Noname manuscript No.
(will be inserted by the editor)

A projection algorithm based on KKT conditions for convex quadratic semidefinite programming with nonnegative constraints

Xiaokai. Chang · Jianchao Bai · Sanyang. Liu · Zhao Deng

Received: date / Accepted: date

Abstract The dual form of convex quadratic semidefinite programming (CQSDP) problem, with nonnegative constraints, is a 4-block separable convex optimization problem. It is known that, the directly extended 4-block alternating direction method of multipliers (ADMM4d) is very efficient to solve the dual, but its convergence is not guaranteed. In this paper, we reformulate the dual as a 3-block convex programming by introducing an extra variable, so as to design a parallel modified 3-block ADMM with larger step size that can exceed the conventional upper bound of $(1+\sqrt{5})/2$. We show that the proposed 3-block ADMM is equivalent to a projection algorithm with two operators projecting onto the positive semidefinite and nonnegative matrix cones respectively. The global convergence and non-ergodic convergence rate $o(1/(k+1))$ are established by using a fixed-point argument and the non-expansion property of the projection operators. Numerical experiments on the various classes of CQSDP problems illustrate that our proposed algorithm performs better than ADMM4d with the aggressive step size of 1.618.

Keywords Quadratic semidefinite programming · Nonnegative constraints · Alternating direction method of multipliers · Projection · Convergence analysis

Mathematics Subject Classification (2000) 90C25 · 90C22 · 65K05 · 47H05 · 47H10

1 Introduction

We consider the following convex quadratic semidefinite programming (CQSDP) with nonnegative constraints on the matrix variable:

$$\begin{aligned} \min \quad & \frac{1}{2}\langle X, \varphi(X)\rangle + \langle C, X\rangle \\ s.t. \quad & \mathcal{A}(X) = b, \\ & X \in \mathcal{S}^n_+, \quad X \in \mathcal{N}^n, \end{aligned} \tag{1}$$

Xiaokai. Chang, Corresponding Author
College of Science, Lanzhou University of Technology, Lanzhou, Gansu 730050, P. R. China.
E-mail: xkchang@lut.cn

Jianchao Bai
School of Mathematics and Statistics, Xi'an Jiaotong University, Xi'an 710049, P.R. China
E-mail: bjc1987@163.com

Sanyang Liu
School of Mathematics and Statistics, Xidian University, Xi'an 710071, P. R. China.
E-mail: liusanyang@126.com

Zhao Deng
School of Mathematics and Statistics, Xidian University, Xi'an 710071, P. R. China.
E-mail: dengzhao2016@163.com

where $\mathcal{S}_+^n$ denotes the symmetric and positive semi-definite matrices cone in the space of $n \times n$ symmetric matrices $\mathcal{S}^n$, endowed with the standard trace inner product $\langle \cdot, \cdot \rangle$ and its induced norm $\| \cdot \|$. $C \in \mathcal{S}^n$, $b \in \mathbb{R}^m$ are given data. $\mathcal{N}^n$ is a closed convex set and

$$\mathcal{N}^n = \{X \in \mathcal{S}^n |\ X \geq 0\}.$$

The operator $\mathcal{A} : \mathcal{S}^n \to \mathbb{R}^m$ is linear, and its adjoint with respect to the standard inner product in $\mathcal{S}^n$ and $\mathbb{R}^m$ is denoted by $\mathcal{A}^*$. $\varphi : \mathcal{S}^n \to \mathcal{S}^n$ is a given self-adjoint positive semidefinite linear operator, for instance, $\varphi(X) = BXB^T$ for a given positive-definite matrix $B \in \mathcal{S}^n$, $\varphi(X) = B \circ X$ for $B \in \mathcal{N}^n$ ("$\circ$" denotes the Hardamard product of two matrices and $\varphi(X) = \frac{BX+XB}{2}$ for $B \in \mathcal{S}_+^n$.

Let $\mathcal{D}^n = \mathcal{S}_+^n \cap \mathcal{N}^n$, named as the doubly nonnegative cone in [1], its dual cone is $\mathcal{S}_+^n + \mathcal{N}^n$. Thus the dual of problem (1) can be formulated as in [2,3]:

$$\begin{array}{ll} \min & \frac{1}{2}\langle W, \varphi(W)\rangle - b^T y \\ s.t. & -\varphi(W) + \mathcal{A}^*(y) + Z + S = C, \\ & W \in \mathcal{S}^n,\ y \in \mathbb{R}^m,\ Z \in \mathcal{S}_+^n,\ S \in \mathcal{N}^n. \end{array} \tag{2}$$

And the Karush-Kuhn-Tuck (KKT) conditions for problem (1) and its dual (2) can be written as follows:

$$\left.\begin{array}{c} \mathcal{A}(X) = b,\ \varphi(W) = \varphi(X), \\ -\varphi(W) + \mathcal{A}^*(y) + Z + S = C, \\ X \in \mathcal{S}_+^n,\ Z \in \mathcal{S}_+^n,\ \langle X, Z\rangle = 0, \\ X \in \mathcal{N}^n,\ S \in \mathcal{N}^n,\ \langle X, S\rangle = 0. \end{array}\right\} \tag{3}$$

By interior-point methods scheme, Toh et al. [4,5,6] proposed inexact primal-dual path following algorithm for solving the CQSDP problems without nonnegative constraints. To handle the CQSDP problems beyond moderate scale can be a challenging task using the interior-point methods, due to the extremely high computational cost per iteration or the inherent ill-conditioning of the linear systems governing the search directions. In addition, there are many methods proposed for solving some special CQSDP problems, see [8,9,10,11,12,13]. Of these methods, ADMM-type algorithms by dealing with the dual reveal excellent numerical results.

Due to the constraints on the doubly nonnegative cone, it is difficult to solve the primal problem (1) directly. In this paper, we pay attention to the dual (2) for its separable structure, which can be expressed in the form of the following convex optimization with four separate blocks in the objective function and a coupling linear equation constraint:

$$\begin{array}{ll} \min & \frac{1}{2}\langle W, \varphi(W)\rangle - b^T y + \delta_{\mathcal{S}_+^n}(Z) + \delta_{\mathcal{N}^n}(S) \\ s.t. & -\varphi(W) + \mathcal{A}^*(y) + Z + S = C, \end{array} \tag{4}$$

where $\delta_{\mathcal{C}}(Y)$ is the indicator function over a given set $\mathcal{C}$ such that $\delta_{\mathcal{C}}(Y) = 0$ if $Y \in \mathcal{C}$ and $+\infty$ otherwise.

Let $\sigma > 0$ be given. The augmented Lagrangian function for (4) reads as

$$\begin{aligned} \mathcal{L}_\sigma(W, y, Z, S, X) = & \frac{1}{2}\langle W, \varphi(W)\rangle - b^T y + \delta_{\mathcal{S}_+^n}(Z) + \delta_{\mathcal{N}^n}(S) \\ & + \langle X, -\varphi(W) + \mathcal{A}^*(y) + Z + S - C\rangle \\ & + \frac{\sigma}{2}\| - \varphi(W) + \mathcal{A}^*(y) + Z + S - C\|^2, \end{aligned} \tag{5}$$

where $(W, y, Z, S, X) \in \mathcal{S}^n \times \mathbb{R}^n \times \mathcal{S}^n_+ \times \mathcal{N}^n \times \mathcal{S}^n$. For a chosen initial point, the directly extended 4-block ADMM (ADMM4d) for (4) consists of the iterations:

$$\left.\begin{array}{l} W^{k+1} = \arg\min_{W \in \mathcal{S}^n} \mathcal{L}_\sigma(W, y^k, Z^k, S^k, X^k), \\ y^{k+1} = \arg\min_{y \in \mathbb{R}^n} \mathcal{L}_\sigma(W^{k+1}, y, Z^k, S^k, X^k), \\ Z^{k+1} = \arg\min_{Z \in \mathcal{S}^n_+} \mathcal{L}_\sigma(W^{k+1}, y^{k+1}, Z, S^k, X^k), \\ S^{k+1} = \arg\min_{S \in \mathcal{N}^n} \mathcal{L}_\sigma(W^{k+1}, y^{k+1}, Z^{k+1}, S, X^k), \\ X^{k+1} = X^k + \tau\sigma\left(-\varphi(W^{k+1}) + \mathcal{A}^*(y^{k+1}) + Z^{k+1} + S^{k+1} - C\right), \end{array}\right\} \quad (6)$$

where $\tau > 0$, e.g., $\tau \in (0, \frac{1+\sqrt{5}}{2})$, is a constant that controls the step size in (6). If choosing $\tau > 1$, the step size of updating the Lagrange multiplier is enlarged, and it is usually beneficial to induce faster convergence.

The direct extension of the classic ADMM to the case of the multi-block convex optimization problem is not necessarily convergent from [15,16], though it often performs very well in practice. With σ being small enough, the convergence of ADMM4d was obtained in [17] for a special 4-block problem with two objective functions being strongly convex. Furthermore, it shown that, the convergence can not be guaranteed only requiring one strongly convex function, by giving a concrete example in [17]. Thus, even for the simplest case with $\varphi(W) = W$ (the objective function $\frac{1}{2}\langle W, W\rangle$ is strongly convex), ADMM4d (6) is not necessarily convergent to solve problem (4).

Generally, there exists two types of methods to develop ADMM's variants, aiming to guarantee convergence and preserve the numerical advantages of the directly extended ADMM. One method is to add a simple correction step, for example, a convergent alternating direction method with a Gaussian back substitution (ADM-G) proposed by He et al. in [18,19], in which each iteration consists of a forward procedure (ADM procedure) and a backward procedure (Gaussian back substitution procedure), the correction step is completely free from step-size computing and its step size is bounded away from zero for all iterates. The other is to employ a simple proximal for solving each subproblem inexactly, which has been suggested by many researchers, see [3,9,21,22,23]. In addition, many modified ADMM-based algorithms were introduced in [24,25,20,28].

More recently, by leveraging on the inexact block symmetric Gauss-Seidel (sGS) decomposition technique, Chen, Sun and Toh [21] had employed the dual approach by proposing an efficient inexact ADMM-type first-oder method (the sGSimsPADMM) for solving problem (2). Furthermore, based on the inexact sGS decomposition technique and the semismooth Newton-CG algorithm, Li, Sun and Toh proposed a two-phase proximal augmented Lagrangian method for convex quadratic semidefinite programming, named QSDPNAL [2]. It extended the ideas from SDPNAL [26] and SDPNAL+ [27] for the linear SDP problems to the QSDP problems.

By making full use of the KKT conditions, Chang et al. [28] presented a modified ADMM to solve the dual of the CQSDP problem in standard form (without nonnegative constraints), which is an extension of the method proposed by Wen et al. [36]. This modified ADMM can always skip the subproblems with respect to the block-variable W, which will save both the computational cost and the memory for variable storage at each iteration. Inspired by the success of modified ADMM [28] as well as aforementioned work on the ADMM-based methods, we present a projection method by modifying 3-block ADMM to solve an equivalent of the KKT system (3). The main contributions of this paper are as follows:

- (1). By introducing an auxiliary variable, we reformulate the 4-block separable convex problem (4) as a 3-block separable convex problem, and apply the directly extended 3-block ADMM (ADMM3d) to this reformulation. Based on the iterative scheme of ADMM3d, we testify the KKT system (3) is equivalent to an equation system having two projection operators onto the positive semidefinite and nonnegative matrix cones respectively.
- (2). We propose a projection method for solving this equation system. Essentially, the proposed method can be explained as a parallel 3-block ADMM with larger step size (can be greater than $(1 + \sqrt{5})/2$), and does not have to solve the subproblem with variable W exactly. Skipping the calculation of W can save $\mathcal{O}(n^3)$ for some operators φ, while the cost is only about $\mathcal{O}(n^2)$ for computing the auxiliary variable

introduced, see Section 3.3. This confirms that at least our methods require less computation than the existing ADMM [3,18,32] in one iteration.

– (3). The global convergence of the proposed method as well as its non-ergodic $o(1/(k+1))$ convergence rate are established to a KKT point by using a fixed-point argument and the projection operator's non-expansion, when the condition on the penalty parameter is satisfied. The numerical experiments show that, our proposed algorithm performs better than ADM-G and ADMM4d with the aggressive step-length of 1.618.

The rest of this paper is organized as follows. Some preliminary results are provided in Section 2. We reformulate the dual (2) as a 3-block convex optimization problem, and introduce how to solve the subproblems from ADMM3d in Section 3. The projection method based on the KKT conditions (3) is presented in Subsection 4. The convergence of the proposed method is analysed in Section 5. Section 6 is devoted to the implementation and numerical experiments to solve the CQSDP problems generated randomly. Finally, the paper is summarized in Section 7.

2 Preliminaries

Most of the definitions and notations used in this paper are standard and can be found in [29]. Throughout, Ω is an arbitrary finite dimensional real Euclidean space with inner product $\langle \cdot, \cdot \rangle$ and its induced norm $\| \cdot \|$. A single-valued mapping $\mathcal{G} : \Omega \to \Omega$ is called β-cocoercive (or β-inverse-strongly monotone), for a certain constant $\beta > 0$, if $\beta\mathcal{G}$ is firmly nonexpansive, i.e.,

$$\langle \mathcal{G}(x) - \mathcal{G}(x'), x - x' \rangle \geq \beta \|\mathcal{G}(x) - \mathcal{G}(x')\|^2, \quad \forall x, x' \in \Omega.$$

Let φ be a self-adjoint non zero positive semidefinite linear operator, we use $\lambda_{\max}(\varphi)$ to denote its largest eigenvalue, then φ is $\frac{1}{\lambda_{\max}(\varphi)}$-cocoercive.

Let $\Gamma_0(\Omega)$ be the class of proper lower semicontinuous convex functions from Ω to $(-\infty, +\infty]$. For any $f \in \Gamma_0(\Omega)$, the subdifferential mapping ∂f of f is then maximal monotone and

$$\mathcal{J}_{\sigma\partial f}(x) = (\mathcal{I} + \sigma\partial f)^{-1}(x) = \operatorname*{argmin}_z \left\{ f(z) + \frac{1}{2\sigma}\|x - z\|^2 \right\}, \quad \forall\, x \in \Omega,$$

where $\mathcal{I} : \Omega \to \Omega$ is the identity operator.

For a given closed convex $\mathcal{C}$, $\delta_{\mathcal{C}}$ is closed proper convex function and $\mathcal{J}_{\sigma\partial\delta_{\mathcal{C}}}(x) = \Pi_{\mathcal{C}}(x)$, i.e. the metric projection of x onto $\mathcal{C}$, and

$$\partial\delta_{\mathcal{C}}(x) = \mathcal{N}_{\mathcal{C}}(x) := \{z \,|\, \langle z, x' - x \rangle \leq 0 \ \forall x' \in \mathcal{C}\},$$

which is a closed convex cone.

We will denote by Fix $\mathcal{T}$ the set of fixed points of operator $\mathcal{T}$, i.e., Fix $\mathcal{T} := \{x^* \in \Omega | x^* = \mathcal{T}(x^*)\}$.

3 Reformulation of (2) and Directly Extended 3-Block ADMM.

Firstly, we make the following assumptions.

Assumption 1 *(i). For the CQSDP problem (1), there exists a feasible solution $X \in \mathcal{S}^n_+$ such that*

$$\mathcal{A}(X) = b, \quad X \in \mathcal{S}^n_{++}, \quad X \in \mathcal{N}^n. \tag{7}$$

(ii). For the dual problem (2), there exists a feasible solution $(W, y, Z, S) \in \mathcal{S}^n \times \mathbb{R}^m \times \mathcal{S}^n_+ \times \mathcal{N}^n$ such that

$$-\varphi(W) + \mathcal{A}^*(y) + Z + S = C, \quad Z \in \mathcal{S}^n_{++}, \quad S \in \mathcal{N}^n. \tag{8}$$

It is known from convex analysis (e.g, Corollary 5.3.6 in [33]) that under Assumption 1, the strong duality for (1) and (2) holds and the KKT conditions (3) have solutions.

Assumption 2 *The linear operator $\mathcal{A}$ is surjective.*

Under Assumption 2, the operator $\mathcal{A}\mathcal{A}^*$ is invertible, then the solution of the subproblem with variable y can be well-defined for ADMM4d.

3.1 3-block Convex Optimization Reformulation

By introducing an auxiliary variable $\widetilde{X}$, we can rewrite (1) equivalently as

$$\begin{aligned} \min \quad & \frac{1}{2}\langle X, \varphi(X)\rangle + \langle C, X\rangle \\ s.t. \quad & X - \widetilde{X} = 0, \mathcal{A}(X) = b, \\ & X \in \mathcal{S}^n_+, \quad \widetilde{X} \in \mathcal{N}. \end{aligned} \tag{9}$$

Defining $U = \begin{pmatrix} X \\ \widetilde{X} \end{pmatrix}$, (9) can be simplied as:

$$\begin{aligned} \min \quad & \theta(U) \\ s.t. \quad & \mathcal{H}(U) = \widetilde{b}, \quad U \in \mathcal{S}^n_+ \times \mathcal{N}^n, \end{aligned} \tag{10}$$

where $\theta(U) = \frac{1}{2}\langle X, \varphi(X)\rangle + \langle C, X\rangle$, and

$$\mathcal{H} = \begin{pmatrix} \mathcal{I} & -\mathcal{I} \\ \mathcal{A} & 0 \end{pmatrix}, \quad \widetilde{b} = \begin{pmatrix} 0 \\ b \end{pmatrix}. \tag{11}$$

By setting

$$\begin{aligned} f(W) &:= \frac{1}{2}\left\langle \begin{pmatrix} \varphi(W) \\ 0 \end{pmatrix}, \begin{pmatrix} W \\ 0 \end{pmatrix} \right\rangle = \frac{1}{2}\langle W, \varphi(W)\rangle, \\ g(Z, U) &:= \delta_{\mathcal{S}^n_+}(Z) + \delta_{\mathcal{N}^n}(U), \\ h(S, y) &:= \left\langle \begin{pmatrix} 0 \\ -b \end{pmatrix}, \begin{pmatrix} S \\ y \end{pmatrix} \right\rangle = \langle 0, S\rangle - b^T y, \end{aligned}$$

and

$$\begin{aligned} \mathcal{F}^*(W) &:= \begin{pmatrix} -\varphi & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} W \\ 0 \end{pmatrix} = \begin{pmatrix} -\varphi(W) \\ 0 \end{pmatrix}, \\ \mathcal{G}^*(Z, U) &:= \begin{pmatrix} \mathcal{I} & 0 \\ 0 & \mathcal{I} \end{pmatrix} \begin{pmatrix} Z \\ U \end{pmatrix} = \begin{pmatrix} Z \\ U \end{pmatrix}, \\ \mathcal{H}^*(S, y) &:= \begin{pmatrix} \mathcal{I} & \mathcal{A}^* \\ -\mathcal{I} & 0 \end{pmatrix} \begin{pmatrix} S \\ y \end{pmatrix} = \begin{pmatrix} S + \mathcal{A}^*(y) \\ -S \end{pmatrix}, \\ \widetilde{C} &:= \begin{pmatrix} C \\ 0 \end{pmatrix}, \end{aligned}$$

then the dual of problem (10) can be reformulated as

$$\begin{aligned} \min \quad & f(W) + g(Z, U) + h(S, y) \\ s.t. \quad & \mathcal{F}^*(W) + \mathcal{G}^*(Z, U) + \mathcal{H}^*(S, y) = \widetilde{C}. \end{aligned} \tag{12}$$

Actually, it is equivalent to (4).

Notice that from Assumption 2 and the definition of $\mathcal{H}$, we can obtain the following results easily.

Lemma 1 *Under Assumption 2, then we have*
(i). $\mathcal{H}\mathcal{H}^*$ is invertible.
(ii). the inverse of $\mathcal{H}\mathcal{H}^*$ can be computed by:

$$(\mathcal{H}\mathcal{H}^*)^{-1} = \begin{pmatrix} \frac{1}{2}\mathcal{I} + \frac{1}{2}\mathcal{A}^*(\mathcal{A}\mathcal{A}^*)^{-1}\mathcal{A} & -\mathcal{A}^*(\mathcal{A}\mathcal{A}^*)^{-1} \\ -(\mathcal{A}\mathcal{A}^*)^{-1}\mathcal{A} & 2(\mathcal{A}\mathcal{A}^*)^{-1} \end{pmatrix}. \tag{13}$$

Specially, for the operator $\mathcal{A} = diag$, we have $\mathcal{A}\mathcal{A}^* = \mathcal{I}$, and

$$(\mathcal{H}\mathcal{H}^*)^{-1} = \begin{pmatrix} \frac{1}{2}\mathcal{I} + \frac{1}{2}\mathcal{A}^*\mathcal{A} & -\mathcal{A}^* \\ -\mathcal{A} & 2\mathcal{I} \end{pmatrix}. \tag{14}$$

From Lemma 1, the solution of the subproblem with variable (S, y) can be well-defined for using ADMM3d to (12). Additionally, reformulation (12) has many advantages for designing efficiency ADMM-based algorithm, for instance, the subproblem with variable (S, y) or (Z, U) can be implemented in parallel and that with variable W can be skipped, the convergence rate of the proposed ADMM can be analysed in non-ergodic case, different with the existing multi-block ADMM, see Section 4 and 5.

3.2 ADMM3d for Solving (12)

Recall that the augmented lagrangian function of the problem (12) has the following form:

$$\begin{aligned} \widehat{\mathcal{L}}_\sigma(W,(Z,U),(S,y),\widehat{X}) = \; & f(W) + g(Z,U) + h(S,y) + \\ & \left\langle \widehat{X}, \mathcal{F}^*(W) + \mathcal{G}^*(Z,U) + \mathcal{H}^*(S,y) - \tilde{C} \right\rangle + \\ & \frac{\sigma}{2}\|\mathcal{F}^*(W) + \mathcal{G}^*(Z,U) + \mathcal{H}^*(S,y) - \tilde{C}\|^2, \end{aligned} \tag{15}$$

where $W \in \mathcal{S}^n$, $(Z,U) \in \mathcal{S}^n_+ \times \mathcal{N}^n$, $(S,y) \in \mathcal{S}^n \times \mathbb{R}^n$ and $\widehat{X} = \begin{pmatrix} X_1 \\ X_2 \end{pmatrix}$ $(X_1, X_2 \in \mathcal{S}^n)$, $\sigma > 0$ is a penalty parameter. Then, the iterative scheme of ADMM3d for solving (12) reads as:

$$W^{k+1} = \arg\min_{W\in\mathcal{S}^n} \widehat{\mathcal{L}}_\sigma\left(W,(Z^k,U^k),(S^k,y^k),\widehat{X}^k\right), \tag{16}$$

$$(S^{k+1},y^{k+1}) = \arg\min_{(S,y)\in\mathcal{S}^n\times\mathbb{R}^m} \widehat{\mathcal{L}}_\sigma\left(W^{k+1},(Z^k,U^k),(S,y),\widehat{X}^k\right), \tag{17}$$

$$(Z^{k+1},U^{k+1}) = \arg\min_{(Z,U)\in\mathcal{S}^n_+\times\mathcal{N}^n} \widehat{\mathcal{L}}_\sigma\left(W^{k+1},(Z,U),(S^{k+1},y^{k+1}),\widehat{X}^k\right), \tag{18}$$

$$\widehat{X}^{k+1} = \widehat{X}^k + \tau\sigma(\mathcal{F}^*(W) + \mathcal{G}^*(Z,U) + \mathcal{H}^*(S,y) - \tilde{C}), \tag{19}$$

where $\tau \in (0, \frac{1+\sqrt{5}}{2})$.

Now, we label $\widehat{\mathcal{L}}_\sigma(W,y,Z,S,X)$ as $\widehat{\mathcal{L}}_\sigma$, and solve these subproblems (17)-(18). From the first-order optimality condition of problem (17), we have

$$\nabla_{(S,y)}\widehat{\mathcal{L}}_\sigma = \mathcal{H}\begin{pmatrix} X_1 - \sigma(\varphi(W) - Z + C) \\ X_2 + \sigma U \end{pmatrix} + \sigma\mathcal{H}\mathcal{H}^*\begin{pmatrix} S \\ y \end{pmatrix} - \begin{pmatrix} 0 \\ b \end{pmatrix} = 0.$$

By Lemma 1, we have

$$\begin{pmatrix} S \\ y \end{pmatrix} = -(\mathcal{H}\mathcal{H}^*)^{-1}\mathcal{H}\begin{pmatrix} \frac{1}{\sigma}X_1 - \varphi(W) + Z - C \\ \frac{1}{\sigma}X_2 + U \end{pmatrix} + \frac{1}{\sigma}(\mathcal{H}\mathcal{H}^*)^{-1}\begin{pmatrix} 0 \\ b \end{pmatrix}, \tag{20}$$

and can compute $S^{k+1} = \phi_S(W^{k+1}, Z^k, U^k, X_1^k, X_2^k)$, where

$$\phi_S(W,Z,U,X_1,X_2) := -\frac{1}{2}\mathcal{Q}_-\left(\frac{1}{\sigma}X_1 - \varphi(W) + Z - C\right) + \frac{1}{2}\mathcal{Q}_+\left(\frac{1}{\sigma}X_2 + U\right) - \frac{1}{\sigma}\mathcal{A}^*(\mathcal{A}\mathcal{A}^*)^{-1}(b), \quad (21)$$

and

$$\mathcal{Q}_- = \mathcal{I} - \mathcal{M}, \quad \mathcal{Q}_+ = \mathcal{I} + \mathcal{M}, \quad \mathcal{M} = \mathcal{A}^*(\mathcal{A}\mathcal{A}^*)^{-1}\mathcal{A}. \quad (22)$$

It is easy to check that the four operators $\mathcal{A}$, $\mathcal{M}$, $\mathcal{Q}_+$ and $\mathcal{Q}_-$ satisfy the following properties:

Lemma 2 *(i).* $\mathcal{M}^* = \mathcal{M},\ \mathcal{Q}_-{}^* = \mathcal{Q}_-,\ \mathcal{Q}_+{}^* = \mathcal{Q}_+$.
(ii). $\mathcal{M}^*\mathcal{M} = \mathcal{M},\ \mathcal{Q}_-{}^*\mathcal{Q}_- = \mathcal{Q}_-,\ \mathcal{Q}_+{}^*\mathcal{Q}_- = \mathcal{Q}_-$.
(iii). $\mathcal{M}\mathcal{A}^* = \mathcal{A}^*,\ \mathcal{A}\mathcal{M} = \mathcal{A},\ \mathcal{A}\mathcal{Q}_- = 0,\ \mathcal{Q}_-\mathcal{A}^* = 0,\ \mathcal{A}\mathcal{Q}_+ = 2\mathcal{A},\ \mathcal{Q}_+\mathcal{A}^* = 2\mathcal{A}^*$.

In addition, it is from Assumption 2 that, we can obtain $y^{k+1} = \phi_y(W^{k+1}, Z^k, U^k, X_1^k, X_2^k)$, where

$$\phi_y(W,Z,U,X_1,X_2) := -(\mathcal{A}\mathcal{A}^*)^{-1}\mathcal{A}\left(\frac{1}{\sigma}X_1 - \varphi(W) + Z - C + \frac{1}{\sigma}X_2 + U\right) + 2\frac{1}{\sigma}(\mathcal{A}\mathcal{A}^*)^{-1}b. \quad (23)$$

Similarly, according to the first-order optimality conditions of problem (18), let

$$\phi_{V_Z}(W,S,y,X_1) := -\frac{1}{\sigma}X_1 + \varphi(W) - S - \mathcal{A}^*(y) + C, \quad (24)$$

$$\phi_{V_U}(S,X_2) := S - \frac{1}{\sigma}X_2, \quad (25)$$

compute $V_Z^{k+1} = \phi_{V_Z}(W^{k+1}, S^{k+1}, y^{k+1}, X_1^k)$ and $V_U^{k+1} = \phi_{V_U}(S^{k+1}, X_2^k)$, then we can get $U^{k+1} = \Pi_{\mathcal{N}^n}(V_U^{k+1})$ and $Z^{k+1} = \Pi_{\mathcal{S}_+^n}(V_Z^{k+1})$ in parallel. By the projection operator's properties, the following properties are easy to obtain by direct computation.

Lemma 3 *Suppose that* $\{Z^{k+1}, U^{k+1}, X_1^{k+1}, X_2^{k+1}\}$ *are generated by (16)-(19) with* $\tau = 1$*, then we have*
(i). $V_Z^{k+1} = Z^{k+1} - \frac{1}{\sigma}X_1^{k+1},\quad Z^{k+1} \in \mathcal{S}_+^n,\quad X_1^{k+1} \in \mathcal{S}_+^n,\quad \langle Z^{k+1}, X_1^{k+1}\rangle = 0;$
(ii). $V_U^{k+1} = U^{k+1} - \frac{1}{\sigma}X_2^{k+1},\quad U^{k+1} \in \mathcal{N}^n,\quad X_2^{k+1} \in \mathcal{N}^n,\quad \langle U^{k+1}, X_2^{k+1}\rangle = 0.$

3.3 Solving the Subproblem (16) with Variable W.

In this section, we introduce how to solve the subproblem (16) efficiently, though it is not necessarily for designing our algorithm. The main objective is to show what our algorithm skips and how to get W^{k+1} for other ADMM-based methods in our numerical experiments.

Since the first-order optimality condition of (16) has the form

$$\nabla_W \widehat{\mathcal{L}}_\sigma = \varphi\left(W - X_1 + \sigma\varphi(W) - Z - S - \mathcal{A}^*(y) + C\right) = 0, \quad (26)$$

the structure of φ is important for computing W^{k+1}. For the simplest $\varphi(W) = W$ as used in the least squares SDP problem [11,13], we can compute W^{k+1} with easy from

$$W^{k+1} = \frac{1}{1+\sigma}(X_1^k + \sigma(Z^k + S^k + \mathcal{A}^*(y^k) - C)), \quad (27)$$

because $\varphi(W) = 0$ if only if $W = 0$.

However, it is not the case for all the operators φ. If $\varphi(W) = \frac{BW+WB}{2}$ as used in [2], for a given matrix $B \in \mathcal{S}_+^n$, the operator φ may not be invertible, then the equation $\varphi(W) = 0$ has many solutions. Although we actually do not need W explicitly in each iterations, only $\varphi(W)$ is needed, but it is generally not easy to obtain $\varphi(W)$ from (26), which will cost $\mathcal{O}(n^3)$ flops for some operators φ. Next, we will introduce how to get $\varphi(W^{k+1})$ effectively from (26).

For the problems with $\varphi(W) = \frac{BW+WB}{2}$, we suppose that the eigenvalue decomposition is $B = P\varLambda P^T$, where $\varLambda = \mathrm{diag}(\lambda)$ and $\lambda = (\lambda_1, \cdots, \lambda_n)^T$ is the vector of eigenvalues of B. Then, we have $\varphi = \mathcal{B}^*\mathcal{B}$, where $\mathcal{B}(X) = H \circ (P^T XP)$, $\mathcal{B}^*(Y) = P(H \circ Y)P^T$ and $H_{ij} = \sqrt{\frac{\lambda_i+\lambda_j}{2}}$, which implies $\mathcal{B}\mathcal{B}^* = H \circ H\circ$. Thus we can obtain the inverse of $\mathcal{I} + \sigma\mathcal{B}\mathcal{B}^*$ by

$$(\mathcal{I} + \sigma\mathcal{B}\mathcal{B}^*)^{-1} = \widehat{H}\circ,$$

with $\widehat{H}_{ij} = \frac{1}{1+\sigma H_{ij}^2}$. By [34, Lemma 4] and setting $\varXi = \widehat{H}\circ$, we have

$$\begin{aligned}(\mathcal{I} + \sigma\varphi)^{-1} &= (\mathcal{I} + \sigma\mathcal{B}^*\mathcal{B})^{-1} \\ &= \mathcal{I} - \sigma\mathcal{B}^*(\mathcal{I} + \sigma\mathcal{B}\mathcal{B}^*)^{-1}\mathcal{B} \\ &= \mathcal{I} - \sigma\mathcal{B}^*\varXi\mathcal{B}.\end{aligned}$$

Thus, $\varphi(W)$ can be computed efficiently by

$$\varphi(W) = (\mathcal{I} - \sigma\mathcal{B}^*\varXi\mathcal{B})\varphi\left(X_1 + \sigma(Z + S + \mathcal{A}^*(y) - C)\right). \tag{28}$$

For the problems with $\varphi(W) = BWB^T$, if the eigenvalue decomposition of B is $P\varLambda P^T$, where $\varLambda = \mathrm{diag}(\lambda)$ and $\lambda = (\lambda_1, \cdots, \lambda_n)^T$ is the vector of eigenvalues of B. We can still write $\varphi = \mathcal{B}^*\mathcal{B}$, where $\mathcal{B}(X) = H \circ (P^T XP)$, $\mathcal{B}^*(Y) = P(H \circ Y)P^T$ but $H_{ij} = \sqrt{\lambda_i\lambda_j}$. By using the same idea as above, $\varphi(W)$ can be computed efficiently by (28).

Suppose that the eigenvalue decomposition of $B = P\varLambda P^T$ is already computed, which is performed only once and needs $9n^3$ flops by the symmetric QR algorithm. If B is a low rank matrix, computing $\varphi(W)$ can be very cheap as the matrix H is sparse, else if B is a positive definite matrix (not an identity matrix), computing $\mathcal{B}(X) = H \circ (P^T XP)$ and $\mathcal{B}^*(Y) = P(H \circ Y)P^T$ needs at least $8n^3$ flops to get $\varphi(W)$ at each iteration.

4 Projection Method

In this section, we define following operators by the metric projection:

$$\mathcal{P}_{\mathcal{S}_+^n}(V) := \left(\varPi_{\mathcal{S}_+^n}(V), \quad \varPi_{\mathcal{S}_+^n}(V) - V\right), \tag{29}$$

$$\mathcal{P}_{\mathcal{N}^n}(V) := \left(\varPi_{\mathcal{N}^n}(V), \quad \varPi_{\mathcal{N}^n}(V) - V\right), \tag{30}$$

$$\mathcal{P}(V) := (\mathcal{P}_{\mathcal{S}_+^n}(V), \quad \mathcal{P}_{\mathcal{N}^n}(V)), \tag{31}$$

for any matrix $V \in \mathcal{S}^n$. By these operators and the iterative scheme (16)-(19), we show the equivalence of the KKT system (3) to an equation system. Then, a projection method for solving the equation system is presented.

In addition, we define a set

$$\mathcal{K} = \left\{(W, Z, U, X_1, X_2) \mid W \in \mathcal{S}^n, Z \in \mathcal{S}_+^n, U \in \mathcal{N}^n, X_1 \in \mathcal{S}^n, X_2 \in \mathcal{S}^n\right\}, \tag{32}$$

which will simplify our analysis.

4.1 Properties

By Moreau decomposition [35] and two operators $\mathcal{P}_{\mathcal{S}_+^n}(\cdot)$ and $\mathcal{P}_{\mathcal{N}^n}(\cdot)$ defined above, we now present the most important conclusion on the KKT system (3) in the following theorem. Based on this conclusion, we propose the projection method to obtain a KKT point.

Theorem 1 *(i) For any* $(W, Z, U, X_1, X_2) \in \mathcal{K}$ *satisfying the following system*

$$\left.\begin{array}{l}\varphi(W) = \varphi(X_1),\ S = \phi_S(W,Z,U,X_1,X_2),\ y = \phi_y(W,Z,U,X_1,X_2),\\ (Z, \frac{1}{\sigma}X_1) = \mathcal{P}_{\mathcal{S}^n_+} \circ \phi_{V_Z}(W,S,y,X_1),\ (U, \frac{1}{\sigma}X_2) = \mathcal{P}_{\mathcal{N}^n} \circ \phi_{V_U}(S,X_2),\end{array}\right\} \tag{33}$$

where the operators ϕ_S, ϕ_y, ϕ_{V_Z} *and* ϕ_{V_U} *are defined in (21), (23), (24) and (25), then* (W, y, Z, S, X_1) *is a solution of the KKT system (3), namely,*

$$\left.\begin{array}{c}\mathcal{A}(X_1) = b,\ \varphi(W) = \varphi(X_1),\\ -\varphi(W) + \mathcal{A}^*(y) + Z + S = C,\\ X_1 \in \mathcal{S}^n_+,\ Z \in \mathcal{S}^n_+,\ \langle Z, X_1\rangle = 0,\\ X_1 \in \mathcal{N}^n,\ S \in \mathcal{N}^n,\ \langle S, X_1\rangle = 0.\end{array}\right\} \tag{34}$$

(ii) If the point (W, y, Z, S, X) *satisfies the KKT system (3), by setting* $X_1 = X_2 = X$ *and* $U = S$*, then* $(W, Z, U, X_1, X_2) \in \mathcal{K}$ *is a solution of the system (33).*

Proof. (i) By $(Z, \frac{1}{\sigma}X_1) = \mathcal{P}_{\mathcal{S}^n_+} \circ \phi_{V_Z}(W,S,y,X_1)$, we have

$$X_1 \in \mathcal{S}^n_+,\ Z \in \mathcal{S}^n_+,\ \langle Z, X_1\rangle = 0,$$

and $\phi_{V_Z}(W,S,y,X_1) = Z - \frac{1}{\sigma}X_1$. With the definition of $\phi_{V_Z}(W,S,y,X_1)$ in (24), we obtain

$$-\varphi(W) + \mathcal{A}^*(y) + Z + S = C. \tag{35}$$

Notice that from $y = \phi_y(W,Z,U,X_1,X_2)$ and $W = X_1$, we deduce that

$$\begin{aligned}(\mathcal{A}\mathcal{A}^*)y &= -\mathcal{A}\left(\frac{1}{\sigma}X_1 - \varphi(W) + Z - C + \frac{1}{\sigma}X_2 + U\right) + 2\frac{1}{\sigma}b\\ &= -\mathcal{A}\left(\frac{1}{\sigma}X_1 - S - \mathcal{A}^*(y) + \frac{1}{\sigma}X_2 + U\right) + 2\frac{1}{\sigma}b.\end{aligned} \tag{36}$$

On the other hand, it follows from Lemma 2 and (21), we have

$$\mathcal{A}(S) = \mathcal{A}(\frac{1}{\sigma}X_2 + U) - \frac{1}{\sigma}b. \tag{37}$$

Substituting (37) into (36), we have $\mathcal{A}(X_1) = b$.

In addition, since $(U, \frac{1}{\sigma}X_2) = \mathcal{P}_{\mathcal{N}^n} \circ \phi_{V_U}(S,X_2)$ and $\phi_{V_U}(S,X_2) = S - \frac{1}{\sigma}X_2$, we have $S = U$, which implies

$$X_2 \in \mathcal{N}^n,\ S \in \mathcal{N}^n,\ \langle S, X_2\rangle = 0.$$

It is follows from $S = \phi_S(W,Z,U,X_1,X_2)$ and (35), that

$$\begin{aligned}S &= -\frac{1}{2}\mathcal{Q}_-\left(\frac{1}{\sigma}X_1 - S - \mathcal{A}^*(y)\right) + \frac{1}{2}\mathcal{Q}_+\left(\frac{1}{\sigma}X_2 + S\right) - \frac{1}{\sigma}\mathcal{M}(X_1)\\ &= -\frac{1}{2}\left(\frac{1}{\sigma}X_1 - S - \mathcal{A}^*(y) - \frac{1}{\sigma}X_2 - S\right) + \frac{1}{2}\mathcal{M}\left(\frac{1}{\sigma}X_1 - S - \mathcal{A}^*(y) + \frac{1}{\sigma}X_2 + S - 2\frac{1}{\sigma}X_1\right)\\ &= \frac{1}{2\sigma}(X_2 - X_1) + \frac{1}{2\sigma}\mathcal{M}(X_2 - X_1) + S,\end{aligned}$$

which means $X_2 = X_1$. Thus, $X_1 \in \mathcal{N}^n,\ S \in \mathcal{N}^n,\ \langle S, X_1\rangle = 0$. Finally, note that $\varphi(W) = \varphi(X_1)$, we obtain (34).

(ii) By (3) and $S = U$, we have

$$\begin{aligned}\phi_{V_Z}(W,S,y,X_1) &= Z - \frac{1}{\sigma}X_1,\\ \phi_{V_U}(S,X_2) &= S - \frac{1}{\sigma}X_2 = U - \frac{1}{\sigma}X_2.\end{aligned}$$

Using Moreau decomposition and $X_1 = X_2 = X$, it is not difficult to obtain $(Z, \frac{1}{\sigma}X_1) = \mathcal{P}_{\mathcal{S}^n_+} \circ \phi_{V_Z}(W, S, y, X_1)$ and $(U, \frac{1}{\sigma}X_2) = \mathcal{P}_{\mathcal{N}^n} \circ \phi_{V_U}(S, X_2)$. In addition, from (23) we have

$$\begin{aligned}
\phi_y(W, Z, U, X_1, X_2) &= -(\mathcal{A}\mathcal{A}^*)^{-1}\mathcal{A}\left(\frac{1}{\sigma}X_1 - \mathcal{A}^*(y) - S + \frac{1}{\sigma}X_2 + S\right) + 2\frac{1}{\sigma}(\mathcal{A}\mathcal{A}^*)^{-1}b \\
&= -(\mathcal{A}\mathcal{A}^*)^{-1}\mathcal{A}\left(\frac{1}{\sigma}X - \mathcal{A}^*(y) + \frac{1}{\sigma}X\right) + 2\frac{1}{\sigma}(\mathcal{A}\mathcal{A}^*)^{-1}b \\
&= (\mathcal{A}\mathcal{A}^*)^{-1}\mathcal{A}\mathcal{A}^*(y) - (\mathcal{A}\mathcal{A}^*)^{-1}\mathcal{A}(\frac{2}{\sigma}X) + 2\frac{1}{\sigma}(\mathcal{A}\mathcal{A}^*)^{-1}b \\
&= y
\end{aligned}$$

and

$$\begin{aligned}
\phi_S(W, Z, U, X_1, X_2) &= -\frac{1}{2}\mathcal{Q}_-\left(\frac{1}{\sigma}X_1 - \mathcal{A}^*(y) - S\right) + \frac{1}{2}\mathcal{Q}_+\left(\frac{1}{\sigma}X_2 + S\right) - \frac{1}{\sigma}\mathcal{A}^*(\mathcal{A}\mathcal{A}^*)^{-1}(b) \\
&= -\frac{1}{2}\mathcal{Q}_-\left(\frac{1}{\sigma}X - S\right) + \frac{1}{2}\mathcal{Q}_+\left(\frac{1}{\sigma}X + S\right) - \frac{1}{\sigma}\mathcal{A}^*(\mathcal{A}\mathcal{A}^*)^{-1}(b) \\
&= S + \frac{1}{\sigma}\mathcal{M}(X) - \frac{1}{\sigma}\mathcal{A}^*(\mathcal{A}\mathcal{A}^*)^{-1}(b) \\
&= S.
\end{aligned}$$

This completes our proof. □

From Theorem 1, we can deduce the following relation:

$$(W, y, Z, S, X) \text{ is a KKT point.} \xLeftrightarrow[X_1 = X_2 = X]{U = S} (W, Z, U, X_1, X_2) \text{ satisfies the system (33),}$$

which implies that the system (33) is nonempty from Assumption 1. The most important thing is that $\varphi(W) = \varphi(X_1)$ in the system (33), which gives us a confidence to believe that, we don't have to solve the subproblem with variable W exactly, even not have to compute $\varphi(W)$, but a KKT point of problem (1) and its dual (2) can be obtained as long as we can get a solution of the system (33). Therefore, we will in the next section design a projection method for solving the system (33).

4.2 Projection Method

Now, we define the following notations to simplify our analysis,

$$w = (Z, \frac{1}{\sigma}X_1, U, \frac{1}{\sigma}X_2), \quad w^k = (Z^k, \frac{1}{\sigma}X_1^k, U^k, \frac{1}{\sigma}X_2^k), \quad w^* = (Z^*, \frac{1}{\sigma}X_1^*, U^*, \frac{1}{\sigma}X_2^*).$$

Since $\varphi(W) = \varphi(X_1)$ in the system (33), we can remove the item $\varphi(W)$, and replace $\varphi(W)$ with $\varphi(X_1)$ in the definition of ϕ_S, ϕ_y, ϕ_{V_Z} and ϕ_{V_U}, e.g.,

$$\left.\begin{aligned}
\phi_S(w) &= -\tfrac{1}{2}\mathcal{Q}_-\left((\tfrac{1}{\sigma}\mathcal{I} - \varphi)(X_1) + Z - C\right) + \tfrac{1}{2}\mathcal{Q}_+\left(\tfrac{1}{\sigma}X_2 + U\right) - \tfrac{1}{\sigma}\mathcal{A}^*(\mathcal{A}\mathcal{A}^*)^{-1}(b), \\
\phi_y(w) &= -(\mathcal{A}\mathcal{A}^*)^{-1}\mathcal{A}\left((\tfrac{1}{\sigma}\mathcal{I} - \varphi)(X_1) + Z - C + \tfrac{1}{\sigma}X_2 + U\right) + 2\tfrac{1}{\sigma}(\mathcal{A}\mathcal{A}^*)^{-1}b, \\
\phi_{V_Z}(S, y, X_1) &= -(\tfrac{1}{\sigma}\mathcal{I} - \varphi)(X_1) - S - \mathcal{A}^*(y) + C, \\
\phi_{V_U}(S, X_2) &= S - \tfrac{1}{\sigma}X_2.
\end{aligned}\right\} \tag{38}$$

Accordingly, $\varphi(W) = \varphi(X_1)$ can be left out, then we can rewrite the system (33) as

$$\left.\begin{aligned}
S = \phi_S(w), \; y = \phi_y(w), \\
(Z, \tfrac{1}{\sigma}X_1) = \mathcal{P}_{\mathcal{S}^n_+} \circ \phi_{V_Z}(S, y, X_1), \\
(U, \tfrac{1}{\sigma}X_2) = \mathcal{P}_{\mathcal{N}^n} \circ \phi_{V_U}(S, X_2).
\end{aligned}\right\} \tag{39}$$

Furthermore, we will use the following notations,

$$v_Z(w)=\phi_{V_Z}(\phi_S(w),\phi_y(w),X_1),\quad v_U(w)=\phi_{V_U}(\phi_S(w),X_2),\quad v(w)=(v_Z(w),\quad v_U(w)),\tag{40}$$

$$\mathcal{J}=\left\{w\ \mid\ Z\in\mathcal{S}^n_+,U\in\mathcal{N}^n,X_1\in\mathcal{S}^n,X_2\in\mathcal{S}^n\right\}.\tag{41}$$

By these notations, the system (39) can be expressed as

$$w=\mathcal{P}\circ v(w)=\left(\mathcal{P}_{\mathcal{S}^n_+}\circ v_Z(w),\quad \mathcal{P}_{\mathcal{N}^n}\circ v_U(w)\right),\tag{42}$$

its solution set can be viewed as the set of fixed points of operator $\mathcal{P}\circ v$, i.e., Fix $\mathcal{P}\circ v:=\{w^*\in\mathcal{J}|w^*=\mathcal{P}\circ v(w^*)\}$. From Assumption 1 and Theorem 1, we have Fix $\mathcal{P}\circ v\neq\emptyset$, so we can design our projection method as follows.

Algorithm 1 (Projection Method for solving the system (39))

Step 0. Let $\sigma\in(0,\frac{2}{\lambda_{max}(\varphi)})$ and $\rho\in\left(0,2-\frac{\sigma\lambda_{max}(\varphi)}{2}\right)$ be given parameters. Choose $w^0\in\mathcal{J}$, set $k=0$.
Step 1. Compute

$$S^{k+1}=\phi_S(w^k),\quad y^{k+1}=\phi_y(w^k);$$

Step 2. Project

$$\widetilde{w}^k=\left[\mathcal{P}_{\mathcal{S}^n_+}\circ\phi_{V_Z}(S^{k+1},y^{k+1},X_1^k),\quad \mathcal{P}_{\mathcal{N}^n}\circ\phi_{V_U}(S^{k+1},X_2^k)\right].\tag{43}$$

Step 3. Generalize

$$w^{k+1}=(1-\rho)w^k+\rho\widetilde{w}^k.$$

Set $k=k+1$, and go to Step 1.

Remark 1 *By (42), we have $\widetilde{w}^k=\mathcal{P}\circ v(w^k)$, and our projection method above is a Krasnosel'skiĭ-Mann algorithm with*

$$w^{k+1}=(1-\rho)w^k+\rho\mathcal{P}\circ v(w^k),$$

for any $\rho\in\left(0,2-\frac{\sigma\lambda_{max}(\varphi)}{2}\right)$. The parameter ρ is similar to the relaxation factor in the generalized Douglas-Rachford operator splitting [30]. Notice that $1<2-\frac{\sigma\lambda_{max}(\varphi)}{2}<2$, it can numerically accelerate our projection method for $\rho>1$.

Remark 2 *From the ADMM perspective, the projection method can be explained as a modified 3-block ADMM with larger step size ρ and a correction step for correcting (Z,U):*

$$\begin{cases} W^{k+1} & =X_1^{k+1}\\ (S^{k+1},y^{k+1}) & =arg\min\limits_{(S,y)\in\mathcal{S}^n\times\mathbb{R}^m}\widehat{\mathcal{L}}_\sigma\left(W^{k+1},(Z^k,U^k),(S,y),\widehat{X}^k\right),\\ (\widetilde{Z}^k,\widetilde{U}^k) & =arg\min\limits_{(Z,U)\in\mathcal{S}^n_+\times\mathcal{N}^n}\widehat{\mathcal{L}}_\sigma\left(W^{k+1},(Z,U),(S^{k+1},y^{k+1}),\widehat{X}^k\right),\\ \widehat{X}^{k+1} & =\widehat{X}^k+\rho\sigma(\mathcal{F}^*(W^{k+1})+\mathcal{G}^*(\widetilde{Z}^k,\widetilde{U}^k)+\mathcal{H}^*(S^{k+1},y^{k+1})-\tilde{C}),\\ \textit{Correction step} & \\ (Z^{k+1},U^{k+1}) & =(1-\rho)(Z^k,U^k)+\rho(\widetilde{Z}^k,\widetilde{U}^k).\end{cases}\tag{44}$$

Remark 3 *Restricting $\sigma\in(0,\frac{2}{\lambda_{max}(\varphi)})$ is to guarantee the convergence of our projection method, which is significantly larger than the range $\sigma\in(0,\frac{2}{5\|\Phi^T\Phi\|})$ shown in [17] for ADMM3d, where $vec(\varphi(X))=\Phi vec(X)$. For some problems with Φ having a larger eigenvalue, the restriction of σ on a small interval may hinder its effective adjustment according the progress of algorithm, and then reduce the convergence speed.*

Remark 4 *The projection method with $\rho = 1$ is not a classic 2-block ADMM, although its computational procedure is $(S, y) \to (Z, U) \to (X_1, X_2)$ when $\varphi(W^k)$ is set to be $\varphi(X_1^k)$. The reason is that, X_1 in our projection method plays a dual role: the Lagrangian multiplier and the variable of the first block.*

Remark 5 *If $\varphi = 0$, problem (1) is the SDP problem with nonnegative constraints and its dual reformulation (12) will be a 2-block convex optimization problem. In this case, our projection method is a classic 2-block ADMM but the step size can close to 2, it is convergent for any $\sigma > 0$.*

5 Convergence Analysis

In this section, we explore the properties of operators v, $\mathcal{P} \circ v$ and $v \circ \mathcal{P}$, and then establish the convergence of our projection method.

Lemma 4 *[36] For any V, $V^* \in \mathcal{S}_+^n$,*
(i). $\|\mathcal{P}_{\mathcal{S}_+^n}(V) - \mathcal{P}_{\mathcal{S}_+^n}(V^*)\|^2 \le \|V - V^*\|^2$, with equality holding if only if

$$\left(\Pi_{\mathcal{S}_+^n}(V)\right)^T (\Pi_{\mathcal{S}_+^n}(V^*) - V^*) = 0 \quad \text{and} \quad \left(\Pi_{\mathcal{S}_+^n}(V) - V\right)^T \Pi_{\mathcal{S}_+^n}(V^*) = 0.$$

(ii). $\|\mathcal{P}_{\mathcal{N}^n}(V) - \mathcal{P}_{\mathcal{N}^n}(V^*)\|^2 \le \|V - V^*\|^2$, with equality holding if only if

$$\left(\Pi_{\mathcal{N}^n}(V)\right)^T (\Pi_{\mathcal{N}^n}(V^*) - V^*) = 0 \quad \text{and} \quad \left(\Pi_{\mathcal{N}^n}(V) - V\right)^T \Pi_{\mathcal{N}^n}(V^*) = 0.$$

Lemma 5 *For any $w, w^* \in \mathcal{J}$, then we have*

$$\begin{aligned} v_Z(w) - v_Z(w^*) = &-\frac{1}{2}\mathcal{Q}_- \left((\frac{1}{\sigma}\mathcal{I} - \varphi)(X_1 - X_1^*) \right) + \frac{1}{2}\mathcal{Q}_+(Z - Z^*) \\ &-\frac{1}{2}\mathcal{Q}_- \left(\frac{1}{\sigma}(X_2 - X_2^*) + (U - U^*) \right), \end{aligned} \tag{45}$$

$$\begin{aligned} v_U(w) - v_U(w^*) = &-\frac{1}{2}\mathcal{Q}_- \left((\frac{1}{\sigma}\mathcal{I} - \varphi)(X_1 - X_1^*) \right) - \frac{1}{2}\mathcal{Q}_-(Z - Z^*) \\ &-\frac{1}{2}\mathcal{Q}_- \left(\frac{1}{\sigma}(X_2 - X_2^*) \right) + \frac{1}{2}\mathcal{Q}_+(U - U^*). \end{aligned} \tag{46}$$

Proof. Using the definition of ϕ_S and ϕ_y in (38), we deduce that

$$\begin{aligned} \phi_y(w) - \phi_y(w^*) = &-(\mathcal{A}\mathcal{A}^*)^{-1}\mathcal{A}\left((\frac{1}{\sigma}\mathcal{I} - \varphi)(X_1 - X_1^*) + (Z - Z^*) \right) \\ &+(\mathcal{A}\mathcal{A}^*)^{-1}\mathcal{A}\left(\frac{1}{\sigma}(X_2 - X_2^*) + (U - U^*) \right), \end{aligned} \tag{47}$$

$$\begin{aligned} \phi_S(w) - \phi_S(w^*) = &-\frac{1}{2}\mathcal{Q}_- \left((\frac{1}{\sigma}\mathcal{I} - \varphi)(X_1 - X_1^*) + (Z - Z^*) \right) \\ &+\frac{1}{2}\mathcal{Q}_+ \left(\frac{1}{\sigma}(X_2 - X_2^*) + (U - U^*) \right). \end{aligned} \tag{48}$$

Together with the definition of ϕ_{V_Z} and ϕ_{V_U} in (38), it is not difficult to get the results. □

Lemma 6 *Suppose that $\varphi \ne 0$, for any $\sigma > 0$ such that $\sigma < \frac{2}{\lambda_{max}(\varphi)}$, then*

$$\left\| (\frac{1}{\sigma}\mathcal{I} - \varphi)(X) \right\|^2 \le \left\| \frac{1}{\sigma} X \right\|^2, \tag{49}$$

with equality holding if and only if $\varphi(X) = 0$.

Proof. Recall that φ is a self-adjoint positive semidefinite linear operator, we have

$$\left\|(\frac{1}{\sigma}\mathcal{I}-\varphi)(X)\right\|^2 = \left\|\frac{1}{\sigma}X\right\|^2 - \left\langle X, (\frac{2}{\sigma}\varphi-\varphi^2)(X)\right\rangle.$$

It follows from $\sigma < \frac{2}{\lambda_{max}(\varphi)}$ that $\frac{2}{\sigma}\mathcal{I}-\varphi$ is positive definite, then $\frac{2}{\sigma}\varphi-\varphi^2$ is positive semidefinite. Namely, $\langle(X), (\frac{2}{\sigma}\varphi-\varphi^2)(X)\rangle \geq 0$, so $\|(\frac{1}{\sigma}\mathcal{I}-\varphi)(X)\|^2 \leq \|\frac{1}{\sigma}X\|^2$.

If the equality in (49) holds, $\langle X, (\frac{2}{\sigma}\varphi-\varphi^2)(X)\rangle = 0$, which implies $\varphi(X) = 0$. □

Lemma 7 *For any $w, w^* \in \mathcal{J}$, we have*

(i).

$$\|v(w)-v(w^*)\|_F^2 \leq \|w-w^*\|_F^2. \tag{50}$$

(ii). If the equality in (50) holds, and $w^* \in Fix\,\mathcal{P}\circ v$ then

$$v_Z(w) = -\frac{1}{\sigma}X_1 + Z, \tag{51}$$

$$v_U(w) = -\frac{1}{\sigma}X_2 + U. \tag{52}$$

Proof. (i). Since for any matrix A, B, $\|A\|_F^2 + \|B\|_F^2 = \frac{1}{2}(\|A+B\|_F^2 + \|A-B\|_F^2)$, by Lemma 5 we infer that

$$\begin{aligned}
&\|v(w)-v(w^*)\|_F^2 = \left\|\begin{matrix} v_Z(w)-v_Z(w^*) \\ v_U(w)-v_U(w^*)\end{matrix}\right\|_F^2 \\
&= \frac{1}{2}\|(v_Z(w)-v_Z(w^*)) + (v_U(w)-v_U(w^*))\|_F^2 + \frac{1}{2}\|(v_Z(w)-v_Z(w^*)) - (v_U(w)-v_U(w^*))\|_F^2 \\
&= \frac{1}{2}\left\|\mathcal{M}((Z-Z^*)+(U-U^*)) - \mathcal{Q}_-\left((\frac{1}{\sigma}\mathcal{I}-\varphi)(X_1-X_1^*) + \frac{1}{\sigma}(X_2-X_2^*)\right)\right\|_F^2 \\
&\quad + \frac{1}{2}\|(Z-Z^*)-(U-U^*)\|_F^2.
\end{aligned}$$

From the definition of $\mathcal{M}$ and $\mathcal{Q}_-$, the spectral radius of the operator $\mathcal{M}$ and $\mathcal{Q}_-$ is no more than 1 and $\mathcal{M}\mathcal{Q}_- = 0$, then

$$\begin{aligned}
&\frac{1}{2}\left\|\mathcal{M}((Z-Z^*)+(U-U^*)) - \mathcal{Q}_-\left((\frac{1}{\sigma}\mathcal{I}-\varphi)(X_1-X_1^*) + \frac{1}{\sigma}(X_2-X_2^*)\right)\right\|_F^2 \\
&\leq \frac{1}{2}\|(Z-Z^*)+(U-U^*)\|_F^2 + \frac{1}{2}\left\|(\frac{1}{\sigma}\mathcal{I}-\varphi)(X_1-X_1^*) + \frac{1}{\sigma}(X_2-X_2^*)\right\|_F^2.
\end{aligned} \tag{53}$$

If the equality above holds, that is,

$$(\mathcal{I}-\mathcal{M})((Z-Z^*)+(U-U^*)) = 0, \tag{54}$$

$$\mathcal{M}\left((\frac{1}{\sigma}\mathcal{I}-\varphi)(X_1-X_1^*) + \frac{1}{\sigma}(X_2-X_2^*)\right) = 0. \tag{55}$$

Thus,

$$\begin{aligned}
&\|v(w)-v(w^*)\|_F^2\\
&\le \frac{1}{2}\|(Z-Z^*)-(U-U^*)\|_F^2+\frac{1}{2}\|(Z-Z^*)+(U-U^*)\|_F^2\\
&\quad +\frac{1}{2}\left\|(\frac{1}{\sigma}\mathcal{I}-\varphi)(X_1-X_1^*)+\frac{1}{\sigma}(X_2-X_2^*)\right\|_F^2\\
&= \|Z-Z^*\|_F^2+\|U-U^*\|_F^2+\frac{1}{2}\left\|(\frac{1}{\sigma}\mathcal{I}-\varphi)(X_1-X_1^*)+\frac{1}{\sigma}(X_2-X_2^*)\right\|_F^2\\
&\le \|Z-Z^*\|_F^2+\|U-U^*\|_F^2+\left\|(\frac{1}{\sigma}\mathcal{I}-\varphi)(X_1-X_1^*)\right\|_F^2+\left\|\frac{1}{\sigma}(X_2-X_2^*)\right\|_F^2. \qquad (56)
\end{aligned}$$

If the equality in (56) holds, that is,

$$(\frac{1}{\sigma}\mathcal{I}-\varphi)(X_1-X_1^*)=\frac{1}{\sigma}(X_2-X_2^*). \qquad (57)$$

For any $\sigma\in(0,\frac{2}{\lambda_{max}(\varphi)})$, it follows from Lemma 6 that

$$\left\|(\frac{1}{\sigma}\mathcal{I}-\varphi)(X_1-X_1^*)\right\|_F^2\le\left\|\frac{1}{\sigma}(X_1-X_1^*)\right\|_F^2 \qquad (58)$$

with equality holding if and only if

$$\varphi(X_1-X_1^*)=0, \qquad (59)$$

which implies $(\frac{1}{\sigma}\mathcal{I}-\varphi)(X_1-X_1^*)=\frac{1}{\sigma}(X_1-X_1^*)$. Thus, we deduce

$$\begin{aligned}
\|v(w)-v(w^*)\|_F^2 &\le \|Z-Z^*\|_F^2+\|U-U^*\|_F^2+\left\|\frac{1}{\sigma}(X_2-X_2^*)\right\|_F^2+\left\|\frac{1}{\sigma}(X_1-X_1^*)\right\|_F^2\\
&= \|w-w^*\|_F^2.
\end{aligned}$$

(ii). If the equality in (50) holds, it also holds in (53), (56) and (58), so we have the conditions (54), (55), (57) and (59). Substituting these conditions into the results of Lemma 5, we have

$$\begin{aligned}
&v_Z(w)-v_Z(w^*)\\
&=-\frac{1}{2}\mathcal{Q}_-\left((\frac{1}{\sigma}\mathcal{I}-\varphi)(X_1-X_1^*)+\frac{1}{\sigma}(X_2-X_2^*)\right)+\frac{1}{2}\mathcal{Q}_+(Z-Z^*)-\frac{1}{2}\mathcal{Q}_-(U-U^*),\\
&=-\frac{1}{2}\left((\frac{1}{\sigma}\mathcal{I}-\varphi)(X_1-X_1^*)+\frac{1}{\sigma}(X_2-X_2^*)\right)+(Z-Z^*),\\
&=-\frac{1}{\sigma}(X_1-X_1^*)+(Z-Z^*)\\
&=\left(-\frac{1}{\sigma}X_1+Z\right)-\left(-\frac{1}{\sigma}X_1^*+Z^*\right).
\end{aligned}$$

By using $w^*\in \text{Fix}\,\mathcal{P}\circ v$ and the proof of Lemma 3, we get $v_Z(w^*)=-\frac{1}{\sigma}X_1^*+Z^*$. Therefore, the equality $v_Z(w)-v_Z(w^*)=\left(-\frac{1}{\sigma}X_1+Z\right)-\left(-\frac{1}{\sigma}X_1^*+Z^*\right)$ implies $v_Z(w)=-\frac{1}{\sigma}X_1+Z$. Similarly, we can obtain $v_U(w)=-\frac{1}{\sigma}X_2+U$ when the equality in (50) holds. The proof is finished. □

Theorem 2 *For the sequence $\{w^k\}$ generated by Algorithm 1 with $\rho=1$, we have*

$$\left\|w^{k+1}-w^*\right\|_F^2=\left\|\mathcal{P}\circ v(w^k)-\mathcal{P}\circ v(w^*)\right\|_F^2\le\left\|w^k-w^*\right\|_F^2,$$

where $w^*\in Fix\,\mathcal{P}\circ v$, Proof. It follows from Remark 1 and Lemma 4 that,

$$\begin{aligned}\left\|w^{k+1}-w^*\right\|_F^2 &= \left\|\mathcal{P}\circ v(w^k)-\mathcal{P}\circ v(w^*)\right\|_F^2\\ &= \left\|\begin{matrix}\mathcal{P}_{\mathcal{S}^n_+}\circ v_Z(w^k)-\mathcal{P}_{\mathcal{S}^n_+}\circ v_Z(w^*)\\ \mathcal{P}_{\mathcal{N}^n}\circ v_U(w^k)-\mathcal{P}_{\mathcal{N}^n}\circ v_U(w^*)\end{matrix}\right\|_F^2\\ &\le \left\|\begin{matrix}v_Z(w^k)-v_Z(w^*)\\ v_U(w^k)-v_U(w^*)\end{matrix}\right\|_F^2\\ &= \left\|v(w^k)-v(w^*)\right\|_F^2.\end{aligned}$$

By Lemma 7, we can obtain the results. □

Theorem 2 show that the operator $\mathcal{P}\circ v$ is quasinonexpansive. Next, we will further explore that the operator $\mathcal{P}\circ v$ is α-averaged with coefficient

$$\alpha := \frac{2}{4-\sigma\lambda_{max}(\varphi)} \in (\frac{1}{2}, 1), \tag{60}$$

for $\sigma \in (0, \frac{2}{\lambda_{max}(\varphi)})$.

Theorem 3 *For the operators v and $\mathcal{P}$ defined in (40) and (42), we have*
(i). $v\circ\mathcal{P} = \mathcal{L}^{-1}\circ\mathcal{T}\circ\mathcal{L}$ with $\mathcal{L} = -\sigma\mathcal{I}$ and

$$\mathcal{T} = \mathcal{I} - \mathcal{J}_{\sigma g} + \mathcal{J}_{\sigma\mathcal{N}_{\mathcal{K}}}\circ\left(2\mathcal{J}_{\sigma g} - \mathcal{I} - \sigma\nabla\theta\circ\mathcal{J}_{\sigma g}\right), \tag{61}$$

where $\mathcal{K} = \{U\in\mathcal{S}^n\times\mathcal{S}^n | \mathcal{H}(U) = \widetilde{b}\}$.
(2). $\mathcal{P}\circ v$ is α-averaged.

Proof. (i). For the sequence $\{w^k\}$ generated by Algorithm 1 with $\rho = 1$, we have

$$\begin{aligned}\mathcal{T}\circ\mathcal{L}\circ v(w^k) &= \mathcal{T}\begin{pmatrix}-\sigma v_Z(w^k)\\ -\sigma v_U(w^k)\end{pmatrix}\\ &= \begin{pmatrix}-\sigma V_Z^{k+1}+X_1^{k+1}\\ -\sigma V_U^{k+1}+X_2^{k+1}\end{pmatrix} + \Pi_{\mathcal{K}}\begin{pmatrix}2X_1^{k+1}+\sigma V_Z^{k+1}-\sigma(\varphi(W^{k+1})+C)\\ 2X_2^{k+1}+\sigma V_U^{k+1}\end{pmatrix}\\ &= \begin{pmatrix}-\sigma Z^{k+1}\\ -\sigma U^{k+1}\end{pmatrix} + \Pi_{\mathcal{K}}\begin{pmatrix}X_1^{k+1}-\sigma(\varphi(W^{k+1})-Z^{k+1}+C)\\ X_2^{k+1}+\sigma U^{k+1}\end{pmatrix}.\end{aligned}$$

Note that

$$\Pi_{\mathcal{K}}(U) = U - \mathcal{H}^*(\mathcal{H}\mathcal{H}^*)^{-1}(\mathcal{H}U-\widetilde{b}), \quad \forall\ U\in\mathcal{S}^n\times\mathcal{S}^n,$$

then, by using (20) we can deduce

$$\begin{aligned}\mathcal{T}\circ\mathcal{L}\circ v(w^k) &= \begin{pmatrix}-\sigma Z^{k+1}\\ -\sigma U^{k+1}\end{pmatrix} + \begin{pmatrix}X_1^{k+1}-\sigma(\varphi(W^{k+1})-Z^{k+1}+C)+\sigma S^{k+2}+\sigma\mathcal{A}^*(y^{k+2})\\ X_2^{k+1}+\sigma U^{k+1}-\sigma S^{k+2}\end{pmatrix}\\ &= \begin{pmatrix}X_1^{k+1}-\sigma(\varphi(W^{k+1})+C)+\sigma S^{k+2}+\sigma\mathcal{A}^*(y^{k+2})\\ X_2^{k+1}-\sigma S^{k+2}\end{pmatrix}\\ &= -\sigma v(w^{k+1})\\ &= -\sigma v\circ\mathcal{P}\circ v(w^k).\end{aligned}$$

By $\mathcal{L} = -\sigma\mathcal{I}$, we have $v\circ\mathcal{P} = \mathcal{L}^{-1}\circ\mathcal{T}\circ\mathcal{L}$.

(2). Note that $\nabla\theta$ is $\frac{1}{\lambda_{max}(\varphi)}$-cocoercive, it follows from [31, Proposition 2.1] that $\mathcal{T}$ is α-averaged. Then, $v\circ\mathcal{P}$ is α-averaged from (i). Since $\mathcal{P}$ is invertible, we have $\mathcal{P}\circ v = \mathcal{P}\circ\mathcal{L}^{-1}\circ\mathcal{T}\circ\mathcal{L}\circ\mathcal{P}^{-1}$, which implies $\mathcal{P}\circ v$ is α-averaged too. □

It follows from [29, Proposition 5.15] and [38, Theorem 1], the convergence result for Algorithm 1 is stated in the following theorem.

Theorem 4 *Under Assumptions 1 and 2, for any* $\sigma \in (0, \frac{2}{\lambda_{max}(\varphi)})$ *and* $\rho \in (0, \frac{1}{\alpha})$*, assume the sequence* $\{w^k\}$ *is generated by Algorithm 1. Then the following results hold:*

(i). For any $w^* \in Fix\, \mathcal{P} \circ v$, $\{\|w^{k+1} - w^*\|\}$ *is monotonically nonincreasing.*

(ii). The fixed-point residual sequence $\{\|\mathcal{P} \circ v(w^k) - w^k\|\}$ *is monotonically nonincreasing and converges to 0.*

(iii). The sequence $\{w^k\}$ *converges to some point* $w^* \in Fix\, \mathcal{P} \circ v$.

(iv). $\|\mathcal{P} \circ v(w^k) - w^k\|^2 \leq \frac{\frac{\alpha^2}{\rho(\alpha-\rho)}\|w^0 - w^*\|^2}{k+1}$ *and* $\|\mathcal{P} \circ v(w^k) - w^k\|^2 = o(\frac{1}{k+1})$.

Theorem 4 shows that, the sequence $\{w^k\}$ generated by Algorithm 1 is convergent to a solution w^* of the system (39). Therefore, by Theorem 1 we can obtain a KKT point $(W^*, Z^*, y^*, S^*, X^*)$ with

$$\varphi(W^*) = \varphi(X^*), \quad X^* = X_1^*, \quad S^* = U^* \quad \text{and} \quad y^* = \phi_y(w^*),$$

where ϕ_y is defined as in (38). Since the simplest choice of W is X satisfying $\varphi(W) = \varphi(X)$, so we set $W^k = X^k = X_1^k$ in the following numerical experiments.

6 Numerical experiments

In this section, we report the numerical performance of our projection method for the CQSDP problems generated randomly in MATLAB R2013B. We denote the random number generator by *seed* for generating data again in MATLAB. 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.

6.1 Doubly non-negative CQSDP problems

In our numerical experiments, we test two types of doubly non-negative CQSDP problems. One is that with $\varphi(X) = \frac{BX+XB}{2}$ for a given matrix $B \in \mathcal{S}_+^n$. So in this case, $\lambda_{max}(\varphi) = \lambda_{max}(B)$. The matrix B is a random symmetric positive semidefinite matrix, generated by `temp=randn(n,r); B=temp*temp';` We set $r = 10$, i.e., rank(B)=10 as in [3]. The other is with $\varphi(X) = X$, as for least squares semidefinite programming in [13,14], then $\lambda_{max}(\varphi) = 1$.

In this paper, we test the problems arising from the relaxation of maximum stable set problems and a binary integer nonconvex quadratic (BIQ) programming. The instances are considered as in [3], [32], and [37]. For instance, we construct QSDP-BIQ problem sets based on the formulation in [3] as follows:

$$\begin{aligned} \min \quad & \frac{1}{2}\langle X, \varphi(X)\rangle + \langle Q, X_0\rangle + \langle c, x\rangle \\ s.t. \quad & \text{diag}(X_0) - x = 0, \ \alpha = 1, \\ & X = \begin{pmatrix} X_0 & x \\ x^T & \alpha \end{pmatrix} \in \mathcal{S}_+^n, \quad X \in \mathcal{C}^n. \end{aligned} \tag{62}$$

The test data for Q and c are taken from Biq Mac Library maintained by Wiegele, which is available at http://biqmac.uni-klu.ac.at/biqmaclib.html. In the same sprit, we construct test problems QSDP-BIQ and QSDP-θ_+.

6.2 Numerical results

In this section, we report the numerical results obtained by our projection method, ADM-G and ADMM4d in solving various instances of the random CQSDP problems with nonnegative constraints.

In order to compare with ADM-G ($\alpha = 0.99$) and ADMM4d ($\tau = 1.618$) for solving 4-block dual (2), we measure the accuracy of the approximate optimal solution (W, y, Z, S, X) (with $W = X$) by using the following relative residual:

$$\delta := \max\ \{\text{pinf},\ \text{dinf},\ p\delta_{\mathcal{S}^n_+},\ p\delta_{\mathcal{N}^n},\ d\delta_{\mathcal{S}^n_+},\ d\delta_{\mathcal{N}^n},\ \delta_{XZ},\ \delta_{XS}\}, \tag{63}$$

where

$$\text{pinf} = \frac{\|\mathcal{A}(X) - b\|}{1 + \|b\|}, \qquad \text{dinf} = \frac{\|C + \varphi(X) - Z - \mathcal{A}^*(y)\|}{1 + \|C\|}, \tag{64}$$

$$p\delta_{\mathcal{S}^n_+} = \frac{\|\varPi_{\mathcal{S}^n_+}(-X)\|}{1 + \|X\|}, \qquad p\delta_{\mathcal{N}^n} = \frac{\|\varPi_{\mathcal{N}^n}(-X)\|}{1 + \|X\|}, \tag{65}$$

$$d\delta_{\mathcal{S}^n_+} = \frac{\|\varPi_{\mathcal{S}^n_+}(-Z)\|}{1 + \|Z\|}, \qquad d\delta_{\mathcal{N}^n} = \frac{\|\varPi_{\mathcal{N}^n}(-S)\|}{1 + \|S\|}, \tag{66}$$

$$\delta_{XZ} = \frac{\left|\langle X, Z\rangle\right|}{1 + \|X\| + \|Z\|}, \qquad \delta_{XS} = \frac{\left|\langle X, S\rangle\right|}{1 + \|X\| + \|S\|}. \tag{67}$$

Additionally, we compute the relative gap by

$$\delta_g = \frac{|\text{pobj} - \text{dobj}|}{1 + |\text{pobj}| + |\text{dobj}|},$$

where $\text{pobj} = \frac{1}{2}\langle X, \varphi(X)\rangle + \langle C, X\rangle$ and $\text{dobj} = -\frac{1}{2}\langle X, \varphi(X)\rangle + b^T y$.

We choose the initial point $X_1^0 = X_2^0 = \mathcal{A}^*(\mathcal{A}\mathcal{A}^*)^{-1}(b)$, and $Z^0 = U^0 = 0$. We terminate all the solvers when $\delta < 10^{-6}$ with the maximum number of iterations set at 25000.

The penalty parameter σ is dynamically adjusted according to the progress of the algorithms, but it satisfies $0 < \sigma < \frac{2}{\lambda_{max}(\varphi)}$ for our projection method from the discussion in Section 5. Thus, we set $\sigma_{max} = \epsilon_0 \frac{2}{\lambda_{max}(\varphi)}$ (ϵ_0 is a constant, e.g., $\epsilon_0 = 0.999$.) and $\sigma_{min} = 10^{-6}$ for our projection method. In our numerical experiments, we use the same adjustment strategy for our projection method, ADM-G and ADMM4d to solve all the tested problems, but $\sigma_{max} = 10^6$ and $\sigma_{min} = 10^{-6}$ for ADM-G and ADMM4d. The key idea for adjusting σ is to balance the progress of primal and dual feasibilities: $\eta_p = \max\{\text{pinf},\ p_{\mathcal{S}^n_+},\ p_{\mathcal{N}^n}\}$ and $\eta_d = \max\{\text{dinf},\ d_{\mathcal{S}^n_+},\ d_{\mathcal{N}^n}\}$. For details, see Appendix 1.

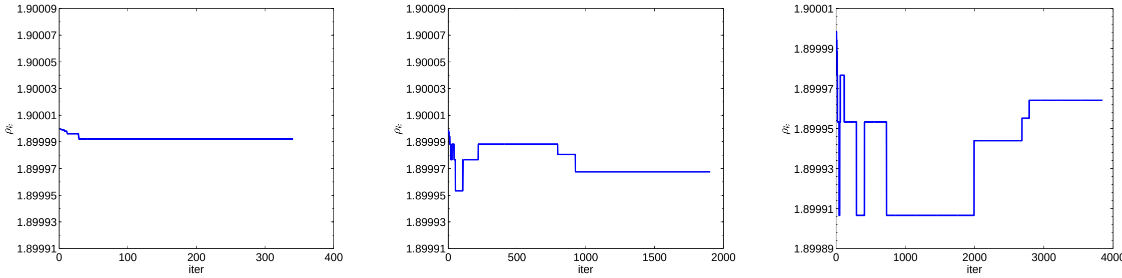
The initial σ_0 for our projection method is chosen to be $10^{-2} \times \sigma_{max}$. For ADM-G and ADMM4d, we set $\sigma_0 = 1$. We use σ_k to denote the penalty parameter at k-th iteration, set

$$\rho_k = \eta \frac{1}{\alpha_k} \quad \text{with} \quad \alpha_k = \frac{2}{4 - \sigma_k \lambda_{max}(\varphi)},$$

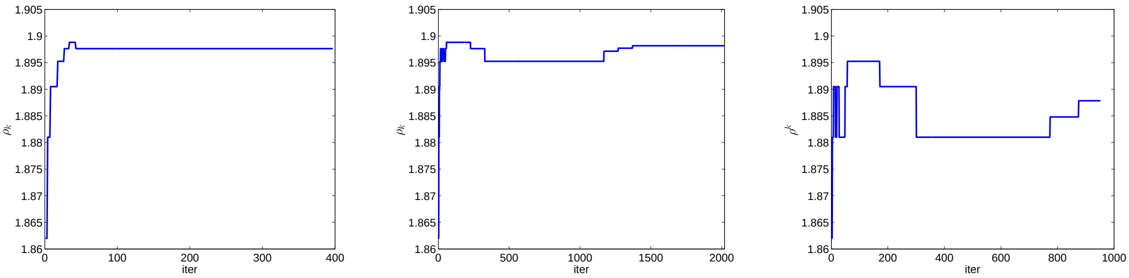
where $\eta \in (0, 1)$ since $\rho_k < \frac{1}{\alpha_k}$. Generally, the larger η can produce better results, so we set $\eta = 0.95$ in this paper. Figure 1 shows the evolutions of ρ_k with respect to iterations for the problems "theta4", "be100.1" and "gka1d", from the results shown we see that the step size ρ_k (and relaxation factor) is always greater than 1.85 in spite of fluctuating with respect to σ_k.

The detailed numerical results are reported in the tables 1-4. Figures 2 and 3 show the performance profiles in terms of the number of iterations and computing time for all the problems tested with $\varphi(X) = \frac{BX+XB}{2}$ and $\varphi(X) = X$, respectively.

Recall that a point (x, y) is in the performance profiles curve of a method if and only if it can solve $(100y)\%$ of all the tested problems no slower than x times of any other methods. We may observe that, our project method takes the least number of iterations and computational time for the majority of the tested

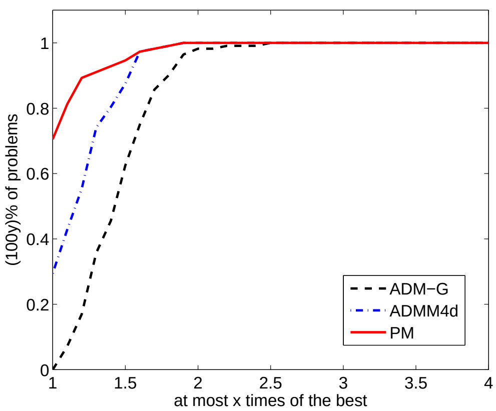


(a) “theta4” with $\varphi(X) = \frac{BX+XB}{2}$. (b) “be100.1” with $\varphi(X) = \frac{BX+XB}{2}$. (c) “gka1d” with $\varphi(X) = \frac{BX+XB}{2}$.

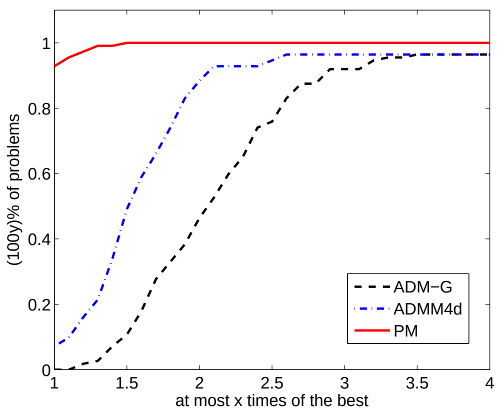


(d) “theta4” with $\varphi(X) = X$. (e) “be100.1” with $\varphi(X) = X$. (f) “gka1d” with $\varphi(X) = X$.

Fig. 1 Evolutions of ρ_k with respect to iterations.



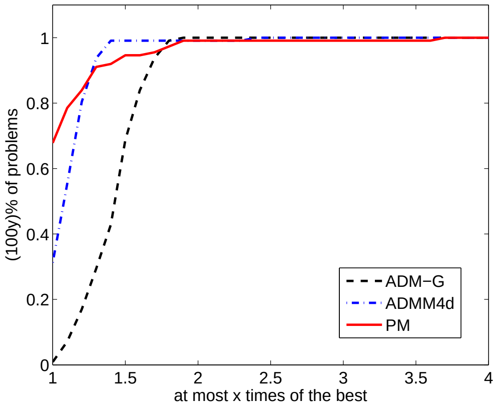
(a) iteration.



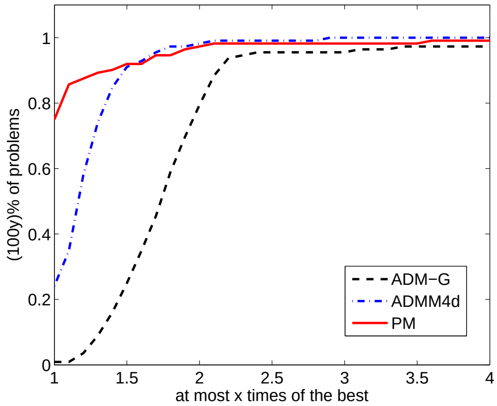
(b) time.

Fig. 2 Performance profiles (iteration and time) of our projection method, ADMM4d and ADM-G for the CQSDP problems with $\varphi(X) = \frac{BX+XB}{2}$ (Tables 1-2).

problems. The main reason behind the efficiency of our projection method, we think, is lager step size (can be greater than $(1+\sqrt{5})/2$) and skipping the computation of W^{k+1}. In addition, our project method and ADMM4d outperform ADM-G in terms of iteration and computational time, even though the convergence of ADMM4d can not be guaranteed.



(a) iteration.



(b) time.

Fig. 3 Performance profiles (iteration and time) of our projection method, ADMM4d and ADM-G for the CQSDP problems with $\varphi(X) = X$ (Tables 3-4).

7 Conclusion

In this paper, we presented a projection method based on the KKT condition for solving the CQSDP problems with nonnegative constraints, and establish its global convergence and $o(\frac{1}{k+1})$ convergence rate. At each iteration, our projection method does not have to solve the subproblem with variable W, compared to the existing multi-block ADMM [3,21] for solving (4) and (12). Numerical experiments on various large scale QSDPs have demonstrated the efficiency of our proposed ADMM in finding medium accuracy solutions.

References

1. Thomas D., Florian J., On the stable solution of large scale problems over the doubly nonnegative cone, Math. Program., Ser. A (2014) 146:299-323
2. Li X., Sun D., Toh K.-C., QSDPNAL: a two-phase augmented Lagrangian method for convex quadratic semidefinite programming. Mathematical Programming Computation, 2018:1-41.
3. Li X., Sun D., Toh K.C., A Schur Complement Based Semi-Proximal ADMM for Convex Quadratic Conic Programming and Extensions, Math. Program. (2014) 1-41.
4. Toh K.-C., Tutuncu R. H.,Todd M. J., Inexact primal-dual path- following algorithms for a special class of convex quadratic SDP and related problems, Pac. J. Optim. 3 (2007) 135-164.
5. Toh K.-C., An inexact primal-dual path-following algorithm for convex quadratic SDP, Math. Program. 112 (2008) 221-254.
6. Li L., Toh K.-C., A polynomial-time inexact primal-dual infeasible path-following algorithm for convex quadratic SDP, Pac. J. Optim. 7(2011) 43-61.
7. Jiang K., Sun D., Toh K.-C., An inexact accelerated proximal gradient method for large scale linearly constrained convex SDP, SIAM J. Optim. 22 (2012) 1042-1064.
8. Sun J.,Zhang S., A modified alternating direction method for convex quadratically constrained quadratic semidefinite programs, EUR. J. Oper. Res. 207(2010)1210-1220.
9. Chang, X., Liu, S., A 2-block semi-proximal ADMM for solving the H-weighted nearest correlation matrix problem, Optimization, 2017, 66(1):1-16
10. Lin H., An inexact spectral bundle method for convex quadratic semidefinite programming, Comput. Optim. Appl. 53 (2012) 45-89.
11. Qi H., Sun D., An augmented Lagrangian dual approach for the H-weighted nearest correlation matrix problem, IMA J. Numer. Anal. 31 (2011) 491-511.
12. Boyd S., Xiao L., Least-squares covariance matrix adjustment, SIAM J. Matrix Anal. Appl. 27 (2005) 532-546.
13. Li G. , Ma A. K. C., Pong T. K., Robust least square semidefinite programming with applications, Comput. Optim. Appl. 58 (2014) 347-379.
14. Gao Y., Sun D., Calibrating least squares semidefinite programming with equality and inequality constraints. SIAM J. Matrix Anal. Appl. 31(3), 1432-1457[J]. SIAM Journal on Matrix Analysis & Applications, 2009, 31(3):1432-1457.
15. Chen C., He B., Ye Y., Yuan X., The direct extension of admm for multi-block convex minimization problems is not necessarily convergent, Mathe. Program. Series A, (2014) 1-23.

16. He B., Tao M., Yuan X., Alternating direction method with gaussian back substitution for separable convex programming, SIAM J. Optim. 22 (2012) 313-340.
17. Tao M., Yuan X, Convergence analysis of the direct extension of ADMM for multiple-block separable convex minimization, arXiv:1609.07221 (2016)
18. He B., Tao M., Yuan X., Alternating direction method with Gaussian back substitution for separable convex programming, SIAM Journal on Optimization, 22, 313-340, 2012
19. He B., Yuan X., A class of ADMM-based algorithms for multi-block separable convex programming, manuscript, 2015;
20. Chang X., Liu S., Zhao P., Li X., Convergent prediction-correction-based ADMM for multi-block separable convex programming. Journal of Computational and Applied Mathematics, 335:270C288, 2018.
21. Chen L., Sun D., Toh K.-C., An efficient inexact symmetric Gauss-Seidel based majorized ADMM for high-dimensional convex composite conic programming, Math. Program., (2017) 161:237-270.
22. He B., Liao L., Han D., Yang H., A new inexact alternating direction method for monotone variational inequalities, Math. Program. 92 (2002) 103-118.
23. Bai J., Li J., Xu F., Generalized symmetric ADMM for separable convex optimization. Computational Optimization & Applications, 2018, 70(1):129-170.
24. He B., Tao M., Yuan X., A splitting method for separable convex programming, IMA Journal of Numerical Analysis, 35(1), 394-426, 2015;
25. Wang W., Hong M., Ma S., Luo Z., Solving multiple-block separable convex minimization problems using two-block alternating direction method of multipliers, Pacific J. Optim., 11(2015): 645-667
26. Zhao W., Sun D., Toh K.-C., A Newton-CG augmented Lagrangian method for semidefinite programming, SIAM Journal on Optimization, 20 (2010), pp. 1737-1765.
27. Yang L., Sun D., Toh K.-C., SDPNAL+: a majorized semismooth Newton-CG augmented Lagrangian method for semidefinite programming with nonnegative constraints, Math. Prog. Comp.
28. Chang X., Liu S., Li X., Modified alternating direction method of multipliers for convex quadratic semidefinite programming, Neurocomputing, 2016,214:575-586
29. Bauschke H.H., Combettes P.L., Convex Analysis and Monotone Operator Theory in Hilbert Spaces. Springer-Verlag, New York, 2011
30. Lions P.L., Mercier B., Splitting algorithms for the sum of two nonlinear operators. SIAM J. Numer. Anal. 16(6), 964–979 (1979)
31. Davis D., Yin W., A Three-Operator Splitting Scheme and its Optimization Applications, UCLA CAM Report 15-13, 2015.
32. Sun D., Toh K.-C., Yang L., A convergent 3-block semi-proximal alternating direction method of multipliers for conic programming with 4-type of constraints, SIAM J. Optim. 25 (2015) 882-915.
33. Borwein J.M., Lewis A.S., Convex Analysis and Nonlinear Optimization: Theory and Examples, vol. 3. Springer, Berlin (2006)
34. Wen Z., Yin W., A Feasible Method for Optimization with Orthogonality Constraints, Math. Program. vol. 142(1-2), pp. 397-434, 2013.
35. Hiriart-Urruty J.-B., Lemaréchal C., Convex analysis and minimization algorithms. I. Fundamentals. In: Grundlehren der Mathematischen Wissenschaften. Fundamental Principles of Mathematical Sciences, vol. 305. Springer, Berlin (1993)
36. Wen Z., Goldfarb D., Yin W., Alternating direction augmented Lagrangian methods for semidefinite programming, Math. Prog. Comp. 2 (2010) 203-230.
37. Monteiro R., Ortiz C., Svaiter B., A first-order block-decomposition method for solving two-easy-block structured semidefinite programs, Mathematical Programming Computation, (2013), pp. 1-48.
38. Davis D., Yin, W., Convergence rate analysis of several splitting schemes. arXiv: 1406.4834v2 (2014)

Appendix 1 *Details on the adjusting of σ.*

The key idea for adjusting σ is to balance the progress of primal and dual feasibilities:

`eta_1`=max$\{pinf,\ p_{\mathcal{S}^n_+},\ p_{\mathcal{N}^n}\}$, `eta_2`=max$\{dinf,\ d_{\mathcal{S}^n_+},\ d_{\mathcal{N}^n}\}$.

Let

`theta`=max $\{pinf,\ dinf,\ p\delta_{\mathcal{S}^n_+},\ p\delta_{\mathcal{N}^n},\ d\delta_{\mathcal{S}^n_+},\ d\delta_{\mathcal{N}^n},\ \delta_{XZ},\ \delta_{XS}\}$,

`gamma=0.5;` `sigma_max`=σ_{max} *and* `sigma_min`=σ_{min}.

the adjusting of σ is expressed as following:

```
 dtmp=eta_1/eta_2;
 if iter<=21;          h=3;
    elseif iter<=61;   h=6;
    elseif iter<=121;  h=50;
    else               h=100;
 end
if       theta<1e-5;   gamma=0.8;
elseif   theta<1e-3;   gamma=0.6;
end



it_pinf = 0;
it_dinf = 0;
if dtmp<=0.8
   it_pinf = it_pinf+1;it_dinf = 0;
      if it_pinf>h
         sigma = min((1/gamma)*sigma,sigma_max);  it_pinf=0;
      end
else if dtmp>1.25
        it_dinf = it_dinf+1;it_pinf = 0;
          if it_dinf>h
              sigma = max(gamma*sigma,sigma_min); it_dinf = 0;
          end
      end
 end
```

Table 1 The performance of our projection method, ADMM4d and ADM-G on the CQSDP problems with $\varphi(X) = \frac{BX+XB}{2}$ ($seed = 1$). In the table, "PM" and "4d" stands for our projection method and ADMM4d, respectively.

			iteration			δ			δ_g			time (second)		
problem	m_E	n_S	ADM-G	4d	PM	ADM-G	4d	PM	ADM-G	4d	PM	ADM-G	4d	PM
bqp50-1	51	51	3176	2602	2262	9.99e-7	9.95e-7	9.97e-7	2.10e-7	9.93e-8	3.46e-7	23.5	11.0	9.3
bqp50-2	51	51	3918	3100	3077	9.98e-7	1.00e-6	9.94e-7	7.24e-7	9.77e-7	4.90e-7	57.1	23.4	29.3
bqp50-3	51	51	2100	1611	1268	9.93e-7	9.97e-7	9.91e-7	1.78e-6	1.64e-6	8.89e-7	40.9	17.8	17.6
bqp50-4	51	51	2643	2197	1812	9.98e-7	9.98e-7	9.96e-7	1.83e-7	5.04e-7	4.42e-7	53.0	27.4	22.9
bqp50-5	51	51	1695	1367	797	9.98e-7	9.94e-7	9.98e-7	1.41e-6	2.76e-6	2.62e-7	33.2	17.6	10.5
bqp50-6	51	51	2125	1690	1326	9.73e-7	9.95e-7	9.96e-7	9.32e-7	1.12e-6	7.83e-7	42.0	23.1	18.6
bqp50-7	51	51	1924	1486	1020	9.98e-7	9.90e-7	9.93e-7	5.28e-8	7.48e-7	3.48e-7	43.1	20.7	13.2
bqp50-8	51	51	2395	2108	1521	9.96e-7	9.96e-7	9.93e-7	2.28e-8	7.30e-8	3.28e-7	48.3	31.1	20.8
bqp50-9	51	51	2097	1664	1106	9.97e-7	9.94e-7	9.99e-7	6.92e-7	1.21e-6	6.63e-7	43.0	24.8	16.1
bqp50-10	51	51	2538	2094	1919	9.97e-7	9.99e-7	9.86e-7	8.18e-8	5.65e-8	2.19e-8	55.8	33.9	28.4
bqp100-1	101	101	3550	3171	1879	9.99e-7	1.00e-6	9.98e-7	1.07e-7	7.39e-9	1.73e-8	92.1	69.8	36.1
bqp100-2	101	101	7794	6106	6652	1.00e-6	1.00e-6	1.00e-6	6.69e-7	7.50e-7	5.28e-7	216.3	141.1	129.0
bqp100-3	101	101	3337	3036	2291	9.99e-7	9.99e-7	9.97e-7	2.95e-7	5.34e-7	3.96e-7	85.1	62.4	45.6
bqp100-4	101	101	2869	2365	1647	9.92e-7	9.93e-7	9.97e-7	7.37e-7	8.53e-7	4.13e-7	76.8	50.0	30.6
bqp100-5	101	101	4215	3418	3026	9.99e-7	9.98e-7	9.99e-7	2.26e-7	4.51e-7	3.82e-7	133.5	73.4	56.6
bqp100-6	101	101	2957	2606	2053	9.98e-7	9.96e-7	9.99e-7	1.83e-6	1.82e-6	1.25e-6	82.0	57.0	39.1
bqp100-7	101	101	3250	2597	2044	9.87e-7	9.95e-7	9.95e-7	7.19e-7	1.05e-6	4.68e-7	95.3	56.5	39.8
bqp100-8	101	101	5029	4562	2957	9.99e-7	9.99e-7	9.99e-7	3.87e-7	2.21e-7	3.27e-7	139.6	100.9	56.1
bqp100-9	101	101	6221	5160	5409	9.99e-7	9.99e-7	9.99e-7	7.29e-7	7.11e-7	5.60e-7	179.6	118.2	102.7
bqp100-10	101	101	4627	4043	3763	1.00e-6	9.99e-7	9.99e-7	9.36e-7	8.47e-7	6.24e-7	129.2	93.5	72.3
bqp250-1	251	251	8252	6536	6309	9.99e-7	1.00e-6	1.00e-6	1.82e-6	1.44e-6	9.31e-7	503.5	428.4	297.2
bqp250-2	251	251	7448	5362	4742	9.96e-7	9.97e-7	9.98e-7	1.17e-6	9.38e-7	3.52e-7	450.2	341.7	221.8
bqp250-3	251	251	7954	5828	5649	9.94e-7	1.00e-6	9.99e-7	7.58e-7	3.89e-7	1.81e-7	477.4	382.5	265.6
bqp250-4	251	251	6593	5354	4872	1.00e-6	9.97e-7	9.98e-7	4.65e-7	2.18e-7	2.31e-7	387.6	349.0	231.4
bqp250-5	251	251	10301	7513	8915	1.00e-6	1.00e-6	1.00e-6	9.75e-7	9.20e-7	5.48e-7	597.7	517.1	275.5
bqp250-6	251	251	5781	4826	4466	9.98e-7	9.99e-7	9.99e-7	2.21e-7	3.74e-7	4.37e-7	346.7	310.1	205.1
bqp250-7	251	251	8950	6942	7758	1.00e-6	9.99e-7	1.00e-6	6.74e-7	5.83e-7	3.17e-7	531.9	465.0	367.7
bqp250-8	251	251	8135	6616	6841	9.99e-7	9.99e-7	9.99e-7	5.50e-7	5.14e-7	2.96e-7	490.0	443.3	325.9
bqp250-9	251	251	8058	5724	5346	9.94e-7	9.94e-7	9.99e-7	6.67e-7	1.22e-6	6.38e-7	482.6	371.7	256.2
bqp250-10	251	251	8342	6584	6747	1.00e-6	1.00e-6	9.99e-7	7.37e-7	7.54e-7	3.80e-7	502.0	444.3	321.7
be100.1	101	101	3164	2688	1906	9.99e-7	9.73e-7	9.99e-7	5.54e-7	7.27e-7	1.97e-7	40.6	49.5	9.3
be100.2	101	101	2985	2766	1788	9.96e-7	9.97e-7	9.97e-7	2.24e-7	8.21e-8	3.97e-9	63.2	59.2	11.7
be100.3	101	101	4889	4245	3341	1.00e-6	1.00e-6	9.97e-7	5.99e-8	2.85e-8	7.18e-8	121.3	95.5	38.4
be100.4	101	101	3270	3022	2294	9.98e-7	1.00e-6	9.98e-7	5.83e-9	1.69e-7	2.97e-7	85.1	66.7	32.9
be100.5	101	101	2607	2430	1530	9.94e-7	9.99e-7	9.94e-7	3.77e-7	3.17e-7	3.52e-7	67.5	52.4	25.1
be100.6	101	101	2812	2434	1555	9.94e-7	9.96e-7	9.99e-7	1.93e-6	8.31e-7	5.05e-7	72.7	51.9	25.5
be100.7	101	101	3144	2459	1675	9.99e-7	9.89e-7	9.94e-7	4.74e-7	2.39e-7	2.25e-7	82.8	51.6	28.9
be100.8	101	101	2808	2375	1556	9.96e-7	9.97e-7	9.96e-7	8.76e-8	4.77e-7	3.67e-7	71.3	49.8	28.4
be100.9	101	101	3131	2862	2252	9.98e-7	9.98e-7	9.98e-7	2.32e-7	1.05e-7	3.01e-7	82.5	63.5	43.0
be100.10	101	101	2681	2318	1520	9.99e-7	9.95e-7	9.97e-7	1.14e-6	3.55e-7	3.91e-7	70.9	47.1	27.5
be120.8.1	121	121	3382	2556	2051	9.79e-7	9.99e-7	9.97e-7	1.13e-6	7.05e-7	6.92e-7	103.1	64.6	45.9
be120.8.2	121	121	3347	2847	2032	9.99e-7	9.99e-7	9.98e-7	4.57e-7	6.20e-7	2.79e-7	98.2	73.7	46.6
be120.8.3	121	121	4987	4323	3882	1.00e-6	9.99e-7	1.00e-6	4.05e-7	1.68e-7	1.34e-7	140.5	114.5	84.5
be120.8.4	121	121	3492	3150	2078	1.00e-6	1.00e-6	9.96e-7	9.67e-8	3.16e-8	7.02e-8	101.4	79.3	44.1
be120.8.5	121	121	3800	3015	2433	9.85e-7	1.00e-6	9.97e-7	7.70e-7	1.08e-6	5.61e-7	119.0	76.9	50.7
be120.8.6	121	121	3114	2476	1599	9.99e-7	9.90e-7	9.99e-7	2.62e-7	4.73e-7	3.95e-7	92.6	62.2	36.7
be120.8.7	121	121	3894	3289	2742	9.97e-7	9.96e-7	9.98e-7	9.35e-7	1.21e-6	5.97e-7	117.1	82.3	59.6
be120.8.8	121	121	3726	3036	2546	9.99e-7	9.75e-7	9.99e-7	9.17e-7	6.45e-7	4.11e-7	111.2	77.5	51.2
be120.8.9	121	121	3274	2720	1916	9.97e-7	9.97e-7	9.96e-7	8.64e-7	6.38e-7	5.61e-7	106.6	73.2	40.1
be120.8.10	121	121	3869	3182	2454	9.98e-7	9.99e-7	9.98e-7	3.84e-7	2.47e-7	4.15e-7	115.3	84.9	50.7
be250.1	251	251	7779	6260	5359	1.00e-6	1.00e-6	9.99e-7	7.69e-7	6.15e-7	1.22e-6	413.4	347.8	176.3
be250.2	251	251	8175	6783	5330	1.00e-6	1.00e-6	9.99e-7	4.25e-7	4.25e-7	8.33e-7	528.2	421.9	244.2
be250.3	251	251	7202	5812	4887	9.99e-7	1.00e-6	1.00e-6	1.33e-6	9.03e-7	1.65e-6	457.2	366.1	229.9
be250.4	251	251	9835	7909	6769	1.00e-6	1.00e-6	1.00e-6	2.79e-7	3.60e-7	9.32e-7	630.0	488.8	313.5
be250.5	251	251	6622	5179	4339	9.92e-7	9.99e-7	9.99e-7	5.50e-7	4.61e-7	1.34e-6	421.0	317.6	201.2
be250.6	251	251	6761	5100	4164	1.00e-6	9.99e-7	1.00e-6	1.25e-6	8.03e-7	5.01e-7	426.5	316.4	208.9
be250.7	251	251	7588	6190	4776	1.00e-6	1.00e-6	1.00e-6	6.07e-9	2.64e-7	9.04e-7	476.1	387.7	225.2
be250.8	251	251	9334	7917	6776	9.99e-7	1.00e-6	1.00e-6	4.44e-7	3.79e-7	8.33e-7	573.5	503.4	314.4
be250.9	251	251	7572	6458	5451	9.99e-7	9.99e-7	1.00e-6	1.34e-6	9.96e-7	2.47e-6	458.1	404.8	253.8
be250.10	251	251	8245	7116	6484	9.99e-7	1.00e-6	1.00e-6	6.18e-7	8.72e-7	2.11e-6	499.1	445.9	302.8

Table 2 The performance of our projection method, ADMM4d and ADM-G on the CQSDP problems with $\varphi(X) = \frac{BX+XB}{2}$ ($seed = 1$). In the table, "PM" and "4d" stands for our projection method and ADMM4d, respectively.

			iteration			δ			δ_g			time (second)		
problem	m_E	n_S	ADM-G	4d	PM	ADM-G	4d	PM	ADM-G	4d	PM	ADM-G	4d	PM
theta4	1949	200	574	494	342	9.90e-7	9.93e-7	9.49e-7	8.88e-7	1.51e-6	1.15e-6	6.8	6.0	3.1
theta42	5986	200	397	335	515	9.90e-7	9.85e-7	9.83e-7	6.68e-7	6.87e-7	3.96e-7	7.2	6.1	4.6
theta6	4375	300	448	389	444	9.47e-7	9.75e-7	9.88e-7	1.41e-6	1.78e-6	2.34e-6	26.5	23.0	9.2
theta62	13390	300	437	415	511	9.79e-7	9.87e-7	9.77e-7	8.63e-7	9.64e-7	6.71e-7	30.7	31.8	19.4
theta8	7905	400	518	478	461	9.92e-7	9.87e-7	8.95e-7	3.82e-7	4.60e-7	8.48e-7	70.7	76.4	39.5
theta82	23872	400	463	392	460	9.88e-7	9.83e-7	9.83e-7	1.26e-6	1.30e-6	1.01e-6	70.9	70.5	47.2
theta10	12470	500	613	553	477	9.95e-7	9.88e-7	8.38e-7	3.75e-7	4.87e-7	7.10e-7	146.4	151.6	78.3
theta102	37467	500	487	459	485	8.88e-7	9.87e-7	9.92e-7	1.01e-6	1.43e-6	1.08e-6	122.3	138.3	88.4
theta103	62516	500	479	463	553	9.80e-7	9.88e-7	9.93e-7	1.85e-6	2.05e-6	1.22e-6	130.1	134.4	102.4
theta104	87254	500	510	493	585	9.84e-7	9.87e-7	9.95e-7	2.50e-6	2.56e-6	2.08e-6	135.6	144.9	103.5
MANN-a27	703	378	1735	1308	700	9.95e-7	9.89e-7	9.85e-7	1.41e-7	6.60e-7	4.91e-7	199.0	152.8	17.2
san200-0.7-1	5971	200	2363	1939	2639	9.52e-7	9.91e-7	9.98e-7	2.45e-6	3.96e-6	2.46e-6	116.7	86.7	41.5
sanr200-0.7	6033	200	376	321	513	9.90e-7	9.85e-7	9.89e-7	7.79e-7	8.28e-7	5.42e-7	17.9	15.0	10.7
c-fat200-1	18367	200	989	801	644	9.98e-7	9.95e-7	9.91e-7	1.00e-6	1.12e-6	8.35e-7	50.6	36.8	14.6
brock200-1	5067	200	390	327	495	9.82e-7	9.74e-7	9.80e-7	5.05e-7	6.17e-7	4.03e-7	18.7	15.6	12.9
brock200-4	6812	200	392	373	471	9.81e-7	9.81e-7	9.94e-7	1.10e-6	1.16e-6	8.25e-7	19.3	18.0	12.1
brock400-1	20078	400	468	393	460	9.89e-7	9.81e-7	9.93e-7	1.07e-6	1.15e-6	1.00e-6	75.5	70.6	38.4
keller4	5101	171	1338	994	873	9.91e-7	9.98e-7	9.96e-7	2.14e-7	3.14e-7	2.35e-7	59.0	42.1	20.9
p-hat300-1	33918	300	1207	980	1023	1.00e-6	9.98e-7	9.98e-7	1.54e-6	1.48e-6	1.18e-6	110.1	99.9	56.7
1dc.128	1472	128	1255	1010	847	9.98e-7	9.99e-7	9.95e-7	1.16e-6	1.28e-6	8.88e-7	36.9	28.2	15.8
1et.128	673	128	7141	5834	4576	1.00e-6	1.00e-6	1.00e-6	7.59e-7	8.46e-7	1.15e-6	216.8	176.7	96.6
1tc.128	513	128	5579	4353	2822	9.99e-7	1.00e-6	1.00e-6	1.05e-6	1.09e-6	1.67e-6	165.4	121.0	62.0
1zc.128	1128	128	586	413	717	9.99e-7	9.93e-7	9.96e-7	4.47e-7	5.85e-8	8.36e-7	16.8	11.2	14.2
1dc.256	3840	256	2286	1870	1510	1.00e-6	9.99e-7	9.99e-7	1.10e-6	1.16e-6	1.51e-6	148.4	135.4	72.7
1et.256	1665	256	6425	5133	4250	1.00e-6	1.00e-6	1.00e-6	4.32e-7	6.69e-7	1.77e-6	423.0	387.3	214.9
1tc.256	1313	256	7636	6094	5283	1.00e-6	1.00e-6	9.78e-7	4.05e-7	2.20e-7	8.98e-7	512.4	458.4	272.4
1zc.256	2817	256	2627	2008	1553	9.99e-7	9.99e-7	9.99e-7	1.19e-7	1.96e-7	4.05e-7	175.9	151.7	80.7
gka1d	101	101	4262	3663	3849	1.00e-6	9.98e-7	1.00e-6	1.04e-7	1.48e-7	8.09e-8	48.2	27.3	23.1
gka1e	201	201	7881	7158	7228	1.00e-6	1.00e-6	9.99e-7	5.15e-7	7.00e-7	5.54e-7	329.7	252.5	210.5
gka1f	501	501	12629	10311	11511	9.99e-7	1.00e-6	1.00e-6	1.23e-6	1.32e-6	6.96e-7	2925.4	2723.4	1897.5
gka2d	101	101	2834	2261	1939	9.95e-7	1.00e-6	9.99e-7	1.25e-6	1.13e-6	8.31e-7	74.7	45.1	32.6
gka2e	201	201	7167	5652	6123	9.99e-7	1.00e-6	9.99e-7	8.39e-7	5.96e-7	3.75e-7	322.4	370.7	196.4
gka2f	501	501	12074	9720	10596	9.95e-7	1.00e-6	1.00e-6	1.14e-6	7.36e-7	6.09e-7	2754.9	2486.0	1706.0
gka3d	101	101	4292	3456	2880	9.98e-7	1.00e-6	1.00e-6	7.66e-7	5.58e-7	5.95e-7	111.1	80.4	50.7
gka3e	201	201	5513	4695	4388	9.99e-7	9.97e-7	9.99e-7	3.48e-7	1.22e-6	5.77e-7	246.5	201.1	143.7
gka3f	501	501	12001	9996	9425	9.96e-7	9.99e-7	9.99e-7	1.66e-6	4.48e-7	6.13e-7	2793.1	2616.8	1512.5
gka4d	101	101	3463	3037	2264	9.95e-7	1.00e-6	9.99e-7	8.38e-7	7.32e-7	4.36e-7	97.5	61.9	41.7
gka4e	201	201	6908	5457	5692	9.99e-7	9.92e-7	9.99e-7	9.90e-7	8.09e-7	6.68e-7	313.2	258.6	189.1
gka4f	501	501	12679	9744	9366	1.00e-6	9.98e-7	9.99e-7	4.77e-7	1.45e-6	3.52e-7	2993.1	2669.6	1480.2
gka5d	101	101	4637	3854	3104	9.99e-7	9.99e-7	9.99e-7	6.91e-7	6.81e-7	1.96e-7	119.6	86.3	54.4
gka5e	201	201	5310	4385	4328	9.99e-7	1.00e-6	1.00e-6	6.01e-7	5.17e-7	3.45e-7	241.4	189.3	143.5
gka5f	501	501	9823	9659	9751	9.99e-7	9.99e-7	1.00e-6	5.99e-6	1.25e-6	3.60e-7	2856.3	2557.7	1569.5
gka6b	71	71	1182	983	1834	9.98e-7	9.96e-7	9.99e-7	5.99e-6	7.49e-6	1.39e-5	6.5	4.8	7.1
gka6d	101	101	3406	2619	2426	1.00e-6	1.00e-6	9.98e-7	7.98e-7	8.84e-7	6.07e-7	54.6	28.0	25.2
gka7b	81	81	425	389	522	9.83e-7	9.89e-7	9.93e-7	5.98e-6	5.62e-6	1.36e-5	7.8	5.7	6.0
gka7d	101	101	2829	2339	1678	1.00e-6	9.99e-7	9.99e-7	3.15e-7	5.08e-7	3.05e-7	63.9	41.3	25.0
gka8b	91	91	1236	1192	1762	9.98e-7	9.95e-7	1.00e-6	2.05e-5	2.41e-6	4.82e-5	27.2	23.7	26.9
gka8d	101	101	3082	2720	1828	9.98e-7	1.00e-6	9.98e-7	5.82e-8	1.47e-7	5.81e-9	76.8	57.0	32.2
gka9b	101	101	2894	2148	3476	9.99e-7	1.00e-6	9.99e-7	1.73e-5	1.61e-5	3.23e-5	68.9	50.1	58.1
gka9d	101	101	3280	2720	1737	9.98e-7	9.97e-7	9.97e-7	3.38e-7	3.47e-7	1.12e-7	82.8	62.7	31.8
gka10b	126	126	2571	2226	3279	1.00e-6	1.00e-6	1.00e-6	2.96e-5	2.65e-5	5.09e-5	67.8	61.6	64.9
gka10d	101	101	4059	3555	2974	9.99e-7	1.00e-6	1.00e-6	5.14e-7	4.28e-7	4.08e-7	104.5	81.2	54.5

Table 3 The performance of our projection method, ADMM4d and ADM-G on the CQSDP problems with $\varphi(X) = X$ ($seed = 1$). In the table, "PM" and "4d" stands for our projection method and ADMM4d, respectively.

			iteration			δ			δ_g			time (second)		
problem	m_E	n_S	4d	PM		4d	PM		4d	PM		4d	PM	
bqp50-1	51	51	2068	1668	2087	1.00e-6	9.99e-7	1.00e-6	2.58e-7	5.73e-7	1.35e-7	10.5	13.3	7.8
bqp50-2	51	51	2184	1859	1941	1.00e-6	9.99e-7	9.98e-7	4.74e-7	4.87e-7	1.29e-6	19.4	24.7	12.5
bqp50-3	51	51	1672	1282	1188	9.98e-7	9.96e-7	9.97e-7	8.41e-7	3.52e-7	1.98e-7	20.0	18.3	11.5
bqp50-4	51	51	1186	795	802	9.38e-7	9.53e-7	9.55e-7	3.39e-7	1.99e-7	5.32e-7	16.6	11.1	7.9
bqp50-5	51	51	1530	1179	1285	9.98e-7	9.96e-7	9.92e-7	4.31e-7	3.03e-7	2.94e-7	22.7	17.7	14.9
bqp50-6	51	51	2735	2276	2882	9.99e-7	9.98e-7	9.99e-7	1.49e-8	7.00e-8	6.78e-8	46.6	34.5	36.3
bqp50-7	51	51	1236	895	933	9.93e-7	9.75e-7	9.98e-7	7.08e-7	3.06e-7	1.26e-7	21.9	13.4	12.6
bqp50-8	51	51	1264	953	875	9.94e-7	9.87e-7	9.95e-7	3.72e-7	3.21e-7	6.34e-7	22.7	15.1	11.6
bqp50-9	51	51	1196	871	765	9.79e-7	9.64e-7	1.00e-6	7.66e-8	2.45e-7	4.25e-7	23.5	12.7	9.8
bqp50-10	51	51	1046	808	857	9.76e-7	9.86e-7	9.99e-7	8.48e-8	3.42e-7	3.02e-8	20.2	12.9	11.5
bqp100-1	101	101	1484	1084	974	9.86e-7	9.94e-7	9.98e-7	1.54e-6	7.14e-7	1.91e-6	36.2	22.5	17.5
bqp100-2	101	101	1976	1728	1474	9.98e-7	9.97e-7	9.97e-7	7.88e-8	2.60e-8	2.12e-7	50.8	35.3	26.7
bqp100-3	101	101	1822	1297	1176	9.95e-7	9.98e-7	9.93e-7	1.50e-6	2.08e-6	1.70e-6	45.3	25.2	22.1
bqp100-4	101	101	2388	1918	1778	9.96e-7	1.00e-6	9.98e-7	1.78e-7	1.81e-7	6.82e-7	61.0	37.2	32.2
bqp100-5	101	101	2434	1934	1634	9.96e-7	9.97e-7	9.96e-7	3.65e-7	6.34e-7	1.48e-6	62.8	38.8	30.3
bqp100-6	101	101	2336	1912	1584	9.98e-7	9.99e-7	9.98e-7	6.32e-7	2.67e-7	4.72e-7	58.4	39.9	29.5
bqp100-7	101	101	1703	1167	1036	9.98e-7	9.96e-7	9.96e-7	9.20e-7	8.12e-8	9.06e-7	41.6	24.1	18.9
bqp100-8	101	101	2405	1853	2099	9.96e-7	9.99e-7	1.00e-6	6.46e-7	4.71e-7	2.05e-9	60.3	38.6	40.6
bqp100-9	101	101	3052	2748	2653	9.93e-7	1.00e-6	9.93e-7	6.61e-7	2.68e-7	9.24e-7	74.7	55.4	50.5
bqp100-10	101	101	2396	1984	1664	9.99e-7	9.98e-7	9.98e-7	6.36e-7	6.56e-7	2.93e-7	58.5	40.6	30.7
bqp250-1	251	251	3349	2598	2148	1.00e-6	9.98e-7	9.99e-7	1.52e-6	7.23e-7	2.00e-6	173.7	121.8	98.2
bqp250-2	251	251	3576	2751	2335	9.93e-7	1.00e-6	9.99e-7	1.05e-6	1.18e-6	1.33e-6	178.5	126.7	111.2
bqp250-3	251	251	3423	2626	2076	9.99e-7	9.97e-7	9.97e-7	5.07e-7	1.08e-6	1.81e-6	171.5	122.0	96.1
bqp250-4	251	251	3187	2406	1864	9.96e-7	1.00e-6	9.96e-7	1.37e-6	1.57e-6	1.36e-6	161.6	114.8	83.3
bqp250-5	251	251	4186	3137	3136	9.99e-7	9.99e-7	9.99e-7	9.44e-7	9.50e-7	9.17e-7	210.1	145.1	143.9
bqp250-6	251	251	3044	2396	1914	9.97e-7	9.96e-7	9.96e-7	1.02e-6	1.62e-6	2.72e-6	155.4	111.6	87.5
bqp250-7	251	251	3446	2637	1960	9.97e-7	9.94e-7	9.96e-7	2.42e-7	8.63e-7	2.17e-6	177.0	126.1	91.5
bqp250-8	251	251	3001	2256	1847	9.98e-7	9.96e-7	9.97e-7	4.15e-7	8.46e-7	1.00e-6	149.7	107.4	84.7
bqp250-9	251	251	3398	2547	1984	1.00e-6	1.00e-6	9.95e-7	4.17e-7	6.05e-7	2.32e-6	174.4	123.7	87.6
bqp250-10	251	251	3108	2433	1883	9.96e-7	9.96e-7	9.99e-7	6.52e-7	1.01e-6	2.14e-6	161.3	117.5	83.5
be100.1	101	101	2489	1858	2018	9.98e-7	9.96e-7	1.00e-6	4.13e-7	8.70e-7	6.45e-7	16.9	19.1	12.6
be100.2	101	101	3436	2598	2071	9.98e-7	1.00e-6	9.97e-7	7.04e-7	8.52e-7	9.03e-7	47.2	49.2	30.1
be100.3	101	101	2205	1716	1317	9.99e-7	9.98e-7	9.93e-7	7.06e-7	8.55e-7	6.23e-7	40.2	35.4	22.1
be100.4	101	101	2075	1566	1484	9.99e-7	9.97e-7	9.99e-7	8.19e-8	1.75e-7	2.94e-7	41.7	30.2	25.8
be100.5	101	101	2250	1728	1601	1.00e-6	9.98e-7	9.99e-7	7.35e-7	5.10e-7	9.83e-8	49.2	36.1	28.2
be100.6	101	101	1937	1509	1699	1.00e-6	9.96e-7	1.00e-6	1.28e-7	2.24e-7	7.07e-7	42.5	31.0	31.6
be100.7	101	101	2059	1505	1335	9.99e-7	9.95e-7	9.98e-7	8.04e-7	1.56e-6	1.25e-6	45.4	29.4	24.6
be100.8	101	101	2015	1443	1207	9.92e-7	9.97e-7	1.00e-6	3.25e-7	7.88e-8	1.54e-7	44.6	30.3	21.7
be100.9	101	101	1833	1407	1484	9.95e-7	9.95e-7	9.95e-7	6.53e-7	8.63e-7	1.35e-6	40.9	29.0	28.0
be100.10	101	101	1676	1274	1254	1.00e-6	9.98e-7	9.91e-7	7.95e-7	7.29e-7	6.95e-8	38.6	25.1	22.6
be120.8.1	121	121	2010	1479	1366	9.95e-7	9.90e-7	9.99e-7	5.66e-7	8.62e-7	1.65e-8	16.4	33.7	24.2
be120.8.2	121	121	2437	1723	1817	9.99e-7	9.97e-7	9.97e-7	2.98e-7	5.87e-7	5.65e-7	41.5	38.9	33.0
be120.8.3	121	121	2376	1859	1567	9.96e-7	9.98e-7	9.99e-7	2.68e-7	5.13e-8	2.10e-8	49.6	41.5	29.1
be120.8.4	121	121	2174	1617	1477	1.00e-6	9.94e-7	9.96e-7	4.52e-7	8.92e-7	6.53e-7	51.1	36.4	28.1
be120.8.5	121	121	2247	1670	1678	9.95e-7	9.93e-7	9.99e-7	6.41e-7	5.99e-7	3.38e-7	54.6	38.1	33.4
be120.8.6	121	121	1965	1524	1368	9.91e-7	9.97e-7	9.95e-7	9.60e-7	1.32e-6	1.96e-8	47.3	33.7	27.4
be120.8.7	121	121	2394	1683	1480	9.75e-7	9.93e-7	9.81e-7	4.43e-7	5.57e-7	1.58e-7	62.6	37.2	29.9
be120.8.8	121	121	2175	1557	1366	9.97e-7	9.92e-7	9.99e-7	2.29e-7	1.29e-7	1.97e-7	56.3	35.0	26.6
be120.8.9	121	121	1972	1496	1296	9.92e-7	9.92e-7	1.00e-6	7.83e-7	9.07e-7	7.65e-7	49.7	35.1	27.1
be120.8.10	121	121	2301	1738	1532	1.00e-6	9.95e-7	9.97e-7	8.41e-7	8.30e-7	1.59e-6	61.7	38.1	30.6
be250.1	251	251	2632	1955	1754	9.97e-7	1.00e-6	9.98e-7	1.09e-6	1.03e-6	1.93e-6	131.5	43.7	43.1
be250.2	251	251	2872	2235	2472	9.99e-7	9.99e-7	9.94e-7	5.06e-7	4.80e-7	3.31e-7	144.7	87.1	99.2
be250.3	251	251	2283	1603	1510	9.78e-7	9.91e-7	9.99e-7	4.03e-7	7.85e-7	2.90e-6	111.2	70.9	60.9
be250.4	251	251	2318	1655	1593	9.90e-7	9.97e-7	9.94e-7	3.87e-7	3.26e-8	2.58e-6	116.2	76.9	67.6
be250.5	251	251	2117	1550	1555	9.96e-7	1.00e-6	9.94e-7	1.32e-7	4.40e-7	2.52e-6	101.5	72.8	64.3
be250.6	251	251	2243	1604	1525	1.00e-6	9.98e-7	9.92e-7	4.02e-8	4.19e-7	2.20e-6	109.4	77.7	64.8
be250.7	251	251	2224	1613	1568	1.00e-6	9.96e-7	9.95e-7	1.42e-7	5.85e-8	2.79e-6	108.9	77.5	68.2
be250.8	251	251	2335	1594	1512	9.96e-7	9.84e-7	1.00e-6	4.21e-7	5.32e-7	2.28e-6	113.4	73.0	67.8
be250.9	251	251	2698	2130	1931	1.00e-6	1.00e-6	1.00e-6	8.26e-7	1.47e-7	5.47e-7	131.8	101.0	84.9
be250.10	251	251	2348	1709	1573	9.99e-7	1.00e-6	9.95e-7	1.20e-6	2.09e-7	2.62e-6	113.5	79.2	70.6

Table 4 The performance of our projection method, ADMM4d and ADM-G on the CQSDP problems with $\varphi(X) = X$ ($seed = 1$). In the table, "PM" and "4d" stands for our projection method and ADMM4d, respectively.

			iteration			δ			δ_g			time (second)		
problem	m_E	n_S	4d	PM		4d	PM		4d	PM		4d	PM	
theta4	1949	200	596	523	402	9.99e-7	9.90e-7	9.91e-7	9.72e-7	1.48e-6	1.50e-6	6.2	4.7	3.6
theta42	5986	200	510	459	461	9.97e-7	9.97e-7	9.91e-7	3.41e-7	9.38e-7	5.34e-7	5.6	4.6	4.1
theta6	4375	300	567	516	452	9.97e-7	9.93e-7	8.82e-7	1.25e-6	1.32e-6	8.74e-7	15.6	15.0	8.4
theta62	13390	300	440	414	513	9.81e-7	9.88e-7	9.91e-7	9.38e-7	1.28e-6	8.33e-7	18.7	16.9	15.0
theta8	7905	400	688	598	462	9.83e-7	1.00e-6	7.75e-7	4.28e-7	1.00e-6	4.22e-7	51.4	47.6	28.9
theta82	23872	400	485	393	490	9.84e-7	9.76e-7	9.99e-7	7.84e-7	1.48e-6	1.02e-6	43.4	37.6	38.5
theta10	12470	500	607	632	530	8.06e-7	9.98e-7	7.48e-7	6.05e-7	1.08e-6	5.57e-7	86.8	95.7	69.6
theta102	37467	500	573	469	518	9.95e-7	9.83e-7	9.95e-7	1.53e-6	1.61e-6	9.54e-7	90.8	74.0	75.8
theta103	62516	500	482	463	679	9.31e-7	9.91e-7	9.85e-7	1.79e-6	2.16e-6	2.56e-6	82.2	74.1	102.3
theta104	87254	500	511	495	896	9.99e-7	9.87e-7	9.89e-7	2.57e-6	2.65e-6	3.75e-6	92.7	79.0	127.7
MANN-a27	703	378	1284	697	2564	9.13e-7	9.74e-7	9.55e-7	2.41e-6	2.11e-6	1.58e-6	89.4	47.3	166.2
san200-0.7-1	5971	200	2551	2029	3564	9.97e-7	8.57e-7	9.94e-7	1.97e-6	2.11e-6	6.61e-6	102.8	72.5	122.8
sanr200-0.7	6033	200	478	440	509	9.82e-7	9.76e-7	9.86e-7	9.44e-7	9.88e-7	6.19e-7	19.2	15.2	16.7
c-fat200-1	18367	200	1015	987	970	9.91e-7	9.91e-7	9.86e-7	2.91e-7	2.20e-7	1.63e-7	39.1	29.8	26.0
brock200-1	5067	200	489	413	474	9.84e-7	9.90e-7	9.95e-7	7.96e-7	8.33e-7	3.89e-7	21.5	14.9	15.9
brock200-4	6812	200	409	382	560	9.88e-7	9.94e-7	9.78e-7	1.09e-6	1.21e-6	1.08e-6	18.2	12.4	17.8
brock400-1	20078	400	570	473	387	9.87e-7	9.87e-7	9.92e-7	6.55e-7	1.29e-6	6.72e-7	62.5	52.1	39.2
keller4	5101	171	819	724	794	9.94e-7	9.96e-7	9.93e-7	4.66e-7	5.57e-7	5.22e-7	25.3	19.4	19.5
p-hat300-1	33918	300	1184	946	1237	9.99e-7	1.00e-6	9.86e-7	1.45e-6	1.44e-6	1.66e-6	80.9	58.8	72.7
1dc.128	1472	128	1129	855	786	9.88e-7	9.92e-7	9.93e-7	3.41e-7	2.98e-6	8.59e-7	29.7	18.4	15.7
1et.128	673	128	1279	1239	997	9.90e-7	9.96e-7	9.99e-7	2.72E-10	1.26e-7	2.55e-7	31.9	26.7	19.0
1tc.128	513	128	1041	928	1161	9.79e-7	9.34e-7	9.98e-7	2.43e-6	4.36e-7	3.92e-6	27.1	20.3	24.5
1zc.128	1128	128	272	251	321	9.65e-7	8.51e-7	7.37e-7	2.46e-6	1.54e-6	4.47e-8	6.9	5.4	5.8
1dc.256	3840	256	2583	2331	1722	1.00e-6	9.98e-7	9.96e-7	3.95e-6	3.92e-6	4.26e-6	130.2	102.9	70.1
1et.256	1665	256	2781	3755	1614	9.99e-7	1.00e-6	9.99e-7	5.85e-7	5.02e-7	7.86e-7	133.9	176.1	61.6
1tc.256	1313	256	5263	4018	2997	1.00e-6	1.00e-6	9.99e-7	1.52e-6	1.68e-6	1.36e-6	254.1	171.1	119.7
1zc.256	2817	256	295	250	307	9.22e-7	9.01e-7	9.49e-7	1.73e-6	9.48e-7	9.18e-7	13.4	9.1	10.6
gka1d	101	101	1356	1074	953	9.90e-7	9.96e-7	9.98e-7	2.02e-7	3.85e-7	8.48e-7	30.9	6.0	4.7
gka1e	201	201	2565	1946	1881	9.97e-7	1.00e-6	1.00e-6	1.11e-6	7.63e-7	1.06e-6	91.8	35.8	27.8
gka1f	501	501	3222	2454	2177	1.00e-6	9.95e-7	9.99e-7	2.04e-7	1.43e-6	3.11e-6	479.5	299.6	290.6
gka2d	101	101	1339	953	855	9.90e-7	9.77e-7	9.96e-7	4.62e-7	5.26e-7	1.38e-6	31.6	15.7	15.3
gka2e	201	201	2328	1743	1434	9.98e-7	9.94e-7	9.94e-7	8.47e-7	7.39e-7	6.16e-7	91.6	52.0	45.8
gka2f	501	501	4937	3907	3063	9.95e-7	1.00e-6	9.99e-7	9.45e-7	2.03e-7	6.75e-7	766.3	532.5	445.3
gka3d	101	101	2339	1944	1877	9.77e-7	9.93e-7	9.65e-7	3.14e-7	2.90e-7	3.86e-7	55.1	33.6	33.9
gka3e	201	201	3458	2785	2477	9.98e-7	9.97e-7	9.99e-7	7.81e-7	9.36e-7	2.80e-7	133.2	88.5	79.0
gka3f	501	501	6603	5240	4600	9.99e-7	9.99e-7	9.99e-7	1.08e-6	9.29e-7	7.40e-7	1009.6	741.2	678.9
gka4d	101	101	1747	1200	1184	9.94e-7	9.95e-7	9.99e-7	7.67e-7	4.85e-7	9.53e-7	39.3	21.1	21.1
gka4e	201	201	2993	2328	1729	9.87e-7	9.89e-7	9.95e-7	8.91e-8	8.71e-7	1.56e-6	110.0	70.1	53.6
gka4f	501	501	6225	5246	4244	1.00e-6	9.99e-7	9.98e-7	8.99e-7	1.57e-6	1.85e-6	962.8	752.6	618.2
gka5d	101	101	1775	1159	1117	9.74e-7	9.99e-7	9.91e-7	7.41e-7	4.86e-9	1.21e-6	42.9	20.1	20.4
gka5e	201	201	3624	2781	2599	9.97e-7	9.99e-7	9.98e-7	7.16e-7	6.03e-7	7.11e-7	132.8	86.1	81.2
gka5f	501	501	6598	5522	4783	9.96e-7	1.00e-6	9.98e-7	1.02e-6	1.53e-6	1.62e-6	982.1	794.6	677.6
gka6b	71	71	784	789	968	9.95e-7	9.90e-7	9.96e-7	1.99e-5	2.29e-5	4.73e-5	15.3	3.5	14.5
gka6d	101	101	1889	1375	1156	9.97e-7	9.92e-7	9.97e-7	3.05e-7	6.49e-7	1.97e-7	45.0	10.5	21.0
gka7b	81	81	799	671	955	9.98e-7	9.94e-7	9.94e-7	1.23e-5	1.33e-5	2.82e-5	16.5	7.7	14.7
gka7d	101	101	1711	1273	1196	9.98e-7	9.95e-7	9.97e-7	5.38e-7	8.34e-7	5.01e-7	39.5	19.4	20.7
gka8b	91	91	922	795	1335	9.98e-7	9.95e-7	9.94e-7	2.72e-5	2.72e-5	5.46e-5	19.9	12.9	20.9
gka8d	101	101	1873	1367	1284	9.94e-7	9.58e-7	9.97e-7	7.97e-7	1.33e-6	5.78e-7	44.1	23.4	23.3
gka9b	101	101	851	751	1313	9.97e-7	9.95e-7	9.96e-7	1.54e-5	1.64e-5	2.63e-5	19.1	13.3	24.4
gka9d	101	101	1713	1293	1215	9.87e-7	9.72e-7	9.95e-7	5.08e-7	1.85e-7	1.06e-7	41.0	24.6	21.4
gka10b	126	126	3619	2319	4351	1.00e-6	1.00e-6	1.00e-6	4.62e-5	4.98e-5	1.12e-4	93.9	47.2	86.7
gka10d	101	101	2257	1697	1519	1.00e-6	9.97e-7	1.00e-6	7.17e-7	3.03e-7	1.01e-6	56.9	33.2	27.4