# A Newton-CG Augmented Lagrangian Method for Semidefinite Programming *

Xin-Yuan Zhao †  Defeng Sun ‡  Kim-Chuan Toh §

March 12, 2008

**Abstract.** We consider a Newton-CG augmented Lagrangian (NCGAL) method for solving semidefinite programming (SDP) problems from the perspective of approximate semismooth Newton methods. In order to analyze the rate of convergence of the method, we characterize the Lipschitz continuity of the corresponding solution mapping at the origin. For the inner problems in the NCGAL method, we show that the positive definiteness of the generalized Hessian of the objective function in these inner problems, a key property for ensuring the efficiency of using an inexact semismooth Newton-CG method to solve the inner problems, is equivalent to the constraint nondegeneracy of the corresponding dual problems. Numerical experiments on a variety of large scale SDPs with matrix dimensions up to $1,600$ and number of equality constraints up to $1,283,258$ show that the proposed method is very efficient.

**Keywords:** Semidefinite programming, Augmented Lagrangian, Semismoothness, Newton's method, Iterative solver.

## 1 Introduction

Let $\mathcal{S}^n$ be the linear space of all $n \times n$ symmetric matrices and $\mathcal{S}^n_+$ be the cone of all $n \times n$ symmetric positive semidefinite matrices. The notation $X \succeq \mathbf{0}$ means that $X$ is a symmetric

---

*A preliminary version of this paper was presented at ICCOPTA 07-MOPTA 07, Hamilton, Canada, August 13, 2007.

†Department of Mathematics, National University of Singapore, 2 Science Drive 2, Singapore 117543 (zhaoxinyuan@nus.edu.sg).

‡Department of Mathematics and Risk Management Institute, National University of Singapore, 2 Science Drive 2, Singapore 117543 (matsundf@nus.edu.sg). This author's research is supported in part by Academic Research Fund under Grant R-146-000-104-112.

§Department of Mathematics, National University of Singapore, 2 Science Drive 2, Singapore 117543 (mattohkc@nus.edu.sg); and Singapore-MIT Alliance, 4 Engineering Drive 3, Singapore 117576. This author's research is supported in part by Academic Research Grant R-146-000-076-112.

positive semidefinite matrix. This paper is devoted to studying an augmented Lagrangian method for solving the following semidefinite programming (SDP) problem

$$(D) \quad \min \Big\{ b^{\mathrm{T}} y \;\mid\; \mathcal{A}^* y - C \succeq \mathbf{0} \Big\},$$

where $C \in \mathcal{S}^n$, $b \in \Re^m$, $\mathcal{A}$ is a linear operator from $\mathcal{S}^n$ to $\Re^m$, and $\mathcal{A}^* : \Re^m \to \mathcal{S}^n$ is the adjoint of $\mathcal{A}$. The dual of $(D)$ takes the form

$$(P) \quad \max \Big\{ \langle\, C, X \rangle \;\mid\; \mathcal{A}(X) = b, \quad X \succeq \mathbf{0} \Big\}.$$

Given a penalty parameter $\sigma > 0$, the *augmented Lagrangian* function for problem $(D)$ is defined as

$$L_\sigma(y, X) = b^{\mathrm{T}} y + \frac{1}{2\sigma}(\|\Pi_{\mathcal{S}^n_+}(X - \sigma(\mathcal{A}^* y - C))\|^2 - \|X\|^2), \quad (y, X) \in \Re^m \times \mathcal{S}^n, \tag{1}$$

where for any closed convex set $\mathcal{D}$ in a finite dimensional real vector space $\mathcal{X}$ equipped with a scalar inner product $\langle \cdot, \cdot \rangle$ and its induced norm $\|\cdot\|$, $\Pi_{\mathcal{D}}(\cdot)$ is the metric projection operator over $\mathcal{D}$, i.e., for any $Y \in \mathcal{X}$, $\Pi_{\mathcal{D}}(Y)$ is the unique optimal solution to the following convex optimization problem

$$\min \Big\{ \frac{1}{2}\langle Z - Y, Z - Y \rangle \;\mid\; Z \in \mathcal{D} \Big\}.$$

Note that, since $\|\Pi_{\mathcal{D}}(\cdot)\|^2$ is continuously differentiable [42], the augmented Lagrangian function defined in (1) is continuously differentiable. In particular, for any given $X \in \mathcal{S}^n$, we have

$$\nabla_y L_\sigma(y, X) = b - \mathcal{A}\Pi_{\mathcal{S}^n_+}(X - \sigma(\mathcal{A}^* y - C)). \tag{2}$$

For given $X^0 \in \mathcal{S}^n$, $\sigma_0 > 0$, and $\rho > 1$, the augmented Lagrangian method for solving problem $(D)$ and its dual $(P)$ generates sequences $\{y^k\} \subset \Re^m$ and $\{X^k\} \subset \mathcal{S}^n$ as follows

$$\left\{ \begin{array}{l} y^{k+1} \approx \arg\min\limits_{y \in \Re^m} L_{\sigma_k}(y, X^k), \\ X^{k+1} = \Pi_{\mathcal{S}^n_+}(X^k - \sigma_k(\mathcal{A}^* y^{k+1} - C)), \qquad k = 0, 1, 2, \ldots \\ \sigma_{k+1} = \rho\sigma_k \text{ or } \sigma_{k+1} = \sigma_k, \end{array} \right. \tag{3}$$

For a general discussion on the augmented Lagrangian method for solving convex optimization problems and beyond, see [30, 31].

For small and medium sized SDP problems, it is widely accepted that interior-point methods (IPMs) with direct solvers are generally very efficient and robust. For large-scale SDP problems with $m$ large and $n$ moderate (say less than $2,000$), the limitations of IPMs with direct solvers become very severe due to the need of computing, storing, and factorizing the $m \times m$ Schur complement matrix. In order to alleviate these difficulties, Toh and Kojima [38] and Toh [39] proposed inexact IPMs using an iterative solver to compute the search direction at each iteration. The approach in [39] was demonstrated to be able to solve large sparse SDPs with $m$ up to $125,000$ in a few hours. Kočvara and Stingl [15] used a modified barrier method (a variant of the augmented Lagrangian method) combined with iterative solvers for linear SDP problems having only inequality constraints and reported computational results in the code PENNON [14] with $m$ up to $125,000$. More recently, Malick, Povh, Rendl, and Wiegele [17] applied regularization approaches to solve SDP problems.

In this paper, we study an augmented Lagrangian dual approach to solve large scale SDPs with $m$ large (say, up to a million) but $n$ moderate (say, up to $2,000$). Our approach is similar in spirit as those in [15] and [17], where the idea of augmented Lagrangian methods (or methods of multipliers in general) was heavily exploited. However, our points of view of employing the augmented Lagrangian methods are fundamentally different from them in solving both the outer and inner problems. It has long been known that the augmented Lagrangian method for convex problems is a gradient ascent method applied to the corresponding dual problems [28]. This inevitably leads to the impression that the augmented Lagrangian method for solving SDPs may converge slowly for the outer iteration sequence $\{X^k\}$. In spite of that, under mild conditions, a linear rate of convergence analysis is available (superlinear convergence is also possible when $\sigma_k$ goes to infinity, which should be avoided in numerical implementations) [31]. However, recent studies conducted by Sun, Sun, and Zhang [36] and Chan and Sun [8] revealed that under the constraint nondegenerate conditions for $(D)$ and $(P)$ (i.e., the dual nondegeneracy and primal nondegeneracy in the IPMs literature, e.g., [1]), respectively, the augmented Lagrangian method can be locally regarded as an approximate generalized Newton method applied to a semismooth equation. It is this connection that inspired us to investigate the augmented Lagrangian method for SDPs.

The objective functions $L_{\sigma_k}(\cdot, X^k)$ in the inner problems of the augmented Lagrangian method (3) are convex and continuously differentiable but not twice continuously differentiable (cf. (2)) due to the fact that $\Pi_{\mathcal{S}^n_+}(\cdot)$ is not continuously differentiable. It seems that Newton's method can not be applied to solve the inner problems. However, since $\Pi_{\mathcal{S}^n_+}(\cdot)$ is strongly semismooth [35], the superlinear (quadratic) convergence analysis of generalized Newton's method established by Kummer [16], and Qi and Sun [24] for solving semismooth equations may be used to get fast convergence for solving the inner problems. In fact, the quadratic convergence and superb numerical results of the generalized Newton's method combined with the conjugate gradient (CG) method reported in [23] for solving a related problem strongly motivated us to study the semismooth Newton-CG method (see Section 3) to solve the inner problems.

In [30, 31], Rockafellar established a general theory on the global convergence and local linear rate of convergence of the sequence generated by the augmented Lagrangian method for solving convex optimization problems including $(D)$ and $(P)$. In order to apply the general results in [30, 31], we characterize the Lipschitz continuity of the solution mapping for $(P)$ defined in [31] at the origin in terms of the second order sufficient condition, and the extended strict constraint qualification for $(P)$. In particular, under the uniqueness of Lagrange multipliers, we establish the equivalence among the Lipschitz continuity of the solution mapping at the origin, the second order sufficient condition, and the strict constraint qualification. As for the inner problems in (3), we show that the constraint nondegeneracy for the corresponding dual problems is equivalent to the positive definiteness of the generalized Hessian of the objective functions in the inner problems. This is important for the success of applying an iterative solver to the generalized Newton equations in solving these inner problems. The differential structure of the nonsmooth metric projection operator $\Pi_{\mathcal{S}^n_+}(\cdot)$ in the augmented Lagrangian function $L_\sigma$ plays a key role in achieving this result.

Besides the theoretical results we establish for the Newton-CG augmented Lagrangian (NCGAL) method proposed in this paper, we also demonstrate convincingly that with efficient implementations, the NCGAL method can solve some very large SDPs much more efficiently than the best alternative methods such as the inexact interior-point methods in [39], the modified barrier method in [15], the boundary-point method in [17], as well as the dedicated augmented Lagrangian method for solving SDPs arising from the lift-and-project procedure of Lovász and Schrijver [5].

The remaining parts of this paper are as follows. In Section 2, we give some preliminaries including a brief introduction about concepts related to the method of multipliers and the characterizations of the Lipschitz continuity of the solution mapping for problem $(P)$ at the origin. In Section 3, we introduce a semismooth Newton-CG method for solving the inner problems and analyze its global and local superlinear (quadratic) convergence. Section 4 presents the Newton-CG augmented Lagrangian dual approach and its linear rate of convergence. Section 5 is on numerical issues of the semismooth Newton-CG algorithm. We report numerical results in Sections 6 and 7 for a variety of large scale linear SDP problems and make final conclusions in Section 8.

# 2 Preliminaries

From [30, 31], we know that the augmented Lagrangian method can be expressed in terms of the method of multipliers for $(D)$. For the sake of subsequent discussions, we introduce related concepts to this.

Let $l(y, X) : \Re^m \times \mathcal{S}^n \to \Re$ be the ordinary Lagrangian function for $(D)$ in extended form:

$$l(y, X) = \begin{cases} b^{\mathrm{T}}y - \langle X, \mathcal{A}^*y - C \rangle & \text{if } y \in \Re^m \text{ and } X \in \mathcal{S}^n_+, \\ -\infty & \text{if } y \in \Re^m \text{ and } X \notin \mathcal{S}^n_+. \end{cases} \tag{4}$$

The essential objective function in $(D)$ is

$$f(y) = \sup_{X \in \mathcal{S}^n} l(y, X) = \begin{cases} b^{\mathrm{T}}y & \text{if } y \in \mathcal{F}_D, \\ +\infty & \text{otherwise,} \end{cases} \tag{5}$$

where $\mathcal{F}_D := \{y \in \Re^m \mid \mathcal{A}^*y - C \succeq \mathbf{0}\}$ is the feasible set of $(D)$, while the essential objective function in $(P)$ is

$$g(X) = \inf_{y \in \Re^m} l(y, X) = \begin{cases} \langle C, X \rangle & \text{if } X \in \mathcal{F}_P, \\ -\infty & \text{otherwise,} \end{cases} \tag{6}$$

where $\mathcal{F}_P := \{X \in \mathcal{S}^n \mid \mathcal{A}(X) = b, X \succeq \mathbf{0}\}$ is the feasible set of $(P)$.

Assume that $\mathcal{F}_D \neq \emptyset$ and $\mathcal{F}_P \neq \emptyset$. As in Rockafellar [31], we define the following three maximal monotone operators

$$\begin{cases} T_g(X) & = & \{U \in \mathcal{S}^n \mid -U \in \partial g(X)\}, & X \in \mathcal{S}^n, \\ T_f(y) & = & \{v \in \Re^m \mid v \in \partial f(y)\}, & y \in \Re^m, \\ T_l(y, X) & = & \{(v, U) \in \Re^m \times \mathcal{S}^n \mid (v, -U) \in \partial l(y, X)\}, & (y, X) \in \Re^m \times \mathcal{S}^n. \end{cases}$$

For each $v \in \Re^m$ and $U \in \mathcal{S}^n$, we consider the following parameterized problem:

$$(P(v, U)) \quad \max \Big\{ \langle C, X \rangle + \langle U, X \rangle \ \mid \ \mathcal{A}(X) + v = b, \quad X \succeq \mathbf{0} \Big\}.$$

From Rockafellar [27, Theorem 23.5] and the definition of $T_g$, we know that for each $U \in \mathcal{S}^n$,

$$T_g^{-1}(U) = \arg \max_{X \in \mathcal{S}^n} \{ g(X) + \langle U,\, X \rangle \}, \tag{7}$$

i.e.,

$$T_g^{-1}(U) = \text{ set of all optimal solutions to } (P(0, U)). \tag{8}$$

Similarly, we have that for each $v \in \Re^m$,

$$T_f^{-1}(v) = \text{ set of all optimal solutions to } (D(v, \mathbf{0})), \tag{9}$$

where for $(v, U) \in \Re^m \times \mathcal{S}^n$, $(D(v, U))$ is the (ordinary) dual of $(P(v, U))$ in the sense that

$$(D(v, U)) \quad \min \Big\{ b^{\mathrm{T}} y - v^{\mathrm{T}} y \ : \ \mathcal{A}^* y - U \succeq C \Big\}.$$

Finally, for any $(v, U) \in \Re^m \times \mathcal{S}^n$, we have that

$$\begin{aligned} T_l^{-1}(v, U) &= \arg \operatorname{minimax} \{ \, l(y, X) - v^{\mathrm{T}} y + \langle U,\, X \rangle \mid y \in \Re^m, X \in \mathcal{S}^n \}, \\ &= \text{set of all}\,(y, X)\,\text{satisfying the Karush-Kuhn-Tucker} \\ &\quad\ \text{conditions for } (P(v, U)).\ \text{(cf. (13))} \end{aligned} \tag{10}$$

**Definition 1.** [30] For a maximal monotone operator $T$ from a finite dimensional linear vector space $\mathcal{X}$ to itself, we say that its inverse $T^{-1}$ is Lipschitz continuous at the origin (with modulus $a \geq 0$) if there is a unique solution $\bar{z}$ to $z = T^{-1}(0)$, and for some $\tau > 0$ we have

$$\|z - \bar{z}\| \leq a \|w\| \quad \text{whenever} \quad z \in T^{-1}(w) \quad \text{and} \quad \|w\| \leq \tau. \tag{11}$$

Throughout this paper, the following generalized Slater condition for $(P)$ is assumed to hold.

**Assumption 1.** Problem $(P)$ satisfies the generalized Slater condition

$$\begin{cases} \mathcal{A} : \mathcal{S}^n \to \Re^m \text{ is onto}, \\ \exists\, X_0 \in \mathcal{S}^n_+ \text{ such that } \mathcal{A}(X_0) = b, X_0 \succ \mathbf{0}, \end{cases} \tag{12}$$

where $X_0 \succ \mathbf{0}$ means that $X_0$ is a symmetric positive definite matrix.

The first order optimality conditions, namely the Karush-Kuhn-Tucker (KKT) conditions, of $(D)$ and $(P)$ are as follows:

$$\begin{cases} \mathcal{A}(X) = b, \\ \mathcal{S}^n_+ \ni (\mathcal{A}^* y - C) \perp X \in \mathcal{S}^n_+, \end{cases} \tag{13}$$

where "$(\mathcal{A}^* y - C) \perp X$" means that $(\mathcal{A}^* y - C)$ and $X$ are orthogonal to each other, i.e., $\langle \mathcal{A}^* y - C, X \rangle = 0$. For any $X \in \mathcal{F}_P$, define the set

$$\mathcal{M}(X) := \{ y \mid \text{any } y \in \Re^m \text{ such that } (y, X) \text{ satisfies the KKT conditions (13)} \}. \tag{14}$$

Let $\overline{X}$ be an optimal solution to $(P)$. Since $(P)$ satisfies the generalized Slater condition (12), $\mathcal{M}(\overline{X})$ is nonempty and bounded [29, Theorems 17 & 18]. Let $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$ be the eigenvalues of $\overline{X}$ being arranged in the nonincreasing order. Let $\alpha := \{ i \mid \lambda_i > 0,\, i = 1, \ldots, n \}$. Since $\overline{X} \in \mathcal{S}^n_+$, there exists an orthogonal matrix $\overline{P} \in \Re^{n \times n}$ such that

$$\Lambda = \overline{P} \begin{bmatrix} \Lambda_\alpha & 0 \\ 0 & 0 \end{bmatrix} \overline{P}^{\mathrm{T}},$$

where $\Lambda_\alpha$ is the diagonal matrix whose diagonal entries are the positive eigenvalues of $\overline{X}$.

Let $y \in \mathcal{M}(\overline{X})$ be arbitrarily chosen. Then, from the second part of (13), we have

$$\overline{X}(\mathcal{A}^* y - C) = 0\,,$$

which, implies

$$\begin{bmatrix} \Lambda_\alpha & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \overline{P}_\alpha^{\mathrm{T}}(\mathcal{A}^* y - C)\overline{P}_\alpha & \overline{P}_\alpha^{\mathrm{T}}(\mathcal{A}^* y - C)\overline{P}_{\bar\alpha} \\ \overline{P}_{\bar\alpha}^{\mathrm{T}}(\mathcal{A}^* y - C)\overline{P}_\alpha & \overline{P}_{\bar\alpha}^{\mathrm{T}}(\mathcal{A}^* y - C)\overline{P}_{\bar\alpha} \end{bmatrix} = 0,$$

where $\bar\alpha := \{1, \cdots, n\} \backslash \alpha$ and $\overline{P} = [\overline{P}_\alpha \; \overline{P}_{\bar\alpha}]$. Thus,

$$\overline{P}_\alpha^{\mathrm{T}}(\mathcal{A}^* y - C)\overline{P}_\alpha = 0, \qquad \overline{P}_\alpha^{\mathrm{T}}(\mathcal{A}^* y - C)\overline{P}_{\bar\alpha} = 0,$$

and

$$\overline{P}^{\mathrm{T}}(\mathcal{A}^* y - C)\overline{P} = \begin{bmatrix} 0 & 0 \\ 0 & \overline{P}_{\bar\alpha}^{\mathrm{T}}(\mathcal{A}^* y - C)\overline{P}_{\bar\alpha} \end{bmatrix}. \tag{15}$$

Let $\mu_1 \le \mu_2 \le \cdots \le \mu_n$ be the eigenvalues of $(\mathcal{A}^*y - C)$ being arranged in the nondecreasing order. Denote $\gamma := \{i \,|\, \mu_i > 0,\, i = 1, \ldots, n\}$. Since $\overline{P}_{\bar\alpha}^{\mathrm{T}}(\mathcal{A}^*y - C)\overline{P}_{\bar\alpha}$ is symmetric and positive semidefinite, from (15) we know that there exists an orthogonal matrix $U \in \Re^{|\bar\alpha|\times|\bar\alpha|}$ such that

$$\overline{P}_{\bar\alpha}^{\mathrm{T}}(\mathcal{A}^*y - C)\overline{P}_{\bar\alpha} = U \begin{bmatrix} 0 & 0 \\ 0 & \Lambda_\gamma \end{bmatrix} U^{\mathrm{T}},$$

where $\Lambda_\gamma$ is the diagonal matrix whose diagonal entries are the positive eigenvalues of $(\mathcal{A}^*y - C)$. Let

$$P := \overline{P} \begin{bmatrix} I_{|\alpha|} & 0 \\ 0 & U \end{bmatrix}, \tag{16}$$

where $I_{|\alpha|}$ is the $|\alpha| \times |\alpha|$ identity matrix. Then

$$\overline{X} = P \begin{bmatrix} \Lambda_\alpha & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} P^{\mathrm{T}} \quad \text{and} \quad (\mathcal{A}^*y - C) = P \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \Lambda_\gamma \end{bmatrix} P^{\mathrm{T}}. \tag{17}$$

Let $A := \overline{X} - (\mathcal{A}^*y - C) \in \mathcal{S}_n$. Then, $A$ has the following spectral decomposition

$$A = P\Lambda P^{\mathrm{T}}, \tag{18}$$

where

$$\Lambda = \begin{bmatrix} \Lambda_\alpha & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & -\Lambda_\gamma \end{bmatrix}.$$

Denote $\beta := \{1, \ldots, n\} \backslash (\alpha \cup \gamma)$. Write $P = [P_\alpha \ P_\beta \ P_\gamma]$ with $P_\alpha \in \Re^{n\times|\alpha|}$, $P_\beta \in \Re^{n\times|\beta|}$, and $P_\gamma \in \Re^{n\times|\gamma|}$. From [2], we know that the tangent cone of $\mathcal{S}^n_+$ at $\overline{X} \in \mathcal{S}^n_+$ can be characterized as follows

$$\mathcal{T}_{\mathcal{S}^n_+}(\overline{X}) = \{B \in \mathcal{S}^n \mid [P_\beta \ P_\gamma]^{\mathrm{T}} B \, [P_\beta \ P_\gamma] \succeq 0\}. \tag{19}$$

Similarly, the tangent cone of $\mathcal{S}^n_+$ at $(\mathcal{A}^*y - C)$ takes the form

$$\mathcal{T}_{\mathcal{S}^n_+}(\mathcal{A}^*y - C) = \{B \in \mathcal{S}^n \mid [P_\alpha \ P_\beta]^{\mathrm{T}} B \, [P_\alpha \ P_\beta] \succeq 0\}. \tag{20}$$

Recall that the *critical cone* of problem $(P)$ at $\overline{X}$ is defined by (cf. [4, p.151])

$$\mathcal{C}(\overline{X}) = \{B \in \mathcal{S}^n \mid \mathcal{A}(B) = 0, \ B \in \mathcal{T}_{\mathcal{S}^n_+}(\overline{X}), \ \langle C, B\rangle = 0\}. \tag{21}$$

Choose an arbitrary element $B \in \mathcal{C}(\overline{X})$. Denote $\widetilde{B} := P^{\mathrm{T}}BP$. Since $\overline{X}$ and $(\mathcal{A}^*y - C)$ have the spectral decompositions as in (17), we obtain that

$$\begin{aligned} 0 = \langle C, B\rangle &= \langle \mathcal{A}^*y - C, B\rangle = \langle P^{\mathrm{T}}(\mathcal{A}^*y - C)P, P^{\mathrm{T}}BP\rangle \\ &= \left\langle \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \Lambda_\gamma \end{bmatrix}, \begin{bmatrix} \widetilde{B}_{\alpha\alpha} & \widetilde{B}_{\alpha\beta} & \widetilde{B}_{\alpha\gamma} \\ \widetilde{B}_{\alpha\beta}^{\mathrm{T}} & \widetilde{B}_{\beta\beta} & \widetilde{B}_{\beta\gamma} \\ \widetilde{B}_{\alpha\gamma}^{\mathrm{T}} & \widetilde{B}_{\beta\gamma}^{\mathrm{T}} & \widetilde{B}_{\gamma\gamma} \end{bmatrix} \right\rangle, \end{aligned}$$

which, together with (19) and (21), implies that $\widetilde{B}_{\gamma\gamma} = 0$. Thus

$$\widetilde{B}_{\beta\gamma} = 0 \quad \text{and} \quad \widetilde{B}_{\gamma\gamma} = 0.$$

Hence, $\mathcal{C}(\overline{X})$ can be rewritten as

$$\mathcal{C}(\overline{X}) = \{B \in \mathcal{S}^n \mid \mathcal{A}(B) = 0,\ P_\beta^{\mathrm{T}} B P_\beta \succeq 0,\ P_\beta^{\mathrm{T}} B P_\gamma = 0,\ P_\gamma^{\mathrm{T}} B P_\gamma = 0\}. \tag{22}$$

By using similar arguments as above, we can also obtain that

$$\mathcal{T}_{\mathcal{S}^n_+}(\mathcal{A}^* y - C) \cap \overline{X}^\perp = \{B \in \mathcal{S}^n \mid P_\alpha^{\mathrm{T}} B P_\alpha = 0,\ \ P_\alpha^{\mathrm{T}} B P_\beta = 0,\ \ P_\beta^{\mathrm{T}} B P_\beta \succeq 0\}. \tag{23}$$

In order to analyze the rate of convergence of the Newton-CG augmented Lagrangian method to be presented in Section 4, we need the following result which characterizes the Lipschitz continuity of $T_g^{-1}$ at the origin.

**Proposition 2.1.** *Suppose that (P) satisfies the generalized Slater condition (12). Let $\overline{X} \in \mathcal{S}^n_+$ be an optimal solution to (P). Then the following conditions are equivalent*

*(i) $T_g^{-1}(\cdot)$ is Lipschitz continuous at the origin.*

*(ii) The second order sufficient condition*

$$\sup_{y \in \mathcal{M}(\overline{X})} \Upsilon_{\overline{X}}(\mathcal{A}^* y - C, H) > 0 \quad \forall\, H \in \mathcal{C}(\overline{X}) \setminus \{0\} \tag{24}$$

*holds at $\overline{X}$, where for any $B \in \mathcal{S}^n$, the linear-quadratic function $\Upsilon_B : \mathcal{S}^n \times \mathcal{S}^n \to \Re$ is defined by*

$$\Upsilon_B(M, H) := 2\,\langle M, HB^\dagger H\rangle, \quad (M, H) \in \mathcal{S}^n \times \mathcal{S}^n \tag{25}$$

*and $B^\dagger$ is the Moore-Penrose pseudo-inverse of $B$.*

*(iii) $\overline{X}$ satisfies the extended strict constraint qualification*

$$\mathcal{A}^*\Re^m + \operatorname{conv}\left(\bigcup_{y \in \mathcal{M}(\overline{X})} \left(\mathcal{T}_{\mathcal{S}^n_+}(\mathcal{A}^* y - C) \cap \overline{X}^\perp\right)\right) = \mathcal{S}^n, \tag{26}$$

*where for any set $\mathcal{W} \subset \mathcal{S}^n$, $\operatorname{conv}(\mathcal{W})$ denotes the convex hull of $\mathcal{W}$.*

*Proof.* "$(i) \Leftrightarrow (ii)$". From [4, Theorem 3.137], we know that $(ii)$ holds if and only if the quadratic growth condition

$$\langle C, \overline{X}\rangle \geq \langle C, X\rangle + c\|X - \overline{X}\|^2 \qquad \forall\, X \in \mathcal{N} \text{ such that } X \in \mathcal{F}_P \tag{27}$$

holds at $\overline{X}$ for some positive constant $c$ and an open neighborhood $\mathcal{N}$ of $\overline{X}$ in $\mathcal{S}^n$. On the other hand, from [31, Proposition 3], we know that $T_g^{-1}(\cdot)$ is Lipschiz continuous at the origin if and only if the quadratic growth condition (27) holds at $\overline{X}$. Hence, $(i) \Leftrightarrow (ii)$.

Next we shall prove that $(ii) \Leftrightarrow (iii)$. For notational convenience, let

$$\Gamma := \operatorname{conv}\left( \bigcup_{y \in \mathcal{M}(\overline{X})} \left( \mathcal{T}_{\mathcal{S}^n_+}(\mathcal{A}^* y - C) \cap \overline{X}^{\perp} \right) \right). \tag{28}$$

"$(ii) \Rightarrow (iii)$". Denote $\mathcal{D} := \mathcal{A}^* \Re^m + \Gamma$. For the purpose of contradiction, we assume that $(iii)$ does not hold, i.e., $\mathcal{D} \neq \mathcal{S}^n$. Let $\mathrm{cl}(\mathcal{D})$ and $\mathrm{ri}(\mathcal{D})$ denote the closure of $\mathcal{D}$ and the relative interior of $\mathcal{D}$, respectively. By [27, Theorem 6.3], since $\mathrm{ri}(\mathcal{D}) = \mathrm{ri}(\mathrm{cl}(\mathcal{D}))$, the relative interior of $\mathrm{cl}(\mathcal{D})$, we know that $\mathrm{cl}(\mathcal{D}) \neq \mathcal{S}^n$. Thus, there exists $B \in \mathcal{S}^n$ such that $B \notin \mathrm{cl}(\mathcal{D})$. Let $\overline{B}$ be the metric projection of $B$ onto $\mathrm{cl}(\mathcal{D})$, i.e., $\overline{B} = \Pi_{\mathrm{cl}(\mathcal{D})}(B)$. Let $H = \overline{B} - B \neq 0$. Since $\mathrm{cl}(\mathcal{D})$ is a nonempty closed convex cone, from Zarantonello [42], we know that

$$\langle H, Z \rangle = \langle \overline{B} - B, Z \rangle \geq 0 \quad \forall\, Z \in \mathrm{cl}(\mathcal{D}).$$

In particular, we have

$$\langle H, \mathcal{A}^* z + Q \rangle \geq 0 \qquad \forall\, z \in \Re^m \text{ and } Q \in \Gamma,$$

which implies (by taking $Q = \mathbf{0}$)

$$\langle \mathcal{A}(H), z \rangle = \langle H, \mathcal{A}^* z \rangle \geq 0 \quad \forall\, z \in \Re^m.$$

Thus

$$\mathcal{A}(H) = 0 \quad \text{and} \quad \langle H, Q \rangle \geq 0 \quad \text{for any } Q \in \Gamma. \tag{29}$$

Let $y \in \Re^m$ be an arbitrary element in $\mathcal{M}(\overline{X})$. Since $(y, \overline{X})$ satisfies the KKT conditions (13), we can assume that $\overline{X}$ and $(\mathcal{A}^* y - C)$ have the spectral decompositions as in (17). Then, we know from (23) that for any $Q \in \mathcal{T}_{\mathcal{S}^n_+}(\mathcal{A}^* y - C) \cap \overline{X}^{\perp}$,

$$\begin{aligned} 0 \;&\leq\; \langle H, Q \rangle = \langle P\widetilde{H}P^{\mathrm{T}}, P\widetilde{Q}P^{\mathrm{T}} \rangle \\ &= \left\langle \begin{bmatrix} \widetilde{H}_{\alpha\alpha} & \widetilde{H}_{\alpha\beta} & \widetilde{H}_{\alpha\gamma} \\ \widetilde{H}_{\alpha\beta}^{\mathrm{T}} & \widetilde{H}_{\beta\beta} & \widetilde{H}_{\beta\gamma} \\ \widetilde{H}_{\alpha\gamma}^{\mathrm{T}} & \widetilde{H}_{\beta\gamma}^{\mathrm{T}} & \widetilde{H}_{\gamma\gamma} \end{bmatrix}, \begin{bmatrix} 0 & 0 & \widetilde{Q}_{\alpha\gamma} \\ 0 & \widetilde{Q}_{\beta\beta} & \widetilde{Q}_{\beta\gamma} \\ \widetilde{Q}_{\alpha\gamma} & \widetilde{Q}_{\beta\gamma} & \widetilde{Q}_{\gamma\gamma} \end{bmatrix} \right\rangle, \end{aligned} \tag{30}$$

where $\widetilde{H} = P^{\mathrm{T}} H P$ and $\widetilde{Q} = P^{\mathrm{T}} Q P$. From (23) and (30), we have

$$\widetilde{H}_{\alpha\gamma} = 0, \quad \widetilde{H}_{\beta\gamma} = 0, \quad \widetilde{H}_{\gamma\gamma} = 0, \quad \text{and} \quad \widetilde{H}_{\beta\beta} \succeq 0. \tag{31}$$

By using (22), (29), and (31), we obtain that $H \in \mathcal{C}(\overline{X})$ and

$$P_\alpha^{\mathrm{T}} H P_\gamma = 0. \tag{32}$$

Since $0 \neq H \in \mathcal{C}(\overline{X})$ and $(ii)$ is assumed to hold, there exists $y \in \mathcal{M}(\overline{X})$ such that

$$\Upsilon_{\overline{X}}(\mathcal{A}^* y - C, H) > 0. \tag{33}$$

By using the fact that $(y, \overline{X})$ satisfies (13), we can assume that $\overline{X}$ and $(\mathcal{A}^* y - C)$ have the spectral decompositions as in (17), i.e., there exists an orthogonal matrix $P \in \Re^{n \times n}$ such that

$$\overline{X} = P \begin{bmatrix} \Lambda_\alpha & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} P^{\mathrm{T}} \quad \text{and} \quad (\mathcal{A}^* y - C) = P \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \Lambda_\gamma \end{bmatrix} P^{\mathrm{T}}. \tag{34}$$

Note that $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$ and $\mu_1 \leq \mu_2 \leq \cdots \leq \mu_n$ are the eigenvalues of $\overline{X}$ and $(\mathcal{A}^* y - C)$, respectively, and $\alpha = \{i \,|\, \lambda_i > 0,\ i = 1, \ldots, n\}$ and $\gamma = \{j \,|\, \mu_j > 0,\ j = 1, \ldots, n\}$. Therefore, from (25), (33), and (34), we obtain that

$$0 < \Upsilon_{\overline{X}}(\mathcal{A}^* y - C, H) = \sum_{i \in \alpha, j \in \gamma} \frac{\mu_j}{\lambda_i} (P_i^{\mathrm{T}} H P_j)^2,$$

which implies $P_\alpha^{\mathrm{T}} H P_\gamma \neq 0$. This contradicts (32) and consequently shows $(ii) \Rightarrow (iii)$.

"$(iii) \Rightarrow (ii)$". Assume that $(ii)$ does not hold at $\overline{X}$. Then there exists $0 \neq H \in \mathcal{C}(\overline{X})$ such that

$$\sup_{y \in \mathcal{M}(\overline{X})} \Upsilon_{\overline{X}}(\mathcal{A}^* y - C, H) = 0. \tag{35}$$

Let $y$ be an arbitrary element in $\mathcal{M}(\overline{X})$. Since $(y, \overline{X})$ satisfies (13), we can assume that there exists an orthogonal matrix $P \in \Re^{n \times n}$ such that $\overline{X}$ and $(\mathcal{A}^* y - C)$ have the spectral decompositions as in (17). From (17), (25), and (35), we have

$$0 \leq \sum_{i \in \alpha, j \in \gamma} \frac{\mu_j}{\lambda_i} (P_i^{\mathrm{T}} H P_j)^2 = \Upsilon_{\overline{X}}(\mathcal{A}^* y - C, H) \leq \sup_{z \in \mathcal{M}(\overline{X})} \Upsilon_{\overline{X}}(\mathcal{A}^* z - C, H) = 0,$$

which implies

$$P_\alpha^{\mathrm{T}} H P_\gamma = 0. \tag{36}$$

Then, by using (22), (23), and (36), we have that

$$\langle Q^y, H \rangle = \langle P^{\mathrm{T}} Q^y P, P^{\mathrm{T}} H P \rangle = \langle P_\beta^{\mathrm{T}} Q^y P_\beta, P_\beta^{\mathrm{T}} H P_\beta \rangle \geq 0 \quad \forall\, Q^y \in \mathcal{T}_{\mathcal{S}_+^n}(\mathcal{A}^* y - C) \cap \overline{X}^{\perp}. \tag{37}$$

Since $(iii)$ is assumed to hold, there exist $z \in \Re^m$ and $Q \in \Gamma$ such that

$$-H = \mathcal{A}^* z + Q. \tag{38}$$

By Carathéodory's Theorem, there exist an integer $k \leq \frac{n(n+1)}{2} + 1$ and scalars $\alpha_i \geq 0$, $i = 1, 2, \ldots, k$, with $\sum_{i=1}^{k} \alpha_i = 1$, and

$$Q_i \in \bigcup_{y \in \mathcal{M}(\overline{X})} \left( \mathcal{T}_{\mathcal{S}^n_+}(\mathcal{A}^* y - C) \cap \overline{X}^{\perp} \right), \quad i = 1, 2, \ldots, k$$

such that $Q$ can be represented as

$$Q = \sum_{i=1}^{k} \alpha_i Q_i.$$

For each $Q_i$, there exists a $y^i \in \mathcal{M}(\overline{X})$ such that $Q_i \in \mathcal{T}_{\mathcal{S}^n_+}(\mathcal{A}^* y^i - C) \cap \overline{X}^{\perp}$. Then by using the fact that $H \in \mathcal{C}(\overline{X})$ and (37), we obtain that

$$\langle H, H \rangle = \langle -\mathcal{A}^* z - Q, H \rangle = -\langle z, \mathcal{A}H \rangle - \langle Q, H \rangle = 0 - \sum_{i=1}^{k} \alpha_i \langle Q_i, H \rangle \leq 0,$$

which contradicts the fact that $H \neq 0$. This contradiction shows that $(ii)$ holds. $\square$

Proposition 2.1 characterizes the Lipschitz continuity of $T_g^{-1}$ at the origin by either the second sufficient condition (24) or the extended strict constraint qualification (26). In particular, if $\mathcal{M}(\overline{X})$ is a singleton, we have the following simple equivalent conditions.

**Corollary 2.2.** *Suppose that (P) satisfies the generalized Slater condition (12). Let $\overline{X}$ be an optimal solution to (P). If $\mathcal{M}(\overline{X}) = \{\bar{y}\}$, then the following are equivalent:*

*(i)* $T_g^{-1}(\cdot)$ *is Lipschitz continuous at the origin.*

*(ii) The second order sufficient condition*

$$\Upsilon_{\overline{X}}(\mathcal{A}^* \bar{y} - C, H) > 0 \quad \forall\, H \in \mathcal{C}(\overline{X}) \setminus \{0\} \tag{39}$$

*holds at* $\overline{X}$.

*(iii)* $\overline{X}$ *satisfies the strict constraint qualification*

$$\mathcal{A}^* \Re^m + \mathcal{T}_{\mathcal{S}^n_+}(\mathcal{A}^* \bar{y} - C) \cap \overline{X}^{\perp} = \mathcal{S}^n. \tag{40}$$

**Remark 1.** Note that in [8, Proposition 15], Chan and Sun proved that if $\mathcal{M}(\overline{X})$ is a singleton, then the *strong* second order sufficient condition (with the set $\mathcal{C}(\overline{X})$ in (39) being replaced by the superset $\{B \in \mathcal{S}^n \mid \mathcal{A}(B) = 0, P_\beta^T B P_\gamma = 0, P_\gamma^T B P_\gamma = 0\}$) is equivalent to the constraint nondegenerate condition, in the sense of Robinson [25, 26], at $\bar{y}$ for $(D)$, i.e,

$$\mathcal{A}^*\Re^m + \operatorname{lin}(\mathcal{T}_{\mathcal{S}_+^n}(\mathcal{A}^*\bar{y} - C)) = \mathcal{S}^n. \tag{41}$$

Corollary 2.2 further establishes the equivalence between the second order sufficient condition (39) and the strict constraint qualification (40) under the condition that $\mathcal{M}(\overline{X})$ is a singleton.

One may observe that the strict constraint qualification condition (40) is weaker than the constraint nondegenerate condition (41). However, if strict complementarity holds, i.e., $\overline{X} + (\mathcal{A}^*\bar{y} - C) \succ 0$ and hence $\beta$ is the empty set, then (40) and (41) coincide.

The constraint nondegenerate condition (41) is equivalent to the dual nondegeneracy stated in [1, Theorem 9]. Note that under such a condition, the optimal solution $\overline{X}$ to $(P)$ is unique.

**Remark 2.** In a similar way, we can establish parallel results for $T_f^{-1}$ as for $T_g^{-1}$ in Proposition 2.1 and Corollary 2.2. For brevity, we omit the details.

# 3 A Semismooth Newton-CG Method for Inner Problems

In this section we introduce a semismooth Newton-CG method for solving the inner problems involved in the augmented Lagrangian method (3). Firstly we present a practical CG method for solving the positive definite linear system. This practical CG method will be used heavily in solving the linear systems arising from applying the semismooth Newton-CG method to these inner problems.

## 3.1 A practical CG method

In this subsection, we consider a practical CG method to solve the following linear equation

$$Ax = b \tag{42}$$

where $b \in \Re^m$ and $A \in \Re^{m\times m}$ is assumed to be a symmetric positive definite matrix. The practical conjugate gradient algorithm [13, Algorithm 10.2.1] depends on two parameters: a maximum number of CG iterations $i_{max} > 0$ and a tolerance $\eta \in (0, \|b\|)$.

**Algorithm 1. Practical CG Algorithm:** $[CG(\eta, i_{max})]$

**Step 0.** Given $x^0 = 0$ and $r^0 = b - Ax^0$.

**Step 1.** While ($\|r^i\| > \eta$) or ($i < i_{max}$)

Step 1.1. $i = i + 1$

Step 1.2. If $i = 1$
$p^1 = r^0$
else
$\beta_i = \|r^{i-1}\|^2 / \|r^{i-2}\|^2$
$p^i = r^{i-1} + \beta_i \, p^{i-1}$
end

Step 1.3. $\alpha_i = \|r^{i-1}\|^2 / \langle p^i, Ap^i \rangle$

Step 1.4. $x^i = x^{i-1} + \alpha_i p^i$

Step 1.5. $r^i = r^{i-1} - \alpha_i A p^i$

**Lemma 3.1.** *Let $0 < \bar{i} \leq i_{\max}$ be the number of iterations when the practical CG Algorithm 1 terminates. For all $i = 1, 2, \cdots, \bar{i}$, the iterates $\{x^i\}$ generated by Algorithm 1 satisfies*

$$\frac{1}{\lambda_{\max}(A)} \leq \frac{\langle x^i, b \rangle}{\|b\|^2} \leq \frac{1}{\lambda_{\min}(A)}, \tag{43}$$

*where $\lambda_{\min}(A)$ and $\lambda_{\max}(A)$ are the smallest and largest eigenvalue of $A$, respectively.*

*Proof.* Let $x^*$ be the exact solution to (42) and $e^i = x^* - x^i$ be the error in the $i$th iteration for $i \geq 0$. From [37, Theorem 38.1], we know that

$$\langle r^i, r^j \rangle = 0 \quad \text{for } j = 1, 2, \ldots, i-1, \tag{44}$$

where $r^i = b - Ax^i$. By using (44), the fact that in Algorithm 1, $r^0 = b$, and the definition of $\beta_i$, we have that

$$\begin{aligned} \langle p^1, b \rangle &= \|r^0\|^2, \\ \langle p^i, b \rangle &= \langle r^{i-1}, b \rangle + \beta_i \langle p^{i-1}, b \rangle = 0 + \prod_{j=2}^{i} \beta_j \langle p^1, b \rangle = \|r^{i-1}\|^2 \quad \forall\, i > 1. \end{aligned} \tag{45}$$

From [37, Theorem 38.2], we know that for $i \geq 1$,

$$\|e^{i-1}\|_A^2 = \|e^i\|_A^2 + \langle \alpha_i p^i, A(\alpha_i p^i) \rangle, \tag{46}$$

which, together with $\alpha_i \|r^{i-1}\|^2 = \langle \alpha_i p^i, A(\alpha_i p^i) \rangle$ (see Step 1.3), implies that

$$\alpha_i \|r^{i-1}\|^2 = \|e^{i-1}\|_A^2 - \|e^i\|_A^2. \tag{47}$$

Here for any $x \in \Re^m$, $\|x\|_A := \sqrt{\langle x, Ax\rangle}$. For any $i \geq 1$, by using (45), (47), and the fact that $x^0 = 0$, we have that

$$\begin{aligned} \langle x^i, b\rangle &= \langle x^{i-1}, b\rangle + \alpha_i \langle p^i, b\rangle = \langle x^0, b\rangle + \sum_{j=1}^{i} \alpha_j \langle p^j, b\rangle = \sum_{j=1}^{i} \alpha_j \|r^{j-1}\|^2 \\ &= \sum_{j=1}^{i} \left[\|e^{j-1}\|_A^2 - \|e^j\|_A^2\right] = \|e^0\|_A^2 - \|e^i\|_A^2, \end{aligned} \tag{48}$$

which, together with (46), implies that

$$\langle x^i,\, b\rangle \geq \langle x^{i-1},\, b\rangle, \quad i = 1, 2, \ldots, \bar{i}.$$

Thus

$$\frac{1}{\lambda_{\max}(A)} \leq \alpha_1 = \frac{\langle x^1, b\rangle}{\|b\|^2} \leq \frac{\langle x^i,\, b\rangle}{\|b\|^2}. \tag{49}$$

Since $e^0 = x^* - x^0 = A^{-1}b$, by (48), we obtain that for $1 \leq i \leq \bar{i}$,

$$\frac{\langle x^i, b\rangle}{\|b\|^2} \leq \frac{\|e^0\|_A^2}{\|b\|^2} = \frac{\|A^{-1}b\|_A^2}{\|b\|^2} \leq \frac{1}{\lambda_{\min}(A)}. \tag{50}$$

By combining (49) and (50), we complete the proof.

$\square$

### 3.2 A Semismooth Newton-CG method

For the augmented Lagrangian method (3), for some fixed $X \in \mathcal{S}^n$ and $\sigma > 0$, we need to consider the following form of inner problems

$$\min\{\varphi(y) := L_\sigma(y, X) \mid y \in \Re^m\}. \tag{51}$$

As explained in the introduction, $\varphi(\cdot)$ is a continuously differentiable convex function, but fails to be twice continuously differentiable because the metric projector $\Pi_{\mathcal{S}^n_+}(\cdot)$ is not continuously differentiable. Fortunately, because $\Pi_{\mathcal{S}^n_+}(\cdot)$ is strongly semismooth [35], we can develop locally a semismooth Newton-CG method to solve the following nonlinear equation

$$\nabla\varphi(y) = b - \mathcal{A}\Pi_{\mathcal{S}^n_+}(X - \sigma(\mathcal{A}^* y - C)) = 0 \tag{52}$$

and expect a superlinear (quadratic) convergence.

**Remark 3.** For a given $X \in \mathcal{S}^n$ and $\sigma > 0$, let $W(y) := X - \sigma(\mathcal{A}^*y - C)$. It is readily shown that the solution $y$ to (52) is part of the root of the following equations:

$$\begin{aligned} \sigma\mathcal{A}\mathcal{A}^*y &= \sigma\mathcal{A}(C+S) - (b - \mathcal{A}(X)); \quad (53) \\ S &= \frac{1}{\sigma}\Big(\Pi_{\mathcal{S}^n_+}(W(y)) - W(y)\Big). \quad (54) \end{aligned}$$

To compute a solution to (53) and (54), one may use an alternating direction method. That is, given an initial vector $y^0 \in \Re^m$ and an accuracy tolerance $\varepsilon$, perform the following loop:

$$\begin{array}{l} W = X - \sigma(\mathcal{A}^*y^j - C); \\ S^j = \frac{1}{\sigma}(\Pi_{\mathcal{S}^n_+}(W) - W); \\ \delta = \sigma\mathcal{A}(C + S^j - \mathcal{A}^*y^j) + \mathcal{A}(X) - b; \\ \text{If } \|\delta\| \le \varepsilon, \text{ stop; Else, compute } y^{j+1} \text{ from (53) with } S = S^j; \end{array} \quad (55)$$

It is readily shown that the above algorithm is actually a gradient based method to solve (52) where $y^{j+1}$ is updated as follows:

$$y^{j+1} = y^j - (\sigma\mathcal{A}\mathcal{A}^*)^{-1}\nabla\varphi(y^j). \quad (56)$$

Since $\Pi_{\mathcal{S}^n_+}(\cdot)$ is Lipschitz continuous with modulus 1, the mapping $\nabla\varphi$ is Lipschitz continuous on $\Re^m$. According to Rademacher's Theorem, $\nabla\varphi$ is almost everywhere Fréchet-differentiable in $\Re^m$. Let $y \in \Re^m$. The generalized Hessian of $\varphi$ at $y$ is defined as

$$\partial^2\varphi(y) := \partial(\nabla\varphi)(y), \quad (57)$$

where $\partial(\nabla\varphi)(y)$ is the Clarke's generalized Jacobian of $\nabla\varphi$ at $y$ [9]. Since it is difficult to express $\partial^2\varphi(y)$ exactly, we define the following alternative for $\partial^2\varphi(y)$

$$\hat{\partial}^2\varphi(y) := \sigma\mathcal{A}\,\partial\Pi_{\mathcal{S}^n_+}(X - \sigma(\mathcal{A}^*y - C))\mathcal{A}^*. \quad (58)$$

From [9, p.75], for $d \in \Re^m$,

$$\partial^2\varphi(y)d \subseteq \hat{\partial}^2\varphi(y)d, \quad (59)$$

which means that if every element in $\hat{\partial}^2\varphi(y)$ is positive definite, so is every element in $\partial^2\varphi(y)$.

For the semismooth Newton-CG method to be presented later, we need to compute an element $V \in \hat{\partial}^2\varphi(y)$. Since $X - \sigma(\mathcal{A}^*y - C)$ is a symmetric matrix in $\Re^{n\times n}$, there exists an orthogonal matrix $Q \in \Re^{n\times n}$ such that

$$X - \sigma(\mathcal{A}^*y - C) = Q\,\Gamma_y\,Q^{\mathrm{T}}, \quad (60)$$

where $\Gamma_y$ is the diagonal matrix with diagonal entries consisting of the eigenvalues $\lambda_1 \ge \lambda_2 \ge \cdots \ge \lambda_n$ of $X - \sigma(\mathcal{A}^*y - C)$ being arranged in the nonincreasing order. Define three index sets

$$\alpha := \{i \mid \lambda_i > 0\}, \quad \beta := \{i \mid \lambda_i = 0\}, \quad \text{and} \quad \gamma := \{i \mid \lambda_i < 0\}.$$

Define the operator $W_y^0 : \mathcal{S}^n \to \mathcal{S}^n$ by

$$W_y^0(H) := Q(\Omega \circ (Q^{\mathrm{T}} H Q))Q^{\mathrm{T}}, \quad H \in \mathcal{S}^n, \tag{61}$$

where

$$\Omega = \begin{bmatrix} E_{\bar{\gamma}\bar{\gamma}} & \nu_{\bar{\gamma}\gamma} \\ \nu_{\bar{\gamma}\gamma}^{\mathrm{T}} & 0 \end{bmatrix}, \quad \nu_{ij} := \frac{\lambda_i}{\lambda_i - \lambda_j}, \ i \in \bar{\gamma}, j \in \gamma, \tag{62}$$

$\bar{\gamma} = \{1, \ldots, n\} \backslash \gamma$, and $E_{\bar{\gamma}\bar{\gamma}} \in \mathcal{S}^{|\bar{\gamma}|}$ is the matrix of ones. Define $V_y^0 : \Re^m \to \mathcal{S}^n$ by

$$V_y^0 d := \sigma \mathcal{A}[\, Q(\Omega \circ (\, Q^{\mathrm{T}} (\mathcal{A}^* d)\, Q))\, Q^{\mathrm{T}}], \quad d \in \Re^m. \tag{63}$$

Since, by Pang, Sun, and Sun [20, Lemma 11],

$$W_y^0 \in \partial_B \Pi_{\mathcal{S}_+^n}(X - \sigma(\mathcal{A}^* y - C)),$$

we know that

$$V_y^0 = \sigma \mathcal{A} W_y^0 \mathcal{A}^* \in \hat{\partial}^2 \varphi(y).$$

Next we shall characterize the positive definiteness of any $V_y \in \hat{\partial}^2 \varphi(y)$. From [31, p.107] and the definitions of $l(y, X)$ in (4), we know that for any $(y, X, \sigma) \in \Re^m \times \mathcal{S}^n \times (0, +\infty)$,

$$L_\sigma(y, X) = \max_{Z \in \mathcal{S}^n} \{ l(y, Z) - \frac{1}{2\sigma} \|Z - X\|^2 \}.$$

Since the generalized Slater condition (12) is assumed to hold, by the definition of $g(\cdot)$ in (6), we can deduce from [29, Theorems 17 and 18] that

$$\begin{aligned} \min_{y \in \Re^m} \varphi(y) &= \min_{y \in \Re^m} \max_{Z \in \mathcal{S}^n} \left\{ l(y, Z) - \frac{1}{2\sigma} \|Z - X\|^2 \right\} = \max_{Z \in \mathcal{S}^n} \left\{ g(Z) - \frac{1}{2\sigma} \|Z - X\|^2 \right\} \\ &= \max_{\mathcal{A}(Z) = b,\, Z \succeq \mathbf{0}} \left\{ \langle C, Z \rangle - \frac{1}{2\sigma} \|Z - X\|^2 \right\}. \end{aligned} \tag{64}$$

Hence, (51) is the dual of

$$\max \left\{ \langle\, C, Z \rangle - \frac{1}{2\sigma} \|Z - X\|^2 \ \mid \ \mathcal{A}(Z) = b, \quad Z \succeq \mathbf{0} \right\}. \tag{65}$$

The KKT conditions of (65) are as follows

$$\left\{ \begin{array}{l} \mathcal{A}(Z) = b, \\ \mathcal{S}_+^n \ni Z \perp [Z - (X - \sigma(\mathcal{A}^* y - C))] \in \mathcal{S}_+^n. \end{array} \right. \tag{66}$$

**Proposition 3.2.** *Suppose that the problem (65) satisfies the generalized Slater condition (12). Let* $(\hat{y}, \widehat{Z}) \in \Re^m \times \mathcal{S}^n$ *be a pair that satisfies the KKT conditions (66) and let* $P$ *be an orthogonal matrix such that* $\widehat{Z}$ *and* $\widehat{Z} - (X - \sigma(\mathcal{A}^*\hat{y} - C))$ *have the spectral decomposition as (17). Then the following conditions are equivalent:*

*(i) The constraint nondegenerate condition*

$$\mathcal{A}\,\mathrm{lin}(\mathcal{T}_{\mathcal{S}^n_+}(\widehat{Z})) = \Re^m \tag{67}$$

*holds at* $\widehat{Z}$*, where* $\mathrm{lin}(\mathcal{T}_{\mathcal{S}^n_+}(\widehat{Z}))$ *denotes the lineality space of* $\mathcal{T}_{\mathcal{S}^n_+}(\widehat{Z})$*, i.e.,*

$$\mathrm{lin}(\mathcal{T}_{\mathcal{S}^n_+}(\widehat{Z})) = \{B \in \mathcal{S}^n \mid [P_\beta \ P_\gamma]^T B \, [P_\beta \ P_\gamma] = 0\}. \tag{68}$$

*(ii) Every* $V_{\hat{y}} \in \hat{\partial}^2\varphi(\hat{y})$ *is symmetric and positive definite.*

*(iii)* $V^0_{\hat{y}} \in \hat{\partial}^2\varphi(\hat{y})$ *is symmetric and positive definite.*

*Proof.* "$(i) \Rightarrow (ii)$". For the sake of contradiction, suppose that $(ii)$ does not hold. Then there exists $V_{\hat{y}} \in \hat{\partial}^2\varphi(\hat{y})$ such that $V_{\hat{y}}$ is not positive definite. By the definition of $\hat{\partial}^2\varphi(\hat{y})$ in (58), there exists a $W_{\hat{y}} \in \partial\Pi_{\mathcal{S}^n_+}(X - \sigma(\mathcal{A}^*\hat{y} - C))$ such that

$$V_{\hat{y}} = \sigma \mathcal{A} W_{\hat{y}} \mathcal{A}^* \not\succ \mathbf{0}.$$

Moreover, by the positive semidefiniteness of $W_{\hat{y}}$ [18, Proposition 1], we know that there exists a $0 \neq d \in \Re^m$ such that

$$\langle d, V_{\hat{y}}\, d\rangle = \sigma\langle \mathcal{A}^* d, W_{\hat{y}}(\mathcal{A}^* d)\rangle = \sigma\langle H, W_{\hat{y}}(H)\rangle = 0, \tag{69}$$

where $H := \mathcal{A}^* d$. Since $W_{\hat{y}} \in \partial\Pi_{\mathcal{S}^n_+}(X - \sigma(\mathcal{A}^*\hat{y} - C))$, there exists a $U_{|\beta|} \in \partial\Pi_{\mathcal{S}^{|\beta|}_+}(\mathbf{0})$ [34, Proposition 2.2] such that

$$\begin{aligned}
0 &= \langle d, V_{\hat{y}}\, d\rangle = \sigma\langle H, W_{\hat{y}}(H)\rangle = \sigma\langle \widetilde{H}, P^{\mathrm{T}} W_{\hat{y}}(H) P\rangle \\
&= \sigma\left\langle \begin{bmatrix} \widetilde{H}_{\alpha\alpha} & \widetilde{H}_{\alpha\beta} & \widetilde{H}_{\alpha\gamma} \\ \widetilde{H}^{\mathrm{T}}_{\alpha\beta} & \widetilde{H}_{\beta\beta} & \widetilde{H}_{\beta\gamma} \\ \widetilde{H}^{\mathrm{T}}_{\alpha\gamma} & \widetilde{H}^{\mathrm{T}}_{\beta\gamma} & \widetilde{H}_{\gamma\gamma} \end{bmatrix}, \begin{bmatrix} \widetilde{H}_{\alpha\alpha} & \widetilde{H}_{\alpha\beta} & \nu_{\alpha\gamma} \circ \widetilde{H}_{\alpha\gamma} \\ \widetilde{H}^{\mathrm{T}}_{\alpha\beta} & U_{|\beta|}(\widetilde{H}_{\beta\beta}) & 0 \\ (\nu_{\alpha\gamma} \circ \widetilde{H}_{\alpha\gamma})^{\mathrm{T}} & 0 & 0 \end{bmatrix} \right\rangle \\
&= \sigma(\langle \widetilde{H}_{\alpha\alpha}, \widetilde{H}_{\alpha\alpha}\rangle + 2\langle \widetilde{H}_{\alpha\beta}, \widetilde{H}_{\alpha\beta}\rangle + 2\langle \widetilde{H}_{\alpha\gamma}, \nu_{\alpha\gamma} \circ \widetilde{H}_{\alpha\gamma}\rangle + \langle \widetilde{H}_{\beta\beta}, U_{|\beta|}(\widetilde{H}_{\beta\beta})\rangle),
\end{aligned}$$

where $\widetilde{H} = P^{\mathrm{T}} H P$ and $\nu_{ij} > 0$, for $i \in \alpha$ and $j \in \gamma$, defined as in (62). Hence, we obtain that

$$\widetilde{H}_{\alpha\alpha} = 0, \quad \widetilde{H}_{\alpha\beta} = 0, \quad \widetilde{H}_{\alpha\gamma} = 0 \quad \text{and} \quad \langle \widetilde{H}_{\beta\beta}, U_{|\beta|}(\widetilde{H}_{\beta\beta})\rangle = 0,$$

which, together with (68), implies that

$$\langle H, Q\rangle = \langle P^{\mathrm{T}} H P, P^{\mathrm{T}} Q P\rangle = 0 \quad \forall\, Q \in \mathrm{lin}(\mathcal{T}_{\mathcal{S}^n_+}(\widehat{Z})). \tag{70}$$

Since the constraint nondegenerate condition (67) holds at $\widehat{Z}$, there exists a matrix $\overline{Q} \in \text{lin}(\mathcal{T}_{\mathcal{S}^n_+}(\widehat{Z}))$ such that $d = \mathcal{A}(\overline{Q})$. By using (70), we have that

$$\langle d, d\rangle = \langle d, \mathcal{A}(\overline{Q}\rangle = \langle \mathcal{A}^* d, \overline{Q}\rangle = \langle H, \overline{Q}\rangle = 0. \tag{71}$$

Thus $d = 0$. This contradicts our assumption. Consequently, $(ii)$ holds.

"$(ii) \Rightarrow (iii)$". This is obvious true since $V^0_{\hat{y}} \in \hat{\partial}^2\varphi(\hat{y})$.

"$(iii) \Rightarrow (i)$". Assume on the contrary that the constraint nondegenerate condition (67) does not hold at $\widehat{Z}$. Then, we have

$$[\mathcal{A}\text{lin}(\mathcal{T}_{\mathcal{S}^n_+}(\widehat{Z}))]^{\perp} \neq \{0\}.$$

Let $0 \neq d \in [\mathcal{A}\text{lin}(\mathcal{T}_{\mathcal{S}^n_+}(\widehat{Z}))]^{\perp}$. Then

$$\langle\, d, \mathcal{A}(Q)\,\rangle = 0 \quad \forall\ Q \in \text{lin}(\mathcal{T}_{\mathcal{S}^n_+}(\widehat{Z})),$$

which can be written as

$$0 = \langle\, \mathcal{A}^* d, Q\,\rangle = \langle\, P^{\mathrm{T}} H P, P^{\mathrm{T}} Q P\,\rangle \quad \forall\ Q \in \text{lin}(\mathcal{T}_{\mathcal{S}^n_+}(\widehat{Z})), \tag{72}$$

where $H := \mathcal{A}^* d$. By using (68) and (72), we obtain that

$$P_\alpha^{\mathrm{T}} H P_\alpha = 0,\ P_\alpha^{\mathrm{T}} H P_\beta = 0,\ \text{and}\ P_\alpha^{\mathrm{T}} H P_\gamma = 0.$$

By the definition of $W^0_{\hat{y}}$ in (61), it follows that $W^0_{\hat{y}}(H) = 0$. Therefore, for the corresponding $V^0_{\hat{y}}$ defined in (63), we have

$$\langle\, d, V^0_{\hat{y}}\, d\,\rangle = \langle\, d, \sigma\mathcal{A}W^0_{\hat{y}}(\mathcal{A}^* d)\,\rangle = \sigma\langle\, H, W^0_{\hat{y}}(H)\,\rangle = 0,$$

which contradicts $(iii)$ since $d \neq 0$. This contradiction shows that $(i)$ holds. $\square$

**Remark 4.** In Proposition 3.2, the "$(i) \Rightarrow (ii)$" part is implied in [3, Proposition 2.8] by the Jacobian amicability of the metric projector $\Pi_{\mathcal{S}^n_+}(\cdot)$. Here we give the proof for completeness.

**Remark 5.** The constraint nondegenerate condition (67) is equivalent to the primal nondegeneracy stated in [1, Theorem 6]. Under this condition, the solution $\hat{y}$ for (66) is unique.

### 3.3 Convergence analysis

In this subsection, we shall introduce the promised semismooth Newton-CG algorithm to solve (51). Choose $y^0 \in \Re^m$. Then the algorithm can be stated as follows.

**Algorithm 2. Semismooth Newton-CG Algorithm** $[NCG(y^0, X, \sigma)]$

**Step 0.** Given $\mu = (0, 1/2)$, $\bar{\eta} \in (0, 1)$, $\tau \in (0, 1]$, $\tau_1, \tau_2 \in (0, 1)$, and $\delta \in (0, 1)$.

**Step 1.** For $j = 0, 1, 2, \ldots$

Step 1.1. Given a maximum number of CG iterations $n_j > 0$ and compute

$$\eta_j := \min(\bar{\eta}, \|\nabla\varphi(y^j)\|^{1+\tau}).$$

Apply the practical CG Algorithm 1 $[CG(\eta_j, n_j)]$ to find an approximation solution $d^j$ to

$$(V_j + \varepsilon_j I)\, d = -\nabla\varphi(y^j), \tag{73}$$

where $V_j \in \hat{\partial}^2\varphi(y^j)$ is defined in (63) and $\varepsilon_j := \tau_1 \min\{\tau_2, \|\nabla\varphi(y^j)\|\}$.

Step 1.2. Set $\alpha_j = \delta^{m_j}$, where $m_j$ is the first nonnegative integer $m$ for which

$$\varphi(y^j + \delta^m d^j) \le \varphi(y^j) + \mu\delta^m \langle \nabla\varphi(y^j), d^j \rangle. \tag{74}$$

Step 1.3. Set $y^{j+1} = y^j + \alpha_j\, d^j$.

**Remark 6.** In Algorithm 2, since $V_j$ is always positive semidefinite, the matrix $V_j + \varepsilon_j I$ is positive definite as long as $\nabla\varphi(y^j) \neq 0$. So we can always apply Algorithm 1 to equation (73).

Now we can analyze the global convergence of Algorithm 2 with the assumption that $\nabla\varphi(y^j) \neq 0$ for any $j \ge 0$. From Theorem 3.1, we know that the search direction $d^j$ generated by Algorithm 2 is always a descent direction. This is stated in the following proposition.

**Proposition 3.3.** *For every $j \ge 0$, the search direction $d^j$ generated in Step 1.2 of Algorithm 2 satisfies*

$$\frac{1}{\lambda_{\max}(\widetilde{V}_j)} \le \frac{\langle -\nabla\varphi(y^j), d^j \rangle}{\|d^j\|^2} \le \frac{1}{\lambda_{\min}(\widetilde{V}_j)}\,, \tag{75}$$

*where $\widetilde{V}_j := V_j + \varepsilon_j I$ and $\lambda_{\max}(\widetilde{V}_j)$ and $\lambda_{\min}(\widetilde{V}_j)$ are the largest and smallest eigenvalues of $\widetilde{V}_j$ respectively.*

**Theorem 3.4.** *Suppose that problem (65) satisfies the generalized Slater condition (12). Then Algorithm 2 is well defined and any accumulation point $\hat{y}$ of $\{y^j\}$ generated by Algorithm 2 is an optimal solution to the inner problem (51).*

*Proof.* By Step 1.1 in Algorithm 2, for any $j \ge 0$, since, by (75), $d^j$ is a descent direction, Algorithm 2 is well defined. Since problem (65) satisfies the generalized Slater condition (12), from [29, Theorems 17 & 18], we know that the level set $\mathcal{L} := \{y \in \Re^m \mid \varphi(y) \le \varphi(y^0)\}$ is a closed and bounded convex set. Therefore, the sequence $\{y^j\}$ is bounded. Let $\hat{y}$ be any accumulation point of $\{y^j\}$.

Assume for the purpose of contradiction that $\hat{y}$ is not a solution. Then, $\nabla\varphi(\hat{y}) \neq 0$. Let $\{y^{j_l}\}$ be a subsequence converging to $\hat{y}$. Then, there exists a constant $\gamma > 0$ such that

$$\|\nabla\varphi(y^{j_l})\| \geq \gamma \quad \text{for all } l \text{ sufficiently large.} \tag{76}$$

From (74), we have

$$\varphi(y^{j_{l+1}}) \leq \cdots \leq \varphi(y^{j_l+1}) \leq \varphi(y^{j_l}) + \mu\alpha_{j_l}\langle\nabla\varphi(y^{j_l}), d^{j_l}\rangle.$$

Since $\varphi(y^{j_{l+1}}) - \varphi(y^{j_l}) \to 0$, we obtain that

$$\alpha_{j_l}\langle\nabla\varphi(y^{j_l}), d^{j_l}\rangle \to 0 \quad \text{as} \quad l \to \infty. \tag{77}$$

From (76) and the definition of $\varepsilon_{j_l}$ in Algorithm 2, we know that $\|(V_{j_l} + \varepsilon_{j_l}I)^{-1}\|$ is uniformly bounded. From the definition of $\eta_{j_l}$, there exist two constants $o < c_1 \leq c_2 < +\infty$ such that

$$c_1 \leq \|d^{j_l}\| \leq c_2. \tag{78}$$

Thus, by Proposition 3.3, there exists a constant $c_3 > 0$ such that

$$\limsup_{l\to\infty}\langle\varphi(y^{j_l}), d^{j_l}\rangle \leq -c_3 < 0, \tag{79}$$

which, together with (77), implies

$$\alpha_{j_l} \to 0. \tag{80}$$

On the other hand, by (74), we know that for all $l$ sufficiently large,

$$\varphi(y^{j_l} + (\alpha_{j_l}/\delta)d^{j_l}) - \varphi(y^{j_l}) > \mu(\alpha_{j_l}/\delta)\langle\nabla\varphi(y^{j_l}), d^{j_l}\rangle,$$

i.e.,

$$\frac{\varphi(y^{j_l} + (\alpha_{j_l}/\delta)d^{j_l}) - \varphi(y^{j_l})}{\alpha_{j_l}/\delta} > \mu\langle\nabla\varphi(y^{j_l}), d^{j_l}\rangle. \tag{81}$$

By using the mean value theorem, we can write (81) as

$$\int_0^1 \langle\nabla\varphi(y^{j_l} + t(\alpha_{j_l}/\delta)d^{j_l}), d^{j_l}\rangle\, dt > \mu\langle\nabla\varphi(y^{j_l}), d^{j_l}\rangle.$$

Since $\Pi_{\mathcal{S}^n_+}(\cdot)$ is Lipschitz continuous, one has

$$\|\nabla\varphi(z) - \nabla\varphi(\tilde{z})\| \leq \sigma\|\mathcal{A}\|\|\mathcal{A}^*\|\|z - \tilde{z}\| \quad \forall\ z, \tilde{z} \in \Re^m. \tag{82}$$

Then, for all $l$ sufficient large, we have

$$\begin{aligned}(\mu - 1)\langle\nabla\varphi(y^{j_l}), d^{j_l}\rangle &< \int_0^1 \langle\nabla\varphi(y^{j_l} + t(\alpha_{j_l}/\delta)d^{j_l}) - \nabla\varphi(y^{j_l}), d^{j_l}\rangle dt \\ &\leq \sigma(\alpha_{j_l}/\delta)\|\mathcal{A}\|\|\mathcal{A}^*\|\|d^{j_l}\|^2,\end{aligned}$$

which, together with (79), implies

$$\alpha_{j_l} > \delta \frac{(\mu - 1)\langle \nabla \varphi(y^{j_l}), d^{j_l} \rangle}{\sigma \|\mathcal{A}\| \|\mathcal{A}^*\| \|d^{j_l}\|^2} \geq \delta \frac{(1-\mu)c_3}{\sigma \|\mathcal{A}\| \|\mathcal{A}^*\| c_2^2} > 0.$$

This contradicts (80). Consequently, we obtain

$$\nabla \varphi(\hat{y}) = \lim_{j_l \to \infty} \nabla \varphi(y^{j_l}) = 0.$$

By the convexity of $\varphi(\cdot)$, $\hat{y}$ is an optimal solution of (51). $\square$

Next we shall discuss the rate of convergence of Algorithm 2.

**Theorem 3.5.** *Assume that problem (65) satisfies the generalized Slater condition (12). Let $\hat{y}$ be an accumulation point of the infinite sequence $\{y^j\}$ generated by Algorithm 2 for solving the inner problem (51). Suppose that at each step $j \geq 0$, when the practical CG Algorithm 1 terminates, the tolerance $\eta_j$ is achieved (e.g., when $n_j = m+1$), i.e.,*

$$\|\nabla \varphi(y^j) + (V_j + \varepsilon_j I)\, d^j\| \leq \eta_j. \tag{83}$$

*Assume that the constraint nondegenerate condition (67) holds at $\widehat{Z} := \Pi_{\mathcal{S}^n_+}(X - \sigma(\mathcal{A}^*\hat{y} - C))$. Then the whole sequence $\{y^j\}$ converges to $\hat{y}$ and*

$$\|y^{j+1} - \hat{y}\| = O(\|y^j - \hat{y}\|^{1+\tau}). \tag{84}$$

*Proof.* By Theorem 3.4, we know that the infinite sequence $\{y^j\}$ is bounded and $\hat{y}$ is an optimal solution to (51) with

$$\nabla \varphi(\hat{y}) = 0.$$

Since the constraint nondegenerate condition (67) is assumed to hold at $\widehat{Z}$, $\hat{y}$ is the unique optimal solution to (51). It then follows from Theorem 3.4 that $\{y^j\}$ converges to $\hat{y}$. From Proposition 3.2, we know that for any $V_{\hat{y}} \in \hat{\partial}^2 \varphi(\hat{y})$ defined in (58), there exists a $W_{\hat{y}} \in \partial \Pi_{\mathcal{S}^n_+}(X - \sigma(\mathcal{A}^*\hat{y} - C))$ such that

$$V_{\hat{y}} = \sigma \mathcal{A} W_{\hat{y}} \mathcal{A}^* \succ \mathbf{0}.$$

Then, for all $j$ sufficiently large, $\{\|(V_j + \varepsilon_j I)^{-1}\|\}$ is uniformly bounded.

For any $V_j$, $j \geq 0$, there exists a $W_j \in \partial \Pi_{\mathcal{S}^n_+}(X - \sigma(\mathcal{A}^* y^j - C))$ such that

$$V_j = \sigma \mathcal{A} W_j \mathcal{A}^*. \tag{85}$$

Since $\Pi_{\mathcal{S}^n_+}(\cdot)$ is strongly semismooth [35], it holds that for all $j$ sufficiently large,

$$
\begin{aligned}
& \|y^j + d^j - \hat{y}\| \ = \ \|y^j + (V_j + \varepsilon_j I)^{-1}((\nabla\varphi(y^j) + (V_j + \varepsilon_j I)\, d^j) - \nabla\varphi(y^j)) - \hat{y}\| \\
& \le \ \|y^j - \hat{y} - (V_j + \varepsilon_j I)^{-1}\nabla\varphi(y^j)\| + \|(V_j + \varepsilon_j I)^{-1}\|\, \|\nabla\varphi(y^j) + (V_j + \varepsilon_j I)\, d^j\| \\
& \le \ \|(V_j + \varepsilon_j I)^{-1}\|\, \|\nabla\varphi(y^j) - \nabla\varphi(\hat{y}) - V_j(y^j - \hat{y})\| \\
& \quad + \|(V_j + \varepsilon_j I)^{-1}\|(\varepsilon_j\|y^j - \hat{y}\| + \eta_j) \\
& \le \ O(\|\mathcal{A}\|\|\Pi_{\mathcal{S}^n_+}(X - \sigma(\mathcal{A}^* y^j - C)) - \Pi_{\mathcal{S}^n_+}(X - \sigma(\mathcal{A}^*\hat{y} - C)) - W_j(\sigma\mathcal{A}^*(y^j - \hat{y}))\|) \\
& \quad + O(\tau_1\|\nabla\varphi(y^j)\|\|y^j - \hat{y}\| + \|\nabla\varphi(y^j)\|^{1+\tau}) \\
& \le \ O(\|\sigma\mathcal{A}^*(y^j - \hat{y})\|^2) + O(\tau_1\, \|\nabla\varphi(y^j) - \nabla\varphi(\hat{y})\|\|y^j - \hat{y}\| + \|\nabla\varphi(y^j) - \nabla\varphi(\hat{y})\|^{1+\tau}) \\
& \le \ O(\|y^j - \hat{y}\|^2) + O(\tau_1\sigma\|\mathcal{A}\|\|\mathcal{A}^*\|\|y^j - \hat{y}\|^2 + (\sigma\|\mathcal{A}\|\|\mathcal{A}^*\|\|y^j - \hat{y}\|)^{1+\tau}) \\
& = \ O(\|y^j - \hat{y}\|^{1+\tau}),
\end{aligned} \tag{86}
$$

which implies that for all $j$ sufficiently large,

$$
y^j - \hat{y} = -d^j + O(\|d^j\|^{1+\tau}) \quad \text{and} \quad \|d^j\| \to 0. \tag{87}
$$

For each $j \ge 0$, let $R^j := \nabla\varphi(y^j) + (V_j + \varepsilon_j I)\, d^j$. Then, for all $j$ sufficiently large,

$$
\begin{aligned}
-\langle\, \nabla\varphi(y^j), d^j \,\rangle \ &= \ \langle\, d^j, (V_j + \varepsilon_j I)\, d^j \,\rangle - \langle\, R^j, d^j \,\rangle \\
&\ge \ \langle\, d^j, (V_j + \varepsilon_j I)\ d^j \,\rangle - \eta_j\|d^j\| \\
&\ge \ \langle\, d^j, (V_j + \varepsilon_j I)\ d^j \,\rangle - \|d^j\|\|\nabla\varphi(y^j)\|^{1+\tau} \\
&= \ \langle\, d^j, (V_j + \varepsilon_j I)\ d^j \,\rangle - \|\nabla\varphi(y^j) - \nabla\varphi(\hat{y})\|^{1+\tau}\|d^j\| \\
&\ge \ \langle\, d^j, (V_j + \varepsilon_j I)\ d^j \,\rangle - \sigma\|d^j\|\|\mathcal{A}\|\|\mathcal{A}^*\|\|y^j - \hat{y}\|^{1+\tau} \\
&\ge \ \langle\, d^j, (V_j + \varepsilon_j I)\, d^j \,\rangle - O(\|d^j\|^{2+\tau})\,,
\end{aligned}
$$

which, together with (87) and the fact that $\|(V_j + \varepsilon_j I)^{-1}\|$ is uniformly bounded, implies that there exists a constant $\hat{\delta} > 0$ such that

$$
-\langle\, \nabla\varphi(y^j), d^j \,\rangle \ge \hat{\delta}\|d^j\|^2 \qquad \text{for all } j \text{ sufficiently large.}
$$

Since $\nabla\varphi(\cdot)$ is (strongly) semismooth at $\hat{y}$ (because $\Pi_{\mathcal{S}^n_+}(\cdot)$ is strongly semismooth everywhere), from [11, Theorem 3.3 & Remark 3.4] or [19], we know that for $\mu \in (0, 1/2)$, there exists an integer $j_0$ such that for any $j \ge j_0$,

$$
\varphi(y^j + d^j) \le \varphi(y^j) + \mu\langle\nabla\varphi(y^j), d^j\rangle,
$$

which means that for all $j \ge j_0$,

$$
y^{j+1} = y^j + d^j.
$$

This, together with (86), completes the proof. $\square$

Theorem 3.5 shows that the rate of convergence for Algorithm 2 is of order $(1+\tau)$. If $\tau = 1$, this corresponds to quadratic convergence. However, this will need more CG iterations in Algorithm 1. To save computational time, in practice we choose $\tau = 0.1 \sim 0.2$, which still ensures that Algorithm 2 achieves superlinear convergence.

# 4 A Newton-CG Augmented Lagrangian Method

In this section, we shall introduce a Newton-CG augmented Lagrangian algorithm for solving problems $(D)$ and $(P)$. For any $k \geq 0$, denote $\varphi_k(\cdot) \equiv L_{\sigma_k}(\cdot, X^k)$. Since the inner problems can not be solved exactly, we will use the following stopping criteria considered by Rockaffellar [30, 31] for terminating Algorithm 2:

$(A)$ $\varphi_k(y^{k+1}) - \inf \varphi_k \leq \epsilon_k^2/2\sigma_k, \qquad \epsilon_k \geq 0, \ \sum_{k=0}^{\infty} \epsilon_k < \infty.$

$(B)$ $\varphi_k(y^{k+1}) - \inf \varphi_k \leq (\delta_k^2/2\sigma_k)\|X^{k+1} - X^k\|^2, \qquad \delta_k \geq 0, \ \sum_{k=0}^{\infty} \delta_k < \infty.$

$(B')$ $\|\nabla \varphi_k(y^{k+1})\| \leq (\delta_k'/\sigma_k)\|X^{k+1} - X^k\|, \qquad 0 \leq \delta_k' \to 0.$

**Algorithm 3.** Newton-CG Augmented Lagrangian (NCGAL) Algorithm

**Step 0.** Given $(y^0, X^0) \in \Re^m \times \mathcal{S}_+^n$, $\sigma_0 > 0$, a threshold $\bar{\sigma} \geq \sigma_0 > 0$ and $\rho > 1$.

**Step 1.** For $k = 0, 1, 2, \ldots$

Step 1.1. Starting with $y^k$ as the initial point, apply Algorithm 2 to $\varphi_k(\cdot)$ to find $y^{k+1} = \mathrm{NCG}(y^k, X^k, \sigma_k)$ and $X^{k+1} = \Pi_{\mathcal{S}_+^n}(X^k - \sigma_k(\mathcal{A}^* y^{k+1} - C))$ satisfying $(A)$, $(B)$ or $(B')$.

Step 1.2. If $\sigma_k \leq \bar{\sigma}$, $\sigma_{k+1} = \rho\, \sigma_k$ or $\sigma_{k+1} = \sigma_k$.

The global convergence of Algorithm 3 follows from Rockafellar [30, Theorem 1] and [31, Theorem 4] without much difficulty.

**Theorem 4.1.** *Let Algorithm 2 be executed with stopping criterion $(A)$. If (D) satisfies the Slater condition, i.e., if there exists $z^0 \in \Re^m$ such that*

$$\mathcal{A}^* z^0 - C \succ \mathbf{0}, \tag{88}$$

*then the sequence $\{X^k\} \subset \mathcal{S}_+^n$ generated by Algorithm 3 is bounded and $\{X^k\}$ converges to $\overline{X}$, where $\overline{X}$ is some optimal solution to $(P)$, and $\{y^k\}$ is asymptotically minimizing for (D) with* $\max(P) = \inf(D)$.

*If $\{X^k\}$ is bounded and (P) satisfies the generalized Slater condition (12), then the sequence $\{y^k\}$ is also bounded, and all of its accumulation points of the sequence $\{y^k\}$ are optimal solutions to $(D)$.*

Next we state the local linear convergence of the Newton-CG augmented Lagrangian algorithm.

**Theorem 4.2.** *Let Algorithm 2 be executed with stopping criteria (A) and (B). Assume that (D) satisfies the Slater condition (88) and (P) satisfies the Slater condition (12). If the extended strict constraint qualification (26) holds at $\overline{X}$, where $\overline{X}$ is an optimal solution to (P), then the generated sequence $\{X^k\} \subset \mathcal{S}^n_+$ is bounded and $\{X^k\}$ converges to the unique solution $\overline{X}$ with $\max(P) = \min(D)$, and*

$$\|X^{k+1} - \overline{X}\| \leq \theta_k \|X^k - \overline{X}\| \quad \textit{for all } k \textit{ sufficiently large,}$$

*where*

$$\theta_k = \left[a_g(a_g^2 + \sigma_k^2)^{-1/2} + \delta_k\right](1 - \delta_k)^{-1} \to \theta_\infty = a_g(a_g^2 + \sigma_\infty^2)^{-1/2} < 1,\ \sigma_k \to \sigma_\infty\,,$$

*and $a_g$ is the Lipschitz constant of $T_g^{-1}$ at the origin (cf. Proposition 2.1). The conclusions of Theorem 4.1 about $\{y^k\}$ are valid.*

*Moreover, if the stopping criterion $(B')$ is also used and the constraint nondegenerate conditions (41) and (67) hold at $\bar{y}$ and $\overline{X}$, respectively, then in addition to the above conclusions the sequence $\{y^k\} \to \bar{y}$, where $\bar{y}$ is the unique optimal solution to (D), and one has*

$$\|y^{k+1} - \bar{y}\| \leq \theta'_k \|X^{k+1} - X^k\| \quad \textit{for all } k \textit{ sufficiently large,}$$

*where $\theta'_k = a_l(1+\delta'_k)/\sigma_k \to \delta_\infty = a_l/\sigma_\infty$ and $a_l$ is the Lipschitz constant of $T_l^{-1}$ at the origin.*

*Proof.* Conclusions of the first part of Theorem 4.2 follow from the results in [30, Theorem 2] and [31, Theorem 5] combining with Proposition 2.1. By using the fact that $T_l^{-1}$ is Lipschitz continuous near the origion under the assumption that the constraint nondegenerate conditions (41) and (67) hold, respectively, at $\bar{y}$ and $\overline{X}$ [8, Theorem 18], we can directly obtain conclusions of the second part of this theorem from [30, Theorem 2] and [31, Theorem 5]. $\square$

**Remark 7.** Note that in (3) we can also add the term $\frac{1}{2\sigma_k}\|y - y^k\|^2$ to $L_{\sigma_k}(y, X^k)$ such that $L_{\sigma_k}(y, X^k) + \frac{1}{2\sigma_k}\|y - y^k\|^2$ is a strongly convex function. This actually corresponds to the proximal method of multipliers considered in [31, Section 5] for which the $k$-th iteration is given by

$$\begin{cases} y^{k+1} \approx \arg\min\limits_{y \in \Re^m} \left\{ L_{\sigma_k}(y, X^k) + \frac{1}{2\sigma_k}\|y - y^k\|^2 \right\}, \\ X^{k+1} = \Pi_{\mathcal{S}^n_+}(X^k - \sigma_k(\mathcal{A}^* y^{k+1} - C)), \\ \sigma_{k+1} = \rho\sigma_k \text{ or } \sigma_{k+1} = \sigma_k. \end{cases} \tag{89}$$

Convergence analysis for (89) can be conducted in a parallel way as for (3).

# 5 Numerical Issues in the Associated Semismooth Newton-CG Algorithm

In applying Algorithm 2 to solve the inner subproblem (51), the most expensive step is in computing the direction $d$ at a given $y$ from the linear system (73). Thus (73) must be solved as efficiently as possible. Let

$$M := \sigma\, AQ \otimes Q\mathrm{diag}(\mathrm{vec}(\Omega))Q^{\mathrm{T}} \otimes Q^{\mathrm{T}} A^{\mathrm{T}},$$

where $Q$ and $\Omega$ are given as in (60) and (62), respectively. Here $A$ denotes the matrix representation of $\mathcal{A}$ with respect to the standard bases of $\Re^{n\times n}$ and $\Re^m$. The direction $d$ is computed from the following linear system:

$$(M + \varepsilon I)\, d \;=\; -\nabla\varphi(y). \tag{90}$$

To achieve faster convergence rate when applying the CG method to solve (90), one may apply a preconditioner to the system. By observing that the matrix $\Omega$ has elements all in the interval $[0, 1]$ and that the elements in the $(\bar{\gamma}, \bar{\gamma})$ block are all ones, one may simply approximate $\Omega$ by the matrix of ones, and hence a natural preconditioner for the coefficient matrix in (90) is simply the matrix

$$\widehat{M} := \sigma AA^{\mathrm{T}} + \varepsilon I.$$

However, using $\widehat{M}$ as the preconditioner may be costly since it requires the Cholesky factorization of $AA^{\mathrm{T}}$ and each preconditioning step requires the solution of two triangular linear systems. The last statement holds in particular when the Cholesky factor has large number of fill-ins. Thus in our implementation, we simply use $\mathrm{diag}(\widehat{M})$ as the preconditioner rather than $\widehat{M}$.

Next we discuss how to compute the matrix-vector multiplication $Md$ for a given $d \in \Re^m$ efficiently by exploiting the structure of $\Omega$. Observe that $Md = \sigma\mathcal{A}(Y)$, where $Y = Q(\Omega \circ (Q^{\mathrm{T}} DQ))Q^{\mathrm{T}}$ with $D = \mathcal{A}^* d$. Thus the efficient computation of $Md$ relies on our ability to compute the matrix $Y$ efficiently. We have

$$Y \;=\; [Q_{\bar{\gamma}}\; Q_{\gamma}] \begin{bmatrix} Q_{\bar{\gamma}}^{\mathrm{T}} DQ_{\bar{\gamma}} & \nu_{\bar{\gamma}\gamma} \circ (Q_{\bar{\gamma}}^{\mathrm{T}} DQ_{\gamma}) \\ \nu_{\bar{\gamma}\gamma}^{\mathrm{T}} \circ (Q_{\gamma}^{\mathrm{T}} DQ_{\bar{\gamma}}) & 0 \end{bmatrix} \begin{bmatrix} Q_{\bar{\gamma}}^{\mathrm{T}} \\ Q_{\gamma}^{\mathrm{T}} \end{bmatrix} \;=\; H + H^{\mathrm{T}}, \tag{91}$$

where $H = Q_{\bar{\gamma}}\Big[\frac{1}{2}(UQ_{\bar{\gamma}})Q_{\bar{\gamma}}^{\mathrm{T}} + (\nu_{\bar{\gamma}\gamma} \circ (UQ_{\gamma}))Q_{\gamma}^{\mathrm{T}}\Big]$ with $U = Q_{\bar{\gamma}}^{\mathrm{T}} D$. Now it is easy to see that $Y$ can be computed in at most $6|\bar{\gamma}|n^2$ arithmetic operations. The above computational complexity shows that the NCGAL algorithm is able to take advantage of any low-rank property of the optimal solution $\overline{X}$ to reduce computational cost. In contrast, for inexact interior-point methods such as those proposed in [39], the matrix-vector multiplication in each CG iteration would require $\Theta(n^3)$ flops.

## 5.1 Conditioning of $M$

Recall that under the conditions stated in Theorem 4.2 where the sequences $\{y^k\}$ and $\{X^k\}$ generated by Algorithm 3 converge to the solution $\bar{y}$ and $\overline{X}$, respectively. Let

$$\overline{S} = \mathcal{A}^*\bar{y} - C.$$

For simplicity, we assume that strict complementarity holds for $\overline{X}, \overline{S}$, i.e., $\overline{X} + \overline{S} \succ 0$. We also assume that the constraint nondegenerate conditions (41) and (67) hold for $\bar{y}$ and $\overline{X}$, respectively.

We shall now analyse the conditioning of the matrix $M$ corresponding to the pair $(\bar{y}, \overline{X})$. Proposition 3.2 assured that $M$ is positive definite, but to estimate the convergence of the CG method for solving (90), we need to estimate the condition number of $M$.

From the fact that $\overline{X}\,\overline{S} = 0$, we have the following eigenvalue decomposition:

$$\overline{X} - \sigma\overline{S} = Q \begin{bmatrix} \Lambda^X & 0 \\ 0 & -\sigma\Lambda^S \end{bmatrix} Q^{\mathrm{T}}, \tag{92}$$

where $\Lambda^X = \mathrm{diag}(\lambda^X) \in \Re^{r\times r}$ and $\Lambda^S = \mathrm{diag}(\lambda^S) \in \Re^{(n-r)\times(n-r)}$ are diagonal matrices of positive eigenvalues of $\overline{X}$ and $\overline{S}$, respectively. Define the index sets

$$\bar{\gamma} := \{1, \ldots, r\}, \quad \gamma := \{r+1, \ldots, n\}.$$

Let

$$\Omega = \begin{bmatrix} E_{\bar{\gamma}\bar{\gamma}} & \nu_{\bar{\gamma}\gamma} \\ \nu_{\bar{\gamma}\gamma}^{\mathrm{T}} & 0 \end{bmatrix}, \quad \nu_{ij} := \frac{\lambda_i^X}{\lambda_i^X + \sigma\lambda_{j-r}^S}, \; i \in \bar{\gamma}, j \in \gamma, \tag{93}$$

and

$$c_1 \; = \; \frac{\min(\lambda^X)}{\min(\lambda^X)/\sigma + \max(\lambda^S)}, \quad c_2 \; = \; \frac{\max(\lambda^X)}{\max(\lambda^X)/\sigma + \min(\lambda^S)} \; < \; \sigma.$$

Then

$$c_1 \; \leq \; \sigma\nu_{ij} \; \leq \; c_2, \;\; i \in \bar{\gamma}, j \in \gamma.$$

Consider the decomposition in (92) for the pair $(\bar{y}, \overline{X})$ and let $\nu$ be defined as in (93). Then we have

$$M \; = \; \sigma \left( \widetilde{A}_1 \widetilde{A}_1^{\mathrm{T}} + \widetilde{A}_2 D_2 \widetilde{A}_2^{\mathrm{T}} + \widetilde{A}_3 D_3 \widetilde{A}_3^{\mathrm{T}} \right) \tag{94}$$

where $\widetilde{A}_1 = AQ_{\bar{\gamma}} \otimes Q_{\bar{\gamma}}$, $\widetilde{A}_2 = AQ_{\gamma} \otimes Q_{\bar{\gamma}}$, $\widetilde{A}_3 = AQ_{\bar{\gamma}}^{\mathrm{T}} \otimes Q_{\gamma}^{\mathrm{T}}$, $D_2 = \mathrm{diag}(\mathrm{vec}(\nu_{\bar{\gamma}\gamma}))$, and $D_3 = \mathrm{diag}(\mathrm{vec}(\nu_{\bar{\gamma}\gamma}^{\mathrm{T}}))$.
Since

$$c_1 I \; \preceq \; \sigma D_2, \; \sigma D_3 \; \preceq \; c_2 I \; \prec \; \sigma I,$$

it is rather easy to deduce from (94) that

$$c_1 \left( \widetilde{A}_1 \widetilde{A}_1^{\mathrm{T}} + \widetilde{A}_2 \widetilde{A}_2^{\mathrm{T}} + \widetilde{A}_3 \widetilde{A}_3 \right) \; \preceq \; M \; \preceq \; \sigma \left( \widetilde{A}_1 \widetilde{A}_1^{\mathrm{T}} + \widetilde{A}_2 \widetilde{A}_2^{\mathrm{T}} + \widetilde{A}_3 \widetilde{A}_3^{\mathrm{T}} \right).$$

Hence we obtain the following bound on the condition number of $M$:

$$\kappa(M) \; \leq \; \frac{\sigma}{c_1}\, \kappa\Big( \widetilde{A}_1 \widetilde{A}_1^{\mathrm{T}} + \widetilde{A}_2 \widetilde{A}_2^{\mathrm{T}} + \widetilde{A}_3 \widetilde{A}_3^{\mathrm{T}} \Big) \; = \; \frac{\sigma}{c_1}\, \kappa\Big( [\widetilde{A}_1, \widetilde{A}_2, \widetilde{A}_3] \Big)^2. \tag{95}$$

The above upper bound suggests that $\kappa(M)$ can potentially be large if any of the following factors are large: (i) $\sigma$; (ii) $c_1$; (iii) $\kappa([\widetilde{A}_1, \widetilde{A}_2, \widetilde{A}_3])$. Observe that $c_1$ is approximately equal to $\min(\lambda^X)/\max(\lambda^S)$ if $\min(\lambda^X)/\sigma \ll \max(\lambda^S)$. Thus we see that a small ratio in $\min(\lambda^X)/\max(\lambda^S)$ can potentially lead to a large $\kappa(M)$. Similarly, even though the constraint nondegenerate condition (67) states that $\kappa([\widetilde{A}_1, \widetilde{A}_2, \widetilde{A}_3])$ is finite (this is an equivalent condition), its actual value can affect the conditioning of $M$ quite dramatically. In particular, if $\overline{X}$ is nearly degenerate, i.e., $\kappa([\widetilde{A}_1, \widetilde{A}_2, \widetilde{A}_3])$ is large, then $\kappa(M)$ can potentially be very large.

# 6 Numerical Experiments

We implemented the Newton-CG augmented Lagrangian (NCGAL) algorithm in Matlab to solve a variety of large SDP problems with $m$ up to $1,283,258$ and $n$ up to $1,600$ on a PC (Intel Xeon 3.2 GHz with 4G of RAM). We measure the infeasibilities and optimality for the primal and dual problems as follows:

$$R_D = \frac{\|C + S - \mathcal{A}^* y\|}{1 + \|C\|}, \quad R_P = \frac{\|b - \mathcal{A}(X)\|}{1 + \|b\|}, \quad \text{gap} = \frac{b^T y - \langle C, X \rangle}{1 + |b^T y| + |\langle C, X \rangle|}, \tag{96}$$

where $S = (\Pi_{\mathcal{S}^n_+}(W) - W)/\sigma$ with $W = X - \sigma(\mathcal{A}^* y - C)$. In our numerical experiments, we stop the NCGAL algorithm when

$$\max\{R_D, R_P\} \;\leq\; 10^{-6}. \tag{97}$$

We choose the initial iterate $y^0 = 0$, $X^0 = 0$, and $\sigma_0 = 1$.

In solving the subproblem (51), we cap the number of Newton iterations to be 40, while in computing the inexact Newton direction from (73), we stop the CG solver when the maximum number of CG steps exceeds 500, or when the convergence is too slow in that the reduction in the residual norm is exceedingly small.

## 6.1 Random sparse SDPs

We first consider the collection of random sparse SDPs tested in [17], which reported the performance of the boundary point method introduced in [22] for solving large SDPs.

In Table 1, we give the results obtained by the NCGAL algorithm for the sparse SDPs considered in [17]. The first three columns give the problem name, the dimension of the variable $y$ ($m$), the size of the matrix $C$ ($n_s$), and the number of linear inequality constraints ($n_l$) in $(D)$, respectively. The middle three columns give the number of outer iterations taken by the NCGAL algorithm, the objective values $\langle C, X \rangle$ and $b^T y$, respectively. The relative infeasibilities and gap, as well as times (in the format hours:minutes:seconds) are listed in the last four columns.

Table 2 lists the results obtained by the boundary-point method [17] that is coded in the Matlab program `mprw.m` downloaded from F. Rendl's web page. It basically implements the following algorithm: given $\sigma_0 > 0$, $X^0 \in \mathcal{S}^n$, $y^0 \in \Re^m$, accuracy level $\varepsilon$, perform the following loop:

$$W = X^j - \sigma_j(\mathcal{A}^* y^j - C),\ X^{j+1} = \Pi_{\mathcal{S}^n_+}(W),\ S = (X^{j+1} - W)/\sigma_j$$
$$y^{j+1} = y^j - (\sigma_j \mathcal{A}\mathcal{A}^*)^{-1}(b - \mathcal{A}(X^{j+1}))$$
$$R_P = \|b - \mathcal{A}(X^{j+1})\|/(1 + \|b\|),\ R_D = \|C + S - \mathcal{A}^* y^{j+1}\|/(1 + \|C\|)$$

If $\max\{R_P, R_D\} \leq \varepsilon$, stop; else, update $\sigma_j$, end

Suppose that the Cholesky factorization of $\mathcal{A}\mathcal{A}^*$ is pre-computed. Then each iteration of the above algorithm requires the solution of two triangular linear systems and one full eigenvalue decomposition of an $n \times n$ symmetric matrix. Thus each iteration of the algorithm may become rather expensive when the Cholesky factor of $\mathcal{A}\mathcal{A}^*$ is fairly dense or when $n \geq 500$, and the whole algorithm may be very expensive if a large number of iterations is needed to reach the desired accuracy. We should mention that the performance of the boundary-point method is quite sensitive to the choice of $\sigma_0$. In the program `mprw.m`, the authors suggested picking a value in $[0.1, 10]$ if the SDP data is normalized. Note that in our experiments, we set the maximum number of iterations allowed in the boundary-point method to $2,000$.

Comparing the results in Tables 1 and 2, we observe that the performance of the NCGAL algorithm is competitive with the boundary-point method in [17]. It is rather surprising that the boundary-point method, being a gradient based method (cf. Remark 3), can be so efficient in solving this class of sparse random SDPs.

Table 1: Results for the NCGAL algorithm on the random sparse SDPs considered in [17].

| problem | $m \mid n_s; n_l$ | it. | $\langle C, X\rangle$ | $b^T y$ | $R_P$ | $R_D$ | gap | time |
|---|---|---|---|---|---|---|---|---|
| Rn3m20p3 | 20000 \| 300; | 7 | 7.61352497 2 | 7.61372137 2 | 5.3-7 | 7.4-7 | -1.3-5 | 2:26 |
| Rn3m25p3 | 25000 \| 300; | 5 | 7.38384451 1 | 7.38580987 1 | 1.5-8 | 7.5-7 | **-1.3-4** | 4:53 |
| Rn3m10p4 | 10000 \| 300; | 13 | 1.65974613 2 | 1.65997083 2 | 5.6-8 | 7.4-7 | -6.7-5 | 1:11 |
| Rn4m30p3 | 30000 \| 400; | 8 | 1.07214028 3 | 1.07217904 3 | 5.8-7 | 9.4-7 | -1.8-5 | 4:08 |
| Rn4m40p3 | 40000 \| 400; | 7 | 8.05768028 2 | 8.05775429 2 | 1.3-6 | 2.0-7 | -4.6-6 | 10:14 |
| Rn4m15p4 | 15000 \| 400; | 13 | -6.55000170 2 | -6.54970725 2 | 1.1-7 | 7.0-7 | -2.2-5 | 2:34 |
| Rn5m30p3 | 30000 \| 500; | 12 | 1.10762691 3 | 1.10766643 3 | 9.2-8 | 9.2-7 | -1.8-5 | 5:19 |
| Rn5m40p3 | 40000 \| 500; | 10 | 8.16611248 2 | 8.16656409 2 | 4.2-7 | 8.6-7 | -2.8-5 | 5:49 |
| Rn5m50p3 | 50000 \| 500; | 8 | 3.64945649 2 | 3.64985540 2 | 4.7-7 | 7.1-7 | -5.5-5 | 8:38 |
| Rn5m20p4 | 20000 \| 500; | 14 | 3.28004266 2 | 3.28053627 2 | 1.9-7 | 8.6-7 | -7.5-5 | 4:29 |
| Rn6m40p3 | 40000 \| 600; | 12 | 3.06617447 2 | 3.06659549 2 | 7.0-8 | 7.4-7 | -6.9-5 | 8:17 |
| Rn6m50p3 | 50000 \| 600; | 12 | -3.86413346 2 | -3.86355420 2 | 3.7-7 | 8.8-7 | -7.5-5 | 11:12 |
| Rn6m60p3 | 60000 \| 600; | 10 | 6.41737228 2 | 6.41802708 2 | 4.5-7 | 9.1-7 | -5.1-5 | 10:18 |
| Rn6m20p4 | 20000 \| 600; | 14 | 1.04526970 3 | 1.04531196 3 | 9.0-8 | 7.4-7 | -2.0-5 | 6:08 |
| Rn7m50p3 | 50000 \| 700; | 13 | 3.13203126 2 | 3.13247647 2 | 6.9-8 | 6.6-7 | -7.1-5 | 12:11 |
| Rn7m70p3 | 70000 \| 700; | 12 | -3.69558049 2 | -3.69486685 2 | 3.4-7 | 8.6-7 | -9.6-5 | 16:15 |

Table 1: Results for the NCGAL algorithm on the random sparse SDPs considered in [17].

| problem | $m \mid n_s; n_l$ | it. | $\langle C, X\rangle$ | $b^T y$ | $R_P$ | $R_D$ | gap | time |
|---|---|---|---|---|---|---|---|---|
| Rn8m70p3 | 70000 \| 800; | 12 | 2.33139629 3 | 2.33147962 3 | 9.9-8 | 9.1-7 | -1.8-5 | 18:29 |
| Rn8m100p3 | 100000 \| 800; | 11 | 2.25928886 3 | 2.25939692 3 | 5.7-8 | 9.9-7 | -2.4-5 | 27:15 |

Table 2: Results obtained by the boundary-point method in [17] on the random sparse SDPs considered therein. Initial regularization parameter value is set to 0.1, which gives better timings than the default initial value of 1.

| problem | $m \mid n_s; n_l$ | it. | $\langle C, X\rangle$ | $b^T y$ | $R_P$ | $R_D$ | gap | time |
|---|---|---|---|---|---|---|---|---|
| Rn3m20p3 | 20000 \| 300; | 162 | 7.61352301 2 | 7.61351956 2 | 9.9-7 | 3.3-8 | 2.3-7 | 1:28 |
| Rn3m25p3 | 25000 \| 300; | 244 | 7.38383593 1 | 7.38384809 1 | 9.3-7 | 4.8-8 | -8.2-7 | 4:23 |
| Rn3m10p4 | 10000 \| 300; | 148 | 1.65974687 2 | 1.65975074 2 | 9.8-7 | 7.3-8 | -1.2-6 | 1:53 |
| Rn4m30p3 | 30000 \| 400; | 143 | 1.07214130 3 | 1.07213935 3 | 9.6-7 | 2.7-8 | 9.1-7 | 2:31 |
| Rn4m40p3 | 40000 \| 400; | 193 | 8.05769815 2 | 8.05768569 2 | 9.3-7 | 3.3-8 | 7.7-7 | 7:23 |
| Rn4m15p4 | 15000 \| 400; | 168 | -6.55000133 2 | -6.54998597 2 | 9.9-7 | 1.1-7 | -1.2-6 | 4:16 |
| Rn5m30p3 | 30000 \| 500; | 151 | 1.10762655 3 | 1.10762734 3 | 9.9-7 | 8.4-8 | -3.6-7 | 4:23 |
| Rn5m40p3 | 40000 \| 500; | 136 | 8.16610180 2 | 8.16610683 2 | 9.6-7 | 3.6-8 | -3.1-7 | 4:08 |
| Rn5m50p3 | 50000 \| 500; | 149 | 3.64945604 2 | 3.64945078 2 | 9.7-7 | 2.3-8 | 7.2-7 | 6:30 |
| Rn5m20p4 | 20000 \| 500; | 196 | 3.28004579 2 | 3.28010479 2 | 9.9-7 | 2.1-7 | -9.0-6 | 8:07 |
| Rn6m40p3 | 40000 \| 600; | 153 | 3.06617946 2 | 3.06618173 2 | 9.5-7 | 8.0-8 | -3.7-7 | 7:40 |
| Rn6m50p3 | 50000 \| 600; | 142 | -3.86413897 2 | -3.86413511 2 | 9.9-7 | 5.7-8 | -5.0-7 | 7:08 |
| Rn6m60p3 | 60000 \| 600; | 137 | 6.41736718 2 | 6.41736746 2 | 9.9-7 | 3.0-8 | -2.2-8 | 7:47 |
| Rn6m20p4 | 20000 \| 600; | 226 | 1.04526808 3 | 1.04528328 3 | 9.9-7 | 3.9-7 | -7.3-6 | 12:03 |
| Rn7m50p3 | 50000 \| 700; | 165 | 3.13202583 2 | 3.13205602 2 | 9.9-7 | 1.1-7 | -4.8-6 | 12:50 |
| Rn7m70p3 | 70000 \| 700; | 136 | -3.69558765 2 | -3.69558700 2 | 9.9-7 | 4.2-8 | -8.9-8 | 11:04 |
| Rn8m70p3 | 70000 \| 800; | 158 | 2.33139551 3 | 2.33139759 3 | 9.9-7 | 8.3-8 | -4.5-7 | 18:23 |
| Rn8m100p3 | 100000 \| 800; | 135 | 2.25928693 3 | 2.25928781 3 | 9.4-7 | 2.9-8 | -1.9-7 | 17:35 |

## 6.2 SDPs arising from the nearest correlation matrix estimation

Given an $n \times n$ symmetric matrix $\widehat{X}$ where each element is the correlation coefficient estimated from a statistical sample, the nearest correlation matrix (NCM) problem is find a correlation matrix nearest to the estimated data $\widehat{X}$. Mathematically, one version of the NCM problem is the following:

$$\min\Big\{ \sum_{i,j=1}^{n} H_{ij}|X_{ij} - \widehat{X}_{ij}| \;:\; \mathrm{diag}(X) = e,\ X \succeq 0 \Big\}, \tag{98}$$

where $H \in \mathcal{S}^n$ is a given non-negative weight matrix. The problem (98) can be reformulated as the following SDP with $m = n + n(n+1)/2$ equality constraints:

$$\min\Big\{ \sum_{i,j=1}^{n} H_{ij}(Y_{ij}^{+} + Y_{ij}^{-}) \;:\; \mathrm{diag}(X) = e,\ X_{ij} - \widehat{X}_{ij} = Y_{ij}^{+} - Y_{ij}^{-},\ X \succeq 0,\ Y_{ij}^{+}, Y_{ij}^{-} \geq 0 \Big\}. \tag{99}$$

In our experiments, we set the data matrix to be $\widehat{X} = B + 0.05 * E$, where $B$ and $E$ are generated as follows:

```
xx = 10.^(4*[-1:1/(n-1):0]); B = gallery('randcorr',n*xx/sum(xx));
E = 2*rand(n)-1; E = triu(E) + triu(E,1)';
```

The weight matrix $H$ is next generated as follows. We first generate a random symmetric matrix `H0` whose elements are picked from the uniform distribution in $[0.1, 10]$. Then for a given $p \in (0,1)$, we randomly set approximately $n^2p$ elements of `H0` to 100 and another $n^2p$ elements to 0.01 to simulate the situation where some of the elements in $\widehat{X}$ are fixed and some others are unrestricted. The resulting matrix is chosen to be the weight matrix $H$. In our experiments, we set $p = 0.01$ or 0.2.

Observe that for the NCM problems, the SDPs contain non-negative vector variables in addition to positive semidefinite matrix variables. However, it is easy to extend the NCGAL algorithm to accommodate the non-negative variables.

Tables 3 and 4 list the results obtained by the NCGAL algorithm and the boundary point method in [17], respectively. Comparing the results in Tables 3 and 4, we observe that the NCGAL algorithm outperformed the boundary-point method in [17]. While the former can solve the problems to the desired accuracy, the latter did not do so in 2,000 iterations. The results in Table 4 demonstrate a phenomenon that is typical of a purely gradient based method, i.e., it may stagnate well before the required accuracy is achieved.

Table 3: Results for the NCGAL algorithm on the NCM problems (99). The problem names ending with "H1" and "H2" mean that the weight matrices corresponding to $p = 0.01$ and $p = 0.2$ are used, respectively.

| problem | $m \mid n_s; n_l$ | it. | $\langle C, X\rangle$ | $b^T y$ | $R_P$ | $R_D$ | gap | time |
|---|---|---|---|---|---|---|---|---|
| NCM1n200H1 | 20300 \| 200; 40200 | 22 | 1.01351328 3 | 1.01351300 3 | 1.7-8 | 7.3-7 | 1.4-7 | 2:27 |
| NCM1n200H2 | 20300 \| 200; 40200 | 26 | 6.66596615 2 | 6.66593032 2 | 2.7-8 | 8.9-7 | 2.7-6 | 3:14 |
| NCM1n400H1 | 80600 \| 400; 160400 | 24 | 5.28205364 3 | 5.28205306 3 | 1.3-8 | 6.9-7 | 5.6-8 | 14:45 |
| NCM1n400H2 | 80600 \| 400; 160400 | 27 | 3.99273124 3 | 3.99272063 3 | 4.1-8 | 9.3-7 | 1.3-6 | 20:07 |
| NCM1n800H1 | 321200 \| 800; 640800 | 24 | 2.60803179 4 | 2.60803153 4 | 4.0-8 | 9.7-7 | 4.9-8 | 1:21:09 |
| NCM1n800H2 | 321200 \| 800; 640800 | 28 | 2.19237140 4 | 2.19236896 4 | 6.6-8 | 7.8-7 | 5.6-7 | 1:44:43 |

Table 4: Results obtained by the boundary-point method in [17] on the NCM problems (99). Initial regularization parameter value is set to 0.1.

| problem | $m \mid n_s; n_l$ | it. | $\langle C, X\rangle$ | $b^T y$ | $R_P$ | $R_D$ | gap | time |
|---|---|---|---|---|---|---|---|---|
| NCM1n200H1 | 20300 \| 200; 40200 | 2000 | 1.01351445 3 | 1.01351186 3 | 6.9-11 | 4.7-6 | 1.3-6 | 4:49 |
| NCM1n200H2 | 20300 \| 200; 40200 | 2000 | 6.66612708 2 | 6.66574400 2 | 2.5-7 | 5.9-6 | 2.9-5 | 4:41 |
| NCM1n400H1 | 80600 \| 400; 160400 | 2000 | 5.28205506 3 | 5.28205092 3 | 3.1-9 | 3.1-6 | 3.9-7 | 32:24 |
| NCM1n400H2 | 80600 \| 400; 160400 | 2000 | 3.99275970 3 | 3.99267939 3 | 3.8-7 | 4.6-6 | 1.0-5 | 32:09 |
| NCM1n800H1 | 321200 \| 800; 640800 | 2000 | 2.60803220 4 | 2.60803086 4 | 7.7-9 | 2.6-6 | 2.6-7 | 4:02:13 |
| NCM1n800H2 | 321200 \| 800; 640800 | 2000 | 2.19237800 4 | 2.19235525 4 | 2.0-7 | 3.8-6 | 5.2-6 | 3:58:39 |

## 6.3 SDPs arising from relaxation of frequency assignment problems

Here we consider SDPs arising from semidefinite relaxation of frequency assignment problems [10]. The explicit description of the SDP in the form $(P)$ is given in [6, equation (5)].

Tables 5 and 6 list the results obtained by the NCGAL algorithm and the boundary-point method for the SDP relaxation of frequency assignment problems tested in [6], respectively. Just like the NCM problems, the NCGAL algorithm outperformed the boundary-point method.

It is interesting to note that for this collection, the SDP problems (D) and (P) are likely to be both degenerate at the optimal solution $\bar{y}$ and $\overline{X}$, respectively. For example, the problem `fap01` is both primal and dual degenerate in that $\kappa(\widetilde{A}_1) \approx 3.9 \times 10^{12}$ and $\kappa([\widetilde{A}_1, \widetilde{A}_2, \widetilde{A}_3]) \approx 1.4 \times 10^{12}$, where $\widetilde{A}_1, \widetilde{A}_2, \widetilde{A}_3$ are defined as in (94). Similarly, for `fap02`, we have $\kappa(\widetilde{A}_1) \approx 2.3 \times 10^{12}$ and $\kappa([\widetilde{A}_1, \widetilde{A}_2, \widetilde{A}_3]) \approx 1.7 \times 10^{12}$. It is surprising that the NCGAL algorithm can attain the required accuracy within moderate CPU time despite the fact that the problems do not satisfy the constraint nondegeneracy conditions (41) and (67) at the optimal solution $\bar{y}$ and $\overline{X}$.

Table 5: Results for the NCGAL algorithm on the frequency assignment problems.

| problem | $m \mid n_s; n_l$ | it. | $\langle C, X\rangle$ | $b^T y$ | $R_P$ | $R_D$ | gap | time |
|---|---|---|---|---|---|---|---|---|
| fap01 | 1378 \| 52; 1160 | 20 | 3.28830158-2 | 3.28831899-2 | 7.3-8 | 9.0-8 | -1.6-7 | 07 |
| fap02 | 1866 \| 61; 1601 | 22 | 6.97570239-4 | 7.05737906-4 | 3.1-8 | 7.6-7 | -8.2-6 | 06 |
| fap03 | 2145 \| 65; 1837 | 20 | 4.93727016-2 | 4.93733696-2 | 9.9-8 | 4.7-7 | -6.1-7 | 09 |
| fap04 | 3321 \| 81; 3046 | 25 | 1.74827538-1 | 1.74848000-1 | 4.4-8 | 7.9-7 | -1.5-5 | 29 |

Table 5: Results for the NCGAL algorithm on the frequency assignment problems.

| problem | $m \mid n_s; n_l$ | it. | $\langle C, X\rangle$ | $b^T y$ | $R_P$ | $R_D$ | gap | time |
|---|---|---|---|---|---|---|---|---|
| fap05 | 3570 \| 84; 3263 | 22 | 3.08286119-1 | 3.08289836-1 | 3.0-6 | 3.2-7 | -2.3-6 | 1:08 |
| fap06 | 4371 \| 93; 3997 | 24 | 4.59327736-1 | 4.59352104-1 | 1.4-8 | 9.9-7 | -1.3-5 | 46 |
| fap07 | 4851 \| 98; 4139 | 24 | 2.11761788 0 | 2.11763637 0 | 9.6-7 | 9.0-7 | -3.5-6 | 49 |
| fap08 | 7260 \| 120; 6668 | 25 | 2.43627971 0 | 2.43629199 0 | 1.5-8 | 9.5-7 | -2.1-6 | 45 |
| fap09 | 15225 \| 174; 14025 | 25 | 1.07978019 1 | 1.07978290 1 | 2.6-8 | 6.9-7 | -1.2-6 | 1:35 |
| fap10 | 14479 \| 183; 13754 | 26 | 9.68503146-3 | 9.74838846-3 | 1.1-7 | 9.8-7 | -6.2-5 | 3:08 |
| fap11 | 24292 \| 252; 23275 | 27 | 2.97816495-2 | 2.98654526-2 | 1.4-7 | 8.3-7 | -7.9-5 | 6:47 |
| fap12 | 26462 \| 369; 24410 | 27 | 2.73263404-1 | 2.73445756-1 | 4.4-7 | 9.5-7 | **-1.2-4** | 21:12 |

Table 6: Results obtained by the boundary-point method in [17] on the frequency assignment problems. Initial regularization parameter value is set to 1 (better than 0.1).

| problem | $m \mid n_s; n_l$ | it. | $\langle C, X\rangle$ | $b^T y$ | $R_P$ | $R_D$ | gap | time |
|---|---|---|---|---|---|---|---|---|
| fap01 | 1378 \| 52; 1160 | 2000 | 3.49239684-2 | 3.87066984-2 | 5.4-6 | **1.7-4** | **-3.5-3** | 18 |
| fap02 | 1866 \| 61; 1601 | 2000 | 4.06570342-4 | 1.07844848-3 | 1.6-5 | 7.5-5 | **-6.7-4** | 22 |
| fap03 | 2145 \| 65; 1837 | 2000 | 5.02426246-2 | 5.47858318-2 | 1.5-5 | **1.5-4** | **-4.1-3** | 22 |
| fap04 | 3321 \| 81; 3046 | 2000 | 1.77516830-1 | 1.84285835-1 | 4.5-6 | **1.7-4** | **-5.0-3** | 35 |
| fap05 | 3570 \| 84; 3263 | 2000 | 3.11422846-1 | 3.18992969-1 | 1.1-5 | **1.6-4** | **-4.6-3** | 37 |
| fap06 | 4371 \| 93; 3997 | 2000 | 4.60368585-1 | 4.64270062-1 | 7.5-6 | 9.8-5 | **-2.0-3** | 45 |
| fap07 | 4851 \| 98; 4139 | 2000 | 2.11768050 0 | 2.11802220 0 | 2.5-6 | 1.5-5 | -6.5-5 | 40 |
| fap08 | 7260 \| 120; 6668 | 2000 | 2.43638729 0 | 2.43773801 0 | 2.6-6 | 3.5-5 | **-2.3-4** | 1:08 |
| fap09 | 15225 \| 174; 14025 | 2000 | 1.07978252 1 | 1.07982903 1 | 9.2-7 | 9.8-6 | -2.1-5 | 2:33 |
| fap10 | 14479 \| 183; 13754 | 2000 | 1.70252739-2 | 2.38972400-2 | 1.1-5 | **1.1-4** | **-6.6-3** | 3:49 |
| fap11 | 24292 \| 252; 23275 | 2000 | 4.22711513-2 | 5.94650102-2 | 8.8-6 | **1.4-4** | **-1.6-2** | 8:48 |
| fap12 | 26462 \| 369; 24410 | 2000 | 2.93446247-1 | 3.26163363-1 | 6.0-6 | **1.5-4** | **-2.0-2** | 21:25 |

## 6.4 SDPs arising from relaxation of maximum stable set problems

For a graph $G$ with edge set $\mathcal{E}$, the stability number $\alpha(G)$ is the cardinality of a maximal stable set of $G$, and $\alpha(G) := \{e^T x \,:\, x_i x_j = 0, (i,j) \in \mathcal{E}, x \in \{0,1\}^n\}$. It is known that $\alpha(G) \leq \theta(G) \leq \theta_+(G)$, where

$$\theta(G) \;=\; \max\{\langle ee^T,\, X\rangle \,:\, \langle E_{ij},\, X\rangle = 0, (i,j) \in \mathcal{E},\ \langle I,\, X\rangle = 1,\ X \succeq 0\}, \tag{100}$$

$$\theta_+(G) \;=\; \max\{\langle ee^T,\, X\rangle \,:\, \langle E_{ij},\, X\rangle = 0, (i,j) \in \mathcal{E},\ \langle I,\, X\rangle = 1,\ X \succeq 0,\ X \geq 0\}, \tag{101}$$

where $E_{ij} = e_i e_j^{\mathrm{T}} + e_j e_i^{\mathrm{T}}$ and $e_i$ denotes column $i$ of the identity matrix $I$. Note that for (101), the problem is reformulated as a standard SDP by replacing the constraint $X \geq 0$ by constraints $X - Y = 0$ and $Y \geq 0$. Thus such a reformulation introduces an additional $n(n+1)/2$ linear equality constraints to the SDP.

Table 7 lists the results obtained by the NCGAL algorithm for the SDPs (100) arising from computing $\theta(G)$ for the maximum stable set problems. The first collection of graph instances in Table 7 are the randomly generated instances considered in [39] whereas the second collection is from the Second DIMACS Challenge on Maximum Clique Problems [40]. The last collection are graphs arising from coding theory, available from N. Sloane's web page [33].

Observe that the NCGAL algorithm is not able to achieve the required accuracy level for some of the SDPs from Sloane's collection. It is not surprising that this may happen because many of these SDPs are degenerate at the optimal solution. For example, the problems `1dc.128` and `2dc.128` are degenerate at the optimal solutions $\bar{y}$ even though they are nondegenerate at the optimal solutions $\overline{X}$.

In [17], the performance of the boundary-point method was compared with that of the iterative solver based primal-dual interior-point method in [39], as well as the iterative solver based modified barrier method in [15], on a subset of the large SDPs arising from the first collection of random graphs. The conclusion was that the boundary-point method was between 5-10 times faster than the methods in [39] and [15]. Since the NCGAL algorithm is at least as efficient as the boundary-point method on the theta problems for random graphs (not reported here in the interest of saving space), it is safe to assume that the NCGAL algorithm would be at least 5-10 times faster than the methods in [39] and [15]. Note that the NCGAL algorithm is more efficient than the boundary-point method on the collection of graphs from DIMACS. For example, the NCGAL algorithm takes less than 900 seconds to solve the problem `G43` to an accuracy of less than $10^{-6}$, while the boundary-point method takes more than 25,000 seconds to achieve an accuracy of $7.0 \times 10^{-6}$. Such a result for `G43` is not surprising because the rank of the optimal $X$ is much smaller than $n$, and as already mentioned in [22], the boundary-point method typically would perform poorly under such a situation.

Table 7: Results for the NCGAL algorithm on computing $\theta(G)$ in (100) for the maximum stable set problems.

| problem | $m \mid n_s; n_l$ | it. | $\langle C, X \rangle$ | $b^T y$ | $R_P$ | $R_D$ | gap | time |
|---|---|---|---|---|---|---|---|---|
| theta4 | 1949 \| 200; | 22 | 5.03212185 1 | 5.03212127 1 | 6.0-8 | 6.6-7 | 5.7-8 | 12 |
| theta42 | 5986 \| 200; | 15 | 2.39317083 1 | 2.39317076 1 | 1.1-8 | 3.8-7 | 1.4-8 | 11 |
| theta6 | 4375 \| 300; | 23 | 6.34770837 1 | 6.34770800 1 | 4.5-8 | 4.6-7 | 2.9-8 | 31 |
| theta62 | 13390 \| 300; | 20 | 2.96412503 1 | 2.96412489 1 | 6.9-8 | 4.9-7 | 2.3-8 | 32 |
| theta8 | 7905 \| 400; | 20 | 7.39535673 1 | 7.39535551 1 | 7.5-8 | 8.1-7 | 8.2-8 | 1:06 |
| theta82 | 23872 \| 400; | 16 | 3.43668925 1 | 3.43668903 1 | 1.3-7 | 8.3-7 | 3.2-8 | 1:08 |
| theta83 | 39862 \| 400; | 20 | 2.03018906 1 | 2.03018893 1 | 9.3-9 | 3.8-7 | 3.1-8 | 1:22 |
| theta10 | 12470 \| 500; | 22 | 8.38059686 1 | 8.38059581 1 | 6.0-8 | 5.7-7 | 6.2-8 | 2:11 |

Table 7: Results for the NCGAL algorithm on computing $\theta(G)$ in (100) for the maximum stable set problems.

| problem | $m \mid n_s; n_l$ | it. | $\langle C, X\rangle$ | $b^T y$ | $R_P$ | $R_D$ | gap | time |
|---|---|---|---|---|---|---|---|---|
| theta102 | 37467 \| 500; | 17 | 3.83905460 1 | 3.83905448 1 | 5.3-8 | 3.6-7 | 1.5-8 | 2:10 |
| theta103 | 62516 \| 500; | 17 | 2.25285688 1 | 2.25285671 1 | 3.9-8 | 5.6-7 | 3.7-8 | 2:26 |
| theta104 | 87245 \| 500; | 10 | 1.33361408 1 | 1.33361365 1 | 5.6-8 | 9.9-7 | 1.5-7 | 2:44 |
| theta12 | 17979 \| 600; | 17 | 9.28016826 1 | 9.28016673 1 | 1.3-7 | 9.2-7 | 8.2-8 | 3:10 |
| theta123 | 90020 \| 600; | 12 | 2.46686512 1 | 2.46686467 1 | 1.6-7 | 7.9-7 | 9.0-8 | 3:36 |
| theta162 | 127600 \| 800; | 14 | 3.70097361 1 | 3.70097344 1 | 2.2-8 | 6.9-7 | 2.3-8 | 8:59 |
| MANN-a27 | 703 \| 378; | 10 | 1.32762891 2 | 1.32762883 2 | 7.3-11 | 2.4-7 | 2.8-8 | 19 |
| johnson8-4 | 561 \| 70; | 4 | 1.40000001 1 | 1.39999998 1 | 1.7-9 | 1.8-8 | 1.1-8 | 01 |
| johnson16- | 1681 \| 120; | 4 | 8.00000503 0 | 7.99999724 0 | 3.1-8 | 4.0-7 | 4.6-7 | 01 |
| san200-0.7 | 5971 \| 200; | 12 | 2.99999996 1 | 2.99998917 1 | 1.3-8 | 7.9-7 | 1.8-6 | 09 |
| c-fat200-1 | 18367 \| 200; | 9 | 1.20000000 1 | 1.19999990 1 | 3.4-9 | 2.1-7 | 3.8-8 | 20 |
| hamming-6- | 1313 \| 64; | 3 | 5.33333334 0 | 5.33333092 0 | 2.4-10 | 4.7-7 | 2.1-7 | 00 |
| hamming-8- | 11777 \| 256; | 5 | 1.59999977 1 | 1.59999854 1 | 9.7-9 | 8.1-7 | 3.7-7 | 07 |
| hamming-9- | 2305 \| 512; | 6 | 2.24000000 2 | 2.24000051 2 | 1.9-10 | 2.5-7 | -1.1-7 | 35 |
| hamming-10 | 23041 \| 1024; | 10 | 1.02400127 2 | 1.02399940 2 | 4.1-8 | 6.0-7 | 9.1-7 | 5:57 |
| hamming-7- | 1793 \| 128; | 4 | 4.26666742 1 | 4.26666565 1 | 5.6-8 | 3.0-7 | 2.1-7 | 02 |
| hamming-8- | 16129 \| 256; | 4 | 2.56000007 1 | 2.55999959 1 | 3.1-9 | 2.1-7 | 9.2-8 | 06 |
| hamming-9- | 53761 \| 512; | 4 | 8.53332207 1 | 8.53333472 1 | 2.4-7 | 2.2-7 | -7.4-7 | 30 |
| brock200-1 | 5067 \| 200; | 14 | 2.74566407 1 | 2.74566365 1 | 1.2-7 | 6.9-7 | 7.7-8 | 11 |
| brock200-4 | 6812 \| 200; | 16 | 2.12934757 1 | 2.12934729 1 | 1.1-7 | 5.8-7 | 6.6-8 | 11 |
| brock400-1 | 20078 \| 400; | 18 | 3.97018968 1 | 3.97018927 1 | 1.2-7 | 9.2-7 | 5.1-8 | 1:07 |
| keller4 | 5101 \| 171; | 15 | 1.40122417 1 | 1.40122389 1 | 9.5-9 | 3.6-7 | 9.5-8 | 07 |
| p-hat300-1 | 33918 \| 300; | 19 | 1.00679653 1 | 1.00679608 1 | 1.8-7 | 5.5-7 | 2.1-7 | 5:17 |
| G43 | 9991 \| 1000; | 16 | 2.80624592 2 | 2.80624537 2 | 7.1-8 | 9.9-7 | 9.9-8 | 13:57 |
| G44 | 9991 \| 1000; | 17 | 2.80583201 2 | 2.80583187 2 | 3.3-8 | 4.8-7 | 2.5-8 | 14:12 |
| G45 | 9991 \| 1000; | 17 | 2.80185128 2 | 2.80185103 2 | 3.1-8 | 5.0-7 | 4.4-8 | 14:26 |
| G46 | 9991 \| 1000; | 17 | 2.79836952 2 | 2.79836940 2 | 3.4-8 | 4.1-7 | 2.1-8 | 14:55 |
| G47 | 9991 \| 1000; | 17 | 2.81893965 2 | 2.81893932 2 | 3.5-8 | 5.3-7 | 5.8-8 | 14:28 |
| 1dc.64 | 544 \| 64; | 22 | 1.00000005 1 | 9.99999143 0 | 1.2-7 | 5.3-7 | 4.3-7 | 08 |
| 1et.64 | 265 \| 64; | 10 | 1.88000007 1 | 1.88000074 1 | 8.2-8 | 4.1-7 | -1.7-7 | 01 |
| 1tc.64 | 193 \| 64; | 14 | 1.99999993 1 | 1.99999963 1 | 5.0-8 | 4.3-7 | 7.2-8 | 01 |
| 1dc.128 | 1472 \| 128; | 27 | 1.68422596 1 | 1.68420034 1 | 5.7-6 | 2.7-7 | 7.4-6 | 49 |
| 1et.128 | 673 \| 128; | 14 | 2.92308886 1 | 2.92308947 1 | 7.4-8 | 7.6-7 | -1.0-7 | 03 |
| 1tc.128 | 513 \| 128; | 13 | 3.80000011 1 | 3.80000033 1 | 2.5-8 | 2.6-7 | -2.9-8 | 04 |
| 1zc.128 | 1121 \| 128; | 9 | 2.06666655 1 | 2.06666724 1 | 2.6-8 | 3.1-7 | -1.6-7 | 03 |
| 1dc.256 | 3840 \| 256; | 22 | 3.00000001 1 | 2.99999402 1 | 2.3-8 | 2.6-7 | 9.8-7 | 1:32 |
| 1et.256 | 1665 \| 256; | 21 | 5.51142505 1 | 5.51142454 1 | 8.7-8 | 9.1-7 | 4.6-8 | 2:09 |
| 1tc.256 | 1313 \| 256; | 27 | 6.33998886 1 | 6.33998726 1 | 1.1-7 | 9.7-7 | 1.3-7 | 3:49 |
| 1zc.256 | 2817 \| 256; | 11 | 3.80000006 1 | 3.80000057 1 | 1.8-8 | 7.0-7 | -6.6-8 | 10 |
| 1dc.512 | 9728 \| 512; | 31 | 5.30312348 1 | 5.30307683 1 | 2.6-6 | 4.1-7 | 4.4-6 | 16:53 |

Table 7: Results for the NCGAL algorithm on computing $\theta(G)$ in (100) for the maximum stable set problems.

| problem | $m \mid n_s; n_l$ | it. | $\langle C, X\rangle$ | $b^T y$ | $R_P$ | $R_D$ | gap | time |
|---|---|---|---|---|---|---|---|---|
| 1et.512 | 4033 \| 512; | 20 | 1.04424039 2 | 1.04424021 2 | 1.2-7 | 7.7-7 | 8.8-8 | 8:57 |
| 1tc.512 | 3265 \| 512; | 25 | 1.13412461 2 | 1.13402968 2 | 2.0-5 | 3.6-7 | 4.2-5 | 23:42 |
| 1zc.512 | 6913 \| 512; | 12 | 6.87499996 1 | 6.87500142 1 | 5.8-8 | 3.6-7 | -1.1-7 | 1:24 |
| 1dc.1024 | 24064 \| 1024; | 27 | 9.59853065 1 | 9.59849166 1 | 1.3-6 | 1.5-7 | 2.0-6 | 1:36:20 |
| 1et.1024 | 9601 \| 1024; | 24 | 1.84227103 2 | 1.84226246 2 | 1.6-6 | 2.0-7 | 2.3-6 | 1:53:13 |
| 1tc.1024 | 7937 \| 1024; | 31 | 2.06308842 2 | 2.06304928 2 | 4.2-6 | 4.2-7 | 9.5-6 | 3:11:12 |
| 1zc.1024 | 16641 \| 1024; | 13 | 1.28666662 2 | 1.28666627 2 | 1.3-8 | 6.8-7 | 1.4-7 | 9:50 |

Table 8: Results for the NCGAL algorithm on computing $\theta_+(G)$ in (101) for the maximum stable set problems.

| problem | $m - n_l \mid n_s; n_l$ | it. | $\langle C, X\rangle$ | $b^T y$ | $R_P$ | $R_D$ | gap | time |
|---|---|---|---|---|---|---|---|---|
| theta4 | 1949 \| 200; 20100 | 17 | 4.98690829 1 | 4.98690202 1 | 7.7-7 | 9.7-7 | 6.2-7 | 1:56 |
| theta42 | 5986 \| 200; 20100 | 17 | 2.37382147 1 | 2.37382087 1 | 2.2-7 | 6.7-7 | 1.2-7 | 1:13 |
| theta6 | 4375 \| 300; 45150 | 16 | 6.29618619 1 | 6.29618467 1 | 1.7-7 | 8.6-7 | 1.2-7 | 2:49 |
| theta62 | 13390 \| 300; 45150 | 15 | 2.93779454 1 | 2.93779426 1 | 8.1-8 | 4.4-7 | 4.7-8 | 2:11 |
| theta8 | 7905 \| 400; 80200 | 13 | 7.34078477 1 | 7.34078472 1 | 6.1-8 | 4.3-7 | 3.2-9 | 4:55 |
| theta82 | 23872 \| 400; 80200 | 15 | 3.40643532 1 | 3.40643458 1 | 8.2-8 | 4.0-7 | 1.1-7 | 5:24 |
| theta83 | 39862 \| 400; 80200 | 12 | 2.01671100 1 | 2.01671061 1 | 8.3-8 | 4.3-7 | 9.4-8 | 5:33 |
| theta10 | 12470 \| 500; 125250 | 14 | 8.31490568 1 | 8.31489958 1 | 1.6-7 | 4.2-7 | 3.6-7 | 13:04 |
| theta102 | 37467 \| 500; 125250 | 14 | 3.80663768 1 | 3.80662531 1 | 3.6-7 | 8.9-7 | 1.6-6 | 12:43 |
| theta103 | 62516 \| 500; 125250 | 13 | 2.23774312 1 | 2.23774226 1 | 1.4-7 | 6.8-7 | 1.9-7 | 10:50 |
| theta104 | 87245 \| 500; 125250 | 10 | 1.32826175 1 | 1.32826105 1 | 1.3-7 | 7.4-7 | 2.6-7 | 12:16 |
| theta12 | 17979 \| 600; 180300 | 12 | 9.20908888 1 | 9.20908747 1 | 1.4-7 | 6.9-7 | 7.6-8 | 17:57 |
| theta123 | 90020 \| 600; 180300 | 13 | 2.44952087 1 | 2.44951528 1 | 2.1-7 | 7.4-7 | 1.1-6 | 21:03 |
| theta162 | 127600 \| 800; 320400 | 14 | 3.67114567 1 | 3.67113758 1 | 1.6-7 | 5.0-7 | 1.1-6 | 59:35 |
| MANN-a27 | 703 \| 378; 71631 | 9 | 1.32762891 2 | 1.32762891 2 | 2.4-8 | 6.1-7 | -2.5-10 | 53 |
| johnson8-4 | 561 \| 70; 2485 | 6 | 1.39999999 1 | 1.40000020 1 | 2.0-8 | 1.1-7 | -7.4-8 | 01 |
| johnson16- | 1681 \| 120; 7260 | 9 | 8.00000006 0 | 8.00000001 0 | 2.4-9 | 2.0-9 | 3.1-9 | 03 |
| san200-0.7 | 5971 \| 200; 20100 | 15 | 3.00000002 1 | 2.99999995 1 | 9.4-9 | 1.3-7 | 1.1-8 | 14 |
| c-fat200-1 | 18367 \| 200; 20100 | 8 | 1.20000008 1 | 1.19999969 1 | 5.2-8 | 4.3-7 | 1.5-7 | 52 |
| hamming-6- | 1313 \| 64; 2080 | 6 | 3.99999951 0 | 3.99999987 0 | 5.4-9 | 1.7-8 | -4.1-8 | 01 |
| hamming-8- | 11777 \| 256; 32896 | 9 | 1.59999998 1 | 1.59999959 1 | 8.7-9 | 1.2-7 | 1.2-7 | 11 |
| hamming-9- | 2305 \| 512; 131328 | 5 | 2.24000002 2 | 2.23999908 2 | 4.6-9 | 6.7-7 | 2.1-7 | 46 |
| hamming-10 | 23041 \| 1024; 524800 | 12 | 8.53330819 1 | 8.53333509 1 | 4.7-8 | 1.5-7 | -1.6-6 | 14:44 |
| hamming-7- | 1793 \| 128; 8256 | 16 | 3.60000009 1 | 3.59999952 1 | 4.2-9 | 1.0-7 | 7.7-8 | 07 |
| hamming-8- | 16129 \| 256; 32896 | 8 | 2.55999998 1 | 2.56000001 1 | 1.4-9 | 3.3-9 | -5.7-9 | 10 |

Table 8: Results for the NCGAL algorithm on computing $\theta_+(G)$ in (101) for the maximum stable set problems.

| problem | $m - n_l \mid n_s; n_l$ | it. | $\langle C, X \rangle$ | $b^T y$ | $R_P$ | $R_D$ | gap | time |
|---|---|---|---|---|---|---|---|---|
| hamming-9- | 53761 \| 512; 131328 | 13 | 5.86666672 1 | 5.86666819 1 | 2.5-8 | 2.7-7 | -1.2-7 | 1:31 |
| brock200-1 | 5067 \| 200; 20100 | 19 | 2.71967235 1 | 2.71967185 1 | 1.6-7 | 6.4-7 | 9.0-8 | 55 |
| brock200-4 | 6812 \| 200; 20100 | 14 | 2.11210825 1 | 2.11210791 1 | 1.8-7 | 9.2-7 | 8.0-8 | 48 |
| brock400-1 | 20078 \| 400; 80200 | 14 | 3.93309356 1 | 3.93309238 1 | 1.1-7 | 4.9-7 | 1.5-7 | 5:32 |
| keller4 | 5101 \| 171; 14706 | 16 | 1.34659300 1 | 1.34659070 1 | 9.7-7 | 5.7-7 | 8.2-7 | 2:21 |
| p-hat300-1 | 33918 \| 300; 45150 | 19 | 1.00202225 1 | 1.00202024 1 | 3.3-7 | 6.3-7 | 9.6-7 | 14:28 |
| G43 | 9991 \| 1000; 500500 | 10 | 2.79735986 2 | 2.79735976 2 | 3.0-8 | 4.1-7 | 1.9-8 | 2:56:14 |
| G44 | 9991 \| 1000; 500500 | 10 | 2.79746098 2 | 2.79746109 2 | 7.5-8 | 7.4-7 | -2.0-8 | 2:09:01 |
| G45 | 9991 \| 1000; 500500 | 10 | 2.79317554 2 | 2.79317557 2 | 1.2-7 | 7.5-7 | -6.3-9 | 2:08:45 |
| G46 | 9991 \| 1000; 500500 | 11 | 2.79032524 2 | 2.79032487 2 | 5.5-8 | 3.1-7 | 6.7-8 | 4:41:11 |
| G47 | 9991 \| 1000; 500500 | 11 | 2.80891718 2 | 2.80891718 2 | 1.6-8 | 4.5-7 | -9.1-10 | 2:16:17 |
| 1dc.64 | 544 \| 64; 2080 | 12 | 9.99999231 0 | 9.99998811 0 | 6.1-7 | 6.1-7 | 2.0-7 | 10 |
| 1et.64 | 265 \| 64; 2080 | 17 | 1.87999999 1 | 1.87999820 1 | 1.5-8 | 6.2-7 | 4.6-7 | 02 |
| 1tc.64 | 193 \| 64; 2080 | 12 | 2.00000037 1 | 1.99999909 1 | 2.6-7 | 8.2-7 | 3.1-7 | 05 |
| 1dc.128 | 1472 \| 128; 8256 | 28 | 1.66795621 1 | 1.66782990 1 | 4.5-5 | 2.5-7 | 3.7-5 | 3:37 |
| 1et.128 | 673 \| 128; 8256 | 14 | 2.92308944 1 | 2.92308883 1 | 1.3-7 | 8.4-7 | 1.0-7 | 10 |
| 1tc.128 | 513 \| 128; 8256 | 15 | 3.79999996 1 | 3.80000004 1 | 7.1-8 | 4.1-7 | -1.0-8 | 12 |
| 1zc.128 | 1121 \| 128; 8256 | 13 | 2.06666660 1 | 2.06666462 1 | 4.6-8 | 7.7-7 | 4.7-7 | 05 |
| 1dc.256 | 3840 \| 256; 32896 | 23 | 2.99999999 1 | 3.00000249 1 | 9.5-9 | 1.3-7 | -4.1-7 | 4:33 |
| 1et.256 | 1665 \| 256; 32896 | 28 | 5.44973478 1 | 5.44696431 1 | **2.2-4** | 7.2-8 | **2.5-4** | 13:51 |
| 1tc.256 | 1313 \| 256; 32896 | 32 | 6.32846828 1 | 6.32431265 1 | **3.9-4** | 2.8-7 | **3.3-4** | 17:09 |
| 1zc.256 | 2817 \| 256; 32896 | 17 | 3.73333377 1 | 3.73333069 1 | 7.7-8 | 9.7-7 | 4.1-7 | 29 |
| 1dc.512 | 9728 \| 512; 131328 | 27 | 5.26953558 1 | 5.26951317 1 | 9.9-7 | 4.3-7 | 2.1-6 | 1:12:31 |
| 1et.512 | 4033 \| 512; 131328 | 23 | 1.03588990 2 | 1.03552157 2 | **1.1-4** | 5.0-7 | **1.8-4** | 1:13:02 |
| 1tc.512 | 3265 \| 512; 131328 | 30 | 1.12772823 2 | 1.12550272 2 | **4.8-4** | 4.1-7 | **9.8-4** | 1:50:52 |
| 1zc.512 | 6913 \| 512; 131328 | 13 | 6.79999947 1 | 6.80000261 1 | 9.3-8 | 6.3-7 | -2.3-7 | 10:09 |
| 1dc.1024 | 24064 \| 1024; 524800 | 27 | 9.55566272 1 | 9.55513337 1 | 9.7-6 | 3.2-7 | 2.8-5 | 5:20:48 |
| 1et.1024 | 9601 \| 1024; 524800 | 23 | 1.82230315 2 | 1.82078294 2 | 6.7-5 | 4.9-7 | **4.2-4** | 8:42:01 |
| 1tc.1024 | 7937 \| 1024; 524800 | 30 | 2.04572051 2 | 2.04236411 2 | **1.8-4** | 4.5-7 | **8.2-4** | 11:08:56 |
| 1zc.1024 | 16641 \| 1024; 524800 | 13 | 1.27999989 2 | 1.28000024 2 | 6.8-8 | 6.3-7 | -1.4-7 | 1:07:02 |

# 7 Applications to Quadratic Assignment and Binary Integer Quadratic Programming Problems

In this section, we apply our NCGAL algorithm to compute lower bounds for quadratic assignment problems (QAPs) and binary integer quadratic (BIQ) problems through SDP relaxations. Our purpose here is to demonstrate that the NCGAL algorithm can potentially be very efficient in solving large SDPs (and hence in computing bounds) arising from hard combinatorial problems.

Let $\Pi$ be the set of $n \times n$ permutation matrices. Given matrices $A, B \in \Re^{n\times n}$, the quadratic assignment problem is:

$$v^*_{\rm QAP} := \min\{\langle X, AXB\rangle \,:\, X \in \Pi\}. \tag{102}$$

For a matrix $X = [x_1, \ldots, x_n] \in \Re^{n\times n}$, we will identify it with the $n^2$-vector $x = [x_1; \ldots; x_n]$. For a matrix $Y \in R^{n^2\times n^2}$, we let $Y^{ij}$ be the $n \times n$ block corresponding to $x_i x_j^T$ in the matrix $xx^T$. It is shown in [21] that $v^*_{\rm QAP}$ is bounded below by the following number:

$$\begin{array}{rll} v := & \min & \langle B \otimes A,\, Y\rangle \\ & \text{s.t.} & \sum_{i=1}^n Y^{ii} = I,\ \langle I,\, Y^{ij}\rangle = \delta_{ij}\ \forall\, 1 \le i \le j \le n, \\ & & \langle E,\, Y^{ij}\rangle = 1,\ \forall\, 1 \le i \le j \le n, \\ & & Y \succeq 0,\ Y \ge 0, \end{array} \tag{103}$$

where $E$ is the matrix of ones, and $\delta_{ij} = 1$ if $i = j$, and 0 otherwise. Note that [21] actually used the constraint $\langle E,\, Y\rangle = n^2$ in place of the last set of the equality constraints. But we prefer to use the formulation here because the associated SDP has slightly better numerical behavior. Note that the SDP problems (103) typically do not satisfy the constraint nondegenerate conditions (41) and (67) at the optimal solutions.

In our experiment, we apply the NCGAL algorithm to the dual of (103) and hence any dual feasible solution would give a lower bound for (103). But in practice, our algorithm only delivers an approximately feasible dual solution $\tilde{y}$. We therefore apply the procedure given in [12, Theorem 2] to $\tilde{y}$ to construct a true lower bound for (103), which we denote by $\underline{v}$.

Table 9 lists the results of the NCGAL algorithm on the quadratic assignment instances (103). The details of the table are the same as for Table 1 except that the objective values are replaced by the best known upper bound on (102) under the column "best upper bound" and the lower bound $\underline{v}$. The entries under the column under "%gap" are calculated as follows:

$$\%gap \;=\; \frac{\text{best upper bound} - \underline{v}}{\text{best upper bound}} \times 100\%.$$

We compare our results with those obtained in [5] which used a dedicated augmented Lagrangian algorithm to solve the SDP arising from applying the lift-and-project procedure of Lovász and Schrijver to (102). As the augmented Lagrangian algorithm in [5] is designed specifically for the SDPs arising the lift-and-project procedure, the details of that algorithm is very different from our NCGAL algorithm. Note that the algorithm in [5] was implemented in C (with LAPACK library) and the results reported were obtained from a 2.4 GHz Pentium 4 PC with 1 GB of RAM (which is about 50% slower than our PC). By comparing the results in Table 9 against those in [5, Tables 6 and 7], we can safely conclude that the

NCGAL algorithm applied to (103) is superior in terms of CPU time and the accuracy of the approximate optimal solution computed. Take for example the SDPs corresponding to the QAPs `nug30` and `tai35b`, the NCGAL algorithm obtains the lower bounds with `%gap` of 2.939 and 5.318 in $28,655$ and $63,487$ seconds respectively, whereas the the algorithm in [5] computes the bounds with `%gap` of 3.10 and 15.42 in $127,011$ and $430,914$ seconds respectively.

The paper [5] also solved the lift-and-project SDP relaxations for the maximum stable set problems (denoted as $N_+$ and is known to be at least as strong as $\theta_+$) using a dedicated augmented Lagrangian algorithm. By comparing the results in Table 8 against those in [5, Table 4], we can again conclude that the NCGAL algorithm applied to (101) is superior in terms of CPU time and the accuracy of the approximate optimal solution computed. Take for example the SDPs corresponding to the graphs `p-hat300-1` and `c-fat200-1`, the NCGAL algorithm obtains the upper bounds of $\theta_+ = 10.0202$ and $\theta_+ = 12.0000$ in 868 and 52 seconds respectively, whereas the the algorithm in [5] computes the bounds of $N_+ = 18.6697$ and $N_+ = 14.9735$ in $322,287$ and $126,103$ seconds respectively.

The BIQ problem we consider is the following:

$$v^*_{\rm BIQ} := \min\{x^TQx \;:\; x \in \{0,1\}^n\}, \tag{104}$$

where $Q$ is a symmetric matrix (non positive semidefinite) of order $n$. A natural SDP relaxation of (104) is the following:

$$\begin{array}{ll} \min & \langle Q, Y\rangle \\ \text{s.t.} & \operatorname{diag}(Y) - y = 0, \quad \alpha = 1, \\ & \begin{bmatrix} Y & y \\ y^T & \alpha \end{bmatrix} \succeq 0, \quad Y \geq 0, \; y \geq 0. \end{array} \tag{105}$$

Table 10 lists the results obtained by the NCGAL algorithm on the SDPs (105) arising from the BIQ instances described in [41]. It is interesting to note that the lower bound obtained from (105) is within 10% of the optimal value $v^*_{\rm BIQ}$ for all the instances tested, and for the instances `gka1b`–`gka9b`, the lower bounds are actually equal to $v^*_{BIQ}$.

Table 9: Results for the NCGAL algorithm on the quadratic assignment problems. The entries under the column "%gap" are calculated with respect to the best solution listed, which is known to be optimal unless the symbol (†) is prefixed.

| problem | $m - n_l \mid n_s; n_l$ | it. | best upper bound | lower bound $\underline{v}$ | $R_P$ | $R_D$ | %gap | time |
|---|---|---|---|---|---|---|---|---|
| bur26a | 1051 \| 676; 228826 | 32 | 5.42667000 6 | 5.42558600 6 | **1.2-3** | 4.8-7 | 0.020 | 3:54:21 |
| bur26b | 1051 \| 676; 228826 | 35 | 3.81785200 6 | 3.81665200 6 | **9.5-4** | 2.3-7 | 0.031 | 4:52:43 |
| bur26c | 1051 \| 676; 228826 | 37 | 5.42679500 6 | 5.42616300 6 | **1.2-3** | 7.2-8 | 0.012 | 6:15:05 |
| bur26d | 1051 \| 676; 228826 | 35 | 3.82122500 6 | 3.81973400 6 | **1.3-3** | 2.6-7 | 0.039 | 4:57:23 |

Table 9: Results for the NCGAL algorithm on the quadratic assignment problems. The entries under the column "%gap" are calculated with respect to the best solution listed, which is known to be optimal unless the symbol (†) is prefixed.

| problem | $m - n_l \mid n_s; n_l$ | it. | best upper bound | lower bound $\underline{v}$ | $R_P$ | $R_D$ | %gap | time |
|---|---|---|---|---|---|---|---|---|
| bur26e | 1051 \| 676; 228826 | 33 | 5.38687900 6 | 5.38670400 6 | **9.1-3** | 3.1-7 | 0.003 | 5:17:37 |
| bur26f | 1051 \| 676; 228826 | 35 | 3.78204400 6 | 3.78195200 6 | **5.1-3** | 1.5-7 | 0.002 | 6:01:06 |
| bur26g | 1051 \| 676; 228826 | 30 | 1.01171720 7 | 1.01167210 7 | **5.3-3** | 3.7-7 | 0.004 | 3:44:49 |
| bur26h | 1051 \| 676; 228826 | 31 | 7.09865800 6 | 7.09856300 6 | **7.9-4** | 1.3-7 | 0.001 | 3:35:19 |
| chr12a | 232 \| 144; 10440 | 30 | 9.55200000 3 | 9.55200000 3 | 7.2-8 | 5.0-12 | 0.000 | 3:20 |
| chr12b | 232 \| 144; 10440 | 30 | 9.74200000 3 | 9.74200000 3 | 6.5-7 | 3.7-10 | 0.000 | 2:18 |
| chr12c | 232 \| 144; 10440 | 36 | 1.11560000 4 | 1.11560000 4 | **1.8-3** | 9.0-9 | 0.000 | 8:17 |
| chr15a | 358 \| 225; 25425 | 34 | 9.89600000 3 | 9.88500000 3 | **1.1-2** | 2.8-7 | 0.111 | 24:01 |
| chr15b | 358 \| 225; 25425 | 31 | 7.99000000 3 | 7.99000000 3 | **6.0-4** | 1.9-8 | 0.000 | 13:48 |
| chr15c | 358 \| 225; 25425 | 27 | 9.50400000 3 | 9.50400000 3 | **2.8-4** | 1.9-8 | 0.000 | 12:14 |
| chr18a | 511 \| 324; 52650 | 32 | 1.10980000 4 | 1.10920000 4 | **6.5-3** | 8.2-8 | 0.054 | 56:35 |
| chr18b | 511 \| 324; 52650 | 36 | 1.53400000 3 | 1.53200000 3 | **4.1-4** | 2.3-7 | 0.130 | 32:22 |
| chr20a | 628 \| 400; 80200 | 31 | 2.19200000 3 | 2.18900000 3 | **1.0-2** | 3.1-6 | 0.137 | 1:22:43 |
| chr20b | 628 \| 400; 80200 | 31 | 2.29800000 3 | 2.29800000 3 | **6.8-4** | 1.5-8 | 0.000 | 1:15:22 |
| chr20c | 628 \| 400; 80200 | 37 | 1.41420000 4 | 1.41390000 4 | **9.3-4** | 3.7-7 | 0.021 | 1:14:00 |
| chr22a | 757 \| 484; 117370 | 31 | 6.15600000 3 | 6.15600000 3 | **1.3-3** | 2.6-8 | 0.000 | 2:09:46 |
| chr22b | 757 \| 484; 117370 | 29 | 6.19400000 3 | 6.19300000 3 | **1.4-2** | 3.7-7 | 0.016 | 2:03:51 |
| chr25a | 973 \| 625; 195625 | 36 | 3.79600000 3 | 3.79600000 3 | **6.6-4** | 3.6-8 | 0.000 | 6:39:56 |
| els19 | 568 \| 361; 65341 | 34 | 1.72125480 7 | 1.72115960 7 | **1.8-4** | 1.3-7 | 0.006 | 1:00:38 |
| esc16a | 406 \| 256; 32896 | 35 | 6.80000000 1 | 6.40000000 1 | **2.4-4** | 4.3-7 | 5.882 | 17:42 |
| esc16b | 406 \| 256; 32896 | 34 | 2.92000000 2 | 2.89000000 2 | **3.8-4** | 2.7-7 | 1.027 | 18:48 |
| esc16c | 406 \| 256; 32896 | 35 | 1.60000000 2 | 1.53000000 2 | **5.6-4** | 4.1-7 | 4.375 | 23:23 |
| esc16d | 406 \| 256; 32896 | 26 | 1.60000000 1 | 1.30000000 1 | 1.9-6 | 3.4-7 | 18.750 | 2:54 |
| esc16e | 406 \| 256; 32896 | 36 | 2.80000000 1 | 2.70000000 1 | 7.3-7 | 5.5-8 | 3.571 | 23:05 |
| esc16g | 406 \| 256; 32896 | 34 | 2.60000000 1 | 2.50000000 1 | 9.8-7 | 1.0-7 | 3.846 | 25:08 |
| esc16h | 406 \| 256; 32896 | 31 | 9.96000000 2 | 9.76000000 2 | **1.2-4** | 4.6-7 | 2.008 | 18:47 |
| esc16i | 406 \| 256; 32896 | 29 | 1.40000000 1 | 1.20000000 1 | 7.0-6 | 2.8-7 | 14.286 | 14:40 |
| esc16j | 406 \| 256; 32896 | 29 | 8.00000000 0 | 8.00000000 0 | 1.8-5 | 2.3-7 | 0.000 | 13:27 |
| esc32a | 1582 \| 1024; 524800 | 38 | † 1.30000000 2 | 1.03000000 2 | **8.5-4** | 3.2-7 | 20.769 | 9:34:34 |
| esc32b | 1582 \| 1024; 524800 | 28 | † 1.68000000 2 | 1.32000000 2 | **1.4-4** | 4.5-7 | 21.429 | 5:38:29 |
| esc32c | 1582 \| 1024; 524800 | 39 | † 6.42000000 2 | 6.15000000 2 | **5.4-4** | 1.3-7 | 4.206 | 15:50:30 |
| esc32d | 1582 \| 1024; 524800 | 36 | † 2.00000000 2 | 1.90000000 2 | **6.2-4** | 3.7-7 | 5.000 | 11:48:37 |
| esc32e | 1582 \| 1024; 524800 | 31 | 2.00000000 0 | 2.00000000 0 | **1.8-4** | 9.5-7 | 0.000 | 8:58:38 |
| esc32f | 1582 \| 1024; 524800 | 31 | 2.00000000 0 | 2.00000000 0 | **1.8-4** | 9.5-7 | 0.000 | 9:01:41 |
| esc32g | 1582 \| 1024; 524800 | 24 | 6.00000000 0 | 6.00000000 0 | 3.1-6 | 3.0-7 | 0.000 | 2:08:59 |
| esc32h | 1582 \| 1024; 524800 | 36 | † 4.38000000 2 | 4.20000000 2 | **8.3-4** | 4.5-7 | 4.110 | 12:22:08 |
| had12 | 232 \| 144; 10440 | 31 | 1.65200000 3 | 1.65200000 3 | **2.1-4** | 9.6-8 | 0.000 | 5:25 |

Table 9: Results for the NCGAL algorithm on the quadratic assignment problems. The entries under the column "%gap" are calculated with respect to the best solution listed, which is known to be optimal unless the symbol (†) is prefixed.

| problem | $m - n_l \mid n_s; n_l$ | it. | best upper bound | lower bound $\underline{v}$ | $R_P$ | $R_D$ | %gap | time |
|---|---|---|---|---|---|---|---|---|
| had14 | 313 \| 196; 19306 | 34 | 2.72400000 3 | 2.72400000 3 | **2.4-3** | 1.6-7 | 0.000 | 12:22 |
| had16 | 406 \| 256; 32896 | 33 | 3.72000000 3 | 3.72000000 3 | **1.1-3** | 8.8-8 | 0.000 | 26:38 |
| had18 | 511 \| 324; 52650 | 36 | 5.35800000 3 | 5.35800000 3 | **1.8-3** | 4.7-7 | 0.000 | 56:08 |
| had20 | 628 \| 400; 80200 | 37 | 6.92200000 3 | 6.92200000 3 | **2.2-3** | 3.0-7 | 0.000 | 1:57:29 |
| kra30a | 1393 \| 900; 405450 | 33 | 8.89000000 4 | 8.64950000 4 | **2.6-4** | 3.9-7 | 2.705 | 7:51:29 |
| kra30b | 1393 \| 900; 405450 | 36 | 9.14200000 4 | 8.75010000 4 | **2.9-4** | 2.2-7 | 4.287 | 7:22:21 |
| kra32 | 1582 \| 1024; 524800 | 34 | 8.89000000 4 | 8.54320000 4 | **5.4-4** | 2.5-7 | 3.901 | 9:26:57 |
| lipa20a | 628 \| 400; 80200 | 22 | 3.68300000 3 | 3.68300000 3 | 8.0-7 | 7.4-10 | 0.000 | 24:28 |
| lipa20b | 628 \| 400; 80200 | 20 | 2.70760000 4 | 2.70760000 4 | 5.4-8 | 4.0-9 | 0.000 | 12:41 |
| lipa30a | 1393 \| 900; 405450 | 24 | 1.31780000 4 | 1.31780000 4 | 1.7-7 | 4.4-10 | 0.000 | 5:31:50 |
| lipa30b | 1393 \| 900; 405450 | 24 | 1.51426000 5 | 1.51426000 5 | 7.0-7 | 8.7-9 | 0.000 | 1:47:16 |
| lipa40a | 2458 \| 1600; 1280800 | 26 | 3.15380000 4 | 3.15380000 4 | 4.9-7 | 4.9-10 | 0.000 | 27:21:52 |
| lipa40b | 2458 \| 1600; 1280800 | 23 | 4.76581000 5 | 4.76580000 5 | 7.5-7 | 1.3-7 | 0.000 | 12:44:32 |
| nug12 | 232 \| 144; 10440 | 30 | 5.78000000 2 | 5.67000000 2 | **2.1-4** | 4.2-7 | 1.903 | 2:23 |
| nug14 | 313 \| 196; 19306 | 33 | 1.01400000 3 | 1.00900000 3 | **1.8-4** | 3.4-7 | 0.493 | 9:04 |
| nug15 | 358 \| 225; 25425 | 29 | 1.15000000 3 | 1.13900000 3 | **2.2-4** | 1.8-6 | 0.957 | 11:02 |
| nug16a | 406 \| 256; 32896 | 32 | 1.61000000 3 | 1.59700000 3 | **3.6-4** | 4.2-7 | 0.807 | 20:46 |
| nug16b | 406 \| 256; 32896 | 32 | 1.24000000 3 | 1.21600000 3 | **1.9-4** | 3.4-7 | 1.935 | 15:13 |
| nug17 | 457 \| 289; 41905 | 30 | 1.73200000 3 | 1.70400000 3 | **1.9-4** | 1.8-6 | 1.617 | 20:18 |
| nug18 | 511 \| 324; 52650 | 32 | 1.93000000 3 | 1.89100000 3 | **1.9-4** | 5.0-7 | 2.021 | 32:11 |
| nug20 | 628 \| 400; 80200 | 33 | 2.57000000 3 | 2.50300000 3 | **1.5-4** | 3.9-7 | 2.607 | 45:51 |
| nug21 | 691 \| 441; 97461 | 29 | 2.43800000 3 | 2.37700000 3 | **2.0-4** | 4.0-6 | 2.502 | 58:22 |
| nug22 | 757 \| 484; 117370 | 36 | 3.59600000 3 | 3.52400000 3 | **4.7-4** | 6.8-8 | 2.002 | 1:43:56 |
| nug24 | 898 \| 576; 166176 | 31 | 3.48800000 3 | 3.39700000 3 | **2.2-4** | 2.0-7 | 2.609 | 1:46:40 |
| nug25 | 973 \| 625; 195625 | 31 | 3.74400000 3 | 3.62200000 3 | **1.9-4** | 3.4-7 | 3.259 | 2:27:41 |
| nug27 | 1132 \| 729; 266085 | 35 | 5.23400000 3 | 5.12200000 3 | **5.5-4** | 2.3-7 | 2.140 | 5:14:39 |
| nug28 | 1216 \| 784; 307720 | 33 | 5.16600000 3 | 5.02000000 3 | **3.5-4** | 4.0-7 | 2.826 | 5:23:20 |
| nug30 | 1393 \| 900; 405450 | 33 | 6.12400000 3 | 5.94400000 3 | **1.5-4** | 3.1-7 | 2.939 | 7:57:35 |
| rou12 | 232 \| 144; 10440 | 33 | 2.35528000 5 | 2.35414000 5 | **3.2-4** | 1.1-7 | 0.048 | 4:55 |
| rou15 | 358 \| 225; 25425 | 33 | 3.54210000 5 | 3.49589000 5 | **2.0-4** | 1.6-7 | 1.305 | 9:05 |
| rou20 | 628 \| 400; 80200 | 34 | 7.25522000 5 | 6.94405000 5 | **1.4-4** | 4.1-7 | 4.289 | 38:40 |
| scr12 | 232 \| 144; 10440 | 24 | 3.14100000 4 | 3.14090000 4 | **3.4-4** | 3.2-7 | 0.003 | 2:50 |
| scr15 | 358 \| 225; 25425 | 26 | 5.11400000 4 | 5.11400000 4 | 2.7-7 | 4.5-9 | 0.000 | 8:59 |
| scr20 | 628 \| 400; 80200 | 31 | 1.10030000 5 | 1.06464000 5 | **3.1-4** | 3.9-7 | 3.241 | 54:24 |
| ste36a | 1996 \| 1296; 840456 | 35 | 9.52600000 3 | 9.23100000 3 | **6.7-4** | 3.7-7 | 3.097 | 24:19:19 |
| ste36b | 1996 \| 1296; 840456 | 34 | 1.58520000 4 | 1.55860000 4 | **9.6-4** | 3.2-7 | 1.678 | 27:12:36 |
| ste36c | 1996 \| 1296; 840456 | 33 | 8.23911000 6 | 8.11367100 6 | **9.0-4** | 3.8-7 | 1.522 | 21:57:42 |

Table 9: Results for the NCGAL algorithm on the quadratic assignment problems. The entries under the column "%gap" are calculated with respect to the best solution listed, which is known to be optimal unless the symbol (†) is prefixed.

| problem | $m - n_l \mid n_s; n_l$ | it. | best upper bound | lower bound $\underline{v}$ | $R_P$ | $R_D$ | %gap | time |
|---|---|---|---|---|---|---|---|---|
| tai12a | 232 \| 144; 10440 | 20 | 2.24416000 5 | 2.24416000 5 | 2.7-7 | 5.7-8 | 0.000 | 1:56 |
| tai12b | 232 \| 144; 10440 | 32 | 3.94649250 7 | 3.94647050 7 | **7.7-4** | 5.4-9 | 0.001 | 5:37 |
| tai15a | 358 \| 225; 25425 | 32 | 3.88214000 5 | 3.76559000 5 | **1.6-4** | 4.9-7 | 3.002 | 5:42 |
| tai15b | 358 \| 225; 25425 | 40 | 5.17652680 7 | 5.17651920 7 | **1.0-4** | 1.2-10 | 0.000 | 12:40 |
| tai17a | 457 \| 289; 41905 | 32 | 4.91812000 5 | 4.75964000 5 | **1.2-4** | 4.3-7 | 3.222 | 11:52 |
| tai20a | 628 \| 400; 80200 | 30 | 7.03482000 5 | 6.70979000 5 | **1.1-4** | 1.1-6 | 4.620 | 23:59 |
| tai20b | 628 \| 400; 80200 | 38 | 1.22455319 8 | 1.22454350 8 | **6.3-4** | 2.3-9 | 0.001 | 1:21:46 |
| tai25a | 973 \| 625; 195625 | 35 | 1.16725600 6 | 1.09888100 6 | 1.4-6 | 1.1-7 | 5.858 | 2:56:09 |
| tai25b | 973 \| 625; 195625 | 36 | 3.44355646 8 | 3.33055439 8 | **9.2-4** | 3.7-7 | 3.282 | 4:07:14 |
| tai30a | 1393 \| 900; 405450 | 32 | † 1.81814600 6 | 1.70568800 6 | 9.3-5 | 4.9-7 | 6.185 | 3:21:21 |
| tai30b | 1393 \| 900; 405450 | 35 | 6.37117113 8 | 5.95633454 8 | **1.1-3** | 3.3-7 | 6.511 | 8:21:32 |
| tai35a | 1888 \| 1225; 750925 | 31 | † 2.42200200 6 | 2.21530300 6 | **1.0-4** | 3.6-7 | 8.534 | 8:01:26 |
| tai35b | 1888 \| 1225; 750925 | 34 | 2.83315445 8 | 2.68248553 8 | **8.5-4** | 3.7-7 | 5.318 | 17:38:07 |
| tai40a | 2458 \| 1600; 1280800 | 32 | † 3.13937000 6 | 2.84106900 6 | **1.3-4** | 3.1-7 | 9.502 | 17:00:27 |
| tai40b | 2458 \| 1600; 1280800 | 33 | 6.37250948 8 | 6.06354605 8 | **1.1-3** | 3.9-7 | 4.848 | 34:11:29 |
| tho30 | 1393 \| 900; 405450 | 32 | 1.49936000 5 | 1.43277000 5 | **2.9-4** | 4.2-7 | 4.441 | 5:42:59 |

Table 10: Results for the NCGAL algorithm on the BIQ problems. The entries under the column "%gap" are calculated with respect to the best solution listed, which is known to be optimal unless the symbol (†) is prefixed.

| problem | $m - n_l \mid n_s; n_l$ | it. | best upper bound | lower bound $\underline{v}$ | $R_P$ | $R_D$ | %gap | time |
|---|---|---|---|---|---|---|---|---|
| be100.1 | 101 \| 101; 5151 | 27 | -1.94120000 4 | -2.00210000 4 | 3.5-7 | 9.3-7 | 3.137 | 2:16 |
| be100.2 | 101 \| 101; 5151 | 29 | -1.72900000 4 | -1.79880000 4 | 1.6-7 | 9.9-7 | 4.037 | 2:03 |
| be100.3 | 101 \| 101; 5151 | 31 | -1.75650000 4 | -1.82310000 4 | 1.1-7 | 7.5-7 | 3.792 | 2:27 |
| be100.4 | 101 \| 101; 5151 | 31 | -1.91250000 4 | -1.98410000 4 | 3.3-7 | 5.9-7 | 3.744 | 3:25 |
| be100.5 | 101 \| 101; 5151 | 30 | -1.58680000 4 | -1.68880000 4 | 1.2-7 | 9.8-7 | 6.428 | 1:39 |
| be100.6 | 101 \| 101; 5151 | 30 | -1.73680000 4 | -1.81480000 4 | 1.5-7 | 7.2-7 | 4.491 | 3:02 |
| be100.7 | 101 \| 101; 5151 | 32 | -1.86290000 4 | -1.97010000 4 | 1.7-7 | 9.7-7 | 5.754 | 1:01 |
| be100.8 | 101 \| 101; 5151 | 29 | -1.86490000 4 | -1.99460000 4 | 4.9-8 | 8.4-7 | 6.955 | 58 |
| be100.9 | 101 \| 101; 5151 | 27 | -1.32940000 4 | -1.42630000 4 | 8.2-8 | 5.8-7 | 7.289 | 2:33 |
| be100.10 | 101 \| 101; 5151 | 29 | -1.53520000 4 | -1.64080000 4 | 1.9-7 | 9.4-7 | 6.879 | 1:35 |
| be120.3.1 | 121 \| 121; 7381 | 30 | -1.30670000 4 | -1.38030000 4 | 1.4-7 | 7.8-7 | 5.633 | 4:13 |
| be120.3.2 | 121 \| 121; 7381 | 28 | -1.30460000 4 | -1.36260000 4 | 4.2-7 | 6.3-7 | 4.446 | 5:47 |

Table 10: Results for the NCGAL algorithm on the BIQ problems. The entries under the column "%gap" are calculated with respect to the best solution listed, which is known to be optimal unless the symbol (†) is prefixed.

| problem | $m - n_l \mid n_s; n_l$ | it. | best upper bound | lower bound $\underline{v}$ | $R_P$ | $R_D$ | %gap | time |
|---|---|---|---|---|---|---|---|---|
| be120.3.3 | 121 \| 121; 7381 | 28 | -1.24180000 4 | -1.29880000 4 | 8.0-8 | 8.0-7 | 4.590 | 1:22 |
| be120.3.4 | 121 \| 121; 7381 | 28 | -1.38670000 4 | -1.45110000 4 | 1.7-7 | 8.7-7 | 4.644 | 2:23 |
| be120.3.5 | 121 \| 121; 7381 | 31 | -1.14030000 4 | -1.19910000 4 | 1.2-5 | 3.0-7 | 5.157 | 4:47 |
| be120.3.6 | 121 \| 121; 7381 | 30 | -1.29150000 4 | -1.34320000 4 | 3.6-8 | 6.7-7 | 4.003 | 2:46 |
| be120.3.7 | 121 \| 121; 7381 | 29 | -1.40680000 4 | -1.45640000 4 | 2.2-7 | 6.8-7 | 3.526 | 6:29 |
| be120.3.8 | 121 \| 121; 7381 | 31 | -1.47010000 4 | -1.53030000 4 | 8.2-5 | 3.7-7 | 4.095 | 5:06 |
| be120.3.9 | 121 \| 121; 7381 | 31 | -1.04580000 4 | -1.12410000 4 | 2.9-5 | 2.3-7 | 7.487 | 3:13 |
| be120.3.10 | 121 \| 121; 7381 | 29 | -1.22010000 4 | -1.29300000 4 | 5.5-7 | 7.5-7 | 5.975 | 5:22 |
| be120.8.1 | 121 \| 121; 7381 | 29 | -1.86910000 4 | -2.01940000 4 | 1.6-7 | 8.9-7 | 8.041 | 2:16 |
| be120.8.2 | 121 \| 121; 7381 | 29 | -1.88270000 4 | -2.00740000 4 | 8.6-6 | 4.9-7 | 6.623 | 5:21 |
| be120.8.3 | 121 \| 121; 7381 | 30 | -1.93020000 4 | -2.05060000 4 | 1.0-7 | 6.3-7 | 6.238 | 1:44 |
| be120.8.4 | 121 \| 121; 7381 | 29 | -2.07650000 4 | -2.17790000 4 | 2.5-7 | 8.3-7 | 4.883 | 5:28 |
| be120.8.5 | 121 \| 121; 7381 | 29 | -2.04170000 4 | -2.13160000 4 | 3.6-7 | 7.1-7 | 4.403 | 5:24 |
| be120.8.6 | 121 \| 121; 7381 | 28 | -1.84820000 4 | -1.96770000 4 | 2.7-7 | 6.8-7 | 6.466 | 4:58 |
| be120.8.7 | 121 \| 121; 7381 | 29 | -2.21940000 4 | -2.37320000 4 | 2.6-7 | 5.8-7 | 6.930 | 2:19 |
| be120.8.8 | 121 \| 121; 7381 | 28 | -1.95340000 4 | -2.12040000 4 | 1.4-7 | 9.7-7 | 8.549 | 55 |
| be120.8.9 | 121 \| 121; 7381 | 29 | -1.81950000 4 | -1.92840000 4 | 1.2-7 | 6.9-7 | 5.985 | 1:38 |
| be120.8.10 | 121 \| 121; 7381 | 29 | -1.90490000 4 | -2.00240000 4 | 5.2-8 | 7.7-7 | 5.118 | 2:38 |
| be150.3.1 | 151 \| 151; 11476 | 32 | -1.88890000 4 | -1.98490000 4 | 7.8-7 | 7.2-7 | 5.082 | 8:34 |
| be150.3.2 | 151 \| 151; 11476 | 29 | -1.78160000 4 | -1.88650000 4 | 9.7-7 | 7.7-7 | 5.888 | 8:58 |
| be150.3.3 | 151 \| 151; 11476 | 32 | -1.73140000 4 | -1.80430000 4 | 8.8-8 | 7.2-7 | 4.210 | 7:28 |
| be150.3.4 | 151 \| 151; 11476 | 32 | -1.98840000 4 | -2.06520000 4 | 5.9-8 | 6.1-7 | 3.862 | 2:51 |
| be150.3.5 | 151 \| 151; 11476 | 29 | -1.68170000 4 | -1.77680000 4 | 1.9-7 | 7.6-7 | 5.655 | 6:50 |
| be150.3.6 | 151 \| 151; 11476 | 32 | -1.67800000 4 | -1.80500000 4 | 5.1-7 | 7.6-7 | 7.569 | 8:33 |
| be150.3.7 | 151 \| 151; 11476 | 30 | -1.80010000 4 | -1.91010000 4 | 1.1-7 | 5.9-7 | 6.111 | 4:36 |
| be150.3.8 | 151 \| 151; 11476 | 32 | -1.83030000 4 | -1.96980000 4 | 3.5-7 | 9.7-7 | 7.622 | 6:48 |
| be150.3.9 | 151 \| 151; 11476 | 32 | -1.28380000 4 | -1.41030000 4 | 1.3-7 | 9.7-7 | 9.854 | 2:38 |
| be150.3.10 | 151 \| 151; 11476 | 32 | -1.79630000 4 | -1.92300000 4 | 3.3-6 | 3.2-7 | 7.053 | 9:55 |
| be150.8.1 | 151 \| 151; 11476 | 30 | -2.70890000 4 | -2.91430000 4 | 5.1-7 | 6.5-7 | 7.582 | 6:51 |
| be150.8.2 | 151 \| 151; 11476 | 29 | -2.67790000 4 | -2.88210000 4 | 1.6-8 | 8.0-7 | 7.625 | 3:05 |
| be150.8.3 | 151 \| 151; 11476 | 30 | -2.94380000 4 | -3.10600000 4 | 2.7-7 | 6.7-7 | 5.510 | 5:58 |
| be150.8.4 | 151 \| 151; 11476 | 28 | -2.69110000 4 | -2.87290000 4 | 9.2-8 | 9.2-7 | 6.756 | 3:32 |
| be150.8.5 | 151 \| 151; 11476 | 29 | -2.80170000 4 | -2.94820000 4 | 2.9-7 | 9.2-7 | 5.229 | 9:36 |
| be150.8.6 | 151 \| 151; 11476 | 29 | -2.92210000 4 | -3.14370000 4 | 8.6-7 | 7.2-7 | 7.584 | 9:27 |
| be150.8.7 | 151 \| 151; 11476 | 30 | -3.12090000 4 | -3.32520000 4 | 5.7-7 | 6.8-7 | 6.546 | 13:32 |
| be150.8.8 | 151 \| 151; 11476 | 30 | -2.97300000 4 | -3.16000000 4 | 3.9-7 | 7.4-7 | 6.290 | 11:03 |
| be150.8.9 | 151 \| 151; 11476 | 30 | -2.53880000 4 | -2.71100000 4 | 1.9-7 | 8.6-7 | 6.783 | 4:35 |

Table 10: Results for the NCGAL algorithm on the BIQ problems. The entries under the column "%gap" are calculated with respect to the best solution listed, which is known to be optimal unless the symbol (†) is prefixed.

| problem | $m - n_l \mid n_s; n_l$ | it. | best upper bound | lower bound $\underline{v}$ | $R_P$ | $R_D$ | %gap | time |
|---|---|---|---|---|---|---|---|---|
| be150.8.10 | 151 \| 151; 11476 | 28 | -2.83740000 4 | -3.00480000 4 | 8.0-8 | 6.2-7 | 5.900 | 3:13 |
| be200.3.1 | 201 \| 201; 20301 | 28 | -2.54530000 4 | -2.77160000 4 | 7.0-7 | 9.3-7 | 8.891 | 13:05 |
| be200.3.2 | 201 \| 201; 20301 | 29 | -2.50270000 4 | -2.67610000 4 | 1.5-7 | 9.0-7 | 6.929 | 6:59 |
| be200.3.3 | 201 \| 201; 20301 | 30 | -2.80230000 4 | -2.94780000 4 | 9.3-7 | 7.8-7 | 5.192 | 22:16 |
| be200.3.4 | 201 \| 201; 20301 | 29 | -2.74340000 4 | -2.91060000 4 | 2.6-7 | 7.5-7 | 6.095 | 18:03 |
| be200.3.5 | 201 \| 201; 20301 | 29 | -2.63550000 4 | -2.80730000 4 | 1.8-7 | 8.6-7 | 6.519 | 11:58 |
| be200.3.6 | 201 \| 201; 20301 | 30 | -2.61460000 4 | -2.79280000 4 | 2.9-7 | 6.7-7 | 6.816 | 10:00 |
| be200.3.7 | 201 \| 201; 20301 | 29 | -3.04830000 4 | -3.16200000 4 | 3.7-7 | 8.3-7 | 3.730 | 14:50 |
| be200.3.8 | 201 \| 201; 20301 | 29 | -2.73550000 4 | -2.92440000 4 | 1.0-7 | 6.2-7 | 6.906 | 8:26 |
| be200.3.9 | 201 \| 201; 20301 | 34 | -2.46830000 4 | -2.64370000 4 | 1.4-5 | 5.0-7 | 7.106 | 12:07 |
| be200.3.10 | 201 \| 201; 20301 | 31 | -2.38420000 4 | -2.57600000 4 | 7.4-6 | 4.0-7 | 8.045 | 23:08 |
| be200.8.1 | 201 \| 201; 20301 | 31 | -4.85340000 4 | -5.08690000 4 | 1.1-7 | 6.3-7 | 4.811 | 19:25 |
| be200.8.2 | 201 \| 201; 20301 | 30 | -4.08210000 4 | -4.43360000 4 | 1.1-7 | 7.6-7 | 8.611 | 3:44 |
| be200.8.3 | 201 \| 201; 20301 | 29 | -4.32070000 4 | -4.62540000 4 | 1.6-7 | 8.4-7 | 7.052 | 12:36 |
| be200.8.4 | 201 \| 201; 20301 | 29 | -4.37570000 4 | -4.66210000 4 | 5.3-8 | 6.7-7 | 6.545 | 6:20 |
| be200.8.5 | 201 \| 201; 20301 | 31 | -4.14820000 4 | -4.42710000 4 | 1.3-6 | 4.3-7 | 6.723 | 21:07 |
| be200.8.6 | 201 \| 201; 20301 | 33 | -4.94920000 4 | -5.12190000 4 | 1.4-5 | 4.2-7 | 3.489 | 19:53 |
| be200.8.7 | 201 \| 201; 20301 | 31 | -4.68280000 4 | -4.93530000 4 | 2.7-8 | 8.9-7 | 5.392 | 5:30 |
| be200.8.8 | 201 \| 201; 20301 | 30 | -4.45020000 4 | -4.76890000 4 | 5.4-8 | 7.1-7 | 7.161 | 6:51 |
| be200.8.9 | 201 \| 201; 20301 | 31 | -4.32410000 4 | -4.54960000 4 | 6.9-7 | 7.3-7 | 5.215 | 27:18 |
| be200.8.10 | 201 \| 201; 20301 | 29 | -4.28320000 4 | -4.57430000 4 | 1.3-7 | 6.7-7 | 6.796 | 18:29 |
| be250.1 | 251 \| 251; 31626 | 30 | -2.40760000 4 | -2.51190000 4 | 1.9-5 | 4.7-7 | 4.332 | 23:40 |
| be250.2 | 251 \| 251; 31626 | 30 | -2.25400000 4 | -2.36810000 4 | 2.0-7 | 8.9-7 | 5.062 | 28:45 |
| be250.3 | 251 \| 251; 31626 | 33 | -2.29230000 4 | -2.40000000 4 | 1.7-5 | 1.5-7 | 4.698 | 22:27 |
| be250.4 | 251 \| 251; 31626 | 31 | -2.46490000 4 | -2.57200000 4 | 3.9-5 | 3.6-7 | 4.345 | 25:15 |
| be250.5 | 251 \| 251; 31626 | 32 | -2.10570000 4 | -2.23740000 4 | 1.5-5 | 3.0-7 | 6.254 | 24:43 |
| be250.6 | 251 \| 251; 31626 | 31 | -2.27350000 4 | -2.40180000 4 | 8.1-5 | 3.2-7 | 5.643 | 19:25 |
| be250.7 | 251 \| 251; 31626 | 33 | -2.40950000 4 | -2.51190000 4 | 2.6-5 | 1.1-7 | 4.250 | 22:07 |
| be250.8 | 251 \| 251; 31626 | 34 | -2.38010000 4 | -2.50200000 4 | 2.5-6 | 3.0-7 | 5.122 | 41:04 |
| be250.9 | 251 \| 251; 31626 | 30 | -2.00510000 4 | -2.13970000 4 | **1.7-4** | 4.4-7 | 6.713 | 18:06 |
| be250.10 | 251 \| 251; 31626 | 31 | -2.31590000 4 | -2.43550000 4 | 5.5-5 | 5.0-7 | 5.164 | 18:15 |
| bqp50-1 | 51 \| 51; 1326 | 31 | -2.09800000 3 | -2.14300000 3 | 8.4-7 | 5.9-7 | 2.145 | 1:21 |
| bqp50-2 | 51 \| 51; 1326 | 29 | -3.70200000 3 | -3.74200000 3 | 7.9-6 | 1.5-7 | 1.080 | 50 |
| bqp50-3 | 51 \| 51; 1326 | 30 | -4.62600000 3 | -4.63700000 3 | 1.2-7 | 7.4-7 | 0.238 | 53 |
| bqp50-4 | 51 \| 51; 1326 | 33 | -3.54400000 3 | -3.58300000 3 | **2.9-4** | 2.8-7 | 1.100 | 1:10 |
| bqp50-5 | 51 \| 51; 1326 | 31 | -4.01200000 3 | -4.07700000 3 | 1.5-7 | 8.5-7 | 1.620 | 1:09 |
| bqp50-6 | 51 \| 51; 1326 | 23 | -3.69300000 3 | -3.71100000 3 | 8.2-5 | 2.4-7 | 0.487 | 39 |

Table 10: Results for the NCGAL algorithm on the BIQ problems. The entries under the column "%gap" are calculated with respect to the best solution listed, which is known to be optimal unless the symbol (†) is prefixed.

| problem | $m - n_l \mid n_s; n_l$ | it. | best upper bound | lower bound $\underline{v}$ | $R_P$ | $R_D$ | %gap | time |
|---|---|---|---|---|---|---|---|---|
| bqp50-7 | 51 \| 51; 1326 | 28 | -4.52000000 3 | -4.64900000 3 | 3.4-8 | 9.4-7 | 2.854 | 23 |
| bqp50-8 | 51 \| 51; 1326 | 26 | -4.21600000 3 | -4.26900000 3 | 1.2-7 | 7.0-7 | 1.257 | 37 |
| bqp50-9 | 51 \| 51; 1326 | 28 | -3.78000000 3 | -3.92100000 3 | 1.3-7 | 8.4-7 | 3.730 | 19 |
| bqp50-10 | 51 \| 51; 1326 | 29 | -3.50700000 3 | -3.62600000 3 | 1.6-7 | 8.9-7 | 3.393 | 17 |
| bqp100-1 | 101 \| 101; 5151 | 27 | -7.97000000 3 | -8.38000000 3 | 1.0-7 | 8.6-7 | 5.144 | 2:04 |
| bqp100-2 | 101 \| 101; 5151 | 24 | -1.10360000 4 | -1.14890000 4 | **3.7-4** | 4.1-7 | 4.105 | 1:44 |
| bqp100-3 | 101 \| 101; 5151 | 29 | -1.27230000 4 | -1.31530000 4 | 4.2-7 | 7.0-7 | 3.380 | 2:11 |
| bqp100-4 | 101 \| 101; 5151 | 30 | -1.03680000 4 | -1.07310000 4 | 5.5-7 | 8.3-7 | 3.501 | 3:55 |
| bqp100-5 | 101 \| 101; 5151 | 30 | -9.08300000 3 | -9.48700000 3 | 1.6-5 | 4.1-7 | 4.448 | 2:36 |
| bqp100-6 | 101 \| 101; 5151 | 30 | -1.02100000 4 | -1.08240000 4 | 1.9-7 | 7.6-7 | 6.014 | 2:31 |
| bqp100-7 | 101 \| 101; 5151 | 29 | -1.01250000 4 | -1.06890000 4 | 3.5-7 | 9.2-7 | 5.570 | 2:15 |
| bqp100-8 | 101 \| 101; 5151 | 30 | -1.14350000 4 | -1.17700000 4 | 2.5-6 | 4.7-7 | 2.930 | 4:44 |
| bqp100-9 | 101 \| 101; 5151 | 30 | -1.14550000 4 | -1.17330000 4 | 5.9-5 | 3.0-7 | 2.427 | 3:27 |
| bqp100-10 | 101 \| 101; 5151 | 35 | -1.25650000 4 | -1.29800000 4 | 1.4-6 | 2.6-7 | 3.303 | 4:43 |
| bqp250-1 | 251 \| 251; 31626 | 31 | -4.56070000 4 | -4.76630000 4 | 1.1-7 | 6.6-7 | 4.508 | 19:52 |
| bqp250-2 | 251 \| 251; 31626 | 34 | -4.48100000 4 | -4.72220000 4 | 1.7-5 | 2.6-7 | 5.383 | 26:09 |
| bqp250-3 | 251 \| 251; 31626 | 29 | -4.90370000 4 | -5.10770000 4 | 1.1-7 | 8.4-7 | 4.160 | 13:53 |
| bqp250-4 | 251 \| 251; 31626 | 31 | -4.12740000 4 | -4.33120000 4 | 3.1-5 | 3.7-7 | 4.938 | 20:54 |
| bqp250-5 | 251 \| 251; 31626 | 32 | -4.79610000 4 | -5.00040000 4 | 3.5-5 | 2.8-7 | 4.260 | 24:06 |
| bqp250-6 | 251 \| 251; 31626 | 29 | -4.10140000 4 | -4.36690000 4 | 1.7-7 | 9.9-7 | 6.473 | 30:21 |
| bqp250-7 | 251 \| 251; 31626 | 33 | -4.67570000 4 | -4.89220000 4 | 1.2-7 | 8.1-7 | 4.630 | 18:58 |
| bqp250-8 | 251 \| 251; 31626 | 31 | -3.57260000 4 | -3.87790000 4 | 2.7-7 | 7.1-7 | 8.546 | 22:07 |
| bqp250-9 | 251 \| 251; 31626 | 32 | -4.89160000 4 | -5.14980000 4 | 9.2-8 | 7.5-7 | 5.278 | 19:01 |
| bqp250-10 | 251 \| 251; 31626 | 29 | -4.04420000 4 | -4.30140000 4 | 6.6-7 | 8.8-7 | 6.360 | 23:15 |
| bqp500-1 | 501 \| 501; 125751 | 30 | -1.16586000 5 | -1.25966000 5 | 9.4-8 | 9.8-7 | 8.046 | 1:20:19 |
| bqp500-2 | 501 \| 501; 125751 | 33 | -1.28223000 5 | -1.36011000 5 | 9.1-5 | 2.6-7 | 6.074 | 1:52:27 |
| bqp500-3 | 501 \| 501; 125751 | 30 | -1.30812000 5 | -1.38455000 5 | 1.2-7 | 9.6-7 | 5.843 | 1:21:08 |
| bqp500-4 | 501 \| 501; 125751 | 30 | -1.30097000 5 | -1.39330000 5 | 6.6-7 | 9.6-7 | 7.097 | 1:55:57 |
| bqp500-5 | 501 \| 501; 125751 | 32 | -1.25487000 5 | -1.34093000 5 | 8.3-5 | 3.4-7 | 6.858 | 1:58:06 |
| bqp500-6 | 501 \| 501; 125751 | 30 | -1.21772000 5 | -1.30766000 5 | 1.5-7 | 8.8-7 | 7.386 | 2:05:49 |
| bqp500-7 | 501 \| 501; 125751 | 31 | -1.22201000 5 | -1.31492000 5 | 8.9-5 | 2.8-7 | 7.603 | 1:48:02 |
| bqp500-8 | 501 \| 501; 125751 | 31 | -1.23559000 5 | -1.33491000 5 | 5.5-8 | 6.8-7 | 8.038 | 1:48:16 |
| bqp500-9 | 501 \| 501; 125751 | 31 | -1.20798000 5 | -1.30289000 5 | 9.4-5 | 2.8-7 | 7.857 | 1:44:31 |
| bqp500-10 | 501 \| 501; 125751 | 32 | -1.30619000 5 | -1.38534000 5 | 2.5-5 | 4.2-8 | 6.060 | 1:30:32 |
| gka1a | 51 \| 51; 1326 | 24 | -3.41400000 3 | -3.53700000 3 | 4.2-6 | 1.7-8 | 3.603 | 29 |
| gka2a | 61 \| 61; 1891 | 26 | -6.06300000 3 | -6.17100000 3 | 1.0-7 | 6.7-7 | 1.781 | 27 |
| gka3a | 71 \| 71; 2556 | 27 | -6.03700000 3 | -6.38600000 3 | 3.6-7 | 6.1-7 | 5.781 | 1:03 |

Table 10: Results for the NCGAL algorithm on the BIQ problems. The entries under the column "%gap" are calculated with respect to the best solution listed, which is known to be optimal unless the symbol (†) is prefixed.

| problem | $m - n_l \mid n_s; n_l$ | it. | best upper bound | lower bound $\underline{v}$ | $R_P$ | $R_D$ | %gap | time |
|---|---|---|---|---|---|---|---|---|
| gka4a | 81 \| 81; 3321 | 32 | -8.59800000 3 | -8.88100000 3 | 7.0-7 | 7.2-7 | 3.291 | 2:55 |
| gka5a | 51 \| 51; 1326 | 29 | -5.73700000 3 | -5.89700000 3 | 1.5-7 | 8.6-7 | 2.789 | 24 |
| gka6a | 31 \| 31; 496 | 24 | -3.98000000 3 | -4.10300000 3 | 1.6-7 | 9.3-7 | 3.090 | 10 |
| gka7a | 31 \| 31; 496 | 31 | -4.54100000 3 | -4.63800000 3 | 9.3-8 | 7.5-7 | 2.136 | 09 |
| gka8a | 101 \| 101; 5151 | 32 | -1.11090000 4 | -1.11970000 4 | 8.4-7 | 9.1-7 | 0.792 | 4:09 |
| gka1b | 21 \| 21; 231 | 10 | -1.33000000 2 | -1.33000000 2 | 6.1-7 | 1.6-7 | 0.000 | 02 |
| gka2b | 31 \| 31; 496 | 20 | -1.21000000 2 | -1.21000000 2 | 6.3-5 | 1.3-7 | 0.000 | 33 |
| gka3b | 41 \| 41; 861 | 14 | -1.18000000 2 | -1.18000000 2 | 4.3-8 | 6.5-9 | 0.000 | 04 |
| gka4b | 51 \| 51; 1326 | 15 | -1.29000000 2 | -1.29000000 2 | 3.1-8 | 5.7-9 | 0.000 | 05 |
| gka5b | 61 \| 61; 1891 | 13 | -1.50000000 2 | -1.50000000 2 | 1.0-7 | 2.5-9 | 0.000 | 06 |
| gka6b | 71 \| 71; 2556 | 14 | -1.46000000 2 | -1.46000000 2 | 1.8-8 | 4.3-10 | 0.000 | 12 |
| gka7b | 81 \| 81; 3321 | 23 | -1.60000000 2 | -1.60000000 2 | 1.6-7 | 8.6-7 | 0.000 | 21 |
| gka8b | 91 \| 91; 4186 | 16 | -1.45000000 2 | -1.45000000 2 | 6.2-8 | 1.6-9 | 0.000 | 33 |
| gka9b | 101 \| 101; 5151 | 22 | -1.37000000 2 | -1.37000000 2 | 4.7-8 | 5.8-11 | 0.000 | 54 |
| gka10b | 126 \| 126; 8001 | 21 | -1.54000000 2 | -1.55000000 2 | **2.0-4** | 1.0-7 | 0.649 | 2:42 |
| gka1c | 41 \| 41; 861 | 30 | -5.05800000 3 | -5.11300000 3 | 5.8-6 | 2.1-7 | 1.087 | 1:19 |
| gka2c | 51 \| 51; 1326 | 27 | -6.21300000 3 | -6.32000000 3 | 3.9-8 | 6.1-7 | 1.722 | 52 |
| gka3c | 61 \| 61; 1891 | 28 | -6.66500000 3 | -6.81300000 3 | 3.4-8 | 9.7-7 | 2.221 | 32 |
| gka4c | 71 \| 71; 2556 | 27 | -7.39800000 3 | -7.56500000 3 | 5.0-7 | 9.7-7 | 2.257 | 2:09 |
| gka5c | 81 \| 81; 3321 | 33 | -7.36200000 3 | -7.57600000 3 | 2.4-6 | 3.8-7 | 2.907 | 1:13 |
| gka6c | 91 \| 91; 4186 | 32 | -5.82400000 3 | -5.96100000 3 | 2.9-5 | 4.1-7 | 2.352 | 3:52 |
| gka7c | 101 \| 101; 5151 | 31 | -7.22500000 3 | -7.31600000 3 | 1.6-5 | 4.9-7 | 1.260 | 2:35 |
| gka1d | 101 \| 101; 5151 | 33 | -6.33300000 3 | -6.52800000 3 | 8.3-6 | 8.2-8 | 3.079 | 3:52 |
| gka2d | 101 \| 101; 5151 | 29 | -6.57900000 3 | -6.99000000 3 | 4.7-7 | 8.3-7 | 6.247 | 2:38 |
| gka3d | 101 \| 101; 5151 | 30 | -9.26100000 3 | -9.73400000 3 | 1.3-5 | 3.0-7 | 5.107 | 3:37 |
| gka4d | 101 \| 101; 5151 | 28 | -1.07270000 4 | -1.12780000 4 | 4.6-7 | 6.7-7 | 5.137 | 2:53 |
| gka5d | 101 \| 101; 5151 | 29 | -1.16260000 4 | -1.23980000 4 | 4.5-7 | 7.5-7 | 6.640 | 3:02 |
| gka6d | 101 \| 101; 5151 | 29 | -1.42070000 4 | -1.49290000 4 | 7.0-7 | 8.2-7 | 5.082 | 3:10 |
| gka7d | 101 \| 101; 5151 | 28 | -1.44760000 4 | -1.53750000 4 | 1.6-7 | 9.9-7 | 6.210 | 43 |
| gka8d | 101 \| 101; 5151 | 29 | -1.63520000 4 | -1.70050000 4 | 1.1-7 | 7.3-7 | 3.993 | 1:36 |
| gka9d | 101 \| 101; 5151 | 29 | -1.56560000 4 | -1.65340000 4 | 2.5-8 | 9.8-7 | 5.608 | 1:18 |
| gka10d | 101 \| 101; 5151 | 30 | -1.91020000 4 | -2.01080000 4 | 1.7-7 | 9.3-7 | 5.266 | 51 |
| gka1e | 201 \| 201; 20301 | 32 | -1.64640000 4 | -1.70690000 4 | 5.4-5 | 2.3-7 | 3.675 | 10:47 |
| gka2e | 201 \| 201; 20301 | 30 | -2.33950000 4 | -2.49170000 4 | 3.7-8 | 7.2-7 | 6.506 | 8:06 |
| gka3e | 201 \| 201; 20301 | 30 | -2.52430000 4 | -2.68980000 4 | 3.6-6 | 4.8-7 | 6.556 | 23:30 |
| gka4e | 201 \| 201; 20301 | 29 | -3.55940000 4 | -3.72250000 4 | 2.2-7 | 7.0-7 | 4.582 | 21:51 |
| gka5e | 201 \| 201; 20301 | 32 | -3.51540000 4 | -3.80020000 4 | 1.9-5 | 1.7-7 | 8.101 | 14:14 |

Table 10: Results for the NCGAL algorithm on the BIQ problems. The entries under the column "%gap" are calculated with respect to the best solution listed, which is known to be optimal unless the symbol (†) is prefixed.

| problem | $m - n_l \mid n_s; n_l$ | it. | best upper bound | lower bound $\underline{v}$ | $R_P$ | $R_D$ | %gap | time |
|---|---|---|---|---|---|---|---|---|
| gka1f | 501 \| 501; 125751 | 33 | †-6.11940000 4 | -6.55590000 4 | 7.4-5 | 1.9-7 | 7.133 | 2:08:44 |
| gka2f | 501 \| 501; 125751 | 31 | †-1.00161000 5 | -1.07932000 5 | 6.7-5 | 1.9-7 | 7.759 | 1:54:39 |
| gka3f | 501 \| 501; 125751 | 33 | †-1.38035000 5 | -1.50151000 5 | 3.8-5 | 1.6-7 | 8.777 | 1:50:44 |
| gka4f | 501 \| 501; 125751 | 33 | †-1.72771000 5 | -1.87088000 5 | 3.0-5 | 1.6-7 | 8.287 | 1:56:51 |
| gka5f | 501 \| 501; 125751 | 33 | †-1.90507000 5 | -2.06915000 5 | 2.1-6 | 4.0-7 | 8.613 | 2:09:56 |

# 8 Conclusion and future work

In this paper, we introduced a Newton-CG augmented Lagrangian algorithm for solving semidefinite programming problems $(D)$ and $(P)$ and analyzed its convergence and rate of convergence. Our convergence analysis is based on classical results of proximal point methods [30, 31] along with recent developments on perturbation analysis of the problems under consideration. Extensive numerical experiments conducted on a variety of large scale SDPs demonstrated that our algorithm is very efficient. This opens up a way to attack problems in which a fast solver for large scale SDPs is crucial, for example, in applications within a branch-and-bound algorithm for solving hard combinatorial problems such as the quadratic assignment problems.

From the surprisingly good numerical performance of the NCGAL algorithm for linear SDP, one may expect the same to hold true for a convex quadratic SDP of the form: $\min\{\frac{1}{2}\langle X,\, Q(X)\rangle + \langle C,\, X\rangle : \mathcal{A}(X) = b, X \succeq 0\}$, where $Q$ is a linear operator defined on $\mathcal{S}^n$; or linearly constrained convex SDP of the form $\min\{f(X) + \langle C,\, X\rangle : \mathcal{A}(X) = b, X \succeq 0\}$, where $f$ is a differentiable convex function. Preliminary research on the performance of the NCGAL algorithm for the above problems indeed confirmed our expectation.

# Acknowledgements

We thank Brian Borchers for providing us the SDP data for the maximum stable set problems collected by N. Sloane [33].

# References

[1] F. Alizadeh, J. P. A. Haeberly, and O. L. Overton, *Complementarity and nondegeneracy in semidefinite programming*, Mathemtical Programming, 77 (1997), pp. 111-128.

[2] V. I. Arnold, *On matrices depending on parameters*, Russian Mathematical Surveys, 26 (1971), pp. 29-43.

[3] Z. J. Bai, D. Chu and D. F. Sun, *A dual optimization approach to inverse quadratic eigenvalue problems with partial eigenstructure*, SIAM Journal on Scientific Computing, 29 (2007), pp. 2531-2561.

[4] J. F. Bonnans and A. Shapiro, *Perturbation Analysis of Optimizaiton Problems*, Springer-Verlag, New York (2000).

[5] S. Burer, and D. Vandenbussche, *Solving Lift-and-Project relaxations of binary integer programs, SIAM Journal on Optimization*, 16 (2006), pp. 726–750.

[6] S. Burer, R. Monterio, and Y. Zhang, *A computational study of a gradient-based log-barrier algorithm for a class of large scale SDPs*, Math. Program., 95 (2003), pp. 359–379.

[7] S. Butenko, P. Pardalos, I. Sergienko, V. Shylo, and P. Stetsyuk, *Finding maximum independent sets in graphs arising from coding theory*, Symposium on Applied Computing, Proceedings of the 2002 ACM symposium on Applied computing, 2002, pp. 542–546.

[8] Z. X. Chan and D. F. Sun, *Constraint nondegeneracy, strong regularity and nonsigularity in semidefinite programming*, SIAM Journal on optimization (2008).

[9] F. H. Clarke, *Optimization and Nonsmooth Analysis*, John Wiley and Sons, New York (1983).

[10] A. Eisenblätter, M. Grötschel, and A. M. C. A. Koster, *Frequency planning and ramification of coloring*, Discuss. Math. Graph Theory, 22 (2002), pp. 51–88.

[11] F. Facchinei, *Minimization of $SC^1$ functions and the Maratos effect*, Operation Research Letters, 17 (1995), 131 - 137.

[12] C. Jansson, *Termination and verification for ill-posed semidefinite programs*, Technical Report, Informatik III, TU Hamburg-Harburg, Hamburg, June 2005.

[13] G. H. Golub and C. F. Van Loan, *Matrix Computations*, The Johns Hopkins University Press, Baltimore and London, 3rd edition, 1996.

[14] M. Kočvara and M. Stingl, *PENNON - a code for convex nonlinear and semidefinite programming*, Optimization Methods and Software, 18 (2003), pp. 317-333.

[15] M. Kočvara and M. Stingl, *On the solution of large-scale SDP problems by the modified barrier method using iterative solvers*, Mathematical Programming, 109 (2007), pp. 413-444.

[16] B. Kummer, *Newton's method for non-differentiable functions*, In: J. Guddat et al.(eds.), Advances in Mathematical Optimization, Akademie-Verlag, Berlin, (1988), pp. 114-125.

[17] J. Malick, J. Povh, F. Rendl and A. Wiegele, *Regularization methods for semidefinite programming*, preprint, 2007.

[18] F. Meng, D. F. Sun and G. Zhao, *Semismoothness of solutions to generalized equation and the Moreau-Yosida regularization*, Mathematical Programming, Series B, 104 (2005), pp. 561-581.

[19] J. S. Pang and L. Qi, *A globally convergent Newton method of convex $SC^1$ minimization problems*, Journal of Optimization Theory and Application, 85 (1995), pp. 633-648.

[20] J. S. Pang, D. F. Sun and J. Sun, *Semismooth homeomorphisms and strong stability of semidefinite and Lorentz complementarity problems*, Mathematics of Operations Research, 28 (2003), pp. 272-284.

[21] J. Povh, and F. Rendl, *Copositive and semidefinite relaxations of the quadratic assignment problem*, Technical Report, Klagenfurt University, July 2006.

[22] J. Povh, F. Rendl, and A. Wiegele, *A boundary point method to solve semidefinite programs*, Computing, 78 (2006), pp. 277-286.

[23] H. D. Qi and D. F. Sun, *A quadratically convergent Newton method for computing the nearest correlation matrix*, SIAM Journal on Matrix Analysis and Applications, 28 (2006), pp. 360-385.

[24] L. Qi and J. Sun, *A nonsmooth version of Newton's method*, Mathematical Programming, 58 (1993), pp. 353-367.

[25] S. M. Robinson, *Local structure of feasible sets in nonlinear programming, Part II: nondegeneracy*, Mathematical Programming Study, 22 (1984), pp. 217-230.

[26] S. M. Robinson, *Local structure of feasible sets in nonlinear programming, Part III: stability and sensitivity*, *Mathematical Programming Study*, 30 (1987), pp. 45-66.

[27] R. T. Rockafellar, *Convex Analysis*, Princenton University Press, New Jersey, 1970.

[28] R. T. Rockafellar, *A dual approach to solving nonlinear programming problems by unconstrained optimization*, Mathematical Programming, 5 (1973), pp. 354-373.

[29] R. T. Rockafellar, *Conjugate Duality and Optimization*, Regional Conference Series in Applied Math., 16 (1974), SIAM Publication.

[30] R. T. Rockafellar, *Monotone operators and the proximal point algorithm*, SIAM Journal on Control and Optimization, 14 (1976), pp. 877-898.

[31] R. T. Rockafellar, *Augmented Lagrangains and applications of the proximal point algorithm in convex programming*, Mathematics of Operation Research, 1 (1976), pp. 97-116.

[32] R. T. Rockafellar and R. J. -B. Wets, *Variational Analysis*, Springer, New York, 1998.

[33] N. Sloane, *Challenge Problems: Independent Sets in Graphs*, http://www.research.att.com/~njas/doc/graphs.html.

[34] D. F. Sun, *The strong second order sufficient condition and constraint nondegeneracy in nonlinear semidefinite programming and their implications*, Mathematics of Operations Research, (2006), pp. 761-776.

[35] D. F. Sun and J. Sun, *Semismooth matrix valued functions*, Mathematics of Operations Research, 27 (2002), pp. 150-169.

[36] D. F. Sun, J. Sun and L. W. Zhang, *The rate of convergence of the augmented Lagrangian method for nonlinear semidefinite programming*, Mathematical Programming (2008).

[37] L. N. Trefethen and D. Bau, *Numerical Linear Algebra*, SIAM, Philadelphia (1997).

[38] K. C. Toh, and M. Kojima, *Solving some large scale semidefinite programs via the conjugate residual method*, SIAM Journal on Optimization, 12 (2002), pp. 669–691.

[39] K. C. Toh, *Solving large scale semidefinite programs via an iterative solver on the augmented systems*, SIAM Journal on Optimization, 14 (2004), pp. 670-698.

[40] M. Trick, V. Chvatal, W. Cook, D. Johnson, C. McGeoch, and R. Tarjan, *The Second DIMACS Implementation Challenge: NP Hard Problems: Maximum Clique, Graph Coloring, and Satisfiability*, Rutgers University, http://dimacs.rutgers.edu/Challenges/ (1992).

[41] A. Wiegele, *Biq Mac library – a collection of Max-Cut and quadratic 0-1 programming instances of medium size*, Preprint, 2007.

[42] E. H. Zarantonello, *Projections on convex sets in Hilbert space and spectral theory I and II*, Contributions to Nonlinear Functional Analysis (E. H. Zarantonello, ed.), Academic Press, New York (1971), pp. 237-424.