{\mathcal{H}}^2+\tfrac{2-\gamma}{2\gamma}\left(\tau\|y_1^k-y_1^{k-1}\|_{\mathcal{D}_0}^2+(1-\tau)\sigma\|B_1(y_1^k-y_1^{k-1})\|^2\right)\\
&\le \|w^1-w^*\|_{\mathcal{H}}^2+\tfrac{2-\gamma}{2\gamma}\left(\tau\|y_1^0-y_1^1\|_{\mathcal{D}_0}^2+(1-\tau)\sigma\|B_1(y_1^0-y_1^1)\|^2\right).
\end{aligned}
\tag{44}
$$

holds for any $w^*\in\mathcal{W}^*$. So, for all $l>k_j$ we have

$$
\begin{aligned}
&\|w^l-w^\infty\|_{\mathcal{H}}^2+\tfrac{2-\gamma}{2\gamma}\left(\tau\|y_1^l-y_1^{l-1}\|_{\mathcal{D}_0}^2+(1-\tau)\sigma\|B_1(y_1^l-y_1^{l-1})\|^2\right)\\
\le\ &\|w^{k_j}-w^\infty\|_{\mathcal{H}}^2+\tfrac{2-\gamma}{2\gamma}\left(\tau\|y_1^{k_j}-y_1^{k_j-1}\|_{\mathcal{D}_0}^2+(1-\tau)\sigma\|B_1(y_1^{k_j}-y_1^{k_j-1})\|^2\right),
\end{aligned}
$$

which together with (41)-(43) and the positive definiteness of $\mathcal{H}$ shows $\lim\limits_{l\to\infty} w^l=w^\infty$. Finally, by (44) and (21), we also have $\lim\limits_{l\to\infty} w^l=\lim\limits_{l\to\infty}\widetilde{w}^l$ which illustrates that the whole sequence $\{\widetilde{w}^k\}$ converges to the solution $w^\infty\in\mathcal{W}^*$. □

Now, we establish the sublinear ergodic convergence rate of LSADMM for the average iterates

$$
\overline{\mathbf{w}}_t:=\frac{1}{t}\sum_{k=1}^{t}\widetilde{w}^k \quad\text{and}\quad \overline{\mathbf{u}}_t:=\frac{1}{t}\sum_{k=1}^{t}\widetilde{u}^k. \tag{45}
$$

Theorem 4 *Let $\overline{\mathbf{w}}_t$ and $\overline{\mathbf{u}}_t$ be defined in (45). Then, for any $w\in\mathcal{W}$ we have*

$$
\theta(\overline{\boldsymbol{u}}_t)-\theta(u)+\langle\overline{\boldsymbol{w}}_t-w,\mathcal{F}(\overline{\boldsymbol{w}}_t)\rangle\le\frac{1}{2t}\left\{\left\|w-w^1\right\|_{\mathcal{H}}^2+\frac{2-\gamma}{\gamma}\left[\tau\|y_1^0-y_1^1\|_{\mathcal{D}_0}^2+(1-\tau)\sigma\|B_1(y_1^0-y_1^1)\|^2\right]\right\}.
$$

Proof. Substituting the inequality (40) into Lemma 4, we have

$$
\begin{aligned}
\theta(u)-\theta(\widetilde{u}^k)+\left\langle w-\widetilde{w}^k,\mathcal{F}(\widetilde{w}^k)\right\rangle\ \ \ge\ \ &\frac{1}{2}\left(\|w-w^{k+1}\|_{\mathcal{H}}^2-\|w-w^k\|_{\mathcal{H}}^2\right)\\
&+\frac{2-\gamma}{2\gamma}\left[\tau\|y_1^k-y_1^{k+1}\|_{\mathcal{D}_0}^2+(1-\tau)\sigma\|B_1(y_1^k-y_1^{k+1})\|^2\right]\\
&-\frac{2-\gamma}{2\gamma}\left[\tau\|y_1^{k-1}-y_1^k\|_{\mathcal{D}_0}^2+(1-\tau)\sigma\|B_1(y_1^{k-1}-y_1^k)\|^2\right],
\end{aligned}
$$

which, by summing it over $k=1,2,\cdots,t$, implies

$$
\begin{aligned}
&\frac{1}{t}\sum_{k=1}^{t}\theta(\widetilde{u}^k)-\theta(u)+\langle\overline{\mathbf{w}}_t-w,\mathcal{F}(\overline{\mathbf{w}}_t)\rangle\\
\le\ \ &\frac{1}{2t}\left\{\left\|w-w^1\right\|_{\mathcal{H}}^2+\frac{2-\gamma}{\gamma}\left[\tau\|y_1^0-y_1^1\|_{\mathcal{D}_0}^2+(1-\tau)\sigma\|B_1(y_1^0-y_1^1)\|^2\right]\right\}.
\end{aligned}
$$

Using the above inequality and the convexity of the function $\theta(u)$, the proof is completed. □

Theorems 3-4 show the global sublinear convergence of LSADMM for problem (1) with $q=1$, when $\rho>p-1$ and $\tau\in(\frac{\gamma+2}{4},1)$ with $\gamma\in(0,2)$. Especially, the indefinite proximal generalized ADMM (IPG-ADMM, [26]) is a special case of LSADMM with $p=1$, implying our algorithm is more general and flexible. Furthermore, we can deduce by the counter-example in [26] that $\frac{2+\gamma}{4}$ is the optimal lower bound of the proximal parameter τ.

3.3.2 Multiple y blocks ($q \geq 2$)

For problem (1) with $q \geq 2$ and any $\tau \in (0, q)$, since there exists $r_j > \sigma\|B_j^* B_j\|$ such that $\tau r_j < q\sigma\|B_j^* B_j\|$, the operator $\widetilde{\mathcal{G}}$ is also positive-indefinite. Hence, in what follows we aim at obtaining a lower bound of τ to ensure the convergence of LSADMM and to save the computational time.

The following result is basic and standard for estimating the lower bound of $\|v^k - \widetilde{v}^k\|_{\widetilde{\mathcal{G}}}^2$.

Lemma 8 *For problem (1) with $q \geq 2$, let the sequence $\{\widetilde{w}^k = (\mathbf{x}^{k+1}, \widetilde{v}^k)\}$ be generated by LSADMM given $\{w^k = (\mathbf{x}^k, v^k)\}$. Then, we have*

$$\begin{aligned} \|v^k - \widetilde{v}^k\|_{\widetilde{\mathcal{G}}}^2 &= \tau\|\mathbf{y}^k - \mathbf{y}^{k+1}\|_{\mathcal{D}_0}^2 + \frac{2-\gamma-\gamma^2}{\gamma^2}\sigma\|\mathcal{B}(\mathbf{y}^k - \mathbf{y}^{k+1})\|^2 + \tau\sigma\|\mathbf{y}^k - \mathbf{y}^{k+1}\|_{\widetilde{B}}^2 \\ &\quad + \frac{2-\gamma}{\sigma\gamma^2}\|\lambda^k - \lambda^{k+1}\|^2 + \frac{2(2-\gamma)}{\gamma^2}\left\langle \lambda^k - \lambda^{k+1}, \mathcal{B}(\mathbf{y}^k - \mathbf{y}^{k+1})\right\rangle. \end{aligned} \tag{46}$$

Proof. In view of the definitions of $\mathcal{D}_0$, $\mathcal{G}$ and $\tilde{v}$, we have

$$\|v^k - \widetilde{v}^k\|_{\widetilde{\mathcal{G}}}^2 = \tau\|\mathbf{y}^k - \mathbf{y}^{k+1}\|_{\mathcal{D}_0}^2 - \sigma\|\mathcal{B}(\mathbf{y}^k - \mathbf{y}^{k+1})\|^2 + \tau\sigma\|\mathbf{y}^k - \mathbf{y}^{k+1}\|_{\widetilde{B}}^2 + \tfrac{(2-\gamma)}{\sigma}\|\lambda^k - \tilde{\lambda}^k\|^2. \tag{47}$$

Meanwhile, it follows from (23) and $\gamma = \alpha + \beta$ that

$$\lambda^k - \tilde{\lambda}^k = \frac{1}{\gamma}(\lambda^k - \lambda^{k+1}) + \frac{\sigma}{\gamma}\mathcal{B}(\mathbf{y}^k - \mathbf{y}^{k+1}),$$

which leads to

$$\begin{aligned} \|\lambda^k - \tilde{\lambda}^k\|^2 &= \frac{1}{\gamma^2}\|\lambda^k - \lambda^{k+1}\|^2 + \frac{\sigma^2}{\gamma^2}\|\mathcal{B}(\mathbf{y}^k - \mathbf{y}^{k+1})\|^2 \\ &\quad + 2\frac{\sigma}{\gamma^2}\left\langle \lambda^k - \lambda^{k+1}, \mathcal{B}(\mathbf{y}^k - \mathbf{y}^{k+1})\right\rangle. \end{aligned} \tag{48}$$

Substituting (48) into (47), the result holds. □

Lemma 9 *The sequence $\{w^{k+1}\}$ be generated by LSADMM satisfies*

$$\begin{aligned} \left\langle \lambda^k - \lambda^{k+1}, \mathcal{B}(\mathbf{y}^k - \mathbf{y}^{k+1})\right\rangle &\geq \frac{\tau}{2}\left(\|\mathbf{y}^k - \mathbf{y}^{k+1}\|_{\mathcal{D}_0}^2 - \|\mathbf{y}^{k-1} - \mathbf{y}^k\|_{\mathcal{D}_0}^2\right) \\ &\quad + \frac{(1-\tau)}{2}\sigma\left(\|\mathbf{y}^k - \mathbf{y}^{k+1}\|_{\widetilde{B}}^2 - \|\mathbf{y}^{k-1} - \mathbf{y}^k\|_{\widetilde{B}}^2\right) \\ &\quad - 2(1-\tau)\sigma\|\mathbf{y}^k - \mathbf{y}^{k+1}\|_{\widetilde{B}}^2. \end{aligned} \tag{49}$$

Proof. By the first-order optimality condition of the y_j-subproblems in Step 2, we have

$$g_j(y_j) - g(y_j^{k+1}) + \left\langle y_j - y_j^{k+1}, R_{y_j}\right\rangle \geq 0 \tag{50}$$

with

$$\begin{aligned} R_{y_j} &= -B_j^*\left[\tilde{\lambda}^k + (\gamma-1)(\tilde{\lambda}^k - \lambda^k)\right] + \tau r_j(y_j^{k+1} - y_j^k) \\ &= -B_j^*\left[\lambda^k + \gamma(\tilde{\lambda}^k - \lambda^k)\right] + \tau r_j(y_j^{k+1} - y_j^k) \\ &= -B_j^*\lambda^{k+1} + \sigma B_j^*\mathcal{B}(\mathbf{y}^k - \mathbf{y}^{k+1}) + \tau r_j(y_j^{k+1} - y_j^k). \end{aligned} \tag{51}$$

Combining (50) and (51) with $y_j = y_j^k$, we have

$$g_j(y_j^k) - g(y_j^{k+1}) + \left\langle y_j^k - y_j^{k+1}, -B_j^*\lambda^{k+1} + \sigma B_j^*\mathcal{B}(\mathbf{y}^k - \mathbf{y}^{k+1}) + \tau r_j(y_j^{k+1} - y_j^k)\right\rangle \geq 0. \tag{52}$$

At the same time, the inequality (50) also implies

$$g_j(y_j) - g(y_j^k) + \left\langle y_j - y_j^k, -B_j^*\lambda^k + \sigma B_j^*\mathcal{B}(\mathbf{y}^{k-1} - \mathbf{y}^k) + \tau r_j(y_j^k - y_j^{k-1})\right\rangle \geq 0,$$

which by setting $y_j = y_j^{k+1}$ leads to

$$g_j(y_j^{k+1}) - g(y_j^k) + \left\langle y_j^{k+1} - y_j^k, -B_j^*\lambda^k + \sigma B_j^*\mathcal{B}(\mathbf{y}^{k-1} - \mathbf{y}^k) + \tau r_j(y_j^k - y_j^{k-1})\right\rangle \geq 0, \tag{53}$$

Then, adding (53) to (52) gives

$$\left\langle y_j^k - y_j^{k+1}, \widetilde{R}_{y_j}\right\rangle \geq 0, \tag{54}$$

where

$$\widetilde{R}_{y_j} = -B_j^*(\lambda^{k+1} - \lambda^k) - \sigma B_j^*\mathcal{B}\left[(\mathbf{y}^{k+1} - \mathbf{y}^k) - (\mathbf{y}^k - \mathbf{y}^{k-1})\right] + \tau r_j\left[(y_j^{k+1} - y_j^k) - (y_j^k - y_j^{k-1})\right].$$

By the notations in (3), (8) and (10), we can rewrite (54) from $j = 1, 2, \cdots, q$ as the following compact variational form

$$\langle \mathbf{y}^k - \mathbf{y}^{k+1}, \ -\mathcal{B}^*(\lambda^{k+1} - \lambda^k) + \mathcal{D}\left[(\mathbf{y}^{k+1} - \mathbf{y}^k) - (\mathbf{y}^k - \mathbf{y}^{k-1})\right]\rangle \geq 0.$$

Therefore, by the relationship $\mathcal{D} = \tau\mathcal{D}_0 - (1-\tau)\sigma\widetilde{B}$ we get

$$\begin{aligned}
& 2\left\langle \lambda^k - \lambda^{k+1}, \ \mathcal{B}(\mathbf{y}^k - \mathbf{y}^{k+1})\right\rangle \\
\geq \ & 2\tau\|\mathbf{y}^k - \mathbf{y}^{k+1}\|_{\mathcal{D}_0}^2 - 2\tau\langle \mathbf{y}^k - \mathbf{y}^{k+1}, \ \mathcal{D}_0(\mathbf{y}^k - \mathbf{y}^{k-1})\rangle \\
& -2(1-\tau)\sigma\|\mathbf{y}^k - \mathbf{y}^{k+1}\|_{\widetilde{B}}^2 + 2(1-\tau)\sigma\langle \mathbf{y}^k - \mathbf{y}^{k+1}, \ \widetilde{B}(\mathbf{y}^k - \mathbf{y}^{k-1})\rangle \\
\geq \ & \tau\|\mathbf{y}^k - \mathbf{y}^{k+1}\|_{\mathcal{D}_0}^2 - \tau\|\mathbf{y}^k - \mathbf{y}^{k-1}\|_{\mathcal{D}_0}^2 - 3(1-\tau)\sigma\|\mathbf{y}^k - \mathbf{y}^{k+1}\|_{\widetilde{B}}^2 \\
& -(1-\tau)\sigma\|\mathbf{y}^k - \mathbf{y}^{k-1}\|_{\widetilde{B}}^2,
\end{aligned}$$

where the above last inequality uses the Cauchy-Schwarz inequality. □

Actually, for any $\delta \in (0,1)$, by the Cauchy-Schwarz inequality we still obtain

$$\langle \lambda^k - \lambda^{k+1}, \mathcal{B}(\mathbf{y}^k - \mathbf{y}^{k+1})\rangle \ \geq \ -\frac{1}{4(1-\delta)}\sigma\|\mathcal{B}(\mathbf{y}^k - \mathbf{y}^{k+1})\|^2 - (1-\delta)\frac{1}{\sigma}\|\lambda^k - \lambda^{k+1}\|^2. \tag{55}$$

Based on the above preparations, we now show a lemma to estimate the lower bound of $\|v^k - \widetilde{v}^k\|_{\widetilde{\mathcal{G}}}^2$ as the following.

Lemma 10 *For problem (1) with $q \geq 2$, let the sequence $\{\widetilde{w}^k\}$ be generated by LSADMM with given $\{w^k\}$. Then, for any*

$$\rho > p - 1, \quad \tau \in \left(\frac{q(\gamma+2)}{4}, q\right) \quad \textit{with} \quad \gamma \in (0,2), \tag{56}$$

we have

$$\begin{aligned}\|v^k-\widetilde{v}^k\|^2_{\widetilde{\mathcal{G}}} \ \geq\ & \tau\|\mathbf{y}^k-\mathbf{y}^{k+1}\|^2_{\mathcal{D}_0}+\frac{\delta(2-\gamma)}{\sigma\gamma^2}\|\lambda^k-\lambda^{k+1}\|^2\\ &+\frac{2-\gamma}{2\gamma^2}\left[\tau\|\mathbf{y}^k-\mathbf{y}^{k+1}\|^2_{\mathcal{D}_0}+(1-\tau)\sigma\|\mathcal{B}(\mathbf{y}^k-\mathbf{y}^{k+1})\|^2\right]\\ &-\frac{2-\gamma}{2\gamma^2}\left[\tau\|\mathbf{y}^{k-1}-\mathbf{y}^k\|^2_{\mathcal{D}_0}+(1-\tau)\sigma\|\mathcal{B}(\mathbf{y}^{k-1}-\mathbf{y}^k)\|^2\right],\end{aligned}$$

where

$$\delta=\frac{\gamma^2-2\gamma-7+\frac{8}{q}}{\gamma^2-2\gamma-8+\frac{8}{q}}. \tag{57}$$

Proof. Since $\gamma\in(0,2)$ and $q\geq 2$, we deduce

$$\delta=1+\frac{1}{(\gamma-1)^2-9+\frac{8}{q}}\in\left(1+\frac{1}{(\frac{1}{q}-1)8},1+\frac{1}{(\frac{1}{q}-1)8-1}\right)\subset\left(\frac{3}{4},\frac{8}{9}\right)\subset(0,1).$$

Substituting both (55) and (49) into (46) gives

$$\begin{aligned}\|v^k-\widetilde{v}^k\|^2_{\widetilde{\mathcal{G}}} \ \geq\ & \tau\|\mathbf{y}^k-\mathbf{y}^{k+1}\|^2_{\mathcal{D}_0}+\frac{\delta(2-\gamma)}{\sigma\gamma^2}\|\lambda^k-\lambda^{k+1}\|^2\\ &+\frac{2-\gamma}{2\gamma^2}\left[\tau\|\mathbf{y}^k-\mathbf{y}^{k+1}\|^2_{\mathcal{D}_0}+(1-\tau)\sigma\|\mathcal{B}(\mathbf{y}^k-\mathbf{y}^{k+1})\|^2\right]\\ &-\frac{2-\gamma}{2\gamma^2}\left[\tau\|\mathbf{y}^{k-1}-\mathbf{y}^k\|^2_{\mathcal{D}_0}+(1-\tau)\sigma\|\mathcal{B}(\mathbf{y}^{k-1}-\mathbf{y}^k)\|^2\right]\\ &+\varphi(\tau,\gamma),\end{aligned}$$

where

$$\varphi(\tau,\gamma)=\phi_1\sigma\|\mathcal{B}(\mathbf{y}^k-\mathbf{y}^{k+1})\|^2+\phi_2\sigma\|\mathbf{y}^k-\mathbf{y}^{k+1}\|^2_{\widetilde{B}},$$

and

$$\phi_1=\frac{2-\gamma-\gamma^2}{\gamma^2}-\frac{2-\gamma}{4\gamma^2(1-\delta)},\quad \phi_2=\tau-\frac{2(2-\gamma)(1-\tau)}{\gamma^2}.$$

Obviously, our result can be achieved if $\varphi(\tau,\gamma)\geq 0$, which motivates us to explore the lower bound of $\varphi(\tau,\gamma)$.

By the operator $\widetilde{B}$ in (10), we deduce $\varphi(\tau,\gamma)=\sigma\|y^k-y^{k+1}\|^2_{\widehat{B}}$ with

$$\widehat{B}=\begin{bmatrix}(\phi_1+\phi_2)B_1^*B_1 & \phi_1B_1^*B_2 & \cdots & \phi_1B_1^*B_q\\ \phi_1B_2^*B_1 & (\phi_1+\phi_2)B_2^*B_2 & \cdots & \phi_1B_2^*B_q\\ \vdots & \vdots & \ddots & \vdots\\ \phi_1B_q^*B_1 & \phi_1B_q^*B_2 & \cdots & (\phi_1+\phi_2)B_q^*B_q\end{bmatrix}.$$

Since $\tau>\frac{q(\gamma+2)}{4}$ and

$$\widehat{B}=\begin{bmatrix}B_1 & & & \\ & B_2 & & \\ & & \ddots & \\ & & & B_q\end{bmatrix}^*[\phi_2\mathcal{I}+\phi_1\mathcal{E}\mathcal{E}^*]\begin{bmatrix}B_1 & & & \\ & B_2 & & \\ & & \ddots & \\ & & & B_q\end{bmatrix},$$

we derive

$$\begin{aligned}\phi_2 - q\phi_1 &= \tau - \frac{2(2-\gamma)(1-\tau)}{\gamma^2} - q\left(\frac{2-\gamma-\gamma^2}{\gamma^2} - \frac{2-\gamma}{4\gamma^2(1-\delta)}\right)\\ &= \frac{1}{\gamma^2}\left(\tau(\gamma^2-2\gamma+4) - 4 + 2\gamma - q(\frac{2-\gamma}{4(1-\delta)} - 2 + \gamma + \gamma^2)\right)\\ &> \frac{q}{\gamma^2}\left[\frac{(\gamma+2)(\gamma^2-2\gamma+4)}{4} - \left(\frac{2-\gamma}{4(1-\delta)} - 2 + \gamma + \gamma^2\right) + \frac{2\gamma-4}{q}\right].\end{aligned}$$

Substituting the definition of δ in (57) into the inequality, we have

$$\begin{aligned}\phi_2 - q\phi_1 &> \frac{q}{\gamma^2}\left[\frac{(\gamma+2)(\gamma^2-2\gamma+4)}{4} - \left(\frac{(\gamma-2)(\gamma^2-2\gamma-8+\frac{8}{q})}{4} - 2 + \gamma + \gamma^2\right) + \frac{2\gamma-4}{q}\right]\\ &= \frac{q}{\gamma^2}\left(\frac{4\gamma^2+4\gamma-\frac{8\gamma-16}{q}-8}{4} + 2 - \gamma - \gamma^2 + \frac{2\gamma-4}{q}\right)\\ &= 0,\end{aligned}$$

which means the operator $\phi_2\mathcal{I} + \phi_1\mathcal{E}\mathcal{E}^*$ is strictly diagonally dominant and thus positive definite. Therefore, by the full column rank assumption on B_j, the operator $\widehat{B}$ is positive definite and then $\varphi(\tau,\gamma) \geq 0$. □

In a similar proof as Theorems 3-4, the following convergence result holds, whose proof is omitted here for the sake of conciseness.

Theorem 5 *For problem (1) with $q \geq 2$, the sequence $\{\widetilde{w}^k\}$ generated by LSADMM converges to a point $w^\infty \in \mathcal{W}^*$ for any (τ,γ,ρ) satisfying (56). Moreover, the iterates $\overline{\boldsymbol{w}}_t$ and $\overline{\boldsymbol{u}}_t$ defined in (45) satisfy*

$$\theta(\overline{\boldsymbol{u}}_t) - \theta(u) + \langle\overline{\boldsymbol{w}}_t - w, \mathcal{F}(\overline{\boldsymbol{w}}_t)\rangle \leq \frac{1}{2t}\left\{\left\|w - w^1\right\|_{\mathcal{H}}^2 + \frac{2-\gamma}{\gamma^2}\left[\tau\|\mathbf{y}^0 - \mathbf{y}^1\|_{\mathcal{D}_0}^2 + (1-\tau)\sigma\|\mathcal{B}(\mathbf{y}^0 - \mathbf{y}^1)\|^2\right]\right\}.$$

At the end of this section, we give two remarks to understand our algorithm and the involved parameters:

Remark 1 *By the whole analysis, the convergence of LSADMM depends mainly on $\gamma = \alpha + \beta \in (0,2)$ and $\tau \in \left(\frac{q(\gamma+2)}{4}, +\infty\right)$. Note that if we take $\beta = 0$, then $\alpha \in (0,2)$ and Step 2 is simplified as*

$$\begin{cases} \text{For } \ j = 1, \cdots, q, \\ \quad y_j^{k+1} = \arg\min_{y_j \in \mathcal{Y}_j} \left\{ g_j(y_j) + \frac{\tau r_j}{2}\left\| y_j - \frac{\tau r_j y_j^k + B_j^*(\lambda^{k+\frac{1}{2}})}{\tau r_j}\right\|^2\right\}, \\ \lambda^{k+1} = \lambda^{k+\frac{1}{2}} - \sigma\mathcal{B}\left(\mathbf{y}^{k+1} - \mathbf{y}^k\right). \end{cases}$$

A larger value of τ would lead to slow convergence for any algorithm with proximal term. Thus, we establish the optimal lower bound of τ as $\frac{q(\gamma+2)}{4}$ to save the computational time in the following subsection.

Remark 2 *Noticing that if $p = 1$, then we can set $\rho = 0$. In this case, we know $\mathcal{Q}_1 = 0$ and the sequence $\{v^k\}$ converges by using Lemma 5 and Lemma 10. Then, with the linear operator A_1 having full column rank, we can derive the convergence of $\{x_1^k\}$. Besides, by the structure of LSADMM and its convergence analysis, one can exchange the design idea for the two grouped variables, and the convergence can be also guaranteed but the convergence region for the parameter maybe need further investigations.*

3.4 Convergence can not be guaranteed for $\tau \in \left(0, \frac{q(\gamma+2)}{4}\right]$

In this section, we show that the convergence of LSADMM can not be guaranteed for any $\tau \in (0, \frac{q(\gamma+2)}{4}]$ by a simple counter-example, in other words, $\frac{q(\gamma+2)}{4}$ is the optimal lower bound of the proximal parameter τ.

Let us see the following example:

$$\begin{aligned} \min \quad & 0 \cdot x + \sum_{j=1}^{q} 0 \cdot y_j \\ s.t. \quad & 0 \cdot x + \sum_{j=1}^{q} y_j = 0, \\ & x \in \{0\},\ y_j \in \mathbb{R},\ j = 1, \cdots, q. \end{aligned} \tag{58}$$

For convenience, we set $\sigma = 1$ and then the augmented Lagrangian function of problem (58) can be simplified as

$$\mathcal{L}(x, \mathbf{y}, \lambda) = -\langle \lambda,\ \sum_{j=1}^{q} y_j \rangle + \frac{1}{2} \| \sum_{j=1}^{q} y_j \|^2.$$

According to Remark 1, the updating scheme of LSADMM with $\beta = 0$ for problem (58) is

$$\begin{cases} x^{k+1} = \arg \min_{x \in \{0\}} \mathcal{L}(x, \mathbf{y}^k, \lambda^k) + \frac{\varrho}{2} \|0 \cdot (x - x^k)\|^2, \\ \lambda^{k+\frac{1}{2}} = \lambda^k - \alpha \sum_{j=1}^{q} y_j^k, \\ y_j^{k+1} = y_j^k + \frac{1}{\tau r_j} \lambda^{k+\frac{1}{2}} \quad \text{for all } j = 1, \cdots, q, \\ \lambda^{k+1} = \lambda^{k+\frac{1}{2}} - \sum_{j=1}^{q} (y_j^{k+1} - y_j^k), \end{cases}$$

where $\gamma \equiv \alpha \in (0, 2)$. By simple calculations, the above scheme can be rewritten as

$$\begin{cases} x^{k+1} = 0, \\ y_j^{k+1} = (1 - \frac{\alpha}{\tau r_j}) y_j^k - \alpha \sum_{i=1, i \neq j}^{q} \frac{1}{\tau r_i} y_i^k + \frac{1}{\tau r_j} \lambda^k, j = 1, \cdots, q, \\ \lambda^{k+1} = \varrho(-\alpha \sum_{j=1}^{q} y_j^k + \lambda^k), \end{cases} \tag{59}$$

where $\varrho = 1 - \sum_{j=1}^{q} \frac{1}{\tau r_j}$.

Note that from $x^k \equiv 0$ for any $k \geq 1$, we only need to study the iterative sequence $\{v^k = (\mathbf{y}^k, \lambda^k)\}$. Since $r_j > \sigma = 1$ for any $i = 1, \cdots, q$, we can set $r = r_1 = r_2 = \cdots = r_q$ to simplify our discussions. We can derive from (59) that

$$v^{k+1} = Mv^k,$$

where

$$\begin{aligned} M &= \begin{bmatrix} 1-\frac{\alpha}{\tau r} & -\frac{\alpha}{\tau r} & \cdots & -\frac{\alpha}{\tau r} & \frac{1}{\tau r} \\ -\frac{\alpha}{\tau r} & 1-\frac{\alpha}{\tau r} & \cdots & -\frac{\alpha}{\tau r} & \frac{1}{\tau r} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ -\frac{\alpha}{\tau r} & -\frac{\alpha}{\tau r} & \cdots & 1-\frac{\alpha}{\tau r} & \frac{1}{\tau r} \\ -\alpha\varrho & -\alpha\varrho & \cdots & -\alpha\varrho & \varrho \end{bmatrix} \\ &= \frac{1}{\tau r}\begin{bmatrix} \tau r-\alpha & -\alpha & \cdots & -\alpha & 1 \\ -\alpha & \tau r-\alpha & \cdots & -\alpha & 1 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ -\alpha & -\alpha & \cdots & \tau r-\alpha & 1 \\ -\alpha(\tau r-q) & -\alpha(\tau r-q) & \cdots & -\alpha(\tau r-q) & \tau r-q \end{bmatrix}. \end{aligned}$$

For any $\alpha \in (0, 2)$ and given $\tau \in \left(0, \frac{q(\alpha+2)}{4}\right]$, there exists a $r = \frac{1}{2} + \frac{q(\alpha+2)}{8\tau}$ such that

$$r > 1 \quad \text{and} \quad \phi := \tau r \leq \frac{q(\alpha+2)}{4}.$$

Then, we have $\phi \in \left(0, \frac{q(\alpha+2)}{4}\right]$ and the matrix M can be rewritten as

$$M = \frac{1}{\phi}\begin{bmatrix} \phi-\alpha & -\alpha & \cdots & -\alpha & 1 \\ -\alpha & \phi-\alpha & \cdots & -\alpha & 1 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ -\alpha(\phi-q) & -\alpha(\phi-q) & \cdots & -\alpha(\phi-q) & \phi-q \end{bmatrix}, \tag{60}$$

whose $q+1$ eigenvalues are

$$\begin{aligned} f_1(\alpha,\phi) &= f_2(\alpha,\phi) = \cdots = f_{q-1}(\alpha,\phi) = 1, \\ f_q(\alpha,\phi) &= \frac{\phi - \frac{q(\alpha+1)}{2} + \sqrt{\left[\phi - \frac{q(\alpha+1)}{2}\right]^2 + \phi(q-\phi)}}{\phi}, \\ f_{q+1}(\alpha,\phi) &= \frac{\phi - \frac{q(\alpha+1)}{2} - \sqrt{\left[\phi - \frac{q(\alpha+1)}{2}\right]^2 + \phi(q-\phi)}}{\phi}. \end{aligned}$$

Notice that

$$\begin{aligned} f_{q+1}(\alpha, \frac{q(\alpha+2)}{4}) &= \frac{-q\alpha - \sqrt{(q\alpha)^2 - q^2(\alpha+2)(\alpha-2)}}{q(\alpha+2)} \\ &= -1 \end{aligned}$$

and the following

$$\frac{\partial f_{q+1}(\alpha,\phi)}{\partial \phi} = \frac{q(\alpha+1)}{2\phi^2} + \frac{\left[\frac{q(\alpha+1)}{2} - \phi\right]\frac{q(\alpha+1)}{\phi^2} + \frac{q}{\phi}}{2\sqrt{\left[\phi - \frac{q(\alpha+1)}{2}\right]^2 + \phi(q-\phi)}} > 0$$

holds for any fixed $\alpha \in (0,2)$ and $\phi \in \left(0, \frac{q(\alpha+2)}{4}\right]$. So, we have $f_{q+1}(\alpha,\phi) \leq -1$.

By the above analysis, for any $\tau \in (0, \frac{q(\alpha+2)}{4}]$, the matrix M defined in (60) has eigenvalues less than or equal to -1. Hence, the iterative scheme (59) is not necessarily convergent.

4 Full Jacobian Splitting of ALM

From the process of Algorithm 1, it is suitable for the problem (1) with $p = 0$ or $q = 0$, though the convergence is discussed only for the case with $p, q \geq 1$ in this paper. Actually, for the problem with $q = 0$, Algorithm 1 is equivalent to the method introduced in [9], which is a proximally regularized full Jacobian splitting version of ALM. Thus, in this section, we focus on a special case of Algorithm 1 with $p = 0$.

Note from Remark 1 that, Algorithm 1 for the problem (1) with $p = 0$ can be simplified as:

Algorithm 2 (LSADMM for solving problem (1) with $p = 0$)

Step 0. Initialize the parameters $\sigma > 0, \tau > 0, \alpha \in (0,2)$ and starting point $w^0 \in \mathcal{W}$. Set $k = 0$.

Step 1. For $j = 1, \cdots, q$, solve y_j-subproblems in parallel:

$$y_j^{k+1} = \arg\min_{y_j \in \mathcal{Y}_j} \left\{ g_j(y_j) + \frac{\tau r_j}{2} \left\| y_j - \frac{\tau r_j y_j^k + B_j^*(\lambda^k)}{\tau r_j} \right\|^2 \right\}.$$

Step 2. Update the Lagrange multipliers:

$$\lambda^{k+1} = \lambda^k - \sigma \left[\mathcal{B}((\alpha+1)\mathbf{y}^{k+1} - \mathbf{y}^k) - \alpha c \right].$$

Step 3. Replace k by $k+1$, and turn to Step 1.

Obviously, all the resulting subproblems are linearized and solved in parallel, so we call Algorithm 2 the linearized full Jacobian splitting version of ALM. The convergence of Algorithm 2 with $\tau > \frac{q(2+\alpha)}{4}$ can be obtained easily from Section 3, by removing the items related to variable $\mathbf{x}$, e.g., $\|\mathbf{x}^k - \mathbf{x}^{k+1}\|_{\mathcal{Q}_1}$.

5 Numerical Experiments

In this section, we apply the proposed LSADMM to solve a class of sparse matrix minimization problems, and report the corresponding results. To further illustrate the numerical efficiency of our algorithm, comparison experiments are also done with some popular methods. In the following, we denote the random number generator by *seed* for generating data in MATLAB R2013B. All experiments are performed on an Intel(R) Core(TM) i5-4590 CPU@ 3.30 GHz PC with 8GB of RAM running on 64-bit Windows operating system.

5.1 Test problem

Consider the following Latent Variable Gaussian Graphical Model Selection (LVGGMS) problem in statistical learning [28]:

$$\begin{array}{ll} \min & F(X,S,L) := \langle X, C\rangle - \log\det(X) + \nu\|S\|_1 + \mu \mathrm{tr}(L) \\ \text{s.t.} & X - S + L = 0, \quad L \succeq 0, \;\; X, S \in \mathbb{R}^{n\times n}, \end{array} \tag{61}$$

where $C \in \mathbb{R}^{n\times n}$ is the covariance matrix obtained from observation; ν and μ are given positive weighting parameters; $\mathrm{tr}(L)$ stands for the trace of the matrix L; and $\|S\|_1$ denotes the l_1-norm of S (defined as the sum of the absolute values of all entries). Throughout the experiments, the parameters of all compared algorithms are the same as in [1] and $(\nu, \mu) = (0.005, 0.05)$. The given data C is randomly generated by the following MATLAB code with $m = 100$, which are downloaded from S. Boyd's homepage:

```
randn(seed,0); rand(seed,0); n=m; N=10*n;
Sinv=diag(abs(ones(n,1)));
idx=randsample(n^2,0.001*n^2);
Sinv(idx)=ones(numel(idx),1); Sinv=Sinv+Sinv;
if min(eig(Sinv))<0
   Sinv=Sinv+1.1*abs(min(eig(Sinv)))*eye(n);
end
S=inv(Sinv);
D=mvnrnd(zeros(1,n),S,N); C=cov(D);
```

5.2 Implementation

For a given $\sigma > 0$, the augmented Lagrangian function of problem (61) is

$$L_\sigma(X,S,L,\Lambda) = F(X,S,L) - \langle \Lambda, X - S + L\rangle + \frac{\sigma}{2}\|X - S + L\|^2, \tag{62}$$

where $(X,S,L,\Lambda) \in \mathbb{R}^{n\times n} \times \mathbb{R}^{n\times n} \times \mathbb{R}^{n\times n} \times \mathbb{R}^{n\times n}$. For the simplicity of computation, we use $\beta = 0$ and then $\alpha \in (0,2)$ in the LSADMM. By different ways of partitioning the variables X, S and L, the LSADMM can lead to many different updating schemes. From the sufficient conditions $\tau > \frac{q(2+\gamma)}{4}$ and $\rho > p-1$ to guarantee the convergence, a larger value of τ (or ρ) should be selected for the full Jacobian splitting version of ALM. Thus, we would show the following two preferred updating schemes. One type partition is (X,S) and L, then the corresponding iterative scheme is

$$\text{LSADMM-2-1} \left\{ \begin{array}{l} X^{k+1} := \arg\min\limits_{X\in\mathbb{R}^{n\times n}} L_\sigma(X,S^k,L^k,\Lambda^k) + \frac{\rho\sigma}{2}\|X - X^k\|^2, \\ S^{k+1} := \arg\min\limits_{S\in\mathbb{R}^{n\times n}} L_\sigma(X^k,S,L^k,\Lambda^k) + \frac{\rho\sigma}{2}\|S - S^k\|^2, \\ \Lambda^{k+\frac{1}{2}} := \Lambda^k - \alpha\sigma(X^{k+1} - S^{k+1} + L^k), \\ L^{k+1} := \arg\min\limits_{L\in\mathbb{R}^{n\times n}, L\succeq 0} \left\{ \mu \mathrm{tr}(L) + \frac{\tau r}{2}\left\| L - \frac{\tau r L^k + \Lambda^{k+\frac{1}{2}}}{\tau r}\right\|^2 \right\}, \\ \Lambda^{k+1} := \Lambda^{k+\frac{1}{2}} - \sigma(L^{k+1} - L^k) \end{array} \right. \tag{63}$$

where $\tau \in (0,1)$, $\rho \in (1,+\infty)$ and $r > \sigma$. We denote this scheme by LSADMM-2-1. The other is X and (S,L), then its iterative scheme is

$$\text{LSADMM-1-2}\begin{cases} X^{k+1} := \arg\min_{X\in\mathbb{R}^{n\times n}} L_\sigma(X,S^k,L^k,\Lambda^k) + \frac{\rho\sigma}{2}\|X - X^k\|^2, \\ \Lambda^{k+\frac{1}{2}} := \Lambda^k - \alpha\sigma(X^{k+1} - S^k + L^k), \\ S^{k+1} := \arg\min_{S\in\mathbb{R}^{n\times n}} \left\{\nu\|S\|_1 + \frac{\tau r_1}{2}\left\|S - \frac{\tau r_1 S^k - \Lambda^{k+\frac{1}{2}}}{\tau r_1}\right\|^2\right\}, \\ L^{k+1} := \arg\min_{L\in\mathbb{R}^{n\times n}, L\succeq 0} \left\{\mu \mathrm{tr}(L) + \frac{\tau r_2}{2}\left\|L - \frac{\tau r_2 L^k + \Lambda^{k+\frac{1}{2}}}{\tau r_2}\right\|^2\right\}, \\ \Lambda^{k+1} := \Lambda^{k+\frac{1}{2}} + \sigma(S^{k+1} - S^k) - \sigma(L^{k+1} - L^k). \end{cases} \quad (64)$$

where $\tau \in (2,+\infty)$, $\rho \in [0,+\infty)$, $r_1 > \sigma$ and $r_2 > \sigma$, which is denoted by LSADMM-1-2.

Next, we take the scheme LSADMM-1-2 for an example to show how to get the explicit solutions of each subproblem. By the first-order optimality condition of the X-subproblem, we derive

$$0 = C - \frac{1}{X} + \sigma(X - S^k + L^k - \frac{\Lambda^k}{\sigma}) + \rho\sigma(X - X^k),$$

namely,

$$(\rho+1)\sigma X^2 + [C + \sigma(L^k - S^k) - \Lambda^k - \rho\sigma X^k]X - I = 0.$$

With the eigenvalue decomposition

$$U\mathrm{Diag}(\zeta)U^\top = C + \sigma(L^k - S^k) - \Lambda^k - \rho\sigma X^k,$$

where $\mathrm{Diag}(\zeta)$ is a diagonal matrix with $\zeta_i (i = 1,\cdots,n)$ on the diagonal, we obtain that the solution of the X-subproblem is

$$X^{k+1} = U\mathrm{Diag}(\eta)U^\top,$$

where $\mathrm{Diag}(\eta)$ is the diagonal matrix with diagonal elements

$$\eta_i = \frac{-\zeta_i + \sqrt{\zeta_i^2 + 4(\rho+1)\sigma}}{2(\rho+1)\sigma}, i = 1,\cdots,n.$$

For the S-subproblem, we have

$$S^{k+1} = \mathrm{Shrink}\left(\frac{\tau r_1 S^k - \Lambda^{k+\frac{1}{2}}}{\tau r_1}, \frac{\nu}{\tau r_1}\right),$$

where $\mathrm{Shrink}(\cdot,\cdot)$ is the soft shrinkage operator. Meanwhile, it is easy to verify that the L-subproblem is equivalent to

$$L^{k+1} = V\mathrm{Diag}(\max\{\zeta, 0\})V^\top,$$

where $\max\{\zeta,0\}$ is taken component-wise and $V\mathrm{Diag}(\zeta)V^\top$ is the eigenvalue decomposition of the matrix $\frac{\tau r_2 L^k + \Lambda^{k+\frac{1}{2}} - \mu I}{\tau r_2}$.

The implementation of these two methods are terminated when the following stopping criterions are satisfied:

$$\begin{aligned} \mathrm{RelChg}: &= \max\left\{\frac{\|X^{k+1} - X^k\|_F}{1 + \|X^k\|_F}, \frac{\|S^{k+1} - S^k\|_F}{1 + \|S^k\|_F}, \frac{\|L^{k+1} - L^k\|_F}{1 + \|L^k\|_F}\right\} < \epsilon_1, \\ \mathrm{IER}: &= \|X^k - S^k + L^k\|_F < \epsilon_2, \end{aligned}$$

which measure the relative changing tendency of the iterative error and infeasibility.

5.3 Numerical results

Since small proximal term can allow selecting relative larger stepsize [17, 26], we always let $\tau = \varsigma\frac{2+\alpha}{4}$ with $\varsigma = 1.001$, $r = \varsigma\sigma$ and $\rho = 1.001$ for LSADMM-2-1. For LSADMM-1-2, we have $q = 2$ and then set $\tau = \varsigma\frac{2+\alpha}{2}$, $r_1 = r_2 = \varsigma\sigma, \rho = 0$. For all algorithms, the starting points are initialized as $(X^0, S^0, L^0, \Lambda^0) = (0, 0, 0, 0)$ and the maximal number of iterations is set as 1000[1].

First, we investigate the performance of LSADMM-2-1 and LSADMM-1-2 for solving the LVGGMS problem with variance of the penalty parameter σ and stepsize α. Numerical results are reported in Table 1 with $\epsilon_1 = 10^{-6}$ and $\epsilon_2 = 10^{-7}$. Here, *iter* and *cpu* respectively denote the iteration number and the CPU time in seconds. The bold letter indicates the best result of each algorithm. From Table 1, we can observe that:

- The iteration number and the CPU time of the two algorithms have a similar changing behavior, which decreases firstly and then increases along with the decrease of the value of σ.
- For LSADMM-2-1, the results with $\sigma = 0.07$ are slightly better than the others, while for LSADMM-1-2, the results with $\sigma = 0.12$ are slightly better than the others.
- For the same setting α, the best result of LSADMM-1-2 is better than that of LSADMM-2-1, in terms of the iteration number and the CPU time.

For LSADMM-2-1, the proximal parameters ρ should be more than 1 for the two subproblems about variables X and S. However, for LSADMM-1-2 we have

$$\frac{\sigma}{2}\|S - S^k\|^2_{D_1} = \frac{\tau r_1 - \sigma}{2}\|S - S^k\|^2 = \frac{(\varsigma^2\frac{\alpha+2}{2} - 1)\sigma}{2}\|S - S^k\|^2,$$
$$\frac{\sigma}{2}\|L - L^k\|^2_{D_2} = \frac{\tau r_2 - \sigma}{2}\|L - L^k\|^2 = \frac{(\varsigma^2\frac{\alpha+2}{2} - 1)\sigma}{2}\|L - L^k\|^2,$$

and $\varsigma^2\frac{\alpha+2}{2} - 1 < 1 < \rho$ for any $\alpha \in (0, \frac{4}{\varsigma^2} - 2)$, which may be the key reason to obtain the better result over LSADMM-2-1, because the smaller proximal parameter tends to accelerate convergence in practical computation.

Furthermore, the results in Table 1 indicate that using a relatively larger stepsize α of the dual variables often improve the convergence speed. In view of all the reported results in Table 1, LSADMM-1-2 with $\sigma = 0.12$ and $\alpha = 1.7$ performs better than other cases. Hence, in the following experiments, we adapt LSADMM-1-2 with default $\sigma = 0.12$.

The results in Table 2 show the performance of LSADMM-1-2 with variance of the proximal parameter τ. From the ratio in the number of iterations, LSADMM-1-2 with $\tau = \varsigma\frac{2+\alpha}{2}$ can achieve an improvement of about 3%-9% reduction over that with $\tau = 2\varsigma$.

Finally, we carry out some numerical comparisons on solving the problem (61) by using LSADMM-1-2 with $\sigma = 0.12$ and $\alpha = 1.7$, and we compare it with four popular methods: the third case of the generalized symmetric ADMM [1] (denoted by GS-ADMM-III); Proximal Jacobian Decomposition of ALM [9] (denoted by PJALM); twisted version of the proximal ADMM [18] (denoted by TADMM) and splitting method in [13] (denoted by HTY), in which the parameters of all comparison algorithms are set as in [1, Section 5.2.2] but the stopping criterions are the same as our LSADMM-1-2.

[1]The codes of LSADMM-1-2 and LSADMM-2-1 are available online at https://github.com/cxk9369010/PID-LSADMM.

Table 1: Performance of LSADMM-2-1 and LSADMM-1-2 on LVGGMS problem with different σ.

LSADMM-2-1						
τ	α	σ	*iter*	*cpu*	IER	RelChg
0.725725	0.9	0.008	540	5.7	1.0e-06	9.5e-09
		0.01	433	4.5	9.9e-07	1.2e-08
		0.03	148	1.6	9.5e-07	3.8e-08
		0.05	91	1.0	9.7e-07	8.3e-08
		0.06	79	0.9	8.4e-07	9.1e-08
		0.07	**73**	1.0	5.3e-07	9.1e-08
		0.08	83	1.2	1.0e-07	8.9e-08
		0.1	113	1.8	6.5e-08	9.1e-08
		0.3	327	5.6	2.9e-08	9.9e-08
		0.5	513	9.8	2.4e-08	9.9e-08
0.775775	1.1	0.008	461	9.5	9.9e-07	1.1e-08
		0.01	369	8.2	9.9e-07	1.4e-08
		0.03	125	3.0	9.4e-07	4.2e-08
		0.05	77	1.8	8.2e-07	8.9e-08
		0.06	68	1.6	5.6e-07	8.6e-08
		0.07	**67**	1.6	2.6e-07	9.6e-08
		0.08	83	1.9	6.1e-08	9.9e-08
		0.1	109	2.8	4.2e-08	9.9e-08
		0.3	311	7.7	1.7e-08	9.7e-08
		0.5	484	12.7	1.6e-08	9.8e-08
0.825825	1.3	0.008	408	10.6	9.9e-07	1.4e-08
		0.01	327	8.3	9.8e-07	1.8e-08
		0.03	110	3.4	9.5e-07	5.6e-08
		0.05	71	2.0	7.5e-07	9.1e-08
		0.06	62	1.7	6.2e-07	9.5e-08
		0.07	**61**	1.5	2.5e-07	8.6e-08
		0.08	73	2.0	5.1e-08	9.0e-08
		0.1	95	2.6	3.6e-08	9.1e-08
		0.3	272	7.4	3.1e-08	9.8e-08
		0.5	428	12.6	3.4e-08	1.0e-07
0.875875	1.5	0.008	380	11.0	9.8e-07	1.6e-08
		0.01	304	8.8	9.8e-07	2.0e-08
		0.03	103	3.0	9.2e-07	6.1e-08
		0.05	67	2.2	5.5e-07	8.6e-08
		0.06	58	2.1	4.5e-07	9.4e-08
		0.07	**58**	1.9	1.4e-07	9.6e-08
		0.08	69	2.5	5.0e-08	9.0e-08
		0.1	89	2.8	4.0e-08	8.9e-08
		0.3	253	8.4	4.1e-08	1.0e-07
		0.5	400	11.6	4.0e-08	9.9e-08
0.925925	1.7	0.008	334	9.2	9.8e-07	1.9e-08
		0.01	267	7.6	9.9e-07	2.4e-08
		0.03	93	3.2	9.1e-07	6.7e-08
		0.05	61	2.1	5.9e-07	8.6e-08
		0.06	**54**	1.8	4.2e-07	9.4e-08
		0.07	56	2.1	1.2e-07	1.0e-07
		0.08	66	1.8	5.9e-08	8.5e-08
		0.1	83	2.9	4.7e-08	9.7e-08
		0.3	235	7.6	4.4e-08	9.6e-08
		0.5	371	11.5	4.6e-08	9.9e-08
0.975975	1.9	0.008	322	10.3	9.7e-07	1.9e-08
		0.01	258	8.6	9.7e-07	2.3e-08
		0.03	92	3.2	9.4e-07	6.9e-08
		0.05	59	1.9	6.3e-07	9.5e-08
		0.06	**53**	1.6	3.9e-07	8.8e-08
		0.07	55	1.8	1.2e-07	8.8e-08
		0.08	64	2.0	6.8e-08	8.2e-08
		0.1	80	2.2	5.2e-08	9.9e-08
		0.3	226	7.0	5.1e-08	9.8e-08
		0.5	358	11.9	5.2e-08	1.0e-07

LSADMM-1-2						
τ	α	σ	*iter*	*cpu*	IER	RelChg
1.45145	0.9	0.01	573	11.1	1.0e-06	8.2e-09
		0.1	75	1.7	9.4e-07	6.8e-08
		0.12	66	1.5	7.8e-07	8.5e-08
		0.13	64	1.5	5.7e-07	9.4e-08
		0.14	**62**	1.5	4.5e-07	9.1e-08
		0.15	65	1.6	1.6e-07	9.0e-08
		0.16	73	1.8	3.6e-08	9.2e-08
		0.2	100	2.5	2.2e-08	9.5e-08
		0.3	156	4.2	1.2e-08	9.5e-08
		0.5	257	6.6	7.3e-09	9.6e-08
1.55155	1.1	0.01	495	14.3	9.9e-07	2.1e-09
		0.1	59	1.5	9.5e-07	4.9e-08
		0.12	**55**	1.5	3.9e-07	6.9e-08
		0.13	58	1.6	5.2e-08	8.9e-08
		0.14	62	1.7	7.4e-08	9.0e-08
		0.15	72	2.1	4.0e-08	8.8e-08
		0.16	79	2.7	3.7e-08	9.7e-08
		0.2	102	3.0	2.6e-08	1.0e-07
		0.3	145	4.3	1.6e-08	9.5e-08
		0.5	172	5.0	1.0e-08	1.0e-07
1.65165	1.3	0.01	417	12.4	9.9e-07	2.5e-09
		0.1	46	1.7	9.3e-07	2.6e-08
		0.12	40	1.3	8.2e-07	2.8e-08
		0.13	37	1.9	8.9e-07	3.7e-08
		0.14	**35**	1.3	8.5e-07	9.7e-08
		0.15	38	1.0	6.2e-07	7.5e-08
		0.16	40	1.2	6.8e-07	8.5e-08
		0.2	49	2.0	6.4e-07	8.8e-08
		0.3	70	2.6	5.9e-07	9.1e-08
		0.5	109	3.1	5.6e-07	9.8e-08
1.75175	1.5	0.01	361	11.3	9.9e-07	2.9e-09
		0.1	39	0.9	9.7e-07	3.4e-08
		0.12	34	1.3	7.4e-07	3.6e-08
		0.13	**33**	1.2	6.8e-07	7.0e-08
		0.14	35	1.1	8.1e-07	8.1e-08
		0.15	37	1.2	9.2e-07	9.5e-08
		0.16	40	1.2	7.3e-07	7.6e-08
		0.2	49	1.8	7.1e-07	7.9e-08
		0.3	70	2.2	7.0e-07	8.4e-08
		0.5	109	3.2	7.1e-07	9.2e-08
1.85185	1.7	0.01	318	10.4	9.9e-07	3.3e-09
		0.1	34	1.0	1.0e-06	4.0e-08
		0.12	**31**	1.0	7.4e-07	6.9e-08
		0.13	33	1.1	9.1e-07	8.2e-08
		0.14	36	1.1	7.4e-07	6.8e-08
		0.15	38	1.3	8.6e-07	8.0e-08
		0.16	40	1.3	9.6e-07	9.1e-08
		0.2	49	1.6	9.4e-07	9.2e-08
		0.3	70	2.5	9.4e-07	9.7e-08
		0.5	110	3.6	8.8e-07	9.6e-08
1.95195	1.9	0.01	284	9.2	9.9e-07	3.8e-09
		0.1	33	1.1	6.4e-07	3.5e-08
		0.12	**32**	1.0	7.5e-07	7.6e-08
		0.13	34	1.3	9.6e-07	8.3e-08
		0.14	37	1.0	8.0e-07	6.9e-08
		0.15	39	1.3	9.4e-07	8.1e-08
		0.16	42	1.4	7.7e-07	6.8e-08
		0.2	51	1.7	8.5e-07	7.6e-08
		0.3	72	2.4	9.8e-07	9.2e-08
		0.5	113	4.1	9.5e-07	9.2e-08

Table 2: Performance of LSADMM-1-2 on LVGGMS problem with $\sigma = 0.12$ and different τ.

LSADMM-1-2						LSADMM-1-2					
α	$\tau = 2\varsigma$	*iter*	*cpu*	IER	RelChg	$\tau = \varsigma\frac{2+\alpha}{2}$	*iter*	*cpu*	IER	RelChg	Ratio
1.5	2.002	35	1.26	8.34e-07	7.11e-08	1.75175	34	1.17	7.36e-07	3.57e-08	0.97
1.55	2.002	35	1.14	7.16e-07	6.17e-08	1.776775	33	1.08	6.82e-07	3.85e-08	0.94
1.6	2.002	34	1.11	8.95e-07	7.86e-08	1.8018	32	1.02	7.19e-07	4.75e-08	0.94
1.65	2.002	34	1.14	7.85e-07	7.06e-08	1.826825	31	0.99	8.35e-07	6.70e-08	0.91
1.7	2.002	34	1.23	7.06e-07	6.93e-08	1.85185	31	0.99	7.42e-07	6.86e-08	0.91
1.75	2.002	33	1.08	9.61e-07	8.53e-08	1.876875	31	1.05	8.04e-07	7.38e-08	0.94
1.8	2.002	33	1.05	8.80e-07	9.45e-08	1.9019	31	1.11	9.00e-07	8.14e-08	0.94
1.85	2.002	33	1.05	8.12e-07	8.71e-08	1.926925	32	1.08	6.61e-07	7.13e-08	0.97
1.9	2.002	33	1.05	7.54e-07	6.39e-08	1.95195	32	1.05	7.55e-07	7.62e-08	0.97
1.95	2.002	33	1.05	7.03e-07	6.04e-08	1.976975	32	1.02	8.65e-07	7.74e-08	0.97

Table 3: Comparative results of different algorithms under different tolerances.

method	ϵ_1	ϵ_2	*iter*	*cpu*	IER	RelChg
LSADMM-1-2	1.0e-06	1.0e-07	**31**	**0.9**	9.7e-07	8.1e-08
GS-ADMM-III			54	1.8	2.2e-07	1.0e-07
PJALM			110	3.9	9.9e-07	6.5e-08
TADMM			76	2.4	7.5e-07	9.7e-08
HTY			99	3.3	9.3e-07	1.0e-08
LSADMM-1-2	1.0e-07	1.0e-08	**37**	**1.1**	8.1e-08	6.8e-09
GS-ADMM-III			63	2.4	2.3e-08	9.8e-09
PJALM			138	4.5	9.2e-08	6.1e-09
TADMM			95	3.3	6.9e-08	8.9e-09
HTY			123	3.9	9.2e-08	9.2e-10
LSADMM-1-2	1.0e-08	1.0e-09	**45**	**1.5**	6.8e-09	5.7e-10
GS-ADMM-III			72	2.7	2.4e-09	9.9e-10
PJALM			165	5.9	9.6e-09	6.4e-10
TADMM			113	3.6	7.3e-09	9.4e-10
HTY			146	4.8	1.0e-08	9.5e-11
LSADMM-1-2	1.0e-09	1.0e-10	**54**	**2.4**	8.6e-10	7.2e-11
GS-ADMM-III			81	3.1	2.5e-10	1.0e-10
PJALM			193	6.2	9.4e-10	6.5e-11
TADMM			132	4.6	7.0e-10	8.9e-11
HTY			170	5.4	9.9e-10	9.1e-12
LSADMM-1-2	1.0e-10	1.0e-11	**62**	**2.2**	9.4e-11	2.1e-12
GS-ADMM-III			91	3.0	2.0e-11	7.9e-12
PJALM			221	7.2	9.5e-11	6.7e-12
TADMM			150	5.1	7.5e-11	9.7e-12
HTY			194	6.3	9.9e-11	8.9e-13

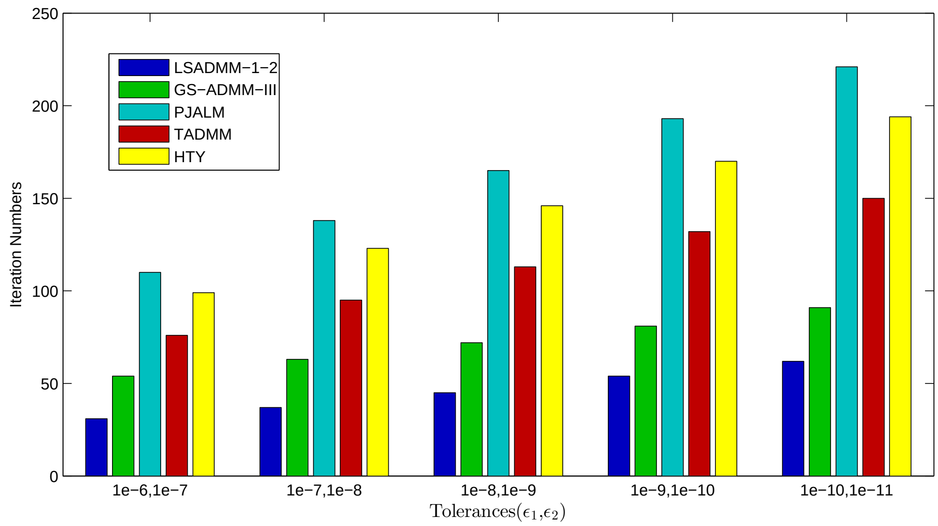


Figure 1: Comparison results of the iteration number vs tolerance by different algorithms.

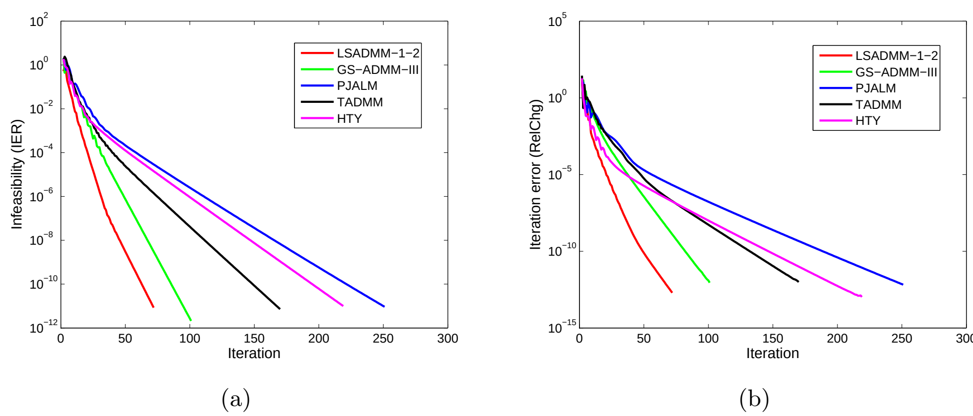
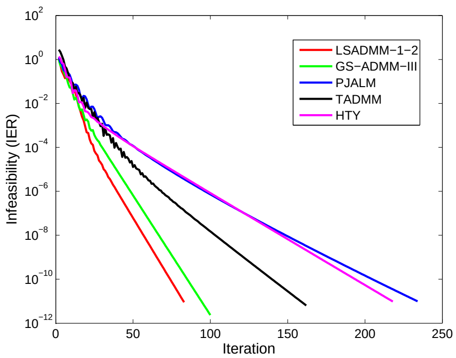
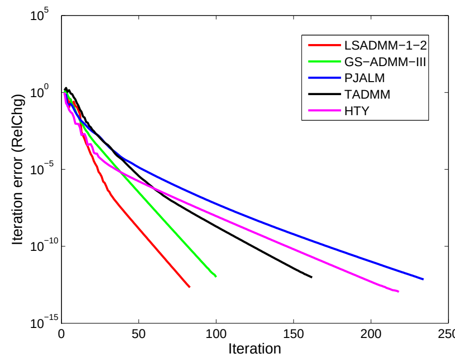


Figure 2: Convergence curves of IER and RelChg with initial values $(X^0, S^0, L^0, \Lambda^0) = (0, 0, 0, 0)$.

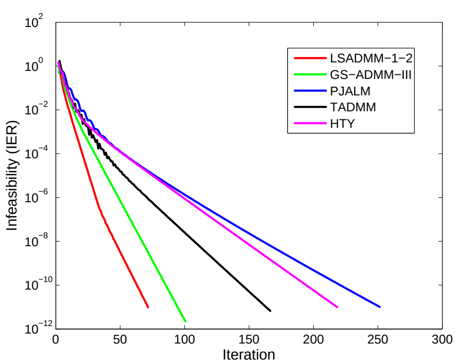


(a)

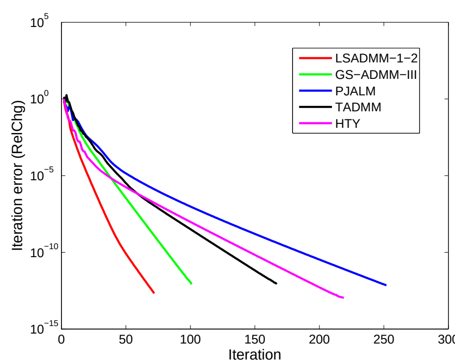


(b)

Figure 3: Convergence curves of IER and RelChg with initial values $(X^0, S^0, L^0, \Lambda^0) = (I, 2I, I, 0)$.



(a)



(b)

Figure 4: Convergence curves of IER and RelChg with initial values $(X^0, S^0, L^0, \Lambda^0) = (I, I, 0, 0)$.

Under different tolerances, the experimental results by the aforementioned five algorithms are reported in Table 3 and Fig.1. In addition, with fixed tolerances $\epsilon_1 = 10^{-11}$ and $\epsilon_2 = 10^{-12}$, convergence behaviors of the error measurements IER and RelChg by using different starting points are shown in Figs.2-4. We may observe from of Table 3 and Figs.1-4 that, LSADMM-1-2 is more efficient than other four algorithms in both the number of iterations, the CPU time and different initial starting points, which is extremely due to the linearized technique, the optimal lower bound of the proximal parameter τ and the proper choice of σ.

6 Brief conclusion

In this paper, a linearized symmetric ADMM, with positive-definite and positive-indefinite proximal regularization terms, is developed for solving a family of multi-block separable convex optimization problems. We achieve the optimal lower bound of the proximal parameter τ, since a smaller proximal parameter would accelerate the algorithm numerically. By the aid of the variational inequality, the global convergence as well as the sublinear ergodic convergence rate of the proposed algorithm are established respectively. Numerical experiments demonstrate the significant improvement of our LSADMM with the optimal proximal parameter on solving the three-block latent variable gaussian graphical model selection problem. However, the performance of the proposed algorithm is sensitive to some parameters like the penalty parameter, which can be dynamically adjusted at each iteration since the adaptive technology has been proved to be effective to enhance the performance of the ADMM-based algorithms. Therefore, in the future we would investigate how to design an adaptive rule for adjusting the parameters. Finally, from the convergence analysis one may construct another linearized ADMM-type algorithm that allows the first grouped subproblems using linearized techniques too. But linearizing the whole objective function of the subproblems, instead of just its quadratic term, may be an interesting work in the future.

References

[1] Jianchao Bai, Jicheng Li, Fengmin Xu, and Hongchao Zhang. Generalized symmetric admm for separable convex optimization. *Computational Optimization & Applications, To appear, DOI:10.1007/s10589-017-9971-0*, 2017.

[2] Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, and Jonathan Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations & Trends in Machine Learning*, 3(1):1–122, 2010.

[3] Brendan O'Donoghue, Giorgos Stathopoulos, and Stephen Boyd. A splitting method for optimal control. *IEEE Transactions on Control Systems Technology*, 21(6):2432–2442, 2013.

[4] M. Partridge and M. Jabri. Robust principal component analysis? *Journal of the ACM*, 58(3):1–37, 2011.

[5] Deren Han, Weiwei Kong, and Wenxing Zhang. A partial splitting augmented lagrangian method for low patch-rank image decomposition. *Journal of Mathematical Imaging & Vision*, 51(1):145–160, 2015.

[6] Roland Glowinski and Americo Marrocco. Sur l'approximation, par *éléments* finis d'ordre un, et la *résolution*, par *pénalisation-dualité*, d'une classe de problèmes de dirichlet non *linéaires*. *Journal of Equine Veterinary Science*, 2(R-2):41–76, 1975.

[7] Caihua Chen, Bingsheng He, Yinyu Ye, and Xiaoming Yuan. The direct extension of admm for multi-block convex minimization problems is not necessarily convergent. *Mathematical Programming*, 155(1-2):57–79, 2016.

[8] Bingsheng He, Min Tao, and Xiaoming Yuan. Alternating direction method with gaussian back substitution for separable convex programming. *SIAM Journal on Optimization*, 22(2):313–340, 2012.

[9] He Bingsheng, Xu Hong, and Yuan Xiaoming. On the proximal jacobian decomposition of alm for multiple-block separable convex minimization problems and its relationship to admm. *Journal of Scientific Computing*, 66:1204–1217, 2016.

[10] Yan Gu, Bo Jiang, and Deren Han. A semi-proximal-based strictly contractive peaceman-rachford splitting method. *arXiv:1506.02221*, 2015.

[11] Jianchao Bai. Hybrid middle proximal admm for linearly constrained convex optimization. *optimization-online, http://www.optimization-online.org/DB_HTML/2017/11/6319.html*, November, 2017.

[12] Xiaokai Chang, Sanyang Liu, Pengjun Zhao, and Xu Li. Convergent predictionccorrection-based admm for multi-block separable convex programming. *Journal of Computational and Applied Mathematics*, 335:270–288, 2018.

[13] Bingsheng He, Min Tao, and Xiaoming Yuan. A splitting method for separable convex programming. *IMA Journal of Numerical Analysis*, 35(1):394–426, 2015.

[14] Xudong Li, Defeng Sun, and Kim Chuan Toh. A schur complement based semi-proximal admm for convex quadratic conic programming and extensions. *Mathematical Programming*, 155(1-2):333–373, 2016.

[15] Defeng Sun, Kim-Chuan Toh, and Liuqin Yang. A convergent 3-block semi-proximal alternating direction method of multipliers for conic programming with 4-type of constraints. *SIAM Journal on Optimization*, 25(2), 2014.

[16] Xiaokai Chang, Sanyang Liu, and Xu Li. Modified alternating direction method of multipliers for convex quadratic semidefinite programming. *NEUROCOMPUTING*, 214:575–586, 2016.

[17] Bin Gao and Feng Ma. Symmetric admm with positive-indefinite proximal regularization for linearly constrained convex optimization. *Journal of Optimization Theory and Applications*, 2017.

[18] Jin Jiang Wang and Wen Song. An algorithm twisted from generalized admm for multi-block separable convex minimization models. *Journal of Computational & Applied Mathematics*, 309:342–358, 2016.

[19] Jonathan Eckstein and Dimitri P. Bertsekas. On the douglas-rachford splitting method and the proximal point algorithm for maximal monotone operators. *Mathematical Programming*, 55(1-3):293–318, 1992.

[20] J. Eckstein. Parallel alternating direction multiplier decomposition of convex programs. *Journal of Optimization Theory & Applications*, 80(1):39–62, 1994.

[21] Junfeng Yang and Xiaoming Yuan. Linearized augmented lagrangian and alternating direction methods for nuclear norm minimization. In *Mathematics of Computation*, pages 301–329, 2013.

[22] Xinxin Li, Lili Mo, Xiaoming Yuan, and Jianzhong Zhang. Linearized alternating direction method of multipliers for sparse group and fused lasso models. *Computational Statistics & Data Analysis*, 79(79):203–221, 2014.

[23] Zhouchen Lin, Risheng Liu, and Huan Li. Linearized alternating direction method with parallel splitting and adaptive penalty for separable convex programs in machine learning. *Machine Learning*, 99(2):287–325, 2015.

[24] Bingsheng He, Feng Ma, and Xiaoming Yuan. Optimal linearized alternating direction method of multipliers for convex programming. *optimization-online, http://www.optimization-online.org/DB_HTML/2017/09/6228.html*, September, 2017.

[25] Min Li, Defeng Sun, and Kim Chuan Toh. A majorized admm with indefinite proximal terms for linearly constrained convex composite optimization. *SIAM Journal on Optimization*, 26(2), 2015.

[26] Fan Jiang, Zhongming Wu, and Xingju Cai. Generalized admm with optimal indefinite proximal term for linearly constrained convex optimization. *optimization-online, http://www.optimization-online.org/DB_HTML/2017/09/6228.html*, October, 2017.

[27] R. Tyrrell Rockafellar. Convex analysis. (17):5–101, 1970.

[28] Shiqian Ma. Alternating proximal gradient method for convex minimization. *Journal of Scientific Computing*, 68(2):546–572, 2016.