GERAD

GROUPE D'ÉTUDES ET DE RECHERCHE
EN ANALYSE DES DÉCISIONS

Les Cahiers du GERAD

G-2012-12
March 2012

CITATION ORIGINALE / ORIGINAL CITATION

Dehghani, A.; Goffin, J.-L.; Orban, Dominique,
A primal-dual regularized interior-point method for semidefinite programming, À paraître dans: Optimization Methods and Software 2016.
DOI:10.1080/10556788.2016.1235708

GERAD HEC Montréal
3000, ch. de la Côte-Sainte-Catherine
Montréal (Québec) Canada H3T 2A7

Tél. : 514 340-6053
Téléc. : 514 340-5665
info@gerad.ca
www.gerad.ca

Les Cahiers du GERAD ISSN: 0711–2440

A Primal-Dual Regularized Interior-Point Method for Semidefinite Programming

A. Dehghani, J.-L. Goffin,
D. Orban

G–2012–12

March 2012

A Primal-Dual Regularized Interior-Point Method for Semidefinite Programming

Ahad Dehghani

GERAD & Department of Mathematics and Industrial Engineering
École Polytechnique de Montréal
Montréal (Québec) Canada
ahad.dehghani@gerad.ca

Jean-Louis Goffin

GERAD & Desautels Faculty of Management, Management Science
McGill University
Montréal (Québec) Canada
jean-louis.goffin@mcgill.ca

Dominique Orban

GERAD & Department of Mathematics and Industrial Engineering
École Polytechnique de Montréal
Montréal (Québec) Canada
dominique.orban@gerad.ca

March 2012

Les Cahiers du GERAD

G–2012–12

Abstract

Interior-point methods in semi-definite programming (SDP) require the solution of a sequence of linear systems which are used to derive the search directions. Safeguards are typically required in order to handle rank-deficient Jacobians and free variables. We propose a primal-dual regularization to the original SDP and show that it is possible to recover an optimal solution of the original SDP via inaccurate solves of a sequence of regularized SDPs for both the NT and dual HKM directions. This work is a generalization of recent work by Friedlander and Orban for quadratic programming.

Résumé

Les méthodes de points intérieurs pour l'optimisation semi-définie nécessitent la solution d'une suite de systèmes linéaires utilisés pour déterminer une direction de recherche. Des garde-fous sont habituellement mis en place pour se prémunir des situations où les contraintes ne sont pas de rang maximal et pour traiter les variables libres. Nous proposons une régularisation primale-duale du SDP d'origine et montrons qu'il est possible de retrouver une solution du SDP initiale. Ceci se fait par le biais de solutions inexactes d'une suite de SDPs régularisés pour les directions NT et HKM duale. Notre algorithme est une généralisation d'une contribution récente de Friedlander et Orban.

Acknowledgments: Research partially supported by NSERC Discovery Grant 299010-04.

1 Introduction

Semidefinite programming (SDP) has been an active topic in optimization for almost two decades. SDP has lots of applications in computational geometry, experiment design, information and communication theory, statistics, control theory, eigenvalue maximization, linear matrix inequalities, and optimal experiment design, to name a few [VB99, VB96].

It has also been recognized in combinatorial optimization as a valuable technique for obtaining bounds on the solution of NP-hard problems [Ali95, Hel00]. There are two handbooks that covers theory, algorithms, applications, and softwares for SDP's [WSV00, FB12].

In this paper, we consider a primal-dual regularization to solve the semidefinite program

$$\begin{aligned} \underset{X\in\mathcal{S}^n}{\text{minimize}} \quad & C \bullet X \\ \text{subject to} \quad & A_i \bullet X = b_i, \quad i = 1, 2, \cdots, m, \\ & X \succcurlyeq 0, \end{aligned} \tag{1}$$

where C, X, and A_i are in $\mathcal{S}^n$, the set of all symmetric matrices in $\mathbb{R}^{n\times n}$, $b_i \in \mathbb{R}$, $C \bullet X = \mathbf{tr}(CX)$ and $X \succcurlyeq 0$ means that X is positive semidefinite. The Lagrangian associated to (1) is

$$\mathcal{L}(X, y, Z) := C \bullet X - \sum_{i=1}^{m} y_i (A_i \bullet X - b_i) - Z \bullet X. \tag{2}$$

It is easy to see that $\frac{\partial}{\partial X}\mathcal{L}(X, y, Z) = 0$ if and only if $C = \sum_{i=1}^{m} y_i A_i + Z$. Substituting this value for C in (2) we get $\mathcal{L}(X, y, Z) = b^T y$ and the dual of (1) is obtained by maximizing $b^T y$ over all $Z \succcurlyeq 0$ that is

$$\begin{aligned} \underset{Z\in\mathcal{S}^n, y\in\mathbb{R}^m}{\text{maximize}} \quad & b^T y \\ \text{subject to} \quad & \sum_{i=1}^{m} y_i A_i + Z = C, \\ & Z \succcurlyeq 0. \end{aligned} \tag{3}$$

Although SDP has remarkable resemblance with linear programming(Lp) and includes Lp, strong duality does not hold in general. It does hold if at least one of the problems (1) and (3) has a strictly feasible point [Ali95]. In interior-point methods for SDP, most of the computational cost lies in the solution of a symmetric indefinite system of linear equations that determines the search direction. At each iteration, a saddle-point system—also known as a KKT system—of the following form must be solved

$$\begin{bmatrix} -\mathcal{D} & \mathcal{A}^* \\ \mathcal{A} & 0 \end{bmatrix} \begin{bmatrix} \Delta X \\ \Delta y \end{bmatrix} = \begin{bmatrix} f \\ g \end{bmatrix}, \tag{4}$$

where $\mathcal{A}$ is a linear operator on $\mathcal{S}^n$, $\mathcal{A}^*$ is its adjoint operator, and $\mathcal{D}$ is a linear operator on $\mathcal{S}^n$ not necessarily symmetric. The operator $\mathcal{D}$ and the right-hand side (f, g) change at each iteration. This system maybe solved using direct or iterative methods, or ΔX and Δy may be solved via

$$\begin{aligned} \mathcal{A}\mathcal{D}^{-1}\mathcal{A}^*\Delta y &= \mathcal{A}\mathcal{D}^{-1} f + g, \\ \mathcal{D}\Delta X &= \mathcal{A}^*\Delta y - f. \end{aligned} \tag{5}$$

2 Motivation and related works

We may cannot solve (4) and (5) efficiently near rank deficiency of $\mathcal{A}$ or near singularity of $\mathcal{D}$. The goal of this paper is to present a *primal-dual regularization* based on modifying the linear systems (4) or (5) in order to alleviate some of these difficulties. Our approach is closely connected to augmented Lagrangian methods for convex programming, except that is based on applying a single Newton iteration on each subproblem, rather than solving it to a certain level of accuracy.

In our algorithms, classical tools from convex optimization such as quadratic regularization and augmented Lagrangian techniques are used and shares many characteristics with algorithms that have already been proposed. For example: a primal-dual regularization by Friedlander and Orban [Orb10] for quadratic programming, an adaptive diagonal regularization for linear and quadratic problems implemented by Altman and Gondzio [AG99], a quadratic regularization based on Moreau-Yosida regularization [MPR+09] for SDP, and a primal regularization for conic optimization by Anjos and Burer [AB07]. They provide a convergence proof for the primal regularization scheme proposed by Mészáros [Més98] in the context of second-order cone programming:

$$\underset{x\in\mathbb{R}^n}{\text{minimize}} \quad \text{subject to} \quad Ax = b, x \in \mathcal{C},$$

where $\mathcal{C}$ is the symmetric self-dual cone. Both Mészáros [Més98] and Anjos and Burer [AB07] motivate this type of regularization as a means to handle free variables. The convergence proof of Anjos and Burer [AB07] is similar to ours but is weaker in at least three respects. The first is that it only considers primal regularization. The second is that it explicitly assumes boundedness of the Newton direction—an assumption done away with in the present paper. The third is that it assumes that A has full row rank. Yet more importantly, their regularization parameter update occurs posthumously in the sense that the parameter value is adjusted if the Newton direction just computed violates some condition. A clear disadvantage is then that each time the parameter value is changed, the Newton direction must be recomputed.

A remedy for an ill-conditioned matrix in block $(1,1)$ in (4) is that to modify $\mathcal{D}$ into $\mathcal{D}+\rho I$, where $\rho > 0$ is a regularization parameter. The new system can be interpreted as the corresponding direction-finding system for the *primal-regularized* SDP

$$\begin{aligned} &\underset{X\in\mathcal{S}^n}{\text{minimize}} && C \bullet X + \tfrac{1}{2}\rho\,\|X - X_k\|_F^2 \\ &\text{subject to} && \mathcal{A}X = b, \quad X \succcurlyeq 0, \end{aligned} \tag{6}$$

where X_k is the current iterate and $\|.\|_F$ is the Frobenius norm.

We face similar difficulties when $\mathcal{A}$ is nearly rank deficient. In this case, the factorizations are not numerically stable and the dual problem (3) does not achieve a unique solution. A remedy of this difficulty is also to add δI to the $(2,2)$ block of (4). The new system can be interpreted as the direction finding-system for the *dual-regularized* SDP

$$\begin{aligned} &\underset{y,Z}{\text{maximize}} && b^T y - \tfrac{1}{2}\delta\|y - y_k\|^2 \\ &\text{subject to} && \mathcal{A}^* y + Z = C, \quad Z \succcurlyeq 0, \end{aligned} \tag{7}$$

where here and throughout $\|\cdot\|$ means the Euclidian norm on $\mathbb{R}^m$. Let X^* and (y^*, Z^*) be solutions of (1)–(3), respectively. It is not difficult to see that X^* and (y^*, Z^*) are also the unique solutions of (6)–(7) in which we set $X_k = X^*$ and $y_k = y^*$ for any nonnegative δ and ρ. Friedlander and Tseng [FT07] term this property exact regularization. In sense above, we can say that these regularized problems are *exact*. We will see that our primal-dual regularized problems are also exact in the next section.

A primal-dual regularization can be obtained by modifying both block $(1,1)$ and $(2,2)$ in (4) simultaneously to get

$$K \begin{bmatrix} \Delta X \\ \Delta y \end{bmatrix} = \begin{bmatrix} f \\ g \end{bmatrix} \quad \text{where} \quad K := \begin{bmatrix} -(\mathcal{D}+\rho I) & \mathcal{A}^* \\ \mathcal{A} & \delta I \end{bmatrix}, \tag{8}$$

called *augmented system*. We derive this system in Section 4 and show that (8) is obtained by using simultaneous regularization to (1)–(3), although the form of simultaneous primal-dual regularization is different than that given by (6)–(7). It worth mentioning that (6)–(7) is not a primal-dual pair, although (8) can be interpreted as a direction-finding system for a *primal-dual regularized* SDP.

The matrix K appears in (8) belongs to the class of symmetric quasi-definite (SQD) matrices and is strongly factorizable [Van94, Van95]. That is for any permutation matrix P, the indefinite matrix PKP^T possesses a Cholesky-type factorization $L\Delta L^T$ where L is unit lower triangular and Δ is diagonal.

The stability of linear system with SQD matrices is analyzed by Gill et al. [GSS96]; Saunders [Sau95, S+96] investigate the use of SQD matrices within interior-point methods for linear programming.

We note that (8) is equivalent to

$$\begin{aligned} \left(\mathcal{A}(\mathcal{D}+\rho I)^{-1}\mathcal{A}^* + \delta I\right)\Delta y &= \mathcal{A}(\mathcal{D}+\rho I)^{-1} f + g, \\ (\mathcal{D}+\rho I)\Delta X &= (\mathcal{D}+\rho I)^{-1}\mathcal{A}^*\Delta y - f. \end{aligned} \tag{9}$$

known as *Schur complement system.* It is clear that (5) can be recovered from (9) by setting $\delta = \rho = 0$. We state and analyze our algorithms based on (8) and test our algorithm by adding regularization parameters ρ and δ to SDPT3 solver [TTT].

3 Notation

For any given symmetric matrix $G \in \mathbb{R}^{n\times n}$, we use $\lambda_n(G)$ and $\lambda_1(G)$ to denote the smallest and the largest eigenvalues of G, respectively. Similarly if G is not symmetric we use $\sigma_n(G)$ and $\sigma_1(G)$ to denote the smallest and the largest singular values of G, respectively.

In order to work with matrices instead of operators we use the following notation. For any $m \times n$ matrix A, $\mathbf{vec}A$ denotes the mn-vector obtained from stacking the columns of A on top of one another. For matrices $A_i \in \mathcal{S}^n (i = 1, 2, \cdots m)$ we use

$$\mathbf{A}^T = [\mathbf{vec}A_1 \ \mathbf{vec}A_2 \ \cdots \ \mathbf{vec}A_m].$$

For any two matrices $B \in \mathbb{R}^{m\times n}$ and $C \in \mathbb{R}^{p\times q}$, the Kronecker product of B and C denoted by $B \otimes C$ and defined as a block matrix consisting mn blocks for which the $ij-$th block is $b_{ij}C$. To see some properties of Kronecker product used in this paper see Appendix A.

4 A Primal-Dual regularization

SDP's are convex problem in the sense that minimizing a linear function over a convex set, but they are not convex in the sense of nonlinear programming, because condition $X \succcurlyeq 0$ generates some nonconvex constraints. In this section, we first state some definitions from [D.G97] to generalize the notion of inequality, convexity, and convex programs in vector spaces and then state our primal-dual regularization for SDP's.

Definition 1 (Generalized inequality) *Let P be a convex cone in a vector space V. For any $x, y \in V$ we write $x \succcurlyeq_P y$ if $x - y \in P$. The cone P defining this relation is called the positive cone in V. To simplify notation and whenever there is no ambiguity, we omit the index P and write $x \succcurlyeq y$ for $x \succcurlyeq_P y$. The cone P is said to be a pointed cone if whenever $x \in P$ and $-x \in P$ then $x = 0$.*

One can easily verify that $x \succcurlyeq y$, and $y \succcurlyeq z$ imply $x \succcurlyeq z$, and since $0 \in P$, $x \succcurlyeq x$ for all $x \in V$. Moreover, if P is pointed then $x \succcurlyeq y$ and $y \succcurlyeq x$ imply $x = y$. Since we have a generalized definition of inequality between vectors, it is possible to generalize the notion of convexity for mappings.

Definition 2 (Generalized convexity) *Let V be a vector space and let W be a vector space having a cone P specified as the positive cone. A mapping $g : V \to W$ is said to be convex if the domain Ω of g is a convex set and if for all $0 \le \alpha \le 1$*

$$g(\alpha x_1 + (1-\alpha)x_2) \preccurlyeq \alpha g(x_1) + (1-\alpha)g(x_2) \quad \text{for all } x_1, x_2 \in \Omega.$$

We note that the convexity is not an intrinsic property of a mapping but it also depends on the specified positive cone. When $V = \mathbb{R}^n$ and

$$P = \{x \in \mathbb{R}^n \mid x_i \ge 0, \ i = 1, 2, \cdots, n\}$$

then the generalized convexity of Definition 2 becomes the usual definition of convexity of functions on $\mathbb{R}^n$. Now, a general convex program can be stated as

$$\begin{aligned} \underset{x\in\Omega}{\text{minimize}} \quad & f(x) \\ \text{subject to} \quad & g(x) \preccurlyeq 0 \\ & Bx = b. \end{aligned} \tag{10}$$

where f and g are convex functions on convex set $\Omega \subset \mathbb{R}^n$, $B \in \mathbb{R}^{m\times n}$, $x \in \mathbb{R}^n$, and $b \in \mathbb{R}^m$. In the sense above, the SDP problem (1) and the following regularization of (1) are convex programs

$$\begin{aligned} \underset{X\in\mathcal{S}^n, r\in\mathbb{R}^m}{\text{minimize}} \quad & C \bullet X + \tfrac{1}{2}\rho \|X - X_k\|_F^2 + \tfrac{1}{2}\delta\|r + y_k\|^2 \\ \text{subject to} \quad & A_i \bullet X + \delta r_i = b_i \quad i = 1, 2, \cdots, m, \\ & X \succcurlyeq 0, \end{aligned} \tag{11}$$

where $\rho \geq 0$ and $\delta \geq 0$ are regularization parameters, and X_k and y_k are current estimates of primal and dual solutions of (1), X^* and y^*, respectively. Note that $\rho = 0$ and $\delta = 0$ recovers the original problem statement (1). It is a convex problem in the variable (X, r) and always strictly feasible when $\delta > 0$. The dual of this problem can be derived by the general duality theory improved for general convex problems in [D.G97, Chapter 8, Theorem 1]. The Lagrangian of (11) is

$$\mathcal{L}_{\rho,\delta}(X, r, y, Z) := C \bullet X + \tfrac{1}{2}\rho \|X - X_k\|_F^2 + \tfrac{1}{2}\delta\|r + y_k\|^2 - \sum_{i=1}^{m} y_i(A_i \bullet X + \delta r_i - b_i) - Z \bullet X. \tag{12}$$

Since $\mathcal{L}_{\rho,\delta}(X, r, y, Z)$ is convex with respect to (X, r), minimizing it is equivalent to setting its derivative to zero, that is

$$\frac{\partial}{\partial X}\mathcal{L}_{\rho,\delta}(X, r, y, Z) = C + \rho(X - X_k) - \sum_{i=1}^{m} y_i A_i - Z = 0, \tag{13}$$

$$\frac{\partial}{\partial r}\mathcal{L}_{\rho,\delta}(X, r, y, Z) = r + y_k - y = 0. \tag{14}$$

Plugging (13) and (14) into (12) and introducing $S := X - X_k$, we get

$$\begin{aligned} \mathcal{L}_{\rho,\delta}(X, r, y, Z) &= \left(-\rho S + \sum_{i=1}^{m} y_i A_i + Z\right) \bullet X + \tfrac{1}{2}\rho \|S\|_F^2 \\ &\quad + \delta\|y\|^2 - \sum_{i=1}^{m} y_i A_i \bullet X - \delta\|y\|^2 + \delta y^T y_k + b^T y - Z \bullet X \\ &= -\rho S \bullet X + \tfrac{1}{2}\rho \|S\|_F^2 - \tfrac{1}{2}\delta\|y - y_k\|^2 - \tfrac{1}{2}\delta\|y_k\| + b^T y \\ &= -\rho S \bullet (S + X_k) + \tfrac{1}{2}\rho \|S\|_F^2 - \tfrac{1}{2}\delta\|y - y_k\|^2 - \tfrac{1}{2}\delta\|y_k\| + b^T y \\ &= -\tfrac{1}{2}\rho (S \bullet S + 2S \bullet X_k) - \tfrac{1}{2}\delta\|y - y_k\|^2 - \tfrac{1}{2}\delta\|y_k\| + b^T y \\ &= -\tfrac{1}{2}\rho \|S + X_k\|_F^2 + \tfrac{1}{2}\rho \|X_k\|_F^2 - \tfrac{1}{2}\delta\|y - y_k\|^2 - \tfrac{1}{2}\delta\|y_k\| + b^T y \\ &= b^T y - \tfrac{1}{2}\rho \|S + X_k\|_F^2 - \tfrac{1}{2}\delta\|y - y_k\|^2 + \tfrac{1}{2}\rho \|X_k\|_F^2 - \tfrac{1}{2}\delta\|y_k\|^2. \end{aligned}$$

Since the last two terms in the last equality are constant, the dual of (11) is

$$\begin{aligned} \underset{S,y,Z}{\text{maximize}} \quad & b^T y - \tfrac{1}{2}\rho \|S + X_k\|_F^2 - \tfrac{1}{2}\delta\|y - y_k\|^2 \\ \text{subject to} \quad & \sum_{i=1}^{m} y_i A_i + Z - \rho S = C \\ & Z \succcurlyeq 0. \end{aligned} \tag{15}$$

We note that if we let $\rho = \delta = 0$ we can recover (1) and (3) from the regularized problems (11) and (15).

4.1 Relation between primal-dual regularized problems and augmented Lagrangian method

There is a nice relation between augmented-Lagrangian approach and ours. Here, we show that it is possible to derive the regularized problems (11)–(15) by applying augmented Lagrangian on (6)–(7). The augmented Lagrangian of (6) defined as

$$\begin{aligned}\mathcal{L}_{au}(X,y,\delta) &= C\bullet X + \tfrac{1}{2}\rho\,\|X-X_k\|_F^2 + y_k^T(b-\mathcal{A}X) + \tfrac{1}{2\delta}\|b-\mathcal{A}X\|^2\\ &= C\bullet X + \tfrac{1}{2}\rho\,\|X-X_k\|_F^2 + y_k^T\bar{r} + \tfrac{1}{2\delta}\|\bar{r}\|^2,\end{aligned}$$

where $\bar{r}=b-\mathcal{A}X$, $\delta>0$ is a penalty parameter, and y_k is the current estimate of the vector of Lagrangian multipliers associated to equality constraints. Augmented Lagrangian method [NW06, Chapter 17] solves a sequence of problems of the form

$$\begin{aligned}\underset{X,\bar{r}}{\text{minimize}}\quad & C\bullet X + \tfrac{1}{2}\rho\,\|X-X_k\|_F^2 + y_k^T\bar{r} + \tfrac{1}{2\delta}\|\bar{r}\|^2\\ \text{subject to}\quad & \mathcal{A}X+\bar{r}=b\\ & Z\succcurlyeq 0.\end{aligned}\tag{16}$$

By introducing $r:=\frac{1}{\delta}\bar{r}$ (16) can be stated as

$$\begin{aligned}\underset{X,r}{\text{minimize}}\quad & C\bullet X + \tfrac{1}{2}\rho\,\|X-X_k\|_F^2 + \tfrac{1}{2}\delta\|r+y_k\|^2\\ \text{subject to}\quad & \mathcal{A}X+\delta r=b\\ & Z\succcurlyeq 0,\end{aligned}\tag{17}$$

which is exactly (11). Therefore, solving (11) is the same as applying augmented lagrangian method on the primal regularized subproblem. Similarly, dual problem (15) can be viewed as applying augmented lagrangian on dual regularized problem (7). If we choose $X_k=X^*$ and $(y_k,Z_k)=(y^*,Z^*)$ then feasibility of X^* implies $r=\bar{r}=0$ and consequently (16) and (17) become (6). By the discussion followed (7) we conclude that primal-dual regularized problems are also exact.

5 An Interior-Point method for the regularized SDP

In this section, we develop a primal-dual interior point method to solve the primal and dual regularization (11) and (15) of (1) and (3) simultaneously. We note that problem (11) and (15) are always strictly feasible when δ and ρ are positive. Therefore, we can eliminate the usual assumption in SDP that the primal or dual problems must have a strictly feasible solutions, i.e, strong duality always holds between (11) and (15). The logarithmic barrier problem associated to (11) is

$$\begin{aligned}\underset{X\in\mathcal{S}^n,r\in\mathbb{R}^m}{\text{minimize}}\quad & C\bullet X + \tfrac{1}{2}\rho\,\|X-X_k\|_F^2 + \tfrac{1}{2}\delta\|r+y_k\|^2 - \mu\log(\det X)\\ \text{subject to}\quad & A_i\bullet X+\delta r_i=b_i \quad i=1,2,\cdots,m,\end{aligned}\tag{18}$$

where μ is a positive real number called the *barrier parameter*. For each $\mu>0$, there is a corresponding Lagrangian:

$$\mathcal{L}_\mu(X,r,y,Z) = C\bullet X + \tfrac{1}{2}\rho\,\|X-X_k\|_F^2 + \tfrac{1}{2}\delta\|r+y_k\|^2 - \mu(\log\det X) - \sum_{i=1}^{m}(y_iA_i\bullet X+\delta y_ir_i-y_ib_i)\,. \tag{19}$$

The first-order optimality conditions for (18) are obtained as:

$$\nabla_X \mathcal{L}_\mu = C + \rho S - \mu X^{-1} - \sum_{i=1}^{m} y_i A_i = 0 \tag{20a}$$

$$\nabla_r \mathcal{L}_\mu = \delta(r + y_k) - \delta y = 0 \tag{20b}$$

$$\nabla_y \mathcal{L}_\mu = -A_i \bullet X - \delta r_i + b_i = 0 \quad i = 1, 2, \cdots, m, \tag{20c}$$

$$X \succ 0. \tag{20d}$$

The strict convexity of (18) implies that there exists a unique solution to (20). The *central path* is defined as the set of all such solutions when $\mu > 0$ varies. If we let $Z := \mu X^{-1}$ then the optimality conditions (20) can be written as

$$\Psi(w) = \Psi(X, r, S, y, Z) := \begin{bmatrix} C + \rho S - Z - \sum_{i=1}^{m} y_i A_i \\ \delta(r + y_k) - \delta y \\ \rho X - \rho(S + X_k) \\ A_1 \bullet X + \delta r_1 - b_1 \\ \vdots \\ A_m \bullet X + \delta r_m - b_m \\ XZ - \mu I \end{bmatrix} = 0, \tag{21}$$

and $(X, Z) \succ 0$ implicitly. One can easily verify that the same optimality conditions can be derived by working with the barrier problem associated with the dual regularized problem (15).

We can simplify the notation slightly by defining the operator $\mathcal{A} : \mathcal{S}^n \to \mathbb{R}^m$ as

$$(\mathcal{A}X)_i = A_i \bullet X, \qquad i = 1, 2, \cdots, m.$$

The adjoint of $\mathcal{A}$ is $\mathcal{A}^* : \mathbb{R}^m \to \mathcal{S}^n$ satisfying

$$\mathcal{A}^* y = \sum_{i=1}^{m} y_i A_i.$$

Using this notation and since $S = X - X_k$, we can rewrite (21) more compactly as

$$C + \rho S - \mathcal{A}^* y - Z = 0 \tag{22a}$$

$$\delta(r + y_k) - \delta y = 0 \tag{22b}$$

$$\rho X - \rho(S + X_k) = 0 \tag{22c}$$

$$\mathcal{A}X + \delta r - b = 0 \tag{22d}$$

$$XZ - \mu I = 0. \tag{22e}$$

Let (X_k, y_k) be temporarily fixed. A primal-dual interior-point method applied to the regularized problems (11) and (15) is based on applying Newton's method to a sequence of nonlinear systems of the form

$$\Psi_k(w) = \Psi_k(X, r, S, y, Z) = \begin{bmatrix} C + \rho S - \mathcal{A}^* y - Z \\ \delta(r + y_k) - \delta y \\ \rho X - \rho(S + X_k) \\ \mathcal{A}X + \delta r - b \\ XZ - \sigma \mu_k I \end{bmatrix} = 0, \quad X \succ 0, Z \succ 0, \tag{23}$$

where $\sigma \in [0, 1]$ is the centering parameter, and $\mu_k = X_k \bullet Z_k / n > 0$ is the current duality measure. When Newton's method applied to (23) direction $(\Delta X, \Delta r, \Delta S, \Delta y, \Delta Z)$ is obtained. It is clear from (26) that if ΔX is symmetric then both ΔZ and ΔS are symmetric too. There are two approaches, which result in symmetric directions ΔX. First, since Ψ_k maps a point $w \in \mathcal{S}^n \times \mathbb{R}^m \times \mathcal{S}^n \times \mathbb{R}^m \times \mathcal{S}^n$ to a point in $\mathcal{S}^n \times \mathbb{R}^m \times \mathcal{S}^n \times \mathbb{R}^m \times \mathbb{R}^{n \times n}$, XZ is usually not symmetric even if X and Z are. In this case, a linearization of $XZ - \sigma \mu_k I$ is transformed by an operator $H : \mathbb{R}^{n \times n} \to \mathcal{S}^n$ and then apply Newton's method to the new system [R.D97, MZ98, AHO98, Zha98]. Second, (23) is solved directly without using any transformation and then ΔX is symmetrized [HRVW96, KSH97]. In this case, the modified direction may be neither a Newton direction nor a descent direction. Therefore, symmetrization should be done in such a way that the modified direction at least be a descent direction. In this paper, we follow the former strategy. A Newton step for (23) when the last row replaced by $H(XZ) = 0$ from the current iterate w_k solves the system

$$\begin{bmatrix} 0 & 0 & \rho I & -\mathcal{A}^* & -I \\ 0 & \delta I & 0 & -\delta I & 0 \\ \rho I & 0 & -\rho I & 0 & 0 \\ \mathcal{A} & \delta I & 0 & 0 & 0 \\ \mathcal{E}_k & 0 & 0 & 0 & \mathcal{F}_k \end{bmatrix} \begin{bmatrix} \Delta X \\ \Delta r \\ \Delta S \\ \Delta y \\ \Delta Z \end{bmatrix} = - \begin{bmatrix} C + \rho S - \mathcal{A}^* y - Z \\ \delta(r + y_k) - \delta y \\ \rho X - \rho(S + X_k) \\ \mathcal{A}X + \delta r - b \\ H(XZ) \end{bmatrix} = 0, \tag{24}$$

where $\mathcal{E}_k := \frac{\partial}{\partial X} H(X_k Z_k)$ and $\mathcal{F}_k := \frac{\partial}{\partial Z} H(X_k Z_k)$. Reducing (24) by eliminating Δr and ΔS, we obtain

$$\begin{bmatrix} -\rho I & \mathcal{A}^* & I \\ \mathcal{A} & \delta I & 0 \\ \mathcal{E}_k & 0 & \mathcal{F}_k \end{bmatrix} \begin{bmatrix} \Delta X \\ \Delta y \\ \Delta Z \end{bmatrix} = \begin{pmatrix} C - \mathcal{A}^* y_k - Z_k \\ b - \mathcal{A} X_k \\ -H(X_k Z_k) \end{pmatrix}, \tag{25}$$

with

$$\Delta r = -r + \Delta y, \tag{26a}$$
$$\Delta S = -S + \Delta X, \tag{26b}$$
$$\Delta Z = C - \mathcal{A}^* y_k - Z_k - \mathcal{A}^* \Delta y + \rho \Delta X. \tag{26c}$$

If $\mathcal{F}$ is a nonsingular operator on $\mathcal{S}^n$ we can eliminate ΔZ from the last row to obtain

$$\Delta Z = -\mathcal{F}_k^{-1} \left(H(X_k Z_k) + \mathcal{E}_k \Delta X \right) \tag{27}$$

and

$$\begin{bmatrix} -\left(\mathcal{F}_k^{-1}\mathcal{E}_k + \rho I\right) & \mathcal{A}^* \\ \mathcal{A} & \delta I \end{bmatrix} \begin{bmatrix} \Delta X \\ \Delta y \end{bmatrix} = \begin{bmatrix} R_d - \mathcal{F}_k^{-1} R_C \\ r_p \end{bmatrix}, \tag{28}$$

where $R_d := C - \mathcal{A}^* y_k - Z_k$, $r_p := b - \mathcal{A}X_k$, and $R_C := -H(X_k Z_k)$

If we work directly with the original primal and dual problems (1) and (3), we get

$$\begin{bmatrix} -\mathcal{F}_k^{-1}\mathcal{E}_k & \mathcal{A}^* \\ \mathcal{A} & 0 \end{bmatrix} \begin{bmatrix} \Delta X \\ \Delta y \end{bmatrix} = \begin{bmatrix} R_d - \mathcal{F}_k^{-1} R_C \\ r_p \end{bmatrix}, \tag{29}$$

which is is precisely (28) with $\rho = \delta = 0$.

Motivated by the works of Alizadeh, Haeberly, and Overton [AHO98] and Monterio [R.D97], Zhang [Zha98] introduced a general symmetrization scheme based on so-called *similar symmetrization* operator $H_P : \mathbb{R}^{n\times n} \to \mathcal{S}^n$ defined as

$$H_P(M) = \tfrac{1}{2}[PMP^{-1} + (PMP^{-1})^T], \tag{30}$$

where P is some nonsingular matrix. Different choices of P lead to different Newton directions For example, Alizadeh et al [AHO98] set $P = I$ to find the direction known the AHO direction. Zhang [Zha98] uses $P = Z^{\frac{1}{2}}$ and establishes the complexity analysis of some path following methods, including an infeasible long-step path-following method. Monteiro [R.D97] uses $P = Z^{\frac{1}{2}}$ and $P = X^{-\frac{1}{2}}$ to establish polynomial complexity of the short- step feasible path-following method. There are many other choices of P as well as other kinds of symmetrization of the linearization of the $XZ - \mu I$ in order to find a suitable search direction for SDP problems. For a comprehensive discussion of these search directions we refer the reader to [M.J99]. In this paper our analysis is based on two well-known directions: the dual HKM direction, with $P = X^{-\frac{1}{2}}$, and the Nesterov-Todd (NT) direction, with $P = W^{-\frac{1}{2}}$, where

$$W = X^{\frac{1}{2}}(X^{\frac{1}{2}} Z X^{\frac{1}{2}})^{-\frac{1}{2}} X^{\frac{1}{2}} = Z^{-\frac{1}{2}}(Z^{\frac{1}{2}} X Z^{\frac{1}{2}})^{\frac{1}{2}} Z^{-\frac{1}{2}}. \tag{31}$$

In order to cast (25) in matrix form, we need to represent the operators $\mathcal{E}$ and $\mathcal{F}$ in matrix form. The first and the second rows of (25) are equivalent to

$$-\rho \mathbf{vec}\Delta X + \mathbf{A}^T \Delta y + \mathbf{vec}\Delta Z = \mathbf{vec}R_d, \tag{32}$$
$$\mathbf{A}^T \mathbf{vec}\Delta X + \delta \Delta y = r_p, \tag{33}$$

respectively, where

$$\mathbf{vec}R_d = \mathbf{vec}C - \mathbf{A}^T y - \mathbf{vec}Z, \tag{34}$$
$$r_p = b - \mathbf{A}\mathbf{vec}X. \tag{35}$$

In the next section, we use some properties of the Kronecker product (see Appendix A) to translate the third row of (25) in matrix form.

6 Linearization

The most obvious linearization of $XZ - \sigma\mu_k I = 0$ is

$$XZ + \Delta XZ + X\Delta Z = \sigma\mu_k I. \tag{36}$$

If we apply the transformation (30) with $P = X^{-\frac{1}{2}}$ to (36), and pre and postmultiplying the equation $H_P(XZ + X\Delta Z + \Delta XZ) = \sigma\mu_k I$ by $X^{\frac{1}{2}}$, we obtain

$$2X(\Delta Z)X + XZ(\Delta X) + (\Delta X)ZX = R_c, \tag{37}$$

where

$$R_c = 2(\sigma\mu_k X - XZX) = 2X^{\frac{1}{2}}(\sigma\mu_k I - X^{\frac{1}{2}}ZX^{\frac{1}{2}})X^{\frac{1}{2}}. \tag{38}$$

Using (91b) in Appendix A we can write (37) in Kronecker product notation as follows:

$$2(X \otimes X)\mathbf{vec}\Delta Z + (XZ \otimes I + I \otimes XZ)\mathbf{vec}\Delta X = \mathbf{vec}R_c. \tag{39}$$

If we define

$$E = XZ \otimes I + I \otimes XZ \tag{40}$$
$$F = 2X \otimes X, \tag{41}$$

then (39) can be written as

$$E\mathbf{vec}\Delta X + F\mathbf{vec}\Delta Z = \mathbf{vec}R_c. \tag{42}$$

Using equations (32), (33), and (42) we can write the optimality condition (25) at $w = w_k$ in matrix form:

$$\begin{bmatrix} -\rho I & \mathbf{A}^T & I \\ \mathbf{A} & \delta I & 0 \\ E & 0 & F \end{bmatrix} \begin{bmatrix} \mathbf{vec}\Delta X \\ \Delta y \\ \mathbf{vec}\Delta Z \end{bmatrix} = \begin{pmatrix} \mathbf{vec}R_d \\ r_p \\ \mathbf{vec}R_c \end{pmatrix}. \tag{43}$$

Eliminating $\mathbf{vec}\Delta Z$ the following system is obtained

$$\begin{bmatrix} -(F^{-1}E + \rho I) & \mathbf{A}^T \\ \mathbf{A} & \delta I \end{bmatrix} \begin{bmatrix} \mathbf{vec}\Delta X \\ \Delta y \end{bmatrix} = \begin{bmatrix} \mathbf{vec}R_d - F^{-1}\mathbf{vec}R_c \\ r_p \end{bmatrix}, \tag{44}$$

where the remaining search directions are recovered via (26a)–(26c). Dropping index k (26c) has matrix form

$$\mathbf{vec}\Delta Z = \mathbf{vec}C - \mathbf{A}^T y - \mathbf{vec}Z - \mathbf{A}^T \Delta y + \rho \mathbf{vec}\Delta X. \tag{45}$$

We note that $F^{-1}E$ and F are symmetric and positive definite, but E may not be symmetric. Using the definitions of $F, \mathbf{vec}R_d$, and $\mathbf{vec}R_c$ we have

$$\begin{aligned}
\mathbf{vec}R_d - F^{-1}\mathbf{vec}R_c &= \mathbf{vec}C - \mathbf{A}^T y - \mathbf{vec}Z - (X^{-1} \otimes X^{-1})\mathbf{vec}R_c \\
&= \mathbf{vec}C - \mathbf{A}^T y - \mathbf{vec}Z - (X^{-1} \otimes X^{-1})\mathbf{vec}(\sigma\mu X - XZX) \\
&= \mathbf{vec}C - \mathbf{A}^T y - \mathbf{vec}Z - \mathbf{vec}(\sigma\mu X^{-1} - Z) \\
&= \mathbf{vec}C - \mathbf{A}^T y - \sigma\mu\mathbf{vec}X^{-1}.
\end{aligned} \tag{46}$$

Therefore, we can rewrite (44) as

$$\begin{bmatrix} -(F^{-1}E + \rho I) & \mathbf{A}^T \\ \mathbf{A} & \delta I \end{bmatrix} \begin{bmatrix} \mathbf{vec}\Delta X \\ \Delta y \end{bmatrix} = \begin{bmatrix} \mathbf{vec}C - \mathbf{A}^T y - \sigma\mu\mathbf{vec}X^{-1} \\ r_p \end{bmatrix}. \tag{47}$$

When the scaling matrix $P = W^{-\frac{1}{2}}$ is used, Todd et al [TTT98] show that the optimality conditions of primal-dual pair (1)–(3) have the following form

$$\mathcal{A}^*\Delta y + \Delta Z = 0, \tag{48a}$$
$$\mathcal{A}\Delta X = 0, \tag{48b}$$
$$W^{-1}\Delta X W^{-1} + \Delta Z = \sigma\mu X^{-1} - Z. \tag{48c}$$

Equivalently, in matrix form,

$$\begin{bmatrix} 0 & \mathbf{A}^T & I \\ \mathbf{A} & 0 & 0 \\ \bar{E} & 0 & I \end{bmatrix} \begin{bmatrix} \mathbf{vec}\Delta X \\ \Delta y \\ \mathbf{vec}\Delta Z \end{bmatrix} = \begin{pmatrix} 0 \\ 0 \\ \mathbf{vec}(\sigma\mu X^{-1} - Z) \end{pmatrix}, \tag{49}$$

where $\bar{E} := W^{-1} \otimes W^{-1}$. If we consider the regularized system corresponding to (49) we get

$$\begin{bmatrix} -\rho I & \mathbf{A}^T & I \\ \mathbf{A} & \delta I & 0 \\ \bar{E} & 0 & I \end{bmatrix} \begin{bmatrix} \mathbf{vec}\Delta X \\ \Delta y \\ \mathbf{vec}\Delta Z \end{bmatrix} = \begin{pmatrix} \mathbf{vec}R_d \\ r_p \\ \mathbf{vec}(\sigma\mu X^{-1} - Z) \end{pmatrix}. \tag{50}$$

Eliminating $\mathbf{vec}\Delta Z$, we get the following system, similar to (47)

$$\begin{bmatrix} -(\bar{E} + \rho I) & \mathbf{A}^T \\ \mathbf{A} & \delta I \end{bmatrix} \begin{bmatrix} \mathbf{vec}\Delta X \\ \Delta y \end{bmatrix} = \begin{bmatrix} \mathbf{vec}(\sigma\mu X^{-1} - Z) \\ r_p \end{bmatrix}. \tag{51}$$

the remaining search directions are recovered from (26). We use (47) and (51) to design our algorithms.

7 A long-step path-following interior-point method

Let $M \in \mathbb{R}^{n\times n}$ be a real matrix. We denote the vector of eigenvalues of M by $\lambda(M) \in \mathbb{C}^n$. If all eigenvalues are real we assume the order

$$\lambda_1(M) \geq \lambda_2(M) \geq \cdots \geq \lambda_n(M).$$

We also define

$$X(\alpha) := X + \alpha\Delta X, \quad Z(\alpha) := Z + \alpha\Delta Z, \quad \mu(\alpha) := \frac{X(\alpha) \bullet Z(\alpha)}{n}.$$

It is not difficult to see that for any matrix $M \in \mathbb{R}^{n\times n}$ with real eigenvalues (e.g., $M = XZ$ with $X, Z \succ 0$) and for any nonsingular matrix $P \in \mathbb{R}^{n\times n}$ and scalar $\tau \in \mathbb{R}$,

$$H_P(M) = \mu I \quad \text{if and only if} \quad M = \mu I,$$

we refer to [Zha98, Proposition 4.1] for the proof. Hence the central path $XZ = \mu I = \frac{X\bullet Z}{n} I$ is equivalently described as $\{(X, y, Z)) \mid H_P(XZ) = \mu I\}$.

Corresponding to one of the most popular centrality condition in linear programming,

$$\gamma_{C}\mu \le x_i z_i \le \bar{\gamma}_{C}\mu, \qquad i = 1, 2, \cdots, n,$$

in SDP we require

$$\gamma_{C}\mu \le \lambda_i(H_P(XZ)) \le \bar{\gamma}_{C}\mu, \qquad i = 1, 2, \cdots, n,$$

where $\bar{\gamma}_{C}$ can be $+\infty$, in which case the condition is one-sided.

Path-following methods generate iterates that remain within a neighborhood of the central path. We define a neighborhood $\mathcal{N}_k$ of the central path as the set of points (X, r, S, y, Z) that satisfy a specific *subset* of the conditions

$$\gamma_{C}\mu e \le \lambda(H_P(XZ)) \le \bar{\gamma}_{C}\mu e, \tag{52a}$$

$$\|\mathbf{A}\mathbf{vec}X + \delta_k r - b\| \le \gamma_{P}\mu, \tag{52b}$$

$$\|\mathbf{vec}C + \rho_k\mathbf{vec}S - \mathbf{A}^T y - \mathbf{vec}Z\| \le \gamma_{D}\mu, \tag{52c}$$

$$\|\delta_k(r + y_k) - \delta_k y\| \le \gamma_{R}\mu, \tag{52d}$$

$$\|\rho_k\mathbf{vec}X - \rho_k\mathbf{vec}(S + X_k)\| \le \gamma_{S}\mu, \tag{52e}$$

where $0 < \gamma_{C} < 1 < \bar{\gamma}_{C} \le +\infty$ and $(\gamma_{P}, \gamma_{D}, \gamma_{R}, \gamma_{S}) > 0$ are given constants. In particular, to stay within this neighborhood, a steplength α must satisfy

$$\gamma_{C}\mu(\alpha)e \le \lambda(H_P(X(\alpha)Z(\alpha))) \le \bar{\gamma}_{C}\mu(\alpha)e. \tag{53}$$

The following lemma demonstrates some nice properties of the transformation H_P.

Lemma 1 *Let $X, Z \in \mathbb{R}^{n\times n}$ be symmetric, $P \in \mathbb{R}^{n\times n}$ nonsingular, and $M \in \mathbb{R}^{n\times n}$ then*

1. *If X and Z are positive definite then $\lambda(H_P(XZ)) = \lambda(XZ)$ for both $P = X^{-\frac{1}{2}}$ and $P = W^{-\frac{1}{2}}$*
2. $\mathbf{tr}(H_P(M)) = \mathbf{tr}(M)$
3. $\|H_P(M)\|_F \le \|M\|_F$

Proof. When $P = X^{-\frac{1}{2}}$ we have $H_P(XZ) = \frac{1}{2}\{X^{-\frac{1}{2}}XZX^{\frac{1}{2}} + X^{\frac{1}{2}}ZXX^{-\frac{1}{2}}\} = X^{\frac{1}{2}}ZX^{\frac{1}{2}}$ and the result is obtained by using (91g) in Appendix A. When $P = W^{-\frac{1}{2}}$ (31) implies that $WZ = XW^{-1}$ and $P^TP = W^{-1}$. Therefore, W and P are symmetric and

$$\begin{aligned} H_p(XZ) &= \tfrac{1}{2}(PXZP^{-1} + P^{-1}ZXP) = \tfrac{1}{2}P(XZ + P^{-2}ZXP^2)P^{-1} \\ &= \tfrac{1}{2}P(XZ + WZXW^{-1})P^{-1} = \tfrac{1}{2}P(XZ + WZWZ)P^{-1} \\ &= P(XZ)P^{-1}. \end{aligned}$$

The above similarity property finishes the proof of the first item. The proofs of the second and third items of the lemma are obvious. □

The following lemma is due to [TTT98, Theorem 3.5] and state the relation between (48c) and the linearization of central path.

Lemma 2 *ΔX and ΔZ satisfy (48c) if and only if ΔX and ΔZ satisfy $H_P(XZ + \Delta XZ + X\Delta S) = \sigma\mu I$, where $P = W^{-\frac{1}{2}}$.*

Our interior-point scheme generates the next iterate w_{k+1} as follows. We compute $\Delta w = (\Delta X, \Delta r, \Delta S, \Delta y, \Delta Z)$ from (44) and (26), and steplength $\alpha_k \in (0, 1]$ such that

$$w_k(\alpha_k) := (X_k + \alpha_k\Delta X, r_k + \alpha_k\Delta r, S_k + \alpha_k\Delta S, y_k + \alpha_k\Delta y, Z_k + \alpha_k\Delta Z) \in \mathcal{N}_{k+1}. \tag{54}$$

Since the neighborhood $\mathcal{N}_{k+1}$ involves ρ_{k+1} and δ_{k+1}, the value of those parameters must be selected together with $\alpha_k \in (0, 1]$ to ensure that (54) is satisfied.

8 Convergence analysis

In this section, we state our algorithms based on the dual HKM and NT directions and provide some tools that we need for the global convergence.

Algorithm 8.1 Primal-dual regularized interior-point algorithm for the dual HKM direction

> ***Step 0. [Initialize]*** *Choose minimum and maximum centering parameters* $0 < \sigma_{min} \leq \sigma_{max} < 1$, *a constant* $\sigma_{max} < \beta < 1$, *proximity parameters* $0 < \gamma_C < 1 < \bar{\gamma}_C, (\gamma_P, \gamma_D, \gamma_R, \gamma_S) > 0$, *initial regularization parameters* $\rho_0 > 0, \delta_0 > 0$, *choose initial primal* $X_0 \succ 0, r_0 \in \mathbb{R}^m$ *and dual guesses* $S_0 \in \mathcal{S}^n, y_0 \in \mathbb{R}^m, Z_0 \succ 0$ *so that* $w_0 \in \mathcal{N}_0$. *Let* $\mu_0 = X_0 \bullet Z_0/n$, *choose a tolerance* $\varepsilon > 0$, *and set* $k = 0$.
>
> ***Step 1. [Test convergence]*** *If* $\mu_k \leq \varepsilon$, *declare convergence and stop.*
>
> ***Step 2. [Step computation]*** *Choose a centering parameter* $\sigma_k \in [\sigma_{min}, \sigma_{max}]$. *Compute the Newton step* $\Delta w_k = (\Delta X_k, \Delta r_k, \Delta S_k, \Delta y_k, \Delta Z_k)$ *from* w_k *by solving*

$$\begin{bmatrix} -(F_k^{-1}E_k + \rho_k I) & \boldsymbol{A}^T \\ \boldsymbol{A} & \delta_k I \end{bmatrix} \begin{bmatrix} \mathbf{vec}\Delta X \\ \Delta y \end{bmatrix} = \begin{bmatrix} \mathbf{vec}C - \boldsymbol{A}^T y_k - \sigma_k \mu_k \mathbf{vec} X_k^{-1} \\ b - \boldsymbol{A}\mathbf{vec}X_k \end{bmatrix}. \tag{55}$$

> *and recover* $\Delta r_k, \Delta S_k$, *and* ΔZ_k *from* (26).
>
> ***Step 3. [Linesearch]*** *Select* $\delta_{k+1} \in (0, \delta_k]$ *and* $\rho_{k+1} \in (0, \rho_k]$ *and compute* α_k *as the largest* $\alpha \in (0, 1]$ *such that*

$$w_k(\alpha_k) \in \mathcal{N}_{k+1}, \tag{56}$$

$$\mu_k(\alpha_k) \leq (1 - \alpha_k(1 - \beta))\mu_k. \tag{57}$$

> *where* $w_k(\alpha) = w_k + \alpha \Delta w_k$ *and* $\mu_k(\alpha) = X_k(\alpha)Z_k(\alpha)/n$.
>
> ***Step 4. [Update iterate]*** *Set* $w_{k+1} = w_k(\alpha_k), \mu_{k+1} = \mu_k(\alpha_k)$. *Increment* k *by* 1 *and go to Step 1.*

Our second algorithm basically is the same as the first one. The only difference is that in Step *2* we solve a system based on the NT direction.

Algorithm 8.2 Primal-dual regularized interior-point algorithm for the NT direction
Apply Algorithm 8.1 where Step 2 is replaced by Step 2′,

> ***Step 2′. [Step computation]*** *Choose a centering parameter* $\sigma_k \in [\sigma_{min}, \sigma_{max}]$. *Compute the Newton step* $\Delta w_k = (\Delta X_k, \Delta r_k, \Delta S_k, \Delta y_k, \Delta Z_k)$ *from* w_k *by solving*

$$\begin{bmatrix} -(\bar{E}_k + \rho_k I) & \boldsymbol{A}^T \\ \boldsymbol{A} & \delta_k I \end{bmatrix} \begin{bmatrix} \boldsymbol{vec}\Delta X \\ \Delta y \end{bmatrix} = \begin{bmatrix} \boldsymbol{vec}(\sigma_k \mu_k X_k^{-1} - Z_k) \\ b - \boldsymbol{Avec}X_k \end{bmatrix}, \tag{58}$$

and recover $\Delta r_k, \Delta S_k$, *and* ΔZ_k *from* (26).

We provide convergence analysis for two particular strategies for updating the regularization parameters. Each strategy uses a different subset of the conditions (52) to define a neighborhood $\mathcal{N}_k$.

8.1 Bounds on matrix coefficients

Since at each iteration of our algorithms we need to solve system (55) or (58), we first provide bounds on the eigenvalues of the matrix

$$K = \begin{bmatrix} -Q & \mathbf{A}^T \\ \mathbf{A} & \delta I \end{bmatrix}$$

where $Q = J + \rho I$ and $J \in \mathbb{R}^{n^2 \times n^2}$ is symmetric and positive definite and $\mathbf{A} \in \mathbb{R}^{m \times n^2}$.The following eigenvalue bounds can be easily derived using Lemma 6 in Appendix A.

$$\lambda_{n^2}(J) + \rho \leq \lambda_{n^2}(Q) \leq \lambda_1(Q) \leq \lambda_1(J) + \rho. \tag{59}$$

It is easy to see that the following congruence relation holds

$$\begin{bmatrix} -Q & \mathbf{A}^T \\ \mathbf{A} & \delta I \end{bmatrix} = \begin{bmatrix} I & 0 \\ -\mathbf{A}Q^{-1} & I \end{bmatrix} \begin{bmatrix} -Q & 0 \\ 0 & \mathbf{A}Q^{-1}\mathbf{A}^T + \delta I \end{bmatrix} \begin{bmatrix} I & -Q^{-1}\mathbf{A}^T \\ 0 & I \end{bmatrix}. \tag{60}$$

Since $Q \in \mathbb{R}^{n^2 \times n^2}$ and $\mathbf{A}Q^{-1}\mathbf{A}^T + \delta I \in \mathbb{R}^{m \times m}$ and are symmetric and positive definite, Silvester's law of inertia implies that K has exactly n^2 negative and m positive eigenvalues. If we denote them in the following order

$$\lambda_{n^2+m} \leq \cdots \leq \lambda_{1+m} < 0 < \lambda_m \leq \lambda_{m-1} \cdots \leq \lambda_2 \leq \lambda_1,$$

and use $\sigma_{\max}(\mathbf{A})$ and $\sigma_{\min}(\mathbf{A})$ for the largest and smallest singular values of $\mathbf{A}$, we have the following result similar to Silvester and Wathen [SW94, Lemma 2.2] and Rusten and Winther [RW92, Lemma 2.1].

Theorem 1 *For all $\rho > 0$ and $\delta > 0$ we have the following eigenvalue bounds for K*

$$\lambda_{n^2+m} \geq \tfrac{1}{2}[\delta - \lambda_1(Q)] - \tfrac{1}{2}[(\lambda_1(Q) - \delta)^2 + 4(\sigma_{max}(\boldsymbol{A})^2 + \lambda_1(Q))\delta]^{1/2} \tag{61a}$$
$$\lambda_1 \leq \tfrac{1}{2}[\delta - \lambda_{n^2}(Q)] + \tfrac{1}{2}[(\lambda_{n^2}(Q) - \delta)^2 + 4(\sigma_{max}(\boldsymbol{A})^2 + \lambda_{n^2}(Q))\delta]^{1/2} \tag{61b}$$
$$\lambda_{1+m} \leq -\lambda_{n^2}(Q) \tag{61c}$$
$$\lambda_m \geq \tfrac{1}{2}[\delta - \lambda_1(Q)] + \tfrac{1}{2}[(\lambda_1(Q) - \delta)^2 + 4(\sigma_{min}(\boldsymbol{A})^2 + \lambda_1(Q))\delta]^{1/2} \tag{61d}$$

Moreover, $\lambda_m = \delta$—the smallest positive eigenvalue of K—if and only if $\boldsymbol{A}$ does not have full row rank. In this case, its associated eigenspace is $\{0\} \times Null(\boldsymbol{A}^T)$. Its geometric multiplicity is thus m—rank($\boldsymbol{A}$).

Proof. Let $(u, v) \neq (0, 0)$ be an eigenvector associated to the eigenvalue λ of K, then

$$-Qu + \mathbf{A}^T v = \lambda u \tag{62a}$$
$$\mathbf{A}u + \delta v = \lambda v. \tag{62b}$$

It is not difficult to see $\lambda = \delta$ if and only if $\mathbf{A}$ does not have full row rank, and in this case $\{0\} \times \mathrm{Null}(\mathbf{A}^T)$ is the eigenspace associated to λ.

If $\lambda > \delta$, from (62b), we have $v = (\lambda - \delta)^{-1}\mathbf{A}u$ with $u \neq 0$. Substituting into (62a) and taking inner product with u yields

$$\lambda\|u\|^2 = -u^T Q u + (\lambda - \delta)^{-1} u^T \mathbf{A}^T \mathbf{A} u$$

from which we enter the following inequality

$$\lambda\|u\|^2 \leq -\lambda_{n^2}(Q)\|u\|^2 + (\lambda - \delta)^{-1}\sigma_{\max}(\mathbf{A})^2\|u\|^2. \tag{63}$$

Since $u \neq 0$ we obtain

$$f(\lambda) = \lambda^2 + (\lambda_{n^2}(Q) - \delta)\,\lambda - \left(\sigma_{\max}(\mathbf{A})^2 + \lambda_{n^2}(Q)\delta\right) \leq 0 \quad \text{for all} \quad \lambda > \delta.$$

In particular, we must have $f(\lambda_1) \leq 0$. This is true only if λ_1 located between two roots of $f(\lambda) = 0$. This yields (61b).

Now, if $\lambda < \delta$ from (63) and the same logic we can find the lower bound to λ_{n^2+m} and establish (61a). When $\lambda < \delta$, the right-hand side of (63) is negative. Therefore, we must have $\lambda < 0$, and so $\lambda_m \geq \delta$. But by the implication drawn from (62), $\lambda_m = \delta$ only if $\mathbf{A}$ is rank deficient. This completes the proof of the last part of the theorem.

We can establish (61c) by taking inner product of (62a) with u and (62b) with v and subtracting, to obtain

$$(\lambda - \delta)\|v\|^2 = u^T Q u + \lambda \|u\|^2 \geq \lambda_{n^2}(Q)\|u\|^2 + \lambda \|u\|^2.$$

This implies that for all $\lambda < 0$ we must have $\lambda \leq -\lambda_{n^2}(Q)$. In particular, this proves (61c).

Having established earlier that $\delta > 0$ was the smallest positive eigenvalue of K if and only if $\mathbf{A}$ does not have full rank, we may assume without loss of generality for the last part that $\mathbf{A}$ has full row rank. Extracting u from (62a) gives $u = (Q + \lambda I)^{-1}\mathbf{A}^T v$ with $v \neq 0$. Injecting u into (62b) and taking inner product with v we get

$$(\lambda - \delta)\|v\|^2 = v^T \mathbf{A}(Q + \lambda I)^{-1}\mathbf{A}^T v \geq (\lambda_1(Q) + \lambda)^{-1}\sigma_{\min}(\mathbf{A})^2\|v\|^2.$$

Since $v \neq 0$ we obtain the following inequality

$$g(\lambda) = \lambda^2 + (\lambda_1(Q) - \delta)\lambda - (\sigma_{\min}(\mathbf{A})^2 + \lambda_1(Q)\delta) \geq 0 \quad \text{for all} \quad \lambda > \delta.$$

Therefore, λ must be at least the right-most root of $g(\lambda) = 0$. In particular, this is true of λ_m, and proves (61d). □

In the proof of Theorem 1, we showed that if $\lambda < \delta$ then λ is negative. Therefore, we must have $\lambda_m \geq \delta$. Using the left-hand side of (59) and the fact that $J \succ 0$, we see that $\lambda_{n^2}(Q) \geq \rho$ and (61c) shows that $\lambda_{m+1} \leq -\rho$. Since $\|K^{-1}\| = \max(\lambda_m^{-1}, |\lambda_{m+1}^{-1}|)$ we have the following corollary.

Corollary 1 *If J is positive definite then for all $\rho > 0$ and $\delta > 0$,*

$$\|K^{-1}\| \leq 1/\min(\rho, \delta). \tag{64}$$

We note that $F^{-1}E$ and $\bar{E}$ are symmetric and positive definite. Therefore, theorem 1 and Corollary 1 guarantee that the coefficient matrices in Algorithms 8.1-8.2 and their inverses are uniformly bounded.

Let $w(\alpha) = w + \alpha\Delta w$. Then Algorithms (8.1) and (8.2) guarantee to achieve the following progress along Newton direction

$$\begin{aligned}\mathbf{A}\mathbf{vec}X(\alpha) + \delta r(\alpha) - b &= \mathbf{A}\mathbf{vec}X + \delta r - b + \alpha(\mathbf{A}\mathbf{vec}\Delta X + \delta\Delta r) \\ &= (1-\alpha)(\mathbf{A}\mathbf{vec}X + \delta r - b),\end{aligned} \tag{65}$$

and using the first row of (24) we obtain

$$\begin{aligned}&C + \rho S(\alpha) + \mathcal{A}^* y(\alpha) - Z(\alpha) \\ &\qquad = C + \rho S + \mathcal{A}^* y - Z + \alpha(\rho\Delta S - \mathcal{A}^*\Delta y - \Delta Z) \\ &\qquad = (1-\alpha)(C + \rho S + \mathcal{A}^* y - Z),\end{aligned}$$

or, equivalently,

$$\mathbf{vec}C + \rho\mathbf{vec}S(\alpha) - \mathbf{A}^T y(\alpha) - \mathbf{vec}Z(\alpha) = (1-\alpha)(\mathbf{vec}C + \rho\mathbf{vec}S - \mathbf{A}^T y - \mathbf{vec}Z). \tag{66}$$

Since the Newton step is computed from the current w_k,

$$\delta r(\alpha) = \delta(r + \alpha\Delta r) = (1-\alpha)\delta r + \alpha\delta\Delta y \tag{67a}$$

$$\rho S(\alpha) = (S + \alpha\Delta S) = (1-\alpha)\rho S + \alpha\rho\Delta X. \tag{67b}$$

If ΔX and ΔZ only satisfy (36) then we obtain

$$\begin{aligned}\mu(\alpha) &= (X + \alpha\Delta X) \bullet (Z + \alpha\Delta Z)/n \\ &= (1-\alpha)\mu + \alpha(X \bullet Z + Z \bullet \Delta X + Z \bullet \Delta Z)/n + \alpha^2(\Delta X \bullet \Delta Z)/n,\end{aligned} \tag{68}$$

and using (36) again we obtain

$$X \bullet Z + Z \bullet \Delta X + Z \bullet \Delta Z = \mathbf{tr}(XZ + \Delta X Z + X\Delta Z) = n\sigma\mu, \tag{69}$$

therefore,

$$\mu(\alpha) = (1 - \alpha + \sigma\alpha)\mu + \alpha^2(\Delta X \bullet \Delta Z)/n. \tag{70}$$

The following lemma shows that if ΔX and ΔZ satisfy (36), then (69) and (70) hold true if (36) is transformed by H_P for any nonsingular matrix P.

Lemma 3 *Let ΔX and ΔZ satisfy (36) and P be any nonsingular matrix. Then (69) and (70) hold true if ΔX and ΔZ satisfy $H_P(XZ + \Delta XZ + X\Delta Z) = \sigma\mu I$ and*

$$H_P(X(\alpha)Z(\alpha)) = (1 - \alpha)H_P(XZ) + \alpha\sigma\mu I + \alpha^2 H_P(\Delta X\Delta Z). \tag{71}$$

Proof. The proof of the first part is an immediate application of Lemma 1 part *2*. For the second part, we have

$$\begin{aligned} X(\alpha)Z(\alpha) &= (X + \alpha\Delta X)(Z + \alpha\Delta Z) &(72)\\ &= XZ + \alpha X\Delta Z + \alpha\Delta XZ + \alpha^2\Delta X\Delta Z &(73)\\ &= (1 - \alpha)XZ + \alpha(XZ + X\Delta z + \Delta XZ) + \alpha^2\Delta X\Delta Z. &(74)\end{aligned}$$

Linearity of H_P and (36) conclude the proof. □

Since the conclusion of Lemma 3 is true for all nonsingular matrix, particular for $P = X^{-\frac{1}{2}}$ and $P = W^{-\frac{1}{2}}$, we have (69), (70), and (71) hold true when $(\Delta X, \Delta Z)$ satisfy Step *2* or Step *2′* of Algorithms 8.1-8.2.

9 Algorithm based on fixed regularization parameters

Our first method leaves the regularization parameters δ_k and ρ_k fixed throughout the iterations. It forces iterate to stay in a neighborhood that satisfy (52a)–(52c).

Algorithm 9.1 Variation of the primal-dual method with constant regularization
Apply Algorithm 8.1 or 8.2 with $\rho_k = \rho_0$ and $\delta_k = \delta_0$ for all k. In Step 3, only conditions (52a), (52b), and (52c) are enforced.

Since regularization parameters are fixed we denote them ρ and δ for readability. Convergence properties rely on the following technical lemma.

Lemma 4 *Suppose $(\Delta X, \Delta y, \Delta Z)$ is given by Step 2 of Algorithms 8.1-8.2, and $\{r_k\}$, $\{S_k\}$, and $\{Z_k\}$ are bounded. Then $(\Delta X, \Delta y, \Delta Z)$ is also bounded.*

Proof. First, we show that the right-hand side of (55)–(58) are bounded. From (52b) we have

$$\begin{aligned} \|\mathbf{A}\mathbf{vec}X_k - b\| &\le \|\mathbf{A}\mathbf{vec}X_k + \delta r_k - b\| + \delta\|r_k\| \\ &\le \gamma_P\mu_k + \delta\sup_k \|r_k\| \\ &\le \gamma_P\mu_0 + \delta\sup_k \|r_k\|, \end{aligned} \tag{75}$$

which shows that the second block of the right-hand side of (55)–(58) are bounded. We also have

$$\begin{aligned} \|\mathbf{vec}C - \mathbf{A}^T y_k - \sigma\mu_k\mathbf{vec}X_k^{-1}\| &\le \|\mathbf{vec}C - \mathbf{A}^T y_k + \rho\mathbf{vec}S_k - \mathbf{vec}Z_k\| \\ &\quad + \rho\sup_k \|S_k\|_F + \|\sigma\mu_k X_k^{-1} - Z_k\|_F \\ &\le \gamma_D\mu_0 + \rho\sup_k \|S_k\|_F + \|\sigma\mu_k X_k^{-1} - Z_k\|_F. \end{aligned} \tag{76}$$

Using the centering condition (52a) and Lemma 1, we have

$$\gamma_C\mu_k \le \lambda_i(X_kZ_k) \le \bar{\gamma}_C\mu_k, \tag{77}$$

reciprocating (77) and multiplying σ to it and subtracting 1 from the result we get

$$\frac{\sigma}{\bar{\gamma}_C} - 1 \leq \sigma\mu_k\lambda_i(X_k^{-1}Z_k^{-1}) - 1 \quad \leq \frac{\sigma}{\gamma_C} - 1, \tag{78}$$

which leads us to

$$\left|\lambda_i(\sigma\mu_k X_k^{-1}Z_k^{-1} - I)\right| \leq M, \quad \text{for all} \quad i = 1, 2, \cdots n,$$

where $M := \max(|\frac{\sigma}{\bar{\gamma}_C} - 1|, |\frac{\sigma}{\gamma_C} - 1|)$. Therefore,

$$\|\sigma\mu_k X_k^{-1}Z_k^{-1} - I\|_F \leq \sqrt{n}M. \tag{79}$$

Using (79) the upper bound on the last term in (76) is obtained

$$\|\sigma\mu_k X_k^{-1} - Z_k\|_F = \|(\sigma\mu_k X_k^{-1}Z_k^{-1} - I)Z_k\|_F \leq \sqrt{n}M \sup_k \|Z_k\|_F. \tag{80}$$

This proves that the first block of the (55)–(58) are also bounded. By Corollary 1 the inverse of the matrix in (55)–(58) are uniformly bounded and so $(\Delta X, \Delta y)$ is bounded. To see that ΔZ is bounded, we note ΔZ satisfy (45), which has a bounded right-hand side. This completes the proof of the lemma. □

In the next lemma, we show the existence of a steplength α such that Step *3* of Algorithm 9.1 is satisfied.

Lemma 5 *Suppose $(\Delta X, \Delta y, \Delta Z)$ is given by Step 2 of Algorithm 9.1, and $\{r_k\}$, $\{S_k\}$, and $\{Z_k\}$ are bounded. Suppose also that there exists an index k_0 such that $\mu_k \geq \varepsilon$ for all $k \geq k_0$. Then there exists $\alpha^* \in (0,1]$ such that for all $\alpha \in (0, \alpha^*]$, step 3 of Algorithm 9.1 is satisfied.*

Proof. Firstly, we need $\alpha \in (0, 1]$ satisfies (56) in Step *3*. Define $G := H_P(X_k(\alpha)Z_k(\alpha)) - \gamma_C\mu_k(\alpha)I$. From (70) and (71) we have

$$G = (1-\alpha)[H_P(X_kZ_k) - \gamma_C\mu_k I] + \alpha\sigma_k\mu_k(1-\gamma_C)I + \alpha^2\left[H_P(\Delta X\Delta Z) - \gamma_C\frac{\Delta X \bullet \Delta Z}{n}I\right],$$

using part 3. of Lemma 1, (92a)–(92d) in Appendix A the following inequality is obtained

$$\lambda_n(G) \geq (1-\alpha)\lambda_n(H_P(X_kZ_k) - \gamma_C\mu_k I) + \alpha\sigma_k\mu_k(1-\gamma_C) - \alpha^2\tau, \tag{81}$$

where $\tau = \left(\gamma_C|\Delta X \bullet \Delta Z| + \frac{1}{n}\|\Delta X\Delta Z\|_F\right)$. Using lemma 4 we can find $\pi_1 > 0$ that depends only on n and

$$|\tau| \leq \pi_1. \tag{82}$$

Therefore, using (81) and (52a) we need to choose $\alpha > 0$ such that $\sigma_{\min}\mu_k(1-\gamma_C) - \alpha\pi_1 \geq 0$, that is we need

$$\alpha \leq \frac{\sigma_{\min}(1-\gamma_C)\mu_k}{\pi_1}. \tag{83}$$

Similarly, $X(\alpha)$ and $Z(\alpha)$ satisfy the right-hand side of (52a) if

$$\alpha \leq \frac{\sigma_{\min}(\bar{\gamma}_C - 1)\mu_k}{\pi_2}, \tag{84}$$

for some $\pi_2 > 0$ which only depends on n.

We need α to satisfy (52b) and (52c). In order to find such an α we may use (52b), (65), and (70) to get

$$\begin{aligned}\gamma_P\mu_k(\alpha) - \|\mathbf{Avec}X_k(\alpha) + \delta r_k(\alpha) - b\| &\geq \alpha\sigma_k\gamma_P\mu_k + \alpha^2\gamma_P\Delta X \bullet \Delta Z/n \\ &\geq \gamma_P(\alpha\sigma_{\min}\mu_k - \alpha^2\pi_3)\end{aligned}$$

and by (66) we get

$$\gamma_D \mu_k(\alpha) - \|\mathbf{vec}C + \rho \mathbf{vec}S_k(\alpha) - \mathbf{A}^T y_k(\alpha) - \mathbf{vec}Z_k(\alpha)\| \geq \alpha\sigma_k\gamma_D\mu_k + \alpha^2\gamma_D \Delta X \bullet \Delta Z/n$$
$$\geq \gamma_D(\alpha\sigma_{\min}\mu_k - \alpha^2\pi_3)$$

where $\pi_3 = \|\Delta X\|_F \|\Delta Z\|_F/n$. Therefore, it is enough to choose

$$\alpha \leq \frac{\sigma_{\min}\mu_k}{\pi_3}, \tag{85}$$

Secondly, the steplength α must satisfy (57). Again from (70) we have

$$(1 - \alpha(1-\beta))\mu_k - \mu_k(\alpha) = \alpha(\beta - \sigma_k)\mu_k + \alpha^2 \Delta X \bullet \Delta Z/n$$
$$\geq \alpha(\beta - \sigma_{\max})\mu_k - \alpha^2\pi_3.$$

Therefore, it is sufficient to choose

$$\alpha \leq \frac{(\beta - \sigma_{\max})\mu_k}{\pi_3} \tag{86}$$

Finally, We need that $X_k(\alpha)$ and $Z_k(\alpha)$ must be positive definite. Since eigenvalues of $X_k(\alpha)$ and $Z_k(\alpha)$ are continuous function respect to α and $X_k(0) = X_k$ and $Z_k(0) = Z_k$ are positive definite, there exists $\bar{\alpha}$ such that $X_k(\alpha)$ and $Z_k(\alpha)$ are positive definite for all $\alpha \in (0, \bar{\alpha}]$. The proof of the lemma is completed if we let

$$\alpha^* = \min\left\{1, \frac{\sigma_{\min}(1-\gamma_C)\varepsilon}{\pi_1}, \frac{\sigma_{\min}(\bar{\gamma}_C - 1)\varepsilon}{\pi_2}, \frac{\sigma_{\min}\varepsilon}{\pi_3}, \frac{(\beta - \sigma_{\max})\varepsilon}{\pi_3}, \bar{\alpha}\right\},$$

and note that Algorithm 9.1 in Step *3* chooses the largest possible α. □

Now, we are in a position to establish the convergence properties of Algorithm 9.1. In the following theorem, we show that the duality measure μ_k converges to zero under a boundedness assumption.

Theorem 2 *Suppose that Algorithm 9.1 generates the sequence $\{w_k\}$, and that the sequence $\{(r_k, S_k, Z_k)\}$ is bounded. Then $\mu_k \to 0$ as $k \to +\infty$.*

Proof. By Lemma 5 for any index k we have $\mu_k \geq 0$. If $\lim_{k\to+\infty} \mu_k \neq 0$, then there exists an index k_0 such that $\mu_k \geq \varepsilon$ for any $k \geq k_0$. Using Lemma 5 again we can find α^* such that Step *3* of Algorithm 9.1 is satisfied. particularly, we have

$$0 < \varepsilon \leq \mu_{k+1} \leq (1 - \alpha^*(1-\beta))\mu_k \leq \cdots \leq (1 - \alpha^*(1-\beta))^{k+1}\mu_0.$$

This is a contradiction because the rightmost term above converges to zero. □

Global convergence of the algorithm is obtained by examine the limiting behavior of the sequences $\{r_k\}$ and $\{S_k\}$.

Theorem 3 *Suppose Algorithm 9.1 with $\varepsilon = 0$ generates the sequence $\{w_k\}$, and that $\{(r_k, S_k, Z_k)\}$ remains bounded. Then if $\hat{r}$ and $\hat{S}$ denote particular limit points of $\{r_k\}$ and $\{S_k\}$ along subsequences index $\mathcal{K} \subset \mathbb{N}$, every limit point of $\{(X_k, Z_k)\}_{\mathcal{K}}$ determines a primal-dual solution of the primal-dual pair*

$$\underset{X}{\textit{minimize}}\ (C + \rho\hat{S}) \bullet X \ \textit{subject to}\ \ \mathcal{A}X = b - \delta\hat{r},\ \ X \succeq 0, \tag{87}$$

$$\underset{y,Z}{\textit{maximize}}\ (b - \delta\hat{r})^T y \quad \textit{subject to}\ \ \mathcal{A}^* y + Z = C + \rho\hat{S},\ \ Z \succeq 0. \tag{88}$$

Proof. Let $\{(X_k, Z_k)\}$ converge to $(\bar{X}, \bar{Z})$ along subsequence index $\mathcal{K}$. Since

$$\|C - \mathcal{A}^* y_k - Z_k\|_F = \|C - \sum_{i=1}^{m} [y_k]_i A_i - Z_k\|_F = \|\mathbf{vec}C - \mathbf{A}^T y_k - \mathbf{vec}Z_k\|_2$$
$$\leq \|\mathbf{vec}C - \mathbf{A}^T y_k - \mathbf{vec}Z_k + \rho S_k\|_2 + \rho \sup_k \|S_k\|_F$$
$$\leq \gamma_{_D} \mu_0 + \rho \sup_k \|S_k\|_F,$$

is bounded, without loss of generality we can assume $(C - \mathcal{A}^* y_k - Z_k) \to (C - \mathcal{A}^* \bar{y} - \bar{Z})$ along $\mathcal{K}$ for some $\bar{y} \in \mathbb{R}^m$. In the limit along $\mathcal{K}$, we have

$$\begin{aligned} \mathcal{A}^* y + Z &= C + \rho \hat{S}, \\ A_i \bullet \bar{X} &= b_i - \delta \hat{r}_i, (i = 1, 2, \cdots, m), \\ \lambda_i(XZ) &= 0, (i = 1, 2, \cdots, m), \\ (X, Z) &\succcurlyeq 0 \end{aligned} \tag{89}$$

which follow form the neighborhood conditions (52a)–(52c), Lemma 1, and the fact that $\mu_k \to 0$. We note that (89) are the optimality conditions for the primal-dual pair (87) and (88). □

In the next theorem, we state some results regarding the limit points of $\{r_k\}$ and $\{S_k\}$ and feasibility of (1) and (3).

Theorem 4 *Suppose Algorithm 9.1 with $\varepsilon = 0$ generates the sequence $\{w_k\}$, and that $\{(r_k, S_k, Z_k)\}$ remains bounded. Then*

i. If $\liminf_{k \in \mathbb{N}} \|r_k\| = 0$, every limit point of $\{X_k\}_{\mathcal{K}}$ is feasible for (1), where $\mathcal{K} \subseteq \mathbb{N}$ is an index set such that $\{r_k\}_{\mathcal{K}} \to 0$.

ii. If $\liminf_{k \in \mathbb{N}} \|S_k\| = 0$, every limit point of $\{(X_k, Z_k)\}_{\mathcal{K}'}$ determines a feasible point for (3), where $\mathcal{K}' \subseteq \mathbb{N}$ is an index set such that $\{S_k\}_{\mathcal{K}'} \to 0$.

iii. If there exists an index set $\mathcal{K}'' \subseteq \mathbb{N}$ such that $\{r_k\}_{\mathcal{K}''} \to 0$ and $\{S_k\}_{\mathcal{K}''} \to 0$, every limit point of $\{X_k, Z_k\}_{\mathcal{K}''}$ determines a primal-dual solution of (1) and (3).

Proof. By assumption in part i, there exists an index set $\mathcal{K} \subseteq \mathbb{N}$ such that $\{r_k\}_{\mathcal{K}} \to 0$. Using Triangular inequality and (52b), we obtain

$$|A_i \bullet X_k - b_i| \leq \|\mathbf{Avec}X_k - b\| \leq \|\mathbf{Avec}X_k + \delta r_k - b\| + \delta \|r_k\|$$
$$\leq \gamma_{_P} \mu_k + \delta \|r_k\| \text{ for all } k \in \mathcal{K} \text{ and } (i = 1, 2, \cdots, m).$$

Now, if we let k go to infinity along $\mathcal{K}$ and use Theorem 2 we obtain $A_i \bullet \bar{X} = b_i$ for all $i = 1, 2, \cdots, m$. Since $X_k \succ 0$ for all $k \in \mathcal{K}$ and eigenvalues are continuous we have $\bar{X} \succeq 0$. This completes the proof of part i.

The proof of part ii is obtained by applying Theorem 3 with $\hat{S} = 0$. The proof of part iii is a direct consequence of parts i-ii and Theorem 3. □

Note that in part iii of Theorem 4 our assumption is that both sequences $\{r_k\}$ and $\{S_k\}$ are going to zero along the index set $\mathcal{K}''$, but in general they may approach zero along different index sets. Therefore, the assumption of part iii has to be slightly restrictive. However, if the algorithm generates a single limit point, then it must be a primal-dual solution. We prove this fact in the next corollary.

Corollary 2 *Suppose Algorithm 9.1 with $\varepsilon = 0$ generates the sequence $\{w_k\}$ such that $\{(X_k, y_k, Z_k)\}$ has the single limit point $(\bar{X}, \bar{y}, \bar{Z})$ and that there exists $\alpha^* > 0$ such that $\alpha_k \geq \alpha^*$ for all sufficiently large k. Then $(\bar{X}, \bar{y}, \bar{Z})$ is a primal-dual solution of (1) and (3).*

Proof. We have $X_{k+1} = X_k + \alpha_k \Delta X_k,\ X_k \to \bar{X}$, and α_k is uniformly bounded away from zero. Therefore, we must have $\Delta X_k \to 0$. Using (67b) we obtain

$$\begin{aligned}\|S_{k+1}\| = \|(1-\alpha_k)S_k + \alpha_k \Delta X_k\| &\le (1-\alpha_k)\|S_k\| + \alpha_k\|\Delta X_k\| \\ &\le (1-\alpha^*)\|S_k\| + \|\Delta X_k\|.\end{aligned}$$

Taking limit superior from both side we obtain $S_k \to 0$. Similarly, $\Delta y_k \to 0$ and from (67a) we obtain

$$\|r_{k+1}\| \le (1-\alpha^*)\|r_k\| + \|\Delta y_k\|,$$

by which we conclude $r_k \to 0$. Theorem 4(iii) completes the proof. □

10 Algorithm based on decreasing regularization parameters

Our second method is based on decreasing the regularization parameter ρ_k and δ_k at each iteration and enforces conditions (52a), (52d), and (52e).

Algorithm 10.1 Variation of the Primal-dual with variable regularization *Apply Algorithm 9.1 with the following specializations. In Step 3, only conditions* (52a)*,* (52d)*, and* (52e) *are enforced, and ρ_{k+1} and δ_{k+1} are chosen so that*

$$\rho_{k+1} \ge \kappa_\rho \rho_k, \tag{90a}$$

$$\delta_{k+1} \ge \kappa_\delta \delta_k, \tag{90b}$$

for some $0 < \kappa_\rho < 1$ and $0 < \kappa_\delta < 1$.

In Step *3* of Algorithm 9.1 we assume that $\rho_{k+1} \le \rho_k$ and $\delta_{k+1} \le \delta_k$, however (90a) and (90b) require they are not decreasing faster than linear rate. This feature of Algorithm 10.1 allow us to establish convergence result of Algorithm 10.1 and recover a solution of (1) and (3). The convergence results of Algorithm 10.1 are similar to those of Algorithm 9.1. In what follows, we establish a result analogous to Theorem 3.

Theorem 5 *Suppose Algorithm 10.1 with $\varepsilon = 0$ generates the sequence $\{w_k\}$ and that $\{(X_k, Z_k)\}$ remains bounded. Suppose also that there exists $k_0 \in \mathbb{N}$ and $\alpha^* \in (0,1]$ such that $\alpha_k \ge \alpha^*$ for all $k \ge k_0$. Then the sequence $\{(X_k, r_k, S_k, Z_k)\}$ is bounded and every limit point of $\{(X_k, Z_k)\}$ determines a primal-dual solution pair* (1) *and* (3)*.*

Proof. Our assumption that $\alpha_k \ge \alpha^* > 0$ for all $k \ge k_o$ and (57) imply that $\mu_k \to 0$. By definition of $\mathcal{N}_k$ we also have $\|\delta_k r_k\| \le \gamma_{R}\mu_k$ and $\|\rho_k \mathbf{vec} S_k\| \le \gamma_{S}\mu_k$, and thus $\delta_k r_k \to 0$ and $\rho_k S_k \to 0$.

We have from (52d), (65), and (90b)

$$\begin{aligned}\|\mathbf{Avec}X_{k+1} + \delta_{k+1}r_{k+1} - b\| &= \|\mathbf{Avec}X_{k+1} + \delta_{k+1}r_{k+1} - b \pm \delta_k r_{k+1}\| \\ &= \|(1-\alpha_k)(\mathbf{Avec}X_k + \delta_k r_k - b) + (\delta_{k+1} - \delta_k)r_{k+1}\| \\ &\le (1-\alpha^*)\|\mathbf{Avec}X_k + \delta_k r_k - b\| + \frac{\delta_k - \delta_{k+1}}{\delta_{k+1}}\gamma_{R}\mu_{k+1} \\ &\le (1-\alpha^*)\|\mathbf{Avec}X_k + \delta_k r_k - b\| + (1-\kappa_\delta)\gamma_{R}\mu_k,\end{aligned}$$

for all $k \ge k_0$. Upon taking the limit superior in the last inequality, and using the fact that $\mu_k \to 0$, we obtain that $(\mathbf{Avec}X_k + \delta_k r_k - b) \to 0$. Similarly, one can show that $(\mathbf{vec}C + \rho_k \mathbf{vec}S_k - \mathbf{A}^T y_k - \mathbf{vec}Z_k) \to 0$. Since $\{(X_k, Z_k)\}$ is bounded, using the above discussion we conclude that $\{\mathbf{A}^T y_k\}$ is bounded. Let $g_k := \mathbf{A}^T y_k$ for all k and consider any limit point $(\bar{X}, \bar{g}, \bar{Z})$ of $\{(X_k, g_k, Z_k)\}$. We note that the range space of $\mathbf{A}^T$ is a closed sub space, therefore $\bar{g} = \mathbf{A}^T \bar{y}$ for some $\bar{y} \in \mathbb{R}^m$. Therefore, in the limit we have

$$\mathbf{Avec}\bar{X} = b, \quad \mathbf{vec}C = \mathbf{A}^T\bar{y} + \mathbf{vec}\bar{Z}, \quad \lambda(\bar{X}\bar{Z}) = 0, \quad (\bar{X}, \bar{Z}) \succeq 0.$$

□

11 Preliminary numerical results

We use MATLAB to code and run our algorithm on Intel Core dual CPU processor $T8300$ @ 2.4 GHZ, with 3 GB of RAM. Algorithm 10.1 is tested on problems from SDPLIB [Bor99] a collection containing a total of 93 SDP problem. Some problems from commonly used DIMACS set of benchmark problems[PS99]. We modify SDPT3 code to incorporate our regularization within an inexact infeasible primal-dual path-following interior-point framework. Ninty problems from above collections are solved and the performance of our regularization is compared to that of SDPT3. The details results of the two solvers SDPT3 and SDPR (regularized SDP) are collected in Table 1–2 and Table 3–4, respectively. For each problem, we select $\delta_0 = 1$ and use the update rules

$$\delta_{k+1} = \frac{\delta}{10}.$$

We set $\rho_k = 0$ for all iterations and let SDPT3 use its own preprocessing to check for independency of A_i. Safeguards are used to ensure that δ_k never take values less than 10^8.

The benchmark optimization software and performance profiles of Dolan and Moré [DM02] is used to compare the performance of SDPT3 and SDPR on these 90 problems. We compare the number of iterations needed to solve an instant problem.

Table 1: Results from SDPT3 on 90 problems.

ProbName	#iter	Gap	P_infeas	D_infeas	CPUTT (secs)
arch0.dat-	26	3.22E-09	3.92E-09	1.27E-11	3.326
arch2.dat-	24	9.42E-09	5.11E-10	2.99E-11	3.198
arch4.dat-	22	1.14E-08	8.52E-10	2.10E-11	2.755
arch8.dat-	25	4.71E-08	1.14E-08	6.20E-11	3.165
control1.d	17	6.55E-08	2.68E-09	2.35E-11	0.248
control10.	25	2.10E-04	4.78E-07	1.51E-09	44.421
control11.	24	7.50E-05	2.41E-07	1.33E-09	66.774
control2.d	21	3.68E-09	2.54E-08	2.98E-11	0.633
control3.d	21	3.52E-06	1.02E-07	1.10E-10	1.52
control4.d	21	9.41E-07	2.13E-07	1.53E-10	3.826
control5.d	23	1.89E-05	5.79E-07	1.93E-10	9.923
control6.d	22	3.77E-05	5.79E-07	5.02E-10	24.267
control7.d	22	1.96E-05	6.36E-08	4.62E-10	42.938
control8.d	23	1.49E-05	2.24E-07	5.39E-10	84.237
control9.d	22	1.96E-05	5.94E-08	5.14E-10	133.717
equalG11.d	17	2.79E-06	1.03E-12	1.00E-12	35.601
equalG51.d	18	3.44E-05	5.78E-11	5.07E-12	64.364
gpp100.dat	14	8.89E-07	2.74E-10	6.17E-11	0.597
gpp124-1.d	18	8.51E-08	3.45E-11	8.85E-12	0.982
gpp124-2.d	15	5.94E-07	1.16E-10	2.21E-11	0.649
gpp124-3.d	14	1.27E-06	4.15E-10	8.55E-11	0.581
gpp124-4.d	15	5.52E-07	5.62E-10	7.44E-11	0.622
gpp250-1.d	17	1.58E-05	5.37E-12	1.31E-12	1.901
gpp250-2.d	15	1.93E-07	1.28E-10	2.61E-11	1.684
gpp250-3.d	15	4.22E-07	3.32E-10	5.38E-11	1.699
gpp250-4.d	14	6.78E-06	2.24E-10	5.28E-11	1.589
gpp500-1.d	19	5.67E-05	9.47E-12	2.15E-12	12.209
gpp500-2.d	16	2.06E-06	3.29E-12	1.65E-12	10.51
gpp500-3.d	16	7.86E-06	4.22E-12	2.23E-12	10.776
gpp500-4.d	17	4.58E-06	8.37E-12	2.68E-12	11.213
hinf1.dat-	26	1.23E-05	3.37E-08	6.34E-09	0.311
hinf10.dat	34	3.92E-04	3.12E-07	1.23E-07	0.468
hinf11.dat	28	1.16E-02	3.53E-07	8.43E-08	0.417
hinf12.dat	60	5.80E-05	3.65E-12	1.98E-06	0.942
hinf13.dat	29	3.04E-02	9.81E-06	6.91E-07	0.625
hinf14.dat	34	1.73E-04	9.91E-08	5.48E-08	0.903
hinf15.dat	26	2.91E-01	7.83E-05	3.61E-06	0.84
hinf2.dat-	16	5.82E-08	3.20E-06	1.45E-11	0.189
hinf3.dat-	20	5.17E-06	6.53E-06	2.42E-10	0.224
hinf4.dat-	21	1.22E-05	8.36E-08	1.43E-09	0.238
hinf5.dat-	21	4.76E-05	9.24E-05	1.06E-10	0.229
hinf6.dat-	22	7.76E-06	4.54E-06	3.50E-11	0.236
hinf7.dat-	18	8.45E-04	5.17E-06	1.78E-10	0.187
hinf8.dat-	21	6.10E-04	1.50E-05	5.71E-09	0.235
hinf9.dat-	22	8.47E-09	5.90E-07	8.97E-15	0.233

Table 2: Results from SDPT3 on 90 problems (cont.)

ProbName	#iter	Gap	P_infeas	D_infeas	CPUTT (secs)
infd1.dat-	11	3.78E+17	3.35E+00	3.22E+01	0.137
infd2.dat-	11	5.33E+18	1.31E+00	1.53E+03	0.142
infp1.dat-	31	2.59E+13	1.47E-01	1.65E+00	0.392
infp2.dat-	31	1.00E+13	3.36E-02	1.28E+00	0.378
maxG11.dat	15	6.10E-06	3.05E-13	1.00E-12	13.694
maxG32.dat	15	3.11E-05	4.01E-12	1.00E-12	120.211
maxG51.dat	17	2.10E-06	1.45E-13	1.00E-12	30.976
mcp100.dat	12	9.00E-07	1.23E-11	1.00E-12	0.41
mcp124-1.d	12	2.12E-06	1.46E-11	1.00E-12	0.428
mcp124-2.d	13	2.23E-07	2.43E-12	1.50E-12	0.479
mcp124-3.d	12	4.63E-06	3.33E-13	1.02E-12	0.527
mcp124-4.d	13	6.96E-07	2.19E-12	1.00E-12	0.496
mcp250-1.d	14	6.17E-07	9.54E-14	1.00E-12	0.858
mcp250-2.d	13	2.80E-06	5.25E-13	1.00E-12	1.047
mcp250-3.d	13	5.51E-06	1.25E-12	1.50E-12	1.149
mcp250-4.d	14	1.53E-05	1.09E-12	1.00E-12	1.269
mcp500-1.d	15	6.58E-07	4.67E-13	1.00E-12	3.817
mcp500-2.d	16	2.55E-06	1.60E-12	1.48E-12	5.469
mcp500-3.d	14	3.39E-05	9.07E-14	1.00E-12	5.455
mcp500-4.d	13	6.25E-05	1.05E-12	1.03E-12	5.857
qap10.dat-	17	2.99E-05	1.98E-07	1.03E-09	4.467
qap5.dat-s	10	6.51E-07	1.71E-11	2.70E-10	0.399
qap6.dat-s	16	2.03E-06	4.73E-07	1.81E-10	1.113
qap7.dat-s	18	3.18E-06	3.51E-07	2.87E-10	2.391
qap8.dat-s	17	4.61E-05	8.80E-07	2.44E-09	0.967
qap9.dat-s	21	3.77E-06	8.43E-08	1.49E-10	4.052
qpG11.dat-	15	4.13E-05	4.57E-13	1.00E-12	13.665
qpG51.dat-	17	5.00E-05	1.36E-11	1.06E-12	30.318
ss30.dat-s	21	1.06E-05	1.39E-07	2.56E-11	12.801
theta1.dat	11	3.85E-07	1.18E-11	4.58E-12	0.553
theta2.dat	13	9.43E-08	1.08E-12	1.00E-12	1.72
theta3.dat	14	2.23E-07	1.16E-11	1.00E-12	4.621
theta4.dat	14	7.14E-07	3.82E-13	1.00E-12	10.85
theta5.dat	14	4.03E-07	4.19E-13	1.00E-12	26.87
theta6.dat	14	1.98E-07	3.39E-13	1.00E-12	62.183
thetaG11.d	18	5.08E-06	4.28E-13	1.01E-12	47.714
truss1.dat	9	1.30E-07	2.31E-09	9.79E-11	0.132
truss2.dat	13	5.35E-07	9.37E-10	5.70E-10	0.281
truss3.dat	12	1.37E-07	5.87E-14	1.00E-12	0.17
truss4.dat	11	1.64E-09	3.85E-09	1.17E-11	0.142
truss5.dat	15	1.53E-07	1.52E-10	2.79E-12	0.691
truss6.dat	24	8.11E-05	1.25E-07	2.89E-11	0.797
truss7.dat	21	1.86E-04	1.13E-08	1.46E-11	0.569
truss8.dat	16	7.48E-08	1.80E-10	1.02E-11	3.115

Table 3: Results from regularized SDP (SDPR) on 90 problems.

ProbName	#iter	Gap	P_infeas	D_infeas	CPUTT (secs)
arch0.dat-	26	3.23E-09	4.52E-09	1.27E-11	5.123
arch2.dat-	24	6.56E-09	4.05E-10	2.08E-11	5.28
arch4.dat-	22	1.13E-08	9.46E-10	1.74E-11	5.138
arch8.dat-	25	1.83E-08	1.39E-08	2.48E-11	4.925
control1.d	17	3.63E-08	2.12E-09	2.13E-10	0.227
control10.	23	4.60E-04	6.75E-08	8.36E-09	77.549
control11.	25	6.93E-05	3.61E-07	2.44E-08	136.909
control2.d	21	2.13E-07	1.41E-08	5.68E-10	0.851
control3.d	25	6.75E-07	1.42E-07	9.54E-09	2.481
control4.d	21	1.58E-06	1.09E-07	9.08E-09	4.572
control5.d	22	1.31E-05	1.05E-07	8.07E-09	10.858
control6.d	22	8.48E-05	1.21E-07	1.12E-08	25.646
control7.d	23	1.79E-05	6.31E-08	8.10E-09	47.769
control8.d	24	1.97E-05	2.86E-07	1.55E-08	93.496
control9.d	23	8.42E-06	9.87E-08	4.99E-09	167.068
equalG11.d	18	6.21E-07	7.07E-11	1.36E-12	50.574
equalG51.d	20	1.38E-05	3.92E-09	3.92E-11	101.035
gpp100.dat	15	1.08E-07	9.58E-09	6.01E-11	0.457
gpp124-1.d	19	5.15E-07	1.71E-09	1.90E-11	0.909
gpp124-2.d	15	5.08E-07	1.46E-09	3.78E-11	0.673
gpp124-3.d	15	1.41E-06	3.81E-09	9.95E-11	0.756
gpp124-4.d	15	3.31E-06	6.57E-09	4.22E-10	0.789
gpp250-1.d	19	5.39E-07	1.39E-09	1.45E-11	2.709
gpp250-2.d	18	9.64E-08	1.23E-09	2.83E-11	4.595
gpp250-3.d	17	4.48E-07	2.65E-09	1.71E-11	4.52
gpp250-4.d	16	7.28E-06	2.28E-09	1.84E-11	2.015
gpp500-1.d	23	8.32E-07	8.00E-10	5.00E-12	16.689
gpp500-2.d	18	2.85E-06	1.78E-09	2.57E-12	13.964
gpp500-3.d	19	5.57E-07	1.66E-10	1.14E-11	14.415
gpp500-4.d	17	1.35E-05	1.06E-10	2.94E-12	13.301
hinf1.dat-	34	1.93E-07	1.34E-07	3.92E-09	0.416
hinf10.dat	31	3.81E-04	7.04E-06	8.83E-08	0.365
hinf11.dat	30	4.46E-04	5.82E-06	1.57E-07	0.482
hinf12.dat	40	8.04E-05	1.32E-05	2.88E-07	0.731
hinf13.dat	23	7.33E-02	2.28E-05	7.37E-07	0.498
hinf14.dat	29	1.75E-05	1.40E-06	4.91E-09	0.918
hinf15.dat	30	1.74E-03	1.01E-05	1.26E-08	1.096
hinf2.dat-	14	2.20E-04	3.38E-07	6.61E-08	0.171
hinf3.dat-	17	4.20E-04	2.84E-06	1.56E-08	0.183
hinf4.dat-	24	1.03E-05	7.08E-07	1.91E-09	0.324
hinf5.dat-	15	6.80E-01	5.74E-05	1.58E-06	0.163
hinf6.dat-	21	9.35E-04	8.93E-06	2.08E-09	0.22
hinf7.dat-	22	4.44E-06	1.63E-05	4.44E-13	0.237
hinf8.dat-	23	1.19E-04	2.19E-06	8.78E-10	0.324
hinf9.dat-	22	8.97E-07	4.58E-07	4.57E-14	0.223

Table 4: Results from regularized SDP (SDPR) on 90 problems (cont.)

ProbName	#iter	Gap	P_infeas	D_infeas	CPUTT (secs)
infd1.dat-	31	1.17E-01	1.93E-01	3.25E-09	0.454
infd2.dat-	31	2.52E-01	2.55E-01	2.16E-08	0.374
infp1.dat-	31	2.59E+13	1.95E-01	1.65E+00	0.481
infp2.dat-	31	9.90E+12	3.94E-02	1.28E+00	0.491
maxG11.dat	15	3.53E-06	1.33E-12	1.00E-12	14.888
maxG32.dat	17	1.01E-05	6.46E-13	1.00E-12	151.427
maxG51.dat	19	2.28E-06	5.16E-14	1.00E-12	35.612
mcp100.dat	13	1.31E-07	2.10E-12	1.00E-12	1.315
mcp124-1.d	13	3.00E-07	1.68E-12	1.00E-12	0.771
mcp124-2.d	13	4.07E-07	2.53E-13	1.00E-12	0.613
mcp124-3.d	13	2.17E-06	3.63E-13	1.00E-12	0.675
mcp124-4.d	13	1.06E-05	1.09E-11	1.00E-12	0.566
mcp250-1.d	14	7.46E-07	1.95E-13	1.00E-12	1.33
mcp250-2.d	13	2.79E-06	5.41E-12	1.00E-12	1.46
mcp250-3.d	13	1.60E-05	1.92E-12	1.00E-12	1.206
mcp250-4.d	16	4.52E-06	1.10E-12	1.00E-12	1.703
mcp500-1.d	15	1.05E-06	1.12E-12	1.00E-12	4.307
mcp500-2.d	16	3.10E-06	2.76E-13	1.00E-12	6.931
mcp500-3.d	15	2.91E-05	1.08E-12	1.00E-12	9.421
mcp500-4.d	15	4.11E-05	3.18E-13	1.04E-12	8.09
qap10.dat-	31	1.55E-04	4.30E-08	7.17E-09	13.298
qap5.dat-s	13	7.18E-07	1.76E-11	9.79E-11	2.124
qap6.dat-s	17	7.89E-04	1.15E-07	2.60E-08	3.183
qap7.dat-s	23	1.91E-04	8.26E-08	1.56E-08	4.176
qap8.dat-s	32	1.56E-04	5.72E-08	1.13E-08	3.293
qap9.dat-s	24	2.39E-04	5.78E-08	1.01E-08	4.03
qpG11.dat-	17	1.79E-04	1.96E-13	1.03E-12	16.422
qpG51.dat-	21	3.89E-03	9.77E-08	5.93E-11	37.824
ss30.dat-s	21	1.44E-06	9.21E-08	6.12E-11	12.27
theta1.dat	33	2.66E-08	2.57E-11	1.28E-10	1.845
theta2.dat	26	1.04E-07	6.44E-12	1.30E-11	1.523
theta3.dat	27	5.42E-08	6.90E-12	1.26E-11	5.153
theta4.dat	21	2.65E-07	1.70E-11	5.38E-12	12.581
theta5.dat	24	3.08E-07	1.93E-11	5.56E-12	40.932
theta6.dat	21	6.81E-07	1.64E-10	1.62E-10	103.082
thetaG11.d	20	1.87E-05	8.70E-12	1.00E-12	50.265
truss1.dat	9	1.87E-08	9.45E-12	6.42E-11	0.195
truss2.dat	12	1.40E-07	4.08E-11	1.50E-10	0.241
truss3.dat	12	5.50E-08	5.09E-12	1.00E-12	0.171
truss4.dat	10	9.55E-09	5.40E-11	2.58E-11	0.134
truss5.dat	15	2.30E-07	2.00E-10	1.51E-11	0.615
truss6.dat	25	6.56E-06	4.68E-08	2.56E-09	0.803
truss7.dat	24	3.31E-05	2.91E-08	5.86E-09	0.604
truss8.dat	16	8.34E-08	6.59E-11	1.19E-11	2.895

The interpretation of Figure 1 is that these two solvers are identical within a factor $\tau < 1.6$. In other words, performance profile saying that the number of iterations needed to solve a problem by one solver is not more than 1.6 time of the iterations needed by the other one.

12 Discussion and future research

We present a primal-dual regularization for SDPs for two directions, NT and dual of HKM. There are about twenty directions and we believe it is possible to extend the convergence analysis to some of these directions. In our analysis, we take advantage of part one of Lemma 1 this property may not hold for other directions. However, conditions $\lambda(H_P(XZ)) = \lambda(XZ)$ may be replaced by weaker conditions to include more directions in the convergence analysis.

The strong similarity between interior-point methods for SDP and for the larger classes of problems, such as conic programming and even in more general convex programming problems suggests that our results can be extended to those general classes. Our approach is based on bounding the eigenvalues of reduced system (44) but we believe that using 3-by-3 system (43) has theoretical advantages over the reduced system (44); the assumptions for convergence are weaker and the proofs are simpler. Orban and Friedlander [Orb10] use 3-by-3 system (43) in quadratic programming context and simplify the convergence analysis using of Armand

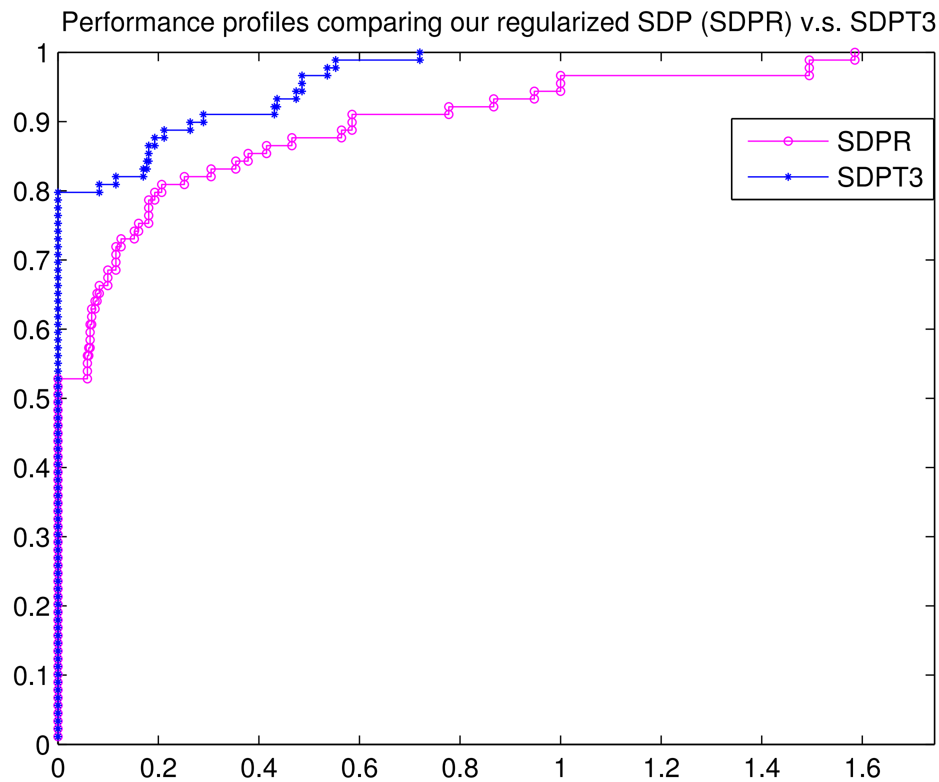

Figure 1: Performance profile comparing our implementation v.s. SDPT3 on a set of 90 problems

and Benoist results [AB11]. It remains to be discover whether or not the Armand and Benoist [AB11] can be extended to SDP and whether using 3-by-3 system has advantages in practice.

A Appendix

We list some important operation rules for Kronecker products that used in this paper. Although some of the proofs are easy to derive, we refer to book [HJ91] for the proofs. $\lambda(A)$ is used for the spectrums of A.

$$A \otimes B = [a_{ij}B], \tag{91a}$$
$$\mathbf{vec}(AXB) = (B^T \otimes A)\mathbf{vec}X, \tag{91b}$$
$$(A \otimes B)^T = A^T \otimes B^T, \tag{91c}$$
$$(A \otimes B)^{-1} = A^{-1} \otimes B^{-1}, \tag{91d}$$
$$(A \otimes B)(C \otimes D) = (AC \otimes BD), \tag{91e}$$
$$\text{If } \lambda(A) = \mu_i \text{ and } \lambda(B) = \nu_j \text{ then } \lambda(A \otimes B) = \mu_i\nu_j, \tag{91f}$$
$$\lambda(AB) = \lambda(BA), \text{ where } A, B \in \mathbb{R}^{n\times n}. \tag{91g}$$

If A is a symmetric $n \times n$ real matrix we order eigenvalues of A by $\lambda_1(A) \geq \lambda_2(A) \geq \cdots \geq \lambda_n(A)$ the following lemma gives us some useful relations between the smallest and the largest eigenvalues of the sum of two matrices.

Lemma 6 *If A and B are $n \times n$ symmetric matrices then*

$$\lambda_n(A+B) \geq \lambda_n(A) + \lambda_n(B), \tag{92a}$$
$$\lambda_1(A+B) \leq \lambda_1(A) + \lambda_1(B), \tag{92b}$$
$$\|A - C\|_F \geq \lambda_n(A) - \lambda_n(C), \tag{92c}$$
$$|\lambda_n(A)| \leq \frac{1}{\sqrt{n}}\|A\|_F. \tag{92d}$$

Proof. By Theorem 8.1.5 of [GCVL96, page 396] we have

$$\lambda_i(A) + \lambda_n(B) \leq \lambda_i(A+B) \leq \lambda_i(A) + \lambda_1(B) \quad \forall i = 1, 2, \ldots n.$$

Particularly, for $i = n$ and $i = 1$ we can easily derive (92a) and (92b). To prove (92d), it is enough to let $A + B = C$ and use (92a) together with the fact that $\lambda_n(B) \geq -\|B\|_F$. □

References

[AB07] M.F. Anjos and S. Burer. On handling free variables in interior-point methods for conic linear optimization. *SIAM Journal on Optimization*, 18(4):1310–1325, 2007.

[AB11] P. Armand and J. Benoist. Uniform boundedness of the inverse of a jacobian matrix arising in regularized interior-point methods. *Mathematical Programming*, 6 pages, DOI: 10.1007/s10107-011-0498-3, 2011.

[AG99] A. Altman and J. Gondzio. Regularized symmetric indefinite systems in interior point methods for linear and quadratic optimization. *Optimization Methods and Software*, 11(1-4):275–302, 1999.

[AHO98] F. Alizadeh, J.P.A. Haeberly, and M.L. Overton. Primal-dual interior-point methods for semidefinite programming: convergence rates, stability and numerical results. *SIAM Journal on Optimization*, 8(3):746–768, 1998.

[Ali95] F. Alizadeh. Interior point methods in semidefinite programming with applications to combinatorial optimization. *SIAM Journal on Optimization*, 5(1):13–51, 1995.

[Bor99] B. Borchers. Sdplib 1.2, a library of semidefinite programming test problems. *Optimization Methods and Software*, 11(1-4):683–690, 1999.

[D.G97] D.G. Luenberger. *Optimization by Vector Space Methods.* Wiley-Interscience, 1997.

[DM02] E. Dolan and J.J. Moré. Benchmarking optimization software with performance profiles. *Mathematical Programming*, 91:201–213, 2002.

[FB12] M.F. Anjos and J.B. Lasserre. *Handbook on Semidefinite, Conic and Polynomial Optimization.* Springer, 2012.

[FT07] M.P. Friedlander and P. Tseng. Exact regularization of convex programs. *SIAM Journal on Optimization*, 18(4):1326–1350, 2007.

[GCVL96] G.H. Golub and C.F. Van Loan. *Matrix Computations*, 3rd edition. Johns Hopkins University Press, 1996.

[GSS96] P.E. Gill, M.A. Saunders, and J.R. Shinnerl. On the stability of cholesky factorization for symmetric quasidefinite systems. *SIAM Journal on Matrix Analysis and Applications*, 17:35, 1996.

[Hel00] C. Helmberg. *Semidefinite programming for combinatorial optimization.* Konrad-Zuse-Zentrum für Informationstechnik Berlin, 2000.

[HJ91] A. Horn and C.R. Johnson. *Topics in Matrix Analysis.* Cambridge University Press, London, 1991.

[HRVW96] C. Helmberg, F. Rendl, R.J. Vanderbei, and H. Wolkowicz. An interior-point method for semidefinite programming. *SIAM Journal on Optimization*, 6(2):342–361, 1996.

[KSH97] M. Kojima, S. Shindoh, and S. Hara. Interior-point methods for the monotone semidefinite linear complementarity problem in symmetric matrices. *SIAM Journal on Optimization*, 7:86, 1997.

[Més98] C. Mészáros. On free variables in interior point methods. *Optimization Methods and Software*, 9(1-3):121–139, 1998.

[M.J99] M.J. Todd. A study of search directions in primal-dual interior-point methods for semidefinite programming. *Optimization Methods and Software*, 11(1-4):1–46, 1999.

[MPR+09] J. Malick, J. Povh, F. Rendl, A. Wiegele, et al. Regularization methods for semidefinite programming. *SIAM Journal on Optimization*, 20(1):336–356, 2009.

[MZ98] R.D.C. Monteiro and Y. Zhang. A unified analysis for a class of long-step primal-dual path-following interior-point algorithms for semidefinite programming. *Mathematical Programming*, 81(3):281–299, 1998.

[NW06] J. Nocedal and S. J. Wright. *Numerical Optimization.* Springer Series in Optimizations Research. Springer, second edition, 2006.

[Orb10] M. P. Friedlander D. Orban. A primal-dual regularized interior-point method for convex quadratic programs. *Les Cahiers du GERAD*, G–2010–47, HEC Montréal, Canada, 2010.

[PS99] G. Pataki and S. Schmieta. The DIMACS library of semidefinite-quadratic-linear programs. *Computational Optimization Research, Columbia University, NY*, 1999.

[R.D97] R.D.C. Monteiro. Primal-dual path-following algorithms for semidefinite programming. *SIAM Journal on Optimization*, 7(3):663–678, 1997.

[RW92] T. Rusten and R. Winther. A preconditioned iterative method for saddlepoint problems. *SIAM Journal on Matrix Analysis and Applications*, 13:887, 1992.

[S$^+$96] M.A. Saunders et al. Cholesky-based methods for sparse least squares: The benefits of regularization. In: L. Adams and J.L. Nazareth (eds.), *Linear and Nonlinear Conjugate Gradient-Related Methods*, SIAM, Philadelphia, pages 92–100, 1996.

[Sau95] M.A. Saunders. Solution of sparse rectangular systems using lsqr and craig. *BIT Numerical Mathematics*, 35(4):588–604, 1995.

[SW94] D. Silvester and A. Wathen. Fast iterative solution of stabilised stokes systems part ii: using general block preconditioners. *SIAM Journal on Numerical Analysis*, 31(5):1352–1367, 1994.

[TTT] K.C. Toh, R.H. Tütüncü, and M.J. Todd. On the implementation and usage of SDPT3–a MATLAB software package for semidefinite-quadratic-linear programming, version 4.0, July 2006.

[TTT98] M.J. Todd, K.C. Toh, and R.H. Tütüncü. On the nesterov-todd direction in semidefinite programming. *SIAM Journal on Optimization*, 8(3):769–796, 1998.

[Van94] R.J. Vanderbei. Commentary – interior-point methods: Algorithms and formulations. *ORSA Journal on Computing*, 6(1):32–34, 1994.

[Van95] R.J. Vanderbei. Symmetric quasi-definite matrices. *SIAM Journal on Optimization*, 5(1):100–113, 1995.

[VB96] L. Vandenberghe and S. Boyd. Semidefinite programming. *SIAM Review*, 38(1):49–95, 1996.

[VB99] L. Vandenberghe and S. Boyd. Applications of semidefinite programming. *Applied Numerical Mathematics*, 29(3):283–299, 1999.

[WSV00] H. Wolkowicz, R. Saigal, and L. Vandenberghe. *Handbook of Semidefinite Programming: Theory, Algorithms, and Applications*, volume 27. Springer, 2000.

[Zha98] Y. Zhang. On extending some primal-dual interior-point algorithms from linear programming to semidefinite programming. *SIAM Journal on Optimization*, 8:365–386, 1998.