JOURNAL OF INDUSTRIAL AND
MANAGEMENT OPTIMIZATION
Volume **12**, Number **1**, January **2016**

doi:10.3934/jimo.2016.12.317

pp. **317–336**

AN ALTERNATING DIRECTION METHOD FOR SOLVING A CLASS OF INVERSE SEMI-DEFINITE QUADRATIC PROGRAMMING PROBLEMS

Yue Lu

Institute of Operations Research and Control Theory, School of Mathematical Sciences
Dalian University of Technology, Dalian 116024, China

Ying-En Ge

School of Transportation and Logistics, Faculty of Infrastructure Engineering
Dalian University of Technology, Dalian 116024, China

Li-Wei Zhang

Institute of Operations Research and Control Theory, School of Mathematical Sciences
Dalian University of Technology, Dalian 116024, China

(Communicated by Shu-Cherng Fang)

Abstract. In this paper, we propose an alternating-direction-type numerical method to solve a class of inverse semi-definite quadratic programming problems. An explicit solution to one direction subproblem is given and the other direction subproblem is proved to be a convex quadratic programming problem over positive semi-definite symmetric matrix cone. We design a spectral projected gradient method for solving the quadratic matrix optimization problem and demonstrate its convergence. Numerical experiments illustrate that our method can solve inverse semi-definite quadratic programming problems efficiently.

1. **Introduction.** In the classic model of optimization problems, some parameters associated with decision variables may appear in the objective function or in the constraint set. Usually, we assume that these parameters are available and then optimize the model to get a solution, which is called a forward optimization problem. But the opposite case often occurs in practice, in which only estimates for parameters are known and optimal solutions are available from experience, observations or experiments. The goal of an inverse optimization problem is to obtain values of parameters that make the known solution optimal and differ from the given estimates as little as possible.

The study of inverse optimization problems was initiated by Burton and Toint [9], in which they investigated an inverse shortest paths problem. After that, there are many important contributions to inverse optimization, for instance, see Cai et al. [10], Zhang and Ma [34], Zhang et al. [33], Ahuja and Orlin [2, 3] and Heuberger

2010 *Mathematics Subject Classification.* Primary: 90C20, 90C22; Secondary: 49M20.

Key words and phrases. Inverse semi-definite programming, alternating direction method, spectral projected gradient method, BB-step, convex semi-definite quadratic programming.

This study is supported by the National Natural Science Foundation of China under projects No.91130007, No.91330206 and No.71171026.

[17]. However, only several papers focus on inverse continuous optimization. Zhang and Liu [31, 32] discussed inverse linear programming problems. Iyengar and Kang [20] proposed several inverse conic programming models for characterizing some portfolio optimization problems. Zhang and Zhang [35] used an augmented Lagrangian method to solve a type of inverse quadratic programming problems.

Recently, Xiao et al. [28, 29] consider a class of semi-definite quadratic programming problems in the form of

$$\mathrm{SDQP}(G,c) \qquad \begin{array}{ll} \min & \frac{1}{2}x^T Gx + c^T x \\ \text{s.t.} & B - \mathcal{A}(x) \in \mathcal{S}^m_+, \end{array} \tag{1}$$

where $x \in \mathcal{R}^n$, $G \in \mathcal{S}^n_+$, $c \in \mathcal{R}^n$ and $\mathcal{S}^m_+$ corresponds to the cone of $m \times m$ positive semi-definite symmetric matrices. The space of $m \times m$ symmetric matrices is denoted by $\mathcal{S}^m$ and $B \in \mathcal{S}^m$. The inner product between two matrices $C, D \in \mathcal{S}^m$ is defined by $\langle C, D\rangle := \mathrm{Tr}(CD)$, where $\mathrm{Tr}(\cdot)$ denotes the trace of a matrix. The Frobenius norm $\|C\|_F := \sqrt{\langle C, C\rangle}$. The operator $\mathcal{A}(\cdot)$ is a linear operator from $\mathcal{R}^n$ to $\mathcal{S}^m$ and its definition is given by

$$\mathcal{A}(y) := \sum_{i=1}^{n} y_i A_i,\ \forall y := (y_1, y_2, \cdots, y_n)^T \in \mathcal{R}^n \text{ and } A_i \in \mathcal{S}^m\ (i = 1, \cdots, n).$$

Then $\mathcal{A}^*(\cdot) : \mathcal{S}^m \to \mathcal{R}^n$ is also a linear operator, that is,

$$\mathcal{A}^*(X) := \begin{bmatrix} \langle A_1, X\rangle \\ \langle A_2, X\rangle \\ \vdots \\ \langle A_n, X\rangle \end{bmatrix}, \quad \forall X \in \mathcal{S}^m. \tag{2}$$

In the papers [28, 29], the authors proposed a smoothing Newton method and an augmented Lagrangian method (ALM) to solve the following inverse semi-definite quadratic programming problem

$$\mathrm{ISDQP}(G,c) \qquad \begin{array}{ll} \min & \frac{1}{2}\|G - G^0\|_F^2 + \frac{1}{2}\|c - c^0\|_2^2 \\ \text{s.t.} & x^0 \in \mathrm{Sol}(\mathrm{SDQP}(G,c)), \\ & (G,c) \in \mathcal{S}^n_+ \times \mathcal{R}^n, \end{array} \tag{3}$$

where $\mathrm{Sol}(\cdot)$ denotes the solution mapping for a given problem, (G^0, c^0) is an estimate to (G, c) and x^0 is a given feasible solution to (1). The inner product of two vectors $c, d \in \mathcal{R}^n$ is denoted by $\langle c, d\rangle_v := c^T d$ and $\|c\|_2 := \sqrt{\langle c, c\rangle_v}$.

The following two reasons stimulate us to propose a new method for the inverse semi-definite quadratic programming problem $\mathrm{ISDQP}(G, c)$.

(i) In theory, the authors in [28, 29] use Newton-type methods to solve the corresponding subproblems. However, a subroutine to analyze the Jacobian matrix for the given systems should be constructed, which is a difficult task and need to acquire techniques from variational analysis [24].
(ii) In algorithm aspect, although Newton method is a very fast local algorithm under some assumptions, the preparations consume a significant amount of time and occupy more storage.

In this paper, we aim to solve $\mathrm{ISDQP}(G, c)$ by an alternating direction method (ADM). The history of ADM dates back to Douglas, Peaceman, Rachford for solving PDE problems [12, 21] and Gabay, Mercier, Glowinski and Tallec for the variational problems [14, 15]. The basic idea of ADM is "Divide and Conquer", that is, divide the original problem into some easy-handled subproblems and design efficient subroutines to solve them. ADMs have been applied to various optimization problems. For instance, Tseng [26, 27] and He et al. [18] used them to handle variational inequality problems. Goldstein and Osher [16], Afonso et al. [1] investigated image reconstructions by ADMs. More theoretical results about ADM can be found in [8, 19, 30].

In our method, there exist two direction subproblems: G-subproblem and $(W, \hat{\Omega})$-subproblem. An explicit solution to G-subproblem can be obtained easily based on eigenvalue decomposition for a given matrix. On the other hand, $(W, \hat{\Omega})$-subproblem is essentially a strictly convex semi-definite quadratic programming problem over lower dimensional positive semi-definite symmetric matrix cone. Consequently, we present a spectral projected gradient type subroutine, which can solve this matrix optimization problem efficiently and does not increase the storage of computing.

The rest of the paper is organized as follows. In Section 2 we present some useful definitions and technical results. In Section 3, we transform (3) into the corresponding alternating-direction-type problem and design our main algorithm ADM-ISDQP. Subsection 3.1 and Subsection 3.2 are devoted to solving the subproblems appeared in the main algorithm. In Subsection 3.3, a subroutine SPGM that uses a spectral projected gradient method for solving $\hat{\Omega}$-subproblem is proposed and its convergence result is also established. In Subsection 3.4, we analyze the stopping criterion of ADM-ISDQP and build up a close connection with ADM for a class of structured convex programming problems. Numerical results implemented by ADM-ISDQP coupled with SPGM are reported in Section 4. Finally, we conclude this paper in Section 5

Before ending this section, we introduce notations used throughout the paper.

- Let $\mathcal{M}_{m\times n}$ denote the space of all $m \times n$ matrices. The set of all $m \times m$ orthogonal matrices is denoted by $\mathcal{O}^m$. Given any vector $x \in \mathcal{R}^m$, let $\mathrm{diag}(x)$ denote an $m \times m$ matrix with entries

$$(\mathrm{diag}(x))_{ij} := \begin{cases} x_i & \text{if } i = j, \\ 0 & \text{otherwise.} \end{cases}$$

  Given an index set $\mathcal{I} \subseteq \{1, \cdots, m\}$, $X_{\mathcal{I}}$ denotes the sub-matrix of X whose columns are indexed by $\mathcal{I}$. For a given scalar $x \in \mathcal{R}$, the function $[x]_+$ is defined by $[x]_+ := \max\{0, x\}$.
- The symbol $\mathcal{N}_{\mathcal{S}^n_+}(\cdot)$ represents the normal cone of $\mathcal{S}^n_+$, that is,

$$\mathcal{N}_{\mathcal{S}^n_+}(X) := \{Y : \langle Y, X' - X\rangle \le 0, \quad \forall X' \in \mathcal{S}^n_+\}.$$

  The normal cone of the product set $\mathcal{S}^n_+ \times \mathcal{S}^p_+$ at the pair $(G^*, \hat{\Omega}^*)$ is denoted by $\mathcal{N}_{\mathcal{S}^n_+\times\mathcal{S}^p_+}(G^*, \hat{\Omega}^*)$.
- Let $\Pi_{\mathcal{S}^n_+}(Y)$ be the metric projection of Y onto $\mathcal{S}^n_+$, that is, $\Pi_{\mathcal{S}^n_+}(Y)$ is the unique optimal solution of the following problem

$$\begin{array}{ll} \min & \frac{1}{2}\|X - Y\|_F^2 \\ \text{s.t.} & X \in \mathcal{S}^n_+. \end{array} \tag{4}$$

- The operators $\mathcal{L}_\xi(\eta)$ and $\mathcal{L}_A(B)$ are defined as follows:

$$\mathcal{L}_\xi(\eta) := \frac{\xi\eta^T + \eta\xi^T}{2} \ (\xi, \eta \in \mathcal{R}^n), \quad \mathcal{L}_A(B) := \frac{AB + BA}{2} \ (A, B \in \mathcal{S}^n).$$

  If A is nonsingular, the inverse operator of $\mathcal{L}_A(\cdot)$ is denoted by $\mathcal{L}_A^{-1}(\cdot)$.

2. **Preliminaries.** Throughout the paper, we use the following definitions and results.

Definition 2.1. Let $F : \mathcal{S}^n \to \mathcal{R}$ and $X, H \in \mathcal{S}^n$. The directional derivative of F at X along H is the following limit (if it exists):

$$F'(X; H) := \lim_{t \downarrow 0} \frac{F(X + tH) - F(X)}{t}.$$

We say that F is directionally differentiable at X if $F'(X; H)$ exists for all H.

Definition 2.2. Let $F : \mathcal{S}^n \to \mathcal{R}$ and $X, H \in \mathcal{S}^n$. F is said to be Gâteaux differentiable at X if F is directionally differentiable at X and the directional derivative $F'(X; H)$ is linear and continuous in H. In other words, there exists an operator $\mathcal{J}F(X)$ such that $\langle \mathcal{J}F(X), H \rangle = F'(X; H)$. We call $\mathcal{J}F(X)$ the derivative of F.

Definition 2.3. Let $F : \mathcal{S}^n \to \mathcal{R}$ and $X, H \in \mathcal{S}^n$. Let F be Gâteaux differentiable in a neighborhood of X. The second-order Gâteaux derivative $\mathcal{J}^2 F(X)$ at X is defined as follows:

$$\mathcal{J}^2 F(X)H := \lim_{t \downarrow 0} \frac{\mathcal{J}F(X + tH) - \mathcal{J}F(X)}{t}.$$

If $F(\cdot)$ is twice continuously differentiable, we obtain

$$F(X + H) = F(X) + \langle \mathcal{J}F(X), H \rangle + \frac{1}{2} \left\langle \mathcal{J}^2 F(X)H, H \right\rangle + o(\|H\|_F^2).$$

The next lemma provides a fundamental connection between normal cone and optimality conditions, which comes from [24, Theorem 6.12].

Lemma 2.4. *Consider a problem of minimizing a differentiable function f_0 over a set C. A necessary condition for $\bar{x}$ to be locally optimal is*

$$0 \in \mathcal{J}f_0(\bar{x}) + \mathcal{N}_C(\bar{x}).$$

When C and f_0 are both convex, the above condition are sufficient for $\bar{x}$ to be globally optimal.

Remark 1. From Lemma 2.4, the optimality condition of (4) is given by

$$0 \in (X - Y) + \mathcal{N}_{\mathcal{S}_+^n}(X) \Leftrightarrow X = \Pi_{\mathcal{S}_+^n}(Y), \tag{5}$$

which characterizes the relation between $\mathcal{N}_{\mathcal{S}_+^n}(\cdot)$ and $\Pi_{\mathcal{S}_+^n}(\cdot)$.

We present a crucial property for eigenvalue decomposition in the following lemma. We refer the interested reader to [25, Section 4] for the proof.

Lemma 2.5. *For any given $A \in S^n$, let A have the eigenvalue decomposition being arranged in non-increasing order*

$$A = P \begin{bmatrix} \Lambda_\alpha & & \\ & 0_\beta & \\ & & \Lambda_\gamma \end{bmatrix} P^T, \tag{6}$$

where $\alpha := \{i | \, \lambda_i(A) > 0\}$, $\beta := \{i | \, \lambda_i(A) = 0\}$, $\gamma := \{i | \, \lambda_i(A) < 0\}$ and $P \in \mathcal{O}^n$. Λ_α and Λ_γ are diagonal matrices whose diagonal entries are positive and negative eigenvalues of A, respectively. Suppose $X \in S_+^n$, $Y \in S_+^n$, $\langle X, Y \rangle = 0$. Let $A = X - Y$ have the eigenvalue decomposition (6), then we can easily get that

$$X = P \begin{bmatrix} \Lambda_\alpha & & \\ & 0_\beta & \\ & & 0_\gamma \end{bmatrix} P^T, \; -Y = P \begin{bmatrix} 0_\alpha & & \\ & 0_\beta & \\ & & \Lambda_\gamma \end{bmatrix} P^T.$$

To end this section, an important fact for a given quadratic function with strict convexity is presented. See [24, Chapter 2] for details.

Lemma 2.6. [1] *Given a function $\phi(x, y)$ being quadratic and strictly convex on (x, y), then the function*

$$\tau(x) := \inf_y \phi(x, y)$$

is also a quadratic and strictly convex function.

3. **Algorithm.** In this section, we first rewrite (3) as an equality constrained optimization problem and then construct the corresponding ADM-type algorithm.

By the fact that $G \in \mathcal{S}^n_+$, the objective function in (1) is a convex function of x. Hence, the constraint $x^0 \in \mathrm{Sol}(\mathrm{SDQP}(G, c))$ is equivalent to the following system

$$\begin{cases} c + Gx^0 - \mathcal{A}^*(\Omega) = 0, \\ \Omega \in \mathcal{S}^m_+, Z^0 \in \mathcal{S}^m_+, \langle \Omega, Z^0 \rangle = 0, \end{cases} \tag{7}$$

where $Z^0 := B - \mathcal{A}(x^0)$. Similar to [28], let $r := \mathrm{rank}(Z^0)$ and assume that Z^0 has the following eigenvalue decomposition

$$Z^0 = [Q_r \ Q_{r^c}] \begin{bmatrix} \Lambda_r & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} Q_r^T \\ Q_{r^c}^T \end{bmatrix},$$

where $p := m - r, Q := [Q_r \ Q_{r^c}] \in \mathcal{O}^m,\ Q_r \in \mathcal{M}_{m\times r},\ Q_{r^c} \in \mathcal{M}_{m\times p},\ \Lambda_r := \mathrm{diag}(\lambda_1, \cdots, \lambda_r)$ and λ_i $(i = 1, \cdots, r)$ are positive eigenvalues of Z^0. Let us denote

$$\hat{A}_i := Q_{r^c}^T A_i Q_{r^c} \quad \text{and} \quad \hat{\Omega} := Q_{r^c}^T \Omega Q_{r^c}.$$

The operator $\hat{\mathcal{A}}^*(\cdot)$ is defined by

$$\hat{\mathcal{A}}^*(\hat{\Omega}) := \begin{bmatrix} \langle \hat{A}_1, \hat{\Omega} \rangle \\ \langle \hat{A}_2, \hat{\Omega} \rangle \\ \vdots \\ \langle \hat{A}_n, \hat{\Omega} \rangle \end{bmatrix}.$$

Hence, the adjoint operator of $\hat{\mathcal{A}}^*$ is given by

$$\hat{\mathcal{A}}(y) := \sum_{i=1}^n y_i \hat{A}_i,\ \forall y := (y_1, y_2, \cdots, y_n)^T \in \mathcal{R}^n.$$

Lemma 3.1. *Problem (3) is equivalent to the following problem*

$$\begin{array}{ll} \min & \frac{1}{2}\|G - G^0\|_F^2 + \frac{1}{2}\|c - c^0\|_2^2 \\ \text{s.t.} & c + Gx^0 - \hat{\mathcal{A}}^*(\hat{\Omega}) = 0, \\ & (G, c, \hat{\Omega}) \in \mathcal{S}^n_+ \times \mathcal{R}^n \times \mathcal{S}^p_+. \end{array} \tag{8}$$

Proof. From (2) and the fact that $Q \in \mathcal{O}^m$, it can be easily verified that

$$\mathcal{A}^*(\Omega) = \left[\langle Q^T A_1 Q, Q^T \Omega Q \rangle, \cdots, \langle Q^T A_n Q, Q^T \Omega Q \rangle\right]^T.$$

The relations $\Omega \in \mathcal{S}^m_+, Z^0 \in \mathcal{S}^m_+, \langle \Omega, Z^0 \rangle = 0$ imply that

$$\Omega = [Q_r \ Q_{r^c}] \begin{bmatrix} 0 & 0 \\ 0 & \hat{\Omega} \end{bmatrix} \begin{bmatrix} Q_r^T \\ Q_{r^c}^T \end{bmatrix},$$

which follows from Lemma 2.5. Hence, for any $i \in \{1, \cdots, n\}$ we have

$$\langle Q^T A_i Q, Q^T \Omega Q \rangle = \langle \hat{A}_i, \hat{\Omega} \rangle.$$

[1] Thank one of the two reviewers to mention the result to us.

From the definition of $\hat{\mathcal{A}}^*(\hat{\Omega})$ and the above equation, we obtain

$$\hat{\mathcal{A}}^*(\hat{\Omega}) = \mathcal{A}^*(\Omega). \tag{9}$$

It follows from (7) and (9) that Problem (3) is equivalent to Problem (8). □

It is clear that Problem (8) can be rewritten as follows:

$$\begin{array}{ll} \min & \frac{1}{2}\|G - G^0\|_F^2 + \frac{1}{2}\|\hat{\mathcal{A}}^*(\hat{\Omega}) - Gx^0 - c^0\|_2^2 \\ \text{s.t.} & (G, \hat{\Omega}) \in \mathcal{S}_+^n \times \mathcal{S}_+^p. \end{array} \tag{10}$$

Let us import an auxiliary variable W and transform (10) into the following problem

$$\begin{array}{ll} \min & \frac{1}{2}\|G - G^0\|_F^2 + \frac{1}{2}\|\hat{\mathcal{A}}^*(\hat{\Omega}) - Wx^0 - c^0\|_2^2 \\ \text{s.t.} & W - G = 0, \\ & (G, W, \hat{\Omega}) \in \mathcal{S}_+^n \times \mathcal{S}^n \times \mathcal{S}_+^p. \end{array} \tag{11}$$

The augmented Lagrangian function of (11) is defined by

$$L_\rho(G, W, \hat{\Omega}, M) := \frac{1}{2}\|G - G^0\|_F^2 + \frac{1}{2}\|\hat{\mathcal{A}}^*(\hat{\Omega}) - Wx^0 - c^0\|_2^2 + \langle M, W - G\rangle + \frac{\rho}{2}\|W - G\|_F^2,$$

where M denotes the multiplier for the equality constraint in (11) and $\rho > 0$ denotes the penalty parameter.

Now, we give the details of our main algorithm for solving (11) as follows:

Algorithm: ADM-ISDQP:

Step 0: *Choose the initial pair* $(W^1, \hat{\Omega}^1, M^1)$ *and set the counter* $k := 1$.

Step 1: *Solve the following* G*-subproblem to obtain* G^{k+1},

$$G^{k+1} := \mathit{argmin}\{L_\rho(G, W^k, \hat{\Omega}^k, M^k) : G \in \mathcal{S}_+^n\}. \tag{12}$$

Step 2: *Obtain* $(W^{k+1}, \hat{\Omega}^{k+1})$ *by solving the following* $(W, \hat{\Omega})$*-subproblem,*

$$(W^{k+1}, \hat{\Omega}^{k+1}) := \mathit{argmin}\{L_\rho(G^{k+1}, W, \hat{\Omega}, M^k) : (W, \hat{\Omega}) \in \mathcal{S}^n \times \mathcal{S}_+^p\}. \tag{13}$$

Step 3: *Update the multiplier* M,

$$M^{k+1} := M^k + \rho(W^{k+1} - G^{k+1}).$$

Step 4: *If the new pair* $(G^{k+1}, W^{k+1}, \hat{\Omega}^{k+1}, M^{k+1})$ *satisfies the given stopping criterion, then stop; Otherwise, go to Step 5.*

Step 5: *Set* $k := k + 1$, *go to Step 1.*

Remark 2. The stopping criterion of ADM-ISDQP will be specified in Section 3.4.

Next, we analyze how to solve the subproblems mentioned in the above algorithm.

3.1. **Solving the subproblem (12).** In this subsection, we derive the explicit expression of G^{k+1}. From (12) and Lemma 2.4, we obtain

$$0 \in (G^{k+1} - G^0) - M^k + \rho(G^{k+1} - W^k) + \mathcal{N}_{\mathcal{S}_+^n}(G^{k+1}),$$

which implies that

$$0 \in G^{k+1} - \left(\frac{1}{1+\rho}(G^0 + M^k + \rho W^k)\right) + \mathcal{N}_{\mathcal{S}_+^n}(G^{k+1}). \tag{14}$$

It follows from (14) and the relation (5) that

$$G^{k+1} = \Pi_{\mathcal{S}_+^n}\left(\frac{1}{1+\rho}(G^0 + M^k + \rho W^k)\right). \tag{15}$$

If the matrix $\frac{1}{1+\rho}(G^0 + M^k + \rho W^k)$ in (15) has the following eigenvalue decomposition

$$\frac{1}{1+\rho}(G^0 + M^k + \rho W^k) = P^k \Xi^k (P^k)^T,$$

where $P^k \in \mathcal{O}^n$, $\Xi^k = \text{diag}\left(\theta_1^k, \cdots, \theta_n^k\right)$ and $\theta_i^k (i = 1, \cdots, n)$ are all eigenvalues of $\frac{1}{1+\rho}(G^0 + M^k + \rho W^k)$. From (15), the explicit expression of G^{k+1} is defined as follows:

$$G^{k+1} = P^k \Xi_+^k (P^k)^T,$$

where $\Xi_+^k = \text{diag}\left([\theta_1^k]_+, \cdots, [\theta_n^k]_+\right)$ and $[\theta_i^k]_+ = \max\{0, \theta_i^k\}$, $(i = 1, \cdots, n)$.

3.2. **Solving the subproblem (13).** Now we turn to the issue of solving the $(W, \hat{\Omega})$-subproblem (13), that is,

$$(W^{k+1}, \hat{\Omega}^{k+1}) = \text{argmin}\{L_\rho(G^{k+1}, W, \hat{\Omega}, M^k) : (W, \hat{\Omega}) \in \mathcal{S}^n \times \mathcal{S}_+^p\}.$$

The reformulation of the above problem is given by

$$\min_{\hat{\Omega} \in \mathcal{S}_+^p} \min_{W \in \mathcal{S}^n} L_\rho(G^{k+1}, W, \hat{\Omega}, M^k). \tag{16}$$

Let us denote

$$W_k(\hat{\Omega}) := \text{argmin}_{W \in \mathcal{S}^n} l_k(W), \tag{17}$$

where $l_k(W)$ is defined by

$$l_k(W) := L_\rho(G^{k+1}, W, \hat{\Omega}, M^k). \tag{18}$$

Hence, we can rewrite (16) as follows:

$$\min_{\hat{\Omega} \in \mathcal{S}_+^p} \left\{ f_k(\hat{\Omega}) := L_\rho(G^{k+1}, W_k(\hat{\Omega}), \hat{\Omega}, M^k) \right\}. \tag{19}$$

It is clear that the function $l_k(W)$ in (18) is a convex quadratic function with respect to the variables $(W, \hat{\Omega})$. In addition, the function $l_k(W)$ is a strictly convex quadratic function of $(W, \hat{\Omega})$, provided that the operator $\hat{\mathcal{A}}^*$ satisfies the following condition:

$$ker(\hat{\mathcal{A}}^*) := \{Z : \hat{\mathcal{A}}^*(Z) = 0\} = \{0\}.$$

Based on the above observation and Lemma 2.6, the following theorem presents the convexity of $f_k(\hat{\Omega})$.

Theorem 3.2. *The objective function $f_k(\hat{\Omega})$ in (19) is a convex quadratic function of $\hat{\Omega}$. In addition, $f_k(\hat{\Omega})$ is a strictly convex quadratic function if $ker(\hat{\mathcal{A}}^*) = \{0\}$.*

It follows from Theorem 3.2 that $f_k(\hat{\Omega})$ takes the form of

$$f_k(\hat{\Omega}) = \varphi_k(\hat{\Omega}) + \text{constant},$$

where $\varphi_k(\hat{\Omega}) := \langle \mathcal{J} f_k(0), \hat{\Omega} \rangle + \frac{1}{2}\langle \hat{\Omega}, \mathcal{J}^2 f_k(0) \hat{\Omega} \rangle$. Before deriving the explicit expressions of $\mathcal{J} f_k(0)$ and $\mathcal{J}^2 f_k(0)$, we present some useful results in the following lemmas.

Lemma 3.3. *The directional derivative of $l_k(W)$ at W along the direction H is expressed by*

$$l_k'(W; H) = \left\langle M^k - \rho G^{k+1} + \mathcal{L}_{x^0}(c^0 - \hat{\mathcal{A}}^*(\hat{\Omega})) + \mathcal{L}_{(x^0(x^0)^T + \rho I_n)}(W), H \right\rangle, \tag{20}$$

where the function $l_k(W)$ is defined in (18). In addition, the optimal solution of (17) is given by

$$W_k(\hat{\Omega}) = -\mathcal{L}^{-1}_{(x^0(x^0)^T+\rho I_n)}\left(M^k - \rho G^{k+1} + \mathcal{L}_{x^0}(c^0 - \hat{\mathcal{A}}^*(\hat{\Omega}))\right). \tag{21}$$

Proof. By the definition of $l_k(W)$, we obtain

$$\begin{aligned}
& l_k(W+tH) - l_k(W)\\
=&\frac{1}{2}\|\hat{\mathcal{A}}^*(\hat{\Omega}) - (W+tH)x^0 - c^0\|_2^2 - \frac{1}{2}\|\hat{\mathcal{A}}^*(\hat{\Omega}) - Wx^0 - c^0\|_2^2 + \langle M^k, tH\rangle\\
&+ \frac{\rho}{2}\|(W+tH) - G^{k+1}\|_F^2 - \frac{\rho}{2}\|W - G^{k+1}\|_F^2\\
=&\frac{1}{2}\langle tHx^0, Wx^0 + c^0 - \hat{\mathcal{A}}^*(\hat{\Omega})\rangle_v + \frac{1}{2}\langle Wx^0 + c^0 - \hat{\mathcal{A}}^*(\hat{\Omega}), tHx^0\rangle_v + \frac{1}{2}\|tHx^0\|_2^2\\
&+ \langle M^k, tH\rangle + \langle \rho(W - G^{k+1}), tH\rangle + \frac{\rho}{2}\|tH\|_F^2\\
=&\langle M^k - \rho G^{k+1}, tH\rangle + \left\langle \frac{(c^0 - \hat{\mathcal{A}}^*(\hat{\Omega}))(x^0)^T + (x^0)(c^0 - \hat{\mathcal{A}}^*(\hat{\Omega}))^T}{2}, tH\right\rangle\\
&+ \left\langle \frac{W(x^0(x^0)^T + \rho I_n) + (x^0(x^0)^T + \rho I_n)W}{2}, tH\right\rangle + \frac{1}{2}\|tHx^0\|_2^2 + \frac{\rho}{2}\|tH\|_F^2,
\end{aligned}$$

which implies that

$$l_k'(W;H) = \left\langle M^k - \rho G^{k+1} + \mathcal{L}_{x^0}(c^0 - \hat{\mathcal{A}}^*(\hat{\Omega})) + \mathcal{L}_{(x^0(x^0)^T+\rho I_n)}(W), H\right\rangle.$$

Now, we turn to the remaining part. It follows from (17) that

$$l_k'(W_k(\hat{\Omega}); H) = 0, \quad \forall H \in \mathcal{S}^n. \tag{22}$$

From (20) and (22), we can easily obtain

$$M^k - \rho G^{k+1} + \mathcal{L}_{x^0}(c^0 - \hat{\mathcal{A}}^*(\hat{\Omega})) + \mathcal{L}_{(x^0(x^0)^T+\rho I_n)}(W_k(\hat{\Omega})) = 0. \tag{23}$$

Hence, we get

$$W_k(\hat{\Omega}) = -\mathcal{L}^{-1}_{(x^0(x^0)^T+\rho I_n)}\left(M^k - \rho G^{k+1} + \mathcal{L}_{x^0}(c^0 - \hat{\mathcal{A}}^*(\hat{\Omega}))\right).$$

The proof is complete. □

One can observe from (21) that $W_k(\hat{\Omega})$ is a linear operator. Hence,

$$W_k(\hat{\Omega} + tZ) = W_k(\hat{\Omega}) + tW_k'(\hat{\Omega}; Z). \tag{24}$$

Combining with the equation (24), the following lemmas present the explicit expressions of the derivative and the second-order derivative of $f_k(\hat{\Omega})$.

Lemma 3.4. *The derivative of $f_k(\hat{\Omega})$ is expressed by*

$$\mathcal{J}f_k(\hat{\Omega}) = \hat{\mathcal{A}}\left(\hat{\mathcal{A}}^*(\hat{\Omega}) - W_k(\hat{\Omega})x^0 - c^0\right),$$

where the function $f_k(\hat{\Omega})$ is defined in (19).

Proof. Let us denote

$$T_1^k := W_k(\hat{\Omega} + tZ)x^0 - W_k(\hat{\Omega})x^0 \quad \text{and} \quad T_2^k := \hat{\mathcal{A}}^*(\hat{\Omega} + tZ) - \hat{\mathcal{A}}^*(\hat{\Omega}).$$

By the definition of $f_k(\hat{\Omega})$, we have

$$\begin{aligned}
& f_k(\hat{\Omega}+tZ)-f_k(\hat{\Omega}) \\
=& \frac{1}{2}\|W_k(\hat{\Omega}+tZ)x^0+c^0-\hat{\mathcal{A}}^*(\hat{\Omega}+tZ)\|_2^2-\frac{1}{2}\|W_k(\hat{\Omega})x^0+c^0-\hat{\mathcal{A}}^*(\hat{\Omega})\|_2^2 \\
& +\langle M^k, W_k(\hat{\Omega}+tZ)-W_k(\hat{\Omega})\rangle+\frac{\rho}{2}\|W_k(\hat{\Omega}+tZ)-G^{k+1}\|_F^2 \\
& -\frac{\rho}{2}\|W_k(\hat{\Omega})-G^{k+1}\|_F^2 \\
=& \langle M^k-\rho G^{k+1}, W_k(\hat{\Omega}+tZ)-W_k(\hat{\Omega})\rangle \\
& +\frac{\rho}{2}\langle W_k(\hat{\Omega}+tZ)+W_k(\hat{\Omega}), W_k(\hat{\Omega}+tZ)-W_k(\hat{\Omega})\rangle \\
& +\frac{1}{2}\left\langle T_1^k-T_2^k, W_k(\hat{\Omega}+tZ)x^0+W_k(\hat{\Omega})x^0-\hat{\mathcal{A}}^*(\hat{\Omega}+tZ)-\hat{\mathcal{A}}^*(\hat{\Omega})+2c^0\right\rangle_v \\
=& \left\langle M^k-\rho G^{k+1}+\frac{W_k(\hat{\Omega})\rho I_n+\rho I_n W_k(\hat{\Omega})}{2}, W_k(\hat{\Omega}+tZ)-W_k(\hat{\Omega})\right\rangle \\
& +\frac{1}{2}\left\langle T_1^k-T_2^k, W_k(\hat{\Omega}+tZ)x^0+W_k(\hat{\Omega})x^0-\hat{\mathcal{A}}^*(\hat{\Omega}+tZ)-\hat{\mathcal{A}}^*(\hat{\Omega})+2c^0\right\rangle_v \\
& +\frac{\rho}{2}\|W_k(\hat{\Omega}+tZ)-W_k(\hat{\Omega})\|_F^2 \\
=& \langle M^k-\rho G^{k+1}, W_k(\hat{\Omega}+tZ)-W_k(\hat{\Omega})\rangle+\frac{\rho}{2}\|W_k(\hat{\Omega}+tZ)-W_k(\hat{\Omega})\|_F^2 \\
=& \frac{\rho}{2}\|W_k'(\hat{\Omega};tZ)\|_F^2+\frac{1}{2}\left\langle W_k'(\hat{\Omega};tZ)x^0, W_k'(\hat{\Omega};tZ)x^0-\hat{\mathcal{A}}^*(tZ)\right\rangle_v \\
& -\frac{1}{2}\left\langle \hat{\mathcal{A}}^*(tZ), 2W_k(\hat{\Omega})x^0+W_k'(\hat{\Omega};tZ)x^0-2\hat{\mathcal{A}}^*(\hat{\Omega})-\hat{\mathcal{A}}^*(tZ)+2c^0\right\rangle_v, \qquad (25)
\end{aligned}$$

where the last equation uses the equation (24). From (25), we obtain

$$f_k'(\hat{\Omega};Z)=\left\langle \hat{\mathcal{A}}^*(\hat{\Omega})-W_k(\hat{\Omega})x^0-c^0, \hat{\mathcal{A}}^*(Z)\right\rangle_v.$$

It follows from $f_k'(\hat{\Omega};Z)=\left\langle \mathcal{J} f_k(\hat{\Omega}), Z\right\rangle$ that

$$\mathcal{J} f_k(\hat{\Omega})=\hat{\mathcal{A}}\left(\hat{\mathcal{A}}^*(\hat{\Omega})-W_k(\hat{\Omega})x^0-c^0\right).$$

The proof is complete. □

Lemma 3.5. *The second-order derivative of $f_k(\hat{\Omega})$ is expressed by*

$$\mathcal{J}^2 f_k(\hat{\Omega})Z=\frac{2\rho}{2\rho+(x^0)^Tx^0}\left(\hat{\mathcal{A}}(\hat{\mathcal{A}}^*(Z))-\frac{1}{2(\rho+(x^0)^Tx^0)}(\hat{\mathcal{A}}^*(Z))^Tx^0\hat{\mathcal{A}}(x^0)\right),$$

where the function $f_k(\hat{\Omega})$ is defined in (19).

Proof. By the definition of $\mathcal{J} f_k(\cdot)$, we have

$$\mathcal{J} f_k(\hat{\Omega}+tZ)-\mathcal{J} f_k(\hat{\Omega})=\hat{\mathcal{A}}\left(\hat{\mathcal{A}}^*(tZ)-W_k'(\hat{\Omega};tZ)x^0\right). \qquad (26)$$

From (26), we obtain

$$\mathcal{J}^2 f_k(\hat{\Omega})Z=\hat{\mathcal{A}}\left(\hat{\mathcal{A}}^*(Z)\right)-\hat{\mathcal{A}}\left(W_k'(\hat{\Omega};Z)x^0\right), \qquad (27)$$

It follows from the equation (23) that

$$M^k-\rho G^{k+1}+\mathcal{L}_{x^0}(c^0-\hat{\mathcal{A}}^*(\hat{\Omega}+Z))+\mathcal{L}_{(x^0(x^0)^T+\rho I_n)}(W_k(\hat{\Omega}+Z))=0. \qquad (28)$$

Combining (23) with (28), we obtain

$$(x^0)^T x^0 (x^0)^T \hat{\mathcal{A}}^*(Z) = \rho (x^0)^T W_k'(\hat{\Omega}; Z) x^0 + (x^0)^T x^0 (x^0)^T W_k'(\hat{\Omega}; Z) x^0 \tag{29}$$

and

$$\begin{aligned} &(x^0)^T x^0 \hat{\mathcal{A}}(\hat{\mathcal{A}}^*(Z)) + \hat{\mathcal{A}}(x^0 (\hat{\mathcal{A}}^*(Z))^T x^0) \\ =&2\rho \hat{\mathcal{A}}(W_k'(\hat{\Omega}; Z) x^0) + (x^0)^T W_k'(\hat{\Omega}; Z) x^0 \hat{\mathcal{A}}(x^0) + (x^0)^T x^0 \hat{\mathcal{A}}(W_k'(\hat{\Omega}; Z) x^0). \end{aligned} \tag{30}$$

From (29), we have

$$(x^0)^T W_k'(\hat{\Omega}; Z) x^0 = \left(\frac{(x^0)^T x^0}{\rho + (x^0)^T x^0} \right) (x^0)^T \hat{\mathcal{A}}^*(Z). \tag{31}$$

The equations (30) and (31) imply that

$$\begin{aligned} &(2\rho + (x^0)^T x^0) \hat{\mathcal{A}}(W_k'(\hat{\Omega}; Z) x^0) \\ =&(x^0)^T x^0 \hat{\mathcal{A}}(\hat{\mathcal{A}}^*(Z)) + \left(1 - \frac{(x^0)^T x^0}{\rho + (x^0)^T x^0} \right) (\hat{\mathcal{A}}^*(Z))^T x^0 \hat{\mathcal{A}}(x^0). \end{aligned} \tag{32}$$

Hence, we can deduce that

$$\begin{aligned} &\mathcal{J}^2 f_k(\hat{\Omega}) Z \\ =&\frac{2\rho}{2\rho + (x^0)^T x^0} \hat{\mathcal{A}}(\hat{\mathcal{A}}^*(Z)) - \frac{\rho}{(2\rho + (x^0)^T x^0)(\rho + (x^0)^T x^0)} (\hat{\mathcal{A}}^*(Z))^T x^0 \hat{\mathcal{A}}(x^0) \\ =&\frac{2\rho}{2\rho + (x^0)^T x^0} \left(\hat{\mathcal{A}}(\hat{\mathcal{A}}^*(Z)) - \frac{1}{2(\rho + (x^0)^T x^0)} (\hat{\mathcal{A}}^*(Z))^T x^0 \hat{\mathcal{A}}(x^0) \right), \end{aligned}$$

which follows from (27) and (32). □

Remark 3. From Lemma 3.4 and Lemma 3.5, we obtain

$$\mathcal{J} f_k(0) = -\hat{\mathcal{A}} \left(W_k(0) x^0 \right) - \hat{\mathcal{A}} \left(c^0 \right), \tag{33}$$

$$\mathcal{J}^2 f_k(0) \hat{\Omega} = \frac{2\rho}{2\rho + (x^0)^T x^0} \left(\hat{\mathcal{A}}(\hat{\mathcal{A}}^*(\hat{\Omega})) - \frac{1}{2(\rho + (x^0)^T x^0)} (\hat{\mathcal{A}}^*(\hat{\Omega}))^T x^0 \hat{\mathcal{A}}(x^0) \right). \tag{34}$$

From (23), we obtain

$$H^k + \mathcal{L}_{(x^0 (x^0)^T + \rho I_n)}(W_k(0)) = 0, \tag{35}$$

where $H^k := M^k - \rho G^{k+1} + \mathcal{L}_{x^0}(c^0)$. As seen in the proof of Lemma 3.5 and together with (35), we have

$$(x^0)^T H^k x^0 + (\rho + (x^0)^T x^0)(x^0)^T W_k(0) x^0 = 0, \tag{36}$$

$$2\hat{\mathcal{A}}(H^k x^0) + (2\rho + (x^0)^T x^0) \hat{\mathcal{A}}(W_k(0) x^0) + (x^0)^T W_k(0) x^0 \hat{\mathcal{A}}(x^0) = 0. \tag{37}$$

It follows from (33), (36) and (37) that

$$\mathcal{J} f_k(0) = \frac{2}{2\rho + (x^0)^T x^0} \left(\hat{\mathcal{A}}(H^k x^0) - \frac{(x^0)^T H^k x^0}{2(\rho + (x^0)^T x^0)} \hat{\mathcal{A}}(x^0) \right) - \hat{\mathcal{A}}(c^0), \tag{38}$$

$$\mathcal{J} f_k(\hat{\Omega}) = \mathcal{J}^2 f_k(0) \hat{\Omega} + \mathcal{J} f_k(0). \tag{39}$$

Hence, Problem (19) is equivalent to the following convex semi-definite quadratic programming problem

$$\begin{array}{ll} \min & \varphi_k(\hat{\Omega}) \\ \text{s.t.} & \hat{\Omega} \in \mathcal{S}_+^p, \end{array} \tag{40}$$

where $\mathcal{J}^2 f_k(0)\hat{\Omega}$ and $\mathcal{J} f_k(0)$ are defined in (34) and (38), respectively. Without loss of generality, we suppose that the assumption $ker(\hat{\mathcal{A}}^*) = \{0\}$ holds in the sequel.

3.3. **Spectral projection gradient method.** In this subsection, we design a subroutine for solving (40) via spectral projection gradient (SPG) method, which was originally proposed to solve convex constraints optimization problems in nonlinear programming by Birgin, Martínez and Raydan [6]. More discussions on SPG methods can be found in [7] and references therein.

The SPG method combines the global Barzila-Borwein (BB) step strategy [23] with the projected gradient (PG) method [5]. The BB step strategy was firstly proposed by Barzila and Borwein [4] for the unconstrained optimization problem. The philosophy of this strategy is to generate the step length by the information of the successive two iterations. The form of BB step is defined by

$$x^{k+1} := x^k - \lambda_k \nabla f(x^k), \quad \lambda_k := \frac{s_{k-1}^T s_{k-1}}{s_{k-1}^T y_{k-1}},$$

where $f : \mathcal{R}^n \to \mathcal{R}$ is a differentiable function and the definitions of s_{k-1} and y_{k-1} are given by

$$s_{k-1} := x^k - x^{k-1}, \quad y_{k-1} := \nabla f(x^k) - \nabla f(x^{k-1}).$$

Barzila and Borwein [4] proved that the BB method converges R-superlinearly for the two dimensional strictly convex quadratic function. The result of Barzila and Borwein has triggered off many researches on the theory based on BB methods during the decades. For instance, Raydan [22] provided global convergence of BB method for the n-dimensional case and the corresponding expected R-linear convergence was established by Dai et al. [11]. Friedlander et al. [13] proposed several useful choices of step length for projected gradient methods.

Before describing the details of our subroutine for solving (40), we define the following notation

$$\Xi(\cdot) := \frac{2\rho}{2\rho + (x^0)^T x^0}\left(\hat{\mathcal{A}}(\hat{\mathcal{A}}^*(\cdot)) - \frac{1}{2(\rho + (x^0)^T x^0)}(\hat{\mathcal{A}}^*(\cdot))^T x^0 \hat{\mathcal{A}}(x^0)\right).$$

Hence, from (34), (39) and the relation between $f_k(\hat{\Omega})$ and $\varphi_k(\hat{\Omega})$, we know

$$\mathcal{J}\varphi_k(\hat{\Omega}) = \mathcal{J} f_k(0) + \Xi(\hat{\Omega}), \quad \varphi_k(\hat{\Omega}) = \frac{1}{2}\left\langle \mathcal{J} f_k(0) + \mathcal{J}\varphi_k(\hat{\Omega}), \hat{\Omega}\right\rangle.$$

The procedure of our subroutine SPGM has the following steps:

Subroutine: SPGM:

Step 0 (Initial): *Set the initial parameters as follows:*

$$\begin{aligned}
&\gamma := 0.1,\ \sigma_1 := 0.1,\ \sigma_2 := 0.9,\ \lambda^k_{min} := 10^{-30},\\
&\lambda^k_{max} := 10^{30},\ tol := 10^{-7},\ \hat{\Omega}^k_0 := \hat{\Omega}^k \in \mathcal{S}^p_+,\\
&\lambda^k_1 := \max\{\lambda^k_{min}, \min\{\lambda^k_{max}, 1/\|\Pi_{\mathcal{S}^p_+}(\hat{\Omega}^k_0 - \mathcal{J}\varphi_k(\hat{\Omega}^k_0)) - \hat{\Omega}^k_0\|_\infty\}\}
\end{aligned}$$

and set the counter $l := 1$.

Step 1 (Termination Test): *If the following condition holds*

$$\|\Pi_{\mathcal{S}^p_+}(\hat{\Omega}^k_l - \mathcal{J}\varphi_k(\hat{\Omega}^k_l)) - \hat{\Omega}^k_l\| \le tol,$$

then stop and declare that $\hat{\Omega}^k_l$ *is the approximate solution of (40).*

Step 2 (Generate the next iterate):

Step 2.1 (Compute the search direction):

$$D_l^k := \Pi_{\mathcal{S}_+^p}(\hat{\Omega}_l^k - \lambda_l^k \mathcal{J}\varphi_k(\hat{\Omega}_l^k)) - \hat{\Omega}_l^k.$$

Step 2.2 (Compute the step using a line search procedure):

Step 2.2.1: *Set* $\alpha^k := 1$.

Step 2.2.2: *If* $\varphi_k(\hat{\Omega}_l^k + \alpha^k D_l^k) \leq \varphi_k(\hat{\Omega}_l^k) + \gamma\alpha^k\langle \mathcal{J}\varphi_k(\hat{\Omega}_l^k), D_l^k\rangle$, *then* $\alpha_l^k := \alpha^k$, $\hat{\Omega}_{l+1}^k := \hat{\Omega}_l^k + \alpha_l^k D_l^k$ *and go to Step 3; Otherwise, go to Step 2.2.3.*

Step 2.2.3: *Compute a new trial step length*

$$\alpha_{tmp}^k := -\frac{(\alpha^k)^2\langle \mathcal{J}\varphi_k(\hat{\Omega}_l^k), D_l^k\rangle}{2[\varphi_k(\hat{\Omega}_l^k + \alpha^k D_l^k) - \varphi_k(\hat{\Omega}_l^k) - \alpha^k\langle \mathcal{J}\varphi_k(\hat{\Omega}_l^k), D_l^k\rangle]}.$$

If $\alpha_{tmp}^k \in [\sigma_1, \sigma_2\alpha^k]$, *then* $\alpha^k := \alpha_{tmp}^k$; *Otherwise, set* $\alpha^k := 0.5\alpha^k$. *Go to Step 2.2.2.*

Step 3 (Generate the next BB step): *Compute* $S_l^k = \alpha_l^k D_l^k$, $Y_l^k = \Xi(S_l^k)$. *Generate the next BB step*

$$\lambda_{l+1}^k := \frac{\|S_l^k\|_F^2}{\langle S_l^k, Y_l^k\rangle}.$$

Step 4 (Update the counter): *Set* $l := l+1$, *go to Step 1.*

Remark 4. Our subroutine is similar to the SPG method in [7, Algorithm 2.1]. However, there are some differences between ours and theirs: (a) In Step 3 of our subroutine SPGM, the safeguard of the next BB step is not required. Because of the strict convexity of the objective function, the next BB step defined in Step 3 is always positive. (b) In our subroutine, the objective function of (40) is a convex function of matrix variable $\hat{\Omega}$ and the convex set is $\mathcal{S}_+^p$.

Before establishing the convergence result of SPGM, we present some useful lemmas, which are similar to [6, Lemma 2.1 and Lemma 2.2]. For notional simplicity, we denote

$$g^{\varphi_k}(X) := \mathcal{J}\varphi_k(X) \quad \text{and} \quad g_t^{\varphi_k}(X) := \Pi_{\mathcal{S}_+^p}(X - t\mathcal{J}\varphi_k(X)) - X.$$

Lemma 3.6. *For all* $X \in \mathcal{S}_+^p$, $t \in (0, \lambda_{\max}^k]$,

(i) $\langle g^{\varphi_k}(X), g_t^{\varphi_k}(X)\rangle \leq -\frac{1}{t}\|g_t^{\varphi_k}(X)\|_F^2 \leq -\frac{1}{\lambda_{\max}^k}\|g_t^{\varphi_k}(X)\|_F^2$.

(ii) The matrix $g_t^{\varphi_k}(X^*)$ *vanishes if and only if* X^* *is a constrained stationary point that satisfies the following relation*

$$\langle g^{\varphi_k}(X^*), X - X^*\rangle \geq 0, \quad \forall\ X \in \mathcal{S}_+^p.$$

Proof. By the definition of $g_t^{\varphi_k}(X)$, we get

$$\Pi_{\mathcal{S}_+^p}(X - tg^{\varphi_k}(X)) = g_t^{\varphi_k}(X) + X. \tag{41}$$

From the property of $\Pi_{\mathcal{S}_+^p}(\cdot)$ and the fact that $X \in \mathcal{S}_+^p$, we deduce that the following condition holds

$$\langle X - tg^{\varphi_k}(X) - \Pi_{\mathcal{S}_+^p}(X - tg^{\varphi_k}(X)), X - \Pi_{\mathcal{S}_+^p}(X - tg^{\varphi_k}(X))\rangle \leq 0.$$

Hence, we obtain

$$\langle g^{\varphi_k}(X), g_t^{\varphi_k}(X)\rangle \leq -\frac{1}{t}\|g_t^{\varphi_k}(X)\|_F^2.$$

Now, we turn to prove the second part of the conclusion. From (41), we know

$$g_t^{\varphi_k}(X^*) = 0 \Leftrightarrow \Pi_{\mathcal{S}_+^p}(X^* - tg^{\varphi_k}(X^*)) = X^*.$$

By the property of $\Pi_{\mathcal{S}^p_+}(\cdot)$, the above right-hand side relation is equivalent to the following condition

$$\langle X^* - tg^{\varphi_k}(X^*) - X^*, X - X^* \rangle \le 0, \quad \forall X \in \mathcal{S}^p_+,$$

that is,

$$\langle g^{\varphi_k}(X^*), X - X^* \rangle \ge 0, \quad \forall X \in \mathcal{S}^p_+.$$

The proof is complete. □

Lemma 3.7. *The following conclusions hold:*

(i) For all $X \in \mathcal{S}^p_+$ and $Z \in \mathcal{S}^p$, the function $h : [0, \infty) \to \mathcal{R}$ given by

$$h(s) = \frac{\left\| \Pi_{\mathcal{S}^p_+}(X + sZ) - X \right\|_F}{s}$$

is monotonically non-increasing.

(ii) For all $X \in \mathcal{S}^p_+$, there exists $s_x > 0$ such that for all $t \in [0, s_x]$ it holds that

$$\varphi_k(X) \ge \varphi_k(\Pi_{\mathcal{S}^p_+}(X - tg^{\varphi_k}(X))) - \gamma \langle g^{\varphi_k}(X), g^{\varphi_k}_t(X) \rangle. \tag{42}$$

Proof. (i) We take two scalars s_1, s_2 and $s_2 > s_1 > 0$. Denote

$$X_1 := X + s_1 Z, \quad X_2 := X + s_2 Z$$

and

$$\bar{X}_1 := \Pi_{\mathcal{S}^p_+}(X_1), \quad \bar{X}_2 := \Pi_{\mathcal{S}^p_+}(X_2).$$

Hence, we only need to show that for arbitrary $s_2 > s_1 > 0$, the following relation holds:

$$\frac{\|\bar{X}_2 - X\|_F}{s_2} \le \frac{\|\bar{X}_1 - X\|_F}{s_1}. \tag{43}$$

(a) If $\bar{X}_1 = \bar{X}_2$, then we can easily deduce that (43) holds.

(b) If $\bar{X}_1 = X$, then $\bar{X}_2 = X$ and (43) holds. By the definition of $\bar{X}_1$, we obtain

$$\bar{X}_1 = X \Leftrightarrow \Pi_{\mathcal{S}^p_+}(X + s_1 Z) = X.$$

The above right-hand side relation is equivalent to the following condition

$$\langle X + s_1 Z - X, Y - X \rangle \le 0, \quad \forall Y \in \mathcal{S}^p_+,$$

namely,

$$\langle Z, Y - X \rangle \le 0, \quad \forall Y \in \mathcal{S}^p_+.$$

By the fact that $s_2 > s_1 > 0$, we get

$$\langle s_2 Z, Y - X \rangle \le 0, \quad \forall Y \in \mathcal{S}^p_+,$$

which implies

$$\langle X + s_2 Z - X, Y - X \rangle \le 0, \quad \forall Y \in \mathcal{S}^p_+.$$

Hence, we can deduce that $\Pi_{\mathcal{S}^p_+}(X + s_2 Z) = X$. That is, $\bar{X}_2 = X$.

(c) If $X_1 \in \mathcal{S}^p_+$, then $\bar{X}_1 = X_1$ and (43) is equivalent to

$$\|\bar{X}_2 - X\|_F \le s_2 \|Z\|_F. \tag{44}$$

By the definitions of $\bar{X}_2$ and X_2, (44) becomes

$$\|\Pi_{\mathcal{S}^p_+}(X + s_2 Z) - X\|_F \le s_2 \|Z\|_F.$$

The above relation can be obtained by the non-expansive property of $\Pi_{\mathcal{S}^p_+}(\cdot)$. Hence, (43) holds in this case.

(d) We only need to show that when $\bar{X}_1 \neq \bar{X}_2$, $\bar{X}_1 \neq X$, $\bar{X}_2 \neq X$ and $X_1 \notin \mathcal{S}_+^p$, (43) also holds. We take W that satisfies $\langle W, \bar{X}_2 - \bar{X}_1\rangle = 0$. Denote

$$l_1 := \left\{Y : \langle W, Y - \bar{X}_1\rangle = 0\right\},\ l_2 := \left\{Y : \langle W, Y - \bar{X}_2\rangle = 0\right\}$$

and

$$l_3 := \{Y : Y = X + \alpha(X_2 - X), \alpha \in R\},\ l_4 := \left\{Y : Y = X + \beta(\bar{X}_1 - X), \beta \in R\right\}.$$

By the definitions of l_1 and l_2, we know that $\bar{X}_1$ lies on l_1, $\bar{X}_2$ lies on l_2, and l_1 parallels l_2. Due to the definitions of $\bar{X}_1$ and $\bar{X}_2$, we obtain the following conditions

$$\langle \bar{X}_2 - \bar{X}_1, X_1 - \bar{X}_1\rangle \leq 0, \quad \langle \bar{X}_2 - \bar{X}_1, X_2 - \bar{X}_2\rangle \geq 0.$$

Hence, neither X_1 nor X_2 lies strictly between l_1 and l_2. Moreover, X lies on the same side of l_1 as X_1. Denote the intersections of l with l_1 and l_2 by X_1^s and X_2^s, respectively. The intersection of l_4 with l_2 is denoted by X_3^s. Hence,

$$\begin{aligned}
\frac{s_2}{s_1} &= \frac{\|X_2 - X\|_F}{\|X_1 - X\|_F} \geq \frac{\|X_2^s - X\|_F}{\|X_1^s - X\|_F} = \frac{\|X_3^s - X\|_F}{\|\bar{X}_1 - X\|_F} \\
&= \frac{\|X_3^s - \bar{X}_1\|_F + \|\bar{X}_1 - X\|_F}{\|\bar{X}_1 - X\|_F} \geq \frac{\|\bar{X}_2 - \bar{X}_1\|_F + \|\bar{X}_1 - X\|_F}{\|\bar{X}_1 - X\|_F} \\
&\geq \frac{\|\bar{X}_2 - X\|_F}{\|\bar{X}_1 - X\|_F},
\end{aligned}$$

which implies that (43) holds in this case.

(ii) (a) If X is a stationary point, then from (ii) of Lemma 3.6 we obtain

$$g_t^{\varphi_k}(X) = 0,$$

which implies that the inequality (42) holds.

(b) If X is not a stationary point, then from (ii) of Lemma 3.6 we obtain

$$g_t^{\varphi_k}(X) \neq 0.$$

Now, by the mean-value theorem, it follows that

$$\begin{aligned}
&\varphi_k(X) - \varphi_k(\Pi_{\mathcal{S}_+^p}(X - t g^{\varphi_k}(X))) \\
=&\langle g^{\varphi_k}(X), -g_t^{\varphi_k}(X)\rangle + \langle g^{\varphi_k}(\Theta) - g^{\varphi_k}(X), -g_t^{\varphi_k}(X)\rangle,
\end{aligned}$$

where Θ lies on the segment joining X and $\Pi_{\mathcal{S}_+^p}(X - t g^{\varphi_k}(X))$. Hence, (42) is equivalent to the following condition

$$\langle g^{\varphi_k}(X), -g_t^{\varphi_k}(X)\rangle + \langle g^{\varphi_k}(\Theta) - g^{\varphi_k}(X), -g_t^{\varphi_k}(X)\rangle \geq \gamma\langle g^{\varphi_k}(X), -g_t^{\varphi_k}(X)\rangle,$$

that is,

$$(1-\gamma)\langle g^{\varphi_k}(X), -g_t^{\varphi_k}(X)\rangle \geq \langle g^{\varphi_k}(X) - g^{\varphi_k}(\Theta), -g_t^{\varphi_k}(X)\rangle.$$

From (i) of Lemma 3.6, we have

$$\langle g^{\varphi_k}(X), -g_t^{\varphi_k}(X)\rangle \geq \frac{1}{t}\|g_t^{\varphi_k}(X)\|_F^2 \geq \|g_1^{\varphi_k}(X)\|_F \cdot \|g_t^{\varphi_k}(X)\|_F.$$

Hence, the inequality (42) is satisfied for all $t \in (0, 1]$ such that

$$(1-\gamma)\|g_1^{\varphi_k}(X)\|_F \cdot \|g_t^{\varphi_k}(X)\|_F \geq \langle g^{\varphi_k}(X) - g^{\varphi_k}(\Theta), -g_t^{\varphi_k}(X)\rangle,$$

which implies that

$$(1-\gamma)\|g_1^{\varphi_k}(X)\|_F \geq \left\langle g^{\varphi_k}(X) - g^{\varphi_k}(\Theta), -\frac{g_t^{\varphi_k}(X)}{\|g_t^{\varphi_k}(X)\|_F}\right\rangle.$$

Therefore, there exists a positive scalar s_x satisfying the above inequality and the inequality (42) holds for all $t \in (0, s_x]$. $\square$

Combining Lemma 3.6 with Lemma 3.7, we can establish the convergence result of SPGM. The proof is similar to [6, Theorem 3.4].

Theorem 3.8. *Subroutine SPGM is well-defined, and any accumulation point of the sequence $\{\hat{\Omega}_l^k\}$ is a constrained stationary point $\hat{\Omega}_*^k$ satisfying the following relation*

$$\langle \mathcal{J}\varphi_k(\hat{\Omega}_*^k), \hat{\Omega} - \hat{\Omega}_*^k \rangle \geq 0, \quad \forall\, \hat{\Omega} \in \mathcal{S}_+^p.$$

3.4. **Stopping criterion of ADM-ISDQP.** Now, the stopping criterion and the convergence result of ADM-ISDQP will be discussed in this subsection. For simplicity, we denote the objective function in (10) by $F(G, \hat{\Omega})$, that is,

$$F(G, \hat{\Omega}) := \frac{1}{2}\|G - G^0\|_F^2 + \frac{1}{2}\|\hat{\mathcal{A}}^*(\hat{\Omega}) - Gx^0 - c^0\|_2^2.$$

Problem (10) can be rewritten as follows:

$$\begin{array}{ll} \min & F(G, \hat{\Omega}) \\ \text{s.t.} & (G, \hat{\Omega}) \in \mathcal{S}_+^n \times \mathcal{S}_+^p. \end{array} \tag{45}$$

Let us define the following notations:

$$\begin{aligned} \mathcal{J}F(G, \hat{\Omega}) &: \text{the derivative of } F \text{ at the pair } (G, \hat{\Omega}), \\ \mathcal{J}_G F(G, \hat{\Omega}) &: \text{the derivative of } F \text{ with respect to the variable } G, \\ \mathcal{J}_{\hat{\Omega}} F(G, \hat{\Omega}) &: \text{the derivative of } F \text{ with respect to the variable } \hat{\Omega}. \end{aligned}$$

If the pair $(G^*, \hat{\Omega}^*)$ is an optimal solution to (45), we obtain

$$0 \in \mathcal{J}F(G^*, \hat{\Omega}^*) + \mathcal{N}_{\mathcal{S}_+^n \times \mathcal{S}_+^p}(G^*, \hat{\Omega}^*), \tag{46}$$

where $\mathcal{J}F(G^*, \hat{\Omega}^*) = \Big(\mathcal{J}_G F(G^*, \hat{\Omega}^*), \mathcal{J}_{\hat{\Omega}} F(G^*, \hat{\Omega}^*)\Big)$, the derivatives $\mathcal{J}_G F(G^*, \hat{\Omega}^*)$ and $\mathcal{J}_{\hat{\Omega}} F(G^*, \hat{\Omega}^*)$ are expressed by

$$\begin{aligned} \mathcal{J}_G F(G^*, \hat{\Omega}^*) &= G^* - G^0 + \mathcal{L}_{x^0}\left(c^0 + G^* x^0 - \hat{\mathcal{A}}^*(\hat{\Omega}^*)\right), \\ \mathcal{J}_{\hat{\Omega}} F(G^*, \hat{\Omega}^*) &= \hat{\mathcal{A}}\left(\hat{\mathcal{A}}^*(\hat{\Omega}^*) - G^* x^0 - c^0\right). \end{aligned}$$

It follows from (5) that (46) is equivalent to

$$\begin{aligned} G^* - \Pi_{\mathcal{S}_+^n}\left(G^0 - \mathcal{L}_{x^0}(c^0 + G^* x^0 - \hat{\mathcal{A}}^*(\hat{\Omega}^*))\right) &= 0, \\ \hat{\Omega}^* - \Pi_{\mathcal{S}_+^p}\left(\hat{\Omega}^* - \hat{\mathcal{A}}(\hat{\mathcal{A}}^*(\hat{\Omega}^*) - G^* x^0 - c^0)\right) &= 0. \end{aligned}$$

In this paper, we set the stopping criterion as follows:

$$\operatorname{res}(G^k, \hat{\Omega}^k) := \max\{r_G^k, r_{\hat{\Omega}}^k\} \leq \epsilon, \tag{47}$$

where ϵ represents the tolerance of the residual and $r_G^k, r_{\hat{\Omega}}^k$ are defined by

$$\begin{aligned} r_G^k &:= \left\| G^k - \Pi_{\mathcal{S}_+^n}\left(G^0 - \mathcal{L}_{x^0}(c^0 + G^k x^0 - \hat{\mathcal{A}}^*(\hat{\Omega}^k))\right)\right\|_F, \\ r_{\hat{\Omega}}^k &:= \left\| \hat{\Omega}^k - \Pi_{\mathcal{S}_+^p}\left(\hat{\Omega}^k - \hat{\mathcal{A}}(\hat{\mathcal{A}}^*(\hat{\Omega}^k) - G^k x^0 - c^0)\right)\right\|_F. \end{aligned}$$

Remark 5. We can establish the convergence result of ADM-ISDQP as the monograph [8] about alternating direction method (ADM). A lot of theoretical results and numerical experiences on ADM have been developed in [8] for solving the following structured convex programming problem

$$\begin{array}{ll} \min & f(x) + g(y), \\ \text{s.t.} & Ax + By = c, \\ & x \in \mathcal{X}, y \in \mathcal{Y}. \end{array}$$

It is clear that (11) can be categorized as the above problem. Note that the stopping criterion of ADM-ISDQP is also similar to the setting in the monograph [8].

4. **Numerical experiments.**

In this section, we report some implementation issues and numerical experiments conducted for testing efficiency of algorithm ADM-ISDQP with subroutine SPGM.

All experiments are performed in MATLAB 2010a on a Intel (R) Core(TM) 2 of 3.3 GHz CPU and 3.23GB RAM under Ubuntu 12.04 LTS operating system.

Some technical details are summarized as follows:

(A) Initial point choice. Our algorithm starts its iterations with the initial iterate $(W^1, \hat{\Omega}^1, M^1) = (W_R, I_p, I_n)$, where the matrix W_R is generated randomly by using the built-in function **rand** in MATLAB and I_p denotes the identity matrix of size p. For convenience, we set all elements of x^0 be 1.

(B) The MATLAB code for generating G^0 is given by

$$G0 = 2.0 * rand(n,n) - ones(n,n); \quad G0 = G0 * G0';$$

$A_1, A_2, \cdots, A_n$ are $m \times m$ symmetric matrices, which have the following two cases:

(a) In Example 4.1, they are generated by using the built-in function **rand** in MATLAB as follows:

$$\begin{array}{l} A(m,m,n) = rand(m,m,n); \\ \text{for } i = 1:n \\ \quad A(:,:,i) = 0.5 * (A(:,:,i) + A(:,:,i)'); \\ \text{end} \end{array}$$

(b) In Example 4.2, they are $m \times m$ symmetric matrices with entries in [-1, 1]. The corresponding MATLAB codes are defined as follows:

$$\begin{array}{l} A(m,m,n) = rand(m,m,n); \\ \text{for } i = 1:n \\ \quad A(:,:,i) = triu(A(:,:,i)) + triu(A(:,:,i))'; \\ \text{end} \end{array}$$

(C) Varying penalty parameter. The initial penalty parameter ρ^1 is 1. The update scheme of ρ^k is similar to [8, Section 3.4.1],

$$\rho^{k+1} := \begin{cases} 5\rho^k & \text{if } r_G^k > 10 r_{\hat{\Omega}}^k \\ \rho^k/5 & \text{if } r_G^k < 10 r_{\hat{\Omega}}^k \\ \rho^k & \text{otherwise.} \end{cases}$$

(D) The stopping criterion. Our algorithm ADM-ISDQP stops when the condition (47) is met, where $\epsilon = 10^{-5}\sqrt{n}$.

TABLE 1. Numerical results of ADM-ISDQP for Example 4.1

n	m	r	it.	time	res.
50	30	10	9	1.98	3.72e-05
50	40	10	9	2.11	3.77e-05
50	50	10	9	3.03	3.74e-05
100	40	10	9	4.44	5.94e-05
100	50	10	9	6.15	5.89e-05
100	60	10	9	7.61	5.76e-05
200	50	10	9	15.95	9.82e-05
200	60	10	9	16.95	9.81e-05
200	70	10	9	20.24	9.83e-05
500	100	30	9	81.62	2.10e-05
500	100	40	9	81.03	2.09e-05
500	100	50	9	80.29	2.10e-05
1000	150	50	10	416.88	9.87e-05
1000	150	60	10	402.59	9.81e-05
1000	150	70	10	394.52	9.87e-05

In all the tables below, "r", "it.", "time" and "res." denote the rank of Z^0, the number of iterations, the total computing time in seconds and the residual at the final iterate, respectively.

Example 4.1. Let G^0, c^0 and x^0 denote a random $n \times n$ symmetric matrix, a random $n \times 1$ vector and a given feasible solution to Problem SDQP(G, c), respectively. The matrices $A_i (i = 1, \cdots, n)$ are defined as case (a) in (B). Numerical results for Example 4.1 are illustrated in Table 1.

Table 1 presents the total computing time and the residual of ADM-ISDQP for solving inverse semi-definite quadratic programming problems with different sizes $n = 50, 100, 200, 500, 1000$. It follows from the above numerical results that the proposed algorithm can solve the test problems successfully.

4.1. **Comparison with other algorithms.** In this subsection, we compare our algorithm (ADM) with the augmented Lagrangian method (ALM) [29] and the smoothing Newton method (SNM) [28] for solving ISDQP(G, c). The results of these methods for a number of scenarios of Example 4.2 are presented in Table 2 and Table 3.

Example 4.2. Let G^0, c^0 and x^0 denote a random $n \times n$ symmetric matrix, a random $n \times 1$ vector and a given feasible solution to Problem SDQP(G, c), respectively. The matrices $A_i (i = 1, \cdots, n)$ are defined as case (b) in (B).

It is not difficult to see that all three algorithms can solve the test problems successfully. In Table 2, we fix $n = 500$ for Example 2 and compare our algorithm with Newton-type methods for different values of m. Numerical results show that our approach behaves similar to the augmented Lagrangian method and the smoothing Newton method. However, the three algorithms show their difference numerically in Table 3. From Table 3, ADM performs better than other algorithms in terms of the total computing time and the corresponding residual.

Table 2. Comparison of different algorithms for Example 4.2 ($n = 500$)

solver	m	r	it.	time	res.
ALM	100	20	5	124.1	7.16e-04
	150	29	4	287.9	2.03e-04
SNM	100	20	7	142.0	4.31e-04
	150	29	6	205.2	1.92e-04
ADM	100	20	12	101.3	1.82e-04
	150	29	12	156.0	1.83e-04

Table 3. Comparison of different algorithms for Example 4.2 ($n = 1000$)

solver	m	r	it.	time	res.
ALM	150	30	5	1329.1	3.38e-04
	200	39	4	1773.8	3.42e-04
SNM	150	30	6	1152.2	1.82e-04
	200	39	7	1648.6	1.75e-04
ADM	150	30	13	507.8	1.30e-04
	200	39	13	752.0	1.31e-04

5. **Conclusions.** In this paper, we consider a class of inverse semi-definite quadratic programming problems. Under mild conditions, we showed that the problem can be solved by an alternating-direction-type numerical method. Our algorithm is easy to implement and code. Numerical experiments indicate that the proposed algorithm is an alternative approach to solve inverse semi-definite quadratic programming problems besides Newton-type methods.

Acknowledgments. The authors would like to thank the referees whose comments have helped us greatly to improve the presentation.

REFERENCES

[1] M. Afonso, J. Bioucas-Dias and M. Figueiredo, Fast image recovery using variable splitting and constrained optimization, *IEEE Transactions on image processing*, **19** (2010), 2345–2356.

[2] R. Ahuja and J. Orlin, Inverse optimization, *Operations Research,* **49** (2001), 771–783.

[3] R. Ahuja and J. Orlin, Combinatorial algorithms for inverse network flow problems, *Networks,* **40** (2002), 181–187.

[4] J. Barzilai and J. M. Borwein, Two point step size gradient methods, *IMA Journal of Numerical Analysis*, **8** (1988), 141–148.

[5] D. Bertsekas, On the Goldstein-Levitin-Polyak gradient projection method, *IEEE Transactions on Automatic Control,* **21** (1976), 174–184.

[6] E. Birgin, J. Martínez and M. Raydan, Nonmonotone spectral projected gradient methods on convex sets, *SIAM Journal on Optimization,* **10** (2000), 1196–1211.

[7] E. Birgin, J. Martínez and M. Raydan, Spectral projected gradient methods: reviews and perspective, Available from: `http://www.ime.usp.br/~egbirgin/publications/bmr5.pdf`.

[8] S. Boyd, N. Parikh, E. Chu, B. Peleato and J. Eckstein, Distributed optimization and statistical learning via the alternating direction method of multipliers, *Foundations and Trends in Machine Learning,* **3** (2011), 1–122.

[9] W. Burton and P. Toint, On an instance of the inverse shortest paths problem, *Mathematical Programming*, **53** (1992), 45–61.

[10] M. Cai, X. Yang and J. Zhang, The complexity analysis of the inverse center location problem, *Journal of Global Optimization*, **15** (1999), 213–218.

[11] Y. Dai and L. Liao, R-linear convergence of the Barzilai and Borwein gradient method, *IMA Journal on Numerical Analysis*, **22** (2002), 1–10.

[12] J. Douglas and H. Rachford, On the numerical solution of the heat conduction problem in two and three space variables, *Transactions of the American Mathematical Society,* **82** (1956), 421–439.

[13] A. Friedlander, J. M. Martínez, B. Molina and M. Raydan, Gradient method with retards and generalizations, *SIAM Journal on Numerical Analysis*, **36** (1999), 275–289.

[14] D. Gabay and B. Mercier, A dual algorithm for the solution of nonlinear variational problems via finite-element approximations, *Computers & Mathematics with Applications,* **2** (1976), 17–40.

[15] R. Glowinski and P. Tallec, *Augmented Lagrangian and Operator-Splitting Methods in Nonlinear Mechanics*, SIAM, Philadelphia, 1989.

[16] T. Goldstein and S. Osher, The split Bregman method for L1-regularized problems, *SIAM Journal on Imaging Sciences*, **2** (2009), 323–343.

[17] C. Heuberger, Inverse combinatorial optimization: A survey on problems, methods and results, *Journal of Combinatorial Optimization,* **8** (2004), 329–361.

[18] B. He, L. Liao, D. Han and H. Yang, A new inexact alternating direction method for monotone variational inequalities, *Mathematical Programming,* **92** (2002), 103–118.

[19] B. He, M. Tao and X. Yuan, Alternating direction method with Gaussian back substitution for separable convex programming, *SIAM Journal on Optimization,* **22** (2012), 313–340.

[20] G. Iyengar and W. Kang, Inverse conic programming and applications, *Operations Research Letters*, **33** (2005), 319–330.

[21] D. Peaceman and H. Rachford, The numerical solution of parabolic elliptic differential equations, *Journal of the Society for Industrial and Applied Mathematics,* **3** (1955), 28–41.

[22] M. Raydan, On the Barzilai and Borwein choice of steplength for the gradient method, *IMA Journal of Numerical Analysis,* **13** (1993), 321–326.

[23] M. Raydan, The Barzilai and Borwein gradient method for the large scale unconstrained minimization problem, *SIAM Journal on Optimization,* **7** (1997), 26–33.

[24] R. Rockafellar and R. Wets, *Variational Analysis*, Springer-Verlag, New York, 1998.

[25] D. Sun, J. Sun and L. Zhang, The rate of convergence of the augmented Lagrangian method for nonlinear semidefinite programming, *Mathematical Programming,* **114** (2008), 349–391.

[26] P. Tseng, Applications of a splitting algorithm to decomposition in convex programming and variational inequalities, *SIAM Journal on Control and Optimization*, **29** (1991), 119–138.

[27] P. Tseng, Further applications of a splitting algorithm to decomposition in variational inequalities and convex programming, *Mathematical Programming,* **48** (1990), 249–263.

[28] X. Xiao, L. Zhang and J. Zhang, A smoothing Newton method for a type of inverse semidefinite quadratic programming problem, *Journal of Computational and Applied Mathematics*, **223** (2009), 485–498.

[29] X. Xiao, L. Zhang and J. Zhang, On convergence of augmented Lagrange method for inverse semi-definite quadratic programming problems, *Journal of Industrial and Management Optimization*, **5** (2009), 319–339.

[30] J. Yang and Y. Zhang, Alternating direction algorithms for l_1-problems in compressive sensing, *SIAM Journal on Scientific Computing,* **33** (2011), 250–278.

[31] J. Zhang and Z. Liu, Calculating some inverse linear programming problems, *Journal of Computational and Applied Mathematics*, **72** (1996), 261–273.

[32] J. Zhang and Z. Liu, A further study on inverse linear programming problems, *Journal of Computational and Applied Mathematics*, **106** (1999), 345–359.

[33] J. Zhang, Z. Liu and Z. Ma, Some reverse location problems, *European Journal of Operations Research*, **124** (2000), 77–88.

[34] J. Zhang and Z. Ma, Solution structure of some inverse combinatorial optimization problems, *Journal of Combinatorial Optimization*, **3** (1999), 127–139.

[35] J. Zhang and L. Zhang, An augmented Lagrangian method for a class of inverse quadratic programming problems, *Applied Mathematics and Optimization,* **61** (2010), 57–83.

Received April 2013; 1st revision March 2014; final revision January 2015.

E-mail address: jinjin403@sina.com
E-mail address: y.e.ge@dlut.edu.cn
E-mail address: lwzhang@dlut.edu.cn