Samuel Burer[1] · Renato D.C. Monteiro[2] · Yin Zhang[3]

Solving a Class of Semidefinite Programs via Nonlinear Programming

September 1999
(Revised December 1999, May 2001 and October 2001)

Abstract. In this paper, we introduce a transformation that converts a class of linear and nonlinear semidefinite programming (SDP) problems into nonlinear optimization problems. For those problems of interest, the transformation replaces matrix-valued constraints by vector-valued ones, hence reducing the number of constraints by an order of magnitude. The class of transformable problems includes instances of SDP relaxations of combinatorial optimization problems with binary variables as well as other important SDP problems. We also derive gradient formulas for the objective function of the resulting nonlinear optimization problem and show that both function and gradient evaluations have affordable complexities that effectively exploit the sparsity of the problem data. This transformation, together with the efficient gradient formulas, enables the solution of very large-scale SDP problems by gradient-based nonlinear optimization techniques. In particular, we propose a first-order log-barrier method designed for solving a class of large-scale linear SDP problems. This algorithm operates entirely within the space of the transformed problem while still maintaining close ties with both the primal and the dual of the original SDP problem. Global convergence of the algorithm is established under mild and reasonable assumptions.

Key words. transformation – semidefinite program – semidefinite relaxation – nonlinear programming – first-order methods – log-barrier algorithms – interior-point methods

1. Introduction

This paper concerns the solution of large-scale linear and nonlinear semidefinite programs. In the past several years, semidefinite programming (SDP) has been one of the most active research areas in mathematical programming. There are two major factors that are responsible for this increased interest in SDP. Firstly, SDP has found numerous applications in various fields, such as statistics, structural design, electrical engineering and combinatorial optimization. Secondly, interior-point methods have proven to be reliable and effective in the solution of small to moderately sized problems. It is often the case, however, that problems arising from applications have very large sizes. This is especially true for SDP problems that arise as relaxations of combinatorial optimization problems.

Generally speaking, the most popular interior-point methods are primal-dual methods. For large-scale problems, however, the capability of these methods is severely limited due to their high demand

1. Dept. of Management Sciences, University of Iowa, Iowa City, IA 52242, USA. e-mail: `samuel-burer@uiowa.edu`.
This author was supported in part by NSF Grants INT-9910084, CCR-9700448 and CCR-9902010.

2. School of ISyE, Georgia Tech, Atlanta, Georgia 30332, USA. e-mail: `monteiro@isye.gatech.edu`.
This author was supported in part by NSF Grants INT-9600343, INT-9910084, CCR-9700448, CCR-9902010.

3. Department of Computational and Applied Mathematics, Rice University, Houston, Texas 77005, USA. e-mail: `yzhang@caam.rice.edu`.
This author was supported in part by DOE Grant DE-FG03-97ER25331, DOE/LANL Contract 03891-99-23 and NSF Grant DMS-9973339.

for storage and computation arising from the use of Newton's method, which requires the solution of large, dense linear systems in each iteration. Even with efforts to exploit sparsity in the problem data (see [6], for example), the practical limitations of primal-dual interior-point methods still remain. Recently, Benson et al. [1] proposed a potential-reduction dual-scaling interior-point method that can better take advantage of the special structure in SDP relaxations of certain combinatorial optimization problems, hence enabling it to solve problems with matrix dimension up to several thousands. Nonetheless, for larger problems this approach still encounters formidable difficulty because it, like the related primal-dual methods, requires the storage and factorization of a dense matrix.

In the past few years, several nonlinear programming methods have been proposed for solving large SDP problems (see[2, 9, 10]), and a common feature of these methods is that each relies only on gradient-based information and consequently avoids costly matrix operations and linear solves. The approach by Helmberg and Rendl [9] for solving a special class of linear SDPs is to optimize a certain partial Lagrangian dual formulation, whose objective function is non-differentiable, using the bundle method for non-smooth convex programming. In contrast, Homer and Peinado [10] use the change of variables $X = VV^T$, $V \in \Re^{n \times n}$, where X is the primal matrix variable of the MAXCUT SDP relaxation (see Section 2), to transform it into an unconstrained, differentiable nonlinear programming problem in the new variable V. More recently, Burer and Monteiro [2] proposed a variant of Homer and Peinado's method by using the change of variable $X = LL^T$, where L is a lower triangular matrix. We note that the substitutions $X = VV^T$ and $X = LL^T$ can be viewed as a matrix analogy to the square-slack variable substitution for scalar inequality constraints.

In this paper, we propose a transformation for converting an SDP problem of a particular form into a nonlinear programming problem. The motivation for introducing such a transformation is two-fold: to reduce the number of constraints and to facilitate the use of gradient-based, nonlinear optimization algorithms that are free from the need for second-order information and linear system solving.

We start with the same change of variables used in [2], that is, we employ the idea of the Cholesky factorization LL^T of a positive semidefinite matrix S. The novelty of our approach, however, lies in the derivation of a fundamental mapping that expresses a subset of the lower entries (say, the off-diagonal) of the Cholesky factor L as a function of the corresponding entries (the off-diagonal) of S and the other lower entries (the diagonal) of L. With the aid of this mapping, we show how to transform a large and important class of linear and/or nonlinear SDP problems into the problem of minimizing a nonlinear function over an "orthant" of the form $\Re^n_{++} \times \Re^N$, where n is the size of L and N is an appropriate nonnegative integer. This transformation has several desirable properties, including that the number of variables of the resulting nonlinear program is minimal in the sense that it equals the dimension of the feasible region of the original SDP problem and that the positive-semidefinite-matrix constraint is replaced by a positive-vector constraint.

The paper is organized as follows. In Section 2, we describe the class of SDP problems to which the aforementioned transformation applies and give several examples of linear and nonlinear SDP problems lying in this class in order to illustrate its generality. In Section 3, we describe the transformation and show that every problem in the above class can be converted to a nonlinear optimization problem over an orthant of the form $\Re^n_{++} \times \Re^N$, where n and N are as described above. In particular, the fundamental mapping that forms the basis of this conversion is also described in this section. In Section 4, we develop explicit formulas for the first derivative of the objective function of the nonlinear optimization problem. In Section 5, we analyze the computational requirements involved in the function and derivative evaluations for both dense and sparse cases. In Section 6, we propose a first-order, or gradient-based, log-barrier algorithm for a class of linear SDP problems and establish its global convergence under very mild conditions. Some final remarks are offered in the last section.

1.1. Preliminary notation

In this paper, $\Re$, $\Re^n$, and $\Re^{n\times n}$ denote the space of real numbers, real n-dimensional column vectors, and real $n \times n$ matrices, respectively. By $\mathcal{S}^n$ we denote the space of real $n \times n$ symmetric matrices, and we define $\mathcal{S}^n_+$ and $\mathcal{S}^n_{++}$ to be the subsets of $\mathcal{S}^n$ consisting of the positive semidefinite and positive definite matrices, respectively. We write $A \succeq 0$ and $A \succ 0$ to indicate that $A \in \mathcal{S}^n_+$ and $A \in \mathcal{S}^n_{++}$, respectively. We let $\mathrm{tr}(A)$ denote the trace of a matrix $A \in \Re^{n\times n}$, namely $\mathrm{tr}(A)$ denotes the sum of the diagonal elements of A. Moreover, for $A, B \in \Re^{n\times n}$, we define $A \bullet B \equiv \mathrm{tr}(A^T B)$. If $\mathcal{I}$ is a finite set, we let $|\mathcal{I}|$ denote its cardinality, that is, the number of elements of $\mathcal{I}$.

This paper is heavy in notation. Instead of collecting all the symbols in one place, we will gradually develop additional notation throughout the course of the paper.

2. The Problem Class and Examples

In this section, we introduce a standard form problem which describes the class of semidefinite programming problems studied in this paper. We then present several specific examples of important linear and nonlinear SDP problems which are members of this class.

Define

$$\mathcal{K} \equiv \{(i,j) : 1 \le j \le i \le n\}, \tag{1}$$

i.e., $\mathcal{K}$ consists of the ordered index pairs corresponding to the lower-triangular positions of an $n \times n$ matrix. Let $\mathcal{D} \subseteq \mathcal{K}$ denote the subset corresponding to the diagonal indices, that is, $\mathcal{D} \equiv \{(i,i) : i = 1, 2, \ldots, n\}$. In addition, for any $\mathcal{A} \subseteq \mathcal{K}$, define

$$\mathcal{S}^{\mathcal{A}} \equiv \{X \in \mathcal{S}^n : X_{ij} = 0 \text{ for every } (i,j) \in \mathcal{K} \setminus \mathcal{A}\},$$

i.e., $\mathcal{S}^{\mathcal{A}}$ is the subspace of $\mathcal{S}^n$ consisting of those matrices which have zeros in the positions corresponding to $\mathcal{K} \setminus \mathcal{A}$. For instance, $\mathcal{S}^{\mathcal{D}}$ is the set of diagonal matrices.

Let $\mathcal{I}$ be a proper subset of $\mathcal{K}$ which contains all the diagonal indices, namely, $\mathcal{D} \subseteq \mathcal{I} \subset \mathcal{K}$, and let m be a nonnegative integer. Given functions $g : \mathcal{S}^{\mathcal{I}} \times \Re^m \to \Re$ and $H : \Re^m \to \mathcal{S}^n$, consider the following semidefinite program (NLSDP):

$$(NLSDP) \qquad \min\left\{ g(Z^{\mathcal{I}}, y) : (Z^{\mathcal{I}}, y) \in \mathcal{F}_{sdp} \right\}, \tag{2}$$

where

$$\mathcal{F}_{sdp} \equiv \left\{ (Z^{\mathcal{I}}, y) \in \mathcal{S}^{\mathcal{I}} \times \Re^m : Z^{\mathcal{I}} + H(y) \succeq 0 \right\}. \tag{3}$$

Note that both functions g and H can be either linear or nonlinear, and $m = 0$ means that the variable y is non-existent. In addition, since $\mathcal{I}$ contains the diagonal indices, for any $y \in \Re^m$, there exists some $Z^{\mathcal{I}} \in \mathcal{S}^{\mathcal{I}}$ such that $Z^{\mathcal{I}} + H(y) \succ 0$, implying that the interior of $\mathcal{F}_{sdp}$, which we denote as $\text{int}\,(\mathcal{F}_{\text{sdp}})$, is nonempty.

Many SDP problems can be put in the form of (2) as the following examples demonstrate.

Example 1. A Class of Dual Linear SDP Problems

Let an index set $\mathcal{I}$ such that $\mathcal{D} \subseteq \mathcal{I} \subset \mathcal{K}$ and a nonnegative integer m be given. Consider the following primal SDP:

$$\max \left\{ C \bullet X : X_{ij} = B^{\mathcal{I}}_{ij}, (i,j) \in \mathcal{I};\ A_k \bullet X = b_k,\ k = 1, \ldots, m;\ X \succeq 0 \right\}, \tag{4}$$

where $C, A_1, \ldots, A_m \in \mathcal{S}^n$, $B^{\mathcal{I}} \in \mathcal{S}^{\mathcal{I}}$, and $b \in \Re^m$ are the data. Its dual is

$$\min \left\{ B^{\mathcal{I}} \bullet Z^{\mathcal{I}} + b^T y : Z^{\mathcal{I}} + \sum_{k=1}^{m} y_k A_k - C \succeq 0;\ (Z^{\mathcal{I}}, y) \in \mathcal{S}^{\mathcal{I}} \times \Re^m \right\}, \tag{5}$$

which is in the form of (2) with $g(Z^{\mathcal{I}}, y) = B^{\mathcal{I}} \bullet Z^{\mathcal{I}} + b^T y$ and $H(y) = \sum_{k=1}^{m} y_k A_k - C$.

Example 2. SDP Relaxations of Binary Combinatorial Optimization Problems

In SDP relaxations of binary combinatorial optimization problems, the binary constraints $x_i^2 = 1,\ i = 1, 2, \ldots, n$, are relaxed into $X_{ii} = 1,\ i = 1, 2, \ldots, n$, (see [19,13,5,16] for the evolution of such relaxations), resulting in primal linear SDP problems in the form of (4) with $\mathcal{I} = \mathcal{D}$ and $B^{\mathcal{I}} = I$, the identity matrix. The dual form of these SDP relaxations are special instances of (5). In particular, when $m = 0$, we obtain the MAXCUT SDP relaxation that forms the basis of the approximation algorithm of Goemans and Williamson in [7]. Another example where $m > 0$ is given next.

Example 3. The Lovász Theta Number

Let a simple graph $G = (V, E)$ with vertex set $V = \{1, \ldots, n\}$ be given. For notational convenience, we assume that the edge set E is given as a subset of $\mathcal{K}$, i.e., an edge between u and v, $u \neq v$, is represented in E as either (u, v) if $u > v$ or (v, u) if $v > u$. Let e_{ijk} denote the $(n+1)$-dimensional column vector of all zeros except ones in the i-th, j-th, and k-th positions, and define $A_{ijk} \equiv e_{ijk} e_{ijk}^T$. Then the (unweighted) Lovász theta number $\vartheta(G)$ of the graph G is the optimal value of the $(n+1)$-dimensional SDP

$$\max \left\{ C \bullet X : X_{ii} = 1, i = 1, \ldots, n+1;\ A_{ij(n+1)} \bullet X = 1, (i,j) \in E;\ X \succeq 0 \right\},$$

where

$$C = \begin{bmatrix} 1/2 & & & 1/4 \\ & \ddots & & \vdots \\ & & 1/2 & 1/4 \\ 1/4 & \dots & 1/4 & \end{bmatrix} \in \mathcal{S}^{n+1}$$

(see Laurent et al. [11]). This problem is an instance of the primal SDP (4) with $\mathcal{I} = \mathcal{D}$ and $m = |E|$. Hence, its dual is an instance of (5).

Example 4. A Second Version of the Lovász Theta Number

Let the graph G be as in the previous example. It is well-known that the Lovász theta number $\vartheta(G)$ of G also equals the maximum value of the n-dimensional SDP

$$\max\left\{J \bullet X : X_{ij} = 0, (i,j) \in E;\ \mathrm{tr}(X) = 1; X \succeq 0\right\},$$

where $J = ee^T$ is the $n \times n$ matrix of all ones. This linear SDP can be reformulated as a nonlinear SDP in the form of (2) with $\mathcal{I} \equiv \mathcal{K} \setminus E$, $m = 0$, and

$$g(X^{\mathcal{I}}) = -\frac{J \bullet X^{\mathcal{I}}}{\mathrm{tr}(X^{\mathcal{I}})}.$$

Example 5. The Positive Semidefinite Matrix Completion Problem

Given an index set $\mathcal{M}$ such that $\mathcal{D} \subseteq \mathcal{M} \subset \mathcal{K}$ and a partially positive semidefinite matrix $A^{\mathcal{M}} \in \mathcal{S}^{\mathcal{M}}$ (i.e., all the principal minors of $A^{\mathcal{M}}$ within $\mathcal{M}$ are positive semidefinite), the positive semidefinite matrix completion problem is either to find a completion matrix $X^{\mathcal{N}} \in \mathcal{S}^{\mathcal{N}}$, where $\mathcal{N} \equiv \mathcal{K} \setminus \mathcal{M}$, such that $A^{\mathcal{M}} + X^{\mathcal{N}} \succeq 0$ or to determine that no such matrix exists.

The problem can be formulated as the nonlinear convex program

$$\min\left\{\|X^{\mathcal{D}}\|_F^2 : X^{\mathcal{D}} + A^{\mathcal{M}} + X^{\mathcal{N}} \succeq 0;\ (X^{\mathcal{D}}, X^{\mathcal{N}}) \in \mathcal{S}^{\mathcal{D}} \times \mathcal{S}^{\mathcal{N}}\right\},$$

which is in the form of (2) with $\mathcal{I} = \mathcal{D} \cup \mathcal{N}$, $m = 0$, and H the constant matrix $A^{\mathcal{M}}$. Whenever the optimal value is zero, a completion matrix is found; otherwise, there exists no such completion matrix.

3. Reduction to Nonlinear Programming

In this section, we introduce a mapping that transforms the semidefinite program (2) into a nonlinear optimization problem whose feasible region is the orthant $\Re^n_{++} \times \Re^N$, where $n = |\mathcal{D}|$ and $N = |\mathcal{I} \setminus \mathcal{D}| + m$. We also discuss the differentiability of this transformation and conclude with an explicit statement of the equivalent nonlinear programming formulation of (2).

3.1. Further notation

We first define the following symbols: $\mathcal{L}^n, \mathcal{L}^n_+, \mathcal{L}^n_{++}, \mathcal{L}^{\mathcal{A}}, \pi^l_{\mathcal{A}}, \pi_{\mathcal{A}}$ and $\mathcal{L}^{\mathcal{I}}_{++}$.

- Let $\mathcal{L}^n \subset \Re^{n\times n}$ denote the set of real $n \times n$ lower triangular matrices, and let $\mathcal{L}^n_+$ and $\mathcal{L}^n_{++}$ denote the subsets of $\mathcal{L}^n$ consisting of those lower triangular matrices with nonnegative and positive diagonal entries, respectively.
- For any $\mathcal{A} \subseteq \mathcal{K}$, define $\mathcal{L}^{\mathcal{A}} \equiv \{L \in \mathcal{L}^n : L_{ij} = 0 \text{ if } (i,j) \notin \mathcal{A}\}$.
- Let $\pi^l_{\mathcal{A}} : \Re^{n\times n} \to \mathcal{L}^{\mathcal{A}}$ and $\pi_{\mathcal{A}} : \Re^{n\times n} \to \mathcal{S}^{\mathcal{A}}$ denote the orthogonal (in Frobenius norm) projection operators from $\Re^{n\times n}$ onto the subspaces $\mathcal{L}^{\mathcal{A}}$ and $\mathcal{S}^{\mathcal{A}}$, respectively.
- For any $\mathcal{I} \subset \mathcal{K}$ containing $\mathcal{D}$, let $\mathcal{L}^{\mathcal{I}}_{++} \equiv \{L \in \mathcal{L}^{\mathcal{I}} : L_{ii} > 0, i = 1, 2, \ldots, n\}$. Note that $\mathcal{L}^{\mathcal{D}}_{++} = \mathcal{L}^n_{++}$.

It is well known that for any $S \in \mathcal{S}^n_{++}$, there exists a unique $L \in \mathcal{L}^n_{++}$ such that $S = LL^T$; L is called the lower Cholesky factor of S. We denote the transformation which maps a positive definite matrix into its lower Cholesky factor by $\text{chol} : \mathcal{S}^n_{++} \to \mathcal{L}^n_{++}$. Namely, for any $S \in \mathcal{S}^n_{++}$,

$$L = \text{chol}(S) \quad \Longleftrightarrow \quad LL^T = S \text{ and } L \in \mathcal{L}^n_{++}.$$

For each $k = 1, \ldots, m$, let $e_k \in \Re^m$ denote the k-th coordinate vector, and for each $(i,j) \in \mathcal{K}$, let E^{ij} denote the $n \times n$ lower triangular matrix with zeros everywhere except a one in position (i,j).

3.2. The Main ideas

From now on, we assume that $\mathcal{I}$ and $\mathcal{J}$ are given and that $\mathcal{D} \subseteq \mathcal{I} \subset \mathcal{K}$ and $\mathcal{J} \equiv \mathcal{K} \backslash \mathcal{I}$. We also assume that a nonnegative integer m and functions $g : \mathcal{S}^n \times \Re^m \to \Re$ and $H : \Re^m \to \mathcal{S}^n$ are given. For the purpose of following the main ideas presented in this subsection, it would be sufficient, and indeed preferable, for the reader to assume that $\mathcal{I}$ represents the diagonal indices and $\mathcal{J}$ the off-diagonal ones.

We employ two key ideas to transform the feasibility set $\mathcal{F}_{sdp}$ of the SDP problem (2) into a certain "orthant":

1. *The square slack variable substitution* $S = LL^T$ for any $S \in \mathcal{S}^n_+$ that turns the positive-semidefinite constraint $Z^{\mathcal{I}} + H(y) \succeq 0$ into the equality constraint

$$Z^{\mathcal{I}} + H(y) = LL^T. \tag{6}$$

2. *An elimination scheme* that uses the equations in (6) to eliminate all the variables L_{ij} for $(i,j) \in \mathcal{J}$, and all the variables in $Z^{\mathcal{I}}$ as well.

The square slack substitution is an old idea in nonlinear programming that has not been regarded as practically useful in general; in part because it can turn a convex feasibility set into a nonconvex one. However, in our context it will be the first step in the process of turning a matrix-valued constraint into a vector-valued one, while still allowing us to construct globally convergent algorithms for linear SDP problems as will be shown in Section 6.

While the meaning of the first idea is clear, that of the second calls for an explanation. Recall that the lower triangular index set $\mathcal{K}$ has been partitioned into $\mathcal{K} = \mathcal{I} \cup \mathcal{J}$. Accordingly, we can partition the lower triangular matrix L into $L = L^{\mathcal{I}} + L^{\mathcal{J}}$, and partition the equations in (6) into two systems of equations:

$$\pi_{\mathcal{J}} \left[(L^{\mathcal{I}} + L^{\mathcal{J}})(L^{\mathcal{I}} + L^{\mathcal{J}})^T - H(y) \right] = 0 \tag{7}$$

corresponding to the index set $\mathcal{J}$, and

$$\pi_{\mathcal{I}}(Z^{\mathcal{I}}) = \pi_{\mathcal{I}} \left[(L^{\mathcal{I}} + L^{\mathcal{J}})(L^{\mathcal{I}} + L^{\mathcal{J}})^T - H(y) \right] \tag{8}$$

corresponding to the index set $\mathcal{I}$. Together, there are $|\mathcal{I}| + |\mathcal{J}| \equiv n(n+1)/2$ equations in (7) and (8); and there are $2|I| + |\mathcal{J}| + m$ variables: $|\mathcal{I}|$ in $L^{\mathcal{I}}$ and also in $Z^{\mathcal{I}}$, $|\mathcal{J}|$ in $L^{\mathcal{J}}$ and m in y. Our strategy is to first use the $|\mathcal{J}|$ equations in (7) to eliminate the $|\mathcal{J}|$ variables in $L^{\mathcal{J}}$, and then to use the $|\mathcal{I}|$ equations in (8) to eliminate the $|\mathcal{I}|$ variables in $Z^{\mathcal{I}}$.

A fundamental observation, which will be formally stated and proved in Lemma 1, is that given any $H(y) \in \mathcal{S}^n$ and $L^{\mathcal{I}} \in \mathcal{L}^{\mathcal{I}}_{++}$, the $|\mathcal{J}|$ variables in $L^{\mathcal{J}}$ can be uniquely determined and explicitly computed from the $|\mathcal{J}|$ equations in (7). In other words, system (7) defines $L^{\mathcal{J}}$ as a function of $(L^{\mathcal{I}}, y)$, namely, $L^{\mathcal{J}} = L^{\mathcal{J}}(L^{\mathcal{I}}, y)$. By substituting $L^{\mathcal{J}} = L^{\mathcal{J}}(L^{\mathcal{I}}, y)$ into (8), we obtain $Z^{\mathcal{I}}$ as a function of $(L^{\mathcal{I}}, y)$ as well. In fact, since $L^{\mathcal{J}}(L^{\mathcal{I}}, y)$ satisfies (7), we have

$$Z^{\mathcal{I}} = (L^{\mathcal{I}} + L^{\mathcal{J}}(L^{\mathcal{I}}, y))(L^{\mathcal{I}} + L^{\mathcal{J}}(L^{\mathcal{I}}, y))^T - H(y) \equiv Z^{\mathcal{I}}(L^{\mathcal{I}}, y) \in \mathcal{S}^{\mathcal{I}}.$$

The above two functions, $L^{\mathcal{J}}(L^{\mathcal{I}}, y)$ and $Z^{\mathcal{I}}(L^{\mathcal{I}}, y)$, allow us to construct a homeomorphism between the sets $\mathcal{L}^{\mathcal{I}}_{++} \times \Re^m$ and $\mathrm{int}\,(\mathcal{F}_{\mathrm{sdp}})$ for sufficiently smooth H, which maps a point $(L^{\mathcal{I}}, y) \in \mathcal{L}^{\mathcal{I}}_{++} \times \Re^m$ into a point $(Z^{\mathcal{I}}, y) \equiv (Z^{\mathcal{I}}(L^{\mathcal{I}}, y), y) \in \mathrm{int}\,(\mathcal{F}_{\mathrm{sdp}})$.

3.3. Transforming the feasibility set into an orthant

In this subsection we formalize the ideas developed in the previous subsection.

Lemma 1. *For each pair $(L^{\mathcal{I}}, H) \in \mathcal{L}^{\mathcal{I}} \times \mathcal{S}^n$ such that $L^{\mathcal{I}}_{ii} \neq 0$ for all $i = 1, \ldots, n$, there exists a unique $L^{\mathcal{J}} \in \mathcal{L}^{\mathcal{J}}$ satisfying*

$$\pi_{\mathcal{J}}(H) = \pi_{\mathcal{J}} \left((L^{\mathcal{I}} + L^{\mathcal{J}})(L^{\mathcal{I}} + L^{\mathcal{J}})^T \right). \tag{9}$$

Proof. System (9) can be expressed as $|\mathcal{J}|$ equations, one for each $(i,j) \in \mathcal{J}$:

$$H_{ij} = \sum_{s=1}^{n} (L^{\mathcal{I}}_{is} + L^{\mathcal{J}}_{is})(L^{\mathcal{I}}_{js} + L^{\mathcal{J}}_{js}) = L^{\mathcal{J}}_{ij} L^{\mathcal{I}}_{jj} + \sum_{s=1}^{j-1} (L^{\mathcal{I}}_{is} + L^{\mathcal{J}}_{is})(L^{\mathcal{I}}_{js} + L^{\mathcal{J}}_{js}),$$

where the second equality follows from the facts that both $L^{\mathcal{I}}$ and $L^{\mathcal{J}}$ are lower triangular matrices, that $i > j$, and that $L^{\mathcal{I}}_{ij} = L^{\mathcal{J}}_{jj} = 0$. It follows that any solution $L^{\mathcal{J}}$ of (9) must satisfy

$$L^{\mathcal{J}}_{ij} = \frac{1}{L^{\mathcal{I}}_{jj}} \left(H_{ij} - \sum_{s=1}^{j-1} (L^{\mathcal{I}}_{is} + L^{\mathcal{J}}_{is})(L^{\mathcal{I}}_{js} + L^{\mathcal{J}}_{js}) \right) \tag{10}$$

for all $(i,j) \in \mathcal{J}$. Notice that $L^{\mathcal{J}}_{ij}$ is expressed completely in terms of H_{ij}, $L^{\mathcal{I}}_{jj}$, and the first $j-1$ columns of $L^{\mathcal{I}}$ and $L^{\mathcal{J}}$. Letting j_0 be the smallest integer such that there exists no index pair $(i,j) \in \mathcal{J}$ with $j < j_0$, we thus see that, for each $(i, j_0) \in \mathcal{J}$,

$$L^{\mathcal{J}}_{ij_0} = \frac{1}{L^{\mathcal{I}}_{j_0 j_0}} \left(H_{ij_0} - \sum_{s=1}^{j_0-1} L^{\mathcal{I}}_{is} L^{\mathcal{I}}_{j_0 s} \right).$$

Hence, column j_0 of $L^{\mathcal{J}}$ has an explicit formula in terms $L^{\mathcal{I}}_{j_0 j_0}$, column j_0 of H, and columns 1 through $j_0 - 1$ of $L^{\mathcal{I}}$. Now using (10) and a simple induction argument, it is straightforward to see that $L^{\mathcal{J}}$ can be explicitly, recursively, and uniquely computed in a "top-to-bottom and left-to-right" order just like in a Cholesky factorization procedure.

Definition 1. *The function $L^{\mathcal{J}} : \mathcal{L}^{\mathcal{I}}_{++} \times \mathcal{S}^n \to \mathcal{L}^{\mathcal{J}}$ is defined by (9), i.e., for any given $(L^{\mathcal{I}}, H) \in \mathcal{L}^{\mathcal{I}}_{++} \times \mathcal{S}^n$, $L^{\mathcal{J}}(L^{\mathcal{I}}, H)$ is the unique solution of (9). Moreover, for any $H : \Re^m \to \mathcal{S}^n$, we also let (with a slight abuse of notation but without confusion) $L^{\mathcal{J}}(L^{\mathcal{I}}, y) \equiv L^{\mathcal{J}}(L^{\mathcal{I}}, H(y))$, which is a mapping from $\mathcal{L}^{\mathcal{I}}_{++} \times \Re^m$ to $\mathcal{L}^{\mathcal{J}}$.*

Proposition 1. *The function $L^{\mathcal{J}}(L^{\mathcal{I}}, H)$ is infinitely differentiable and analytic in $\mathcal{L}^{\mathcal{I}}_{++} \times \mathcal{S}^n$. Furthermore, $L^{\mathcal{J}}(L^{\mathcal{I}}, y)$ is as smooth in $\mathcal{L}^{\mathcal{I}}_{++} \times \Re^m$ as $H(y)$ is in $\Re^m$. In particular, if $H(y)$ is analytic, then so is $L^{\mathcal{J}}(L^{\mathcal{I}}, y)$.*

Proof. Equation (10) developed in the proof of Lemma 1 implies that $L^{\mathcal{J}}(L^{\mathcal{I}}, H)$ is infinitely differentiable in its domain. The statements regarding the differentiability of $L^{\mathcal{J}}(L^{\mathcal{I}}, y)$ follow directly from the differentiability of $L^{\mathcal{J}}(L^{\mathcal{I}}, H)$ and $H(y)$, and the definition of $L^{\mathcal{J}}(L^{\mathcal{I}}, y)$.

Definition 2. *The function $\Psi : \mathcal{L}^{\mathcal{I}}_{++} \times \Re^m \to \mathcal{S}^{\mathcal{I}} \times \Re^m$ is defined by $\Psi(L^{\mathcal{I}}, y) \equiv (Z^{\mathcal{I}}(L^{\mathcal{I}}, y), y)$, where the function $Z^{\mathcal{I}} : \mathcal{L}^{\mathcal{I}}_{++} \times \Re^m \to \mathcal{S}^{\mathcal{I}}$ is given by*

$$Z^{\mathcal{I}}(L^{\mathcal{I}}, y) \equiv (L^{\mathcal{I}} + L^{\mathcal{J}}(L^{\mathcal{I}}, y))(L^{\mathcal{I}} + L^{\mathcal{J}}(L^{\mathcal{I}}, y))^T - H(y). \tag{11}$$

Note that $Z^{\mathcal{I}}(L^{\mathcal{I}}, y) \in \mathcal{S}^{\mathcal{I}}$ due to the fact that (9) holds with $H = H(y)$ and $L^{\mathcal{J}} = L^{\mathcal{J}}(L^{\mathcal{I}}, y)$. In addition, it is straightforward to verify that the function $\Psi^{-1} : \mathrm{int}\,(\mathcal{F}_{\mathrm{sdp}}) \to \mathcal{L}^{\mathcal{I}}_{++} \times \Re^{\mathrm{m}}$ defined by

$$\Psi^{-1}(Z^{\mathcal{I}}, y) \equiv \left(\pi^l_{\mathcal{I}}[\mathrm{chol}(Z^{\mathcal{I}} + H(y))], y\right)$$

is in fact the inverse of Ψ, and so the following result holds.

Lemma 2. *The mapping Ψ is a bijection between the sets $\mathcal{L}^{\mathcal{I}}_{++} \times \Re^m$ and $\mathrm{int}\,(\mathcal{F}_{\mathrm{sdp}})$.*

Moreover, we note that all the mappings involved in the definitions of Ψ and Ψ^{-1} are infinitely differentiable in the respective domains, except possibly for H. Therefore, both are as smooth as H is in the respective domains. We state this fact as the following proposition.

Proposition 2. *The maps Ψ and Ψ^{-1} are as smooth in $\mathcal{L}^{\mathcal{I}}_{++} \times \Re^m$ and $\mathrm{int}\,(\mathcal{F}_{\mathrm{sdp}})$, respectively, as H is in $\Re^m$.*

3.4. The nonlinear programming formulation

Consider the function $f : \mathcal{L}^{\mathcal{I}}_{++} \times \Re^m \to \Re$ defined by

$$f(L^{\mathcal{I}}, y) \equiv g(\Psi(L^{\mathcal{I}}, y)), \quad \forall\, (L^{\mathcal{I}}, y) \in \mathcal{L}^{\mathcal{I}}_{++} \times \Re^m, \tag{12}$$

and the following nonlinear program (NLP):

$$(NLP) \qquad\qquad \inf \left\{ f(L^{\mathcal{I}}, y) : (L^{\mathcal{I}}, y) \in \mathcal{F}_{nlp} \right\}, \tag{13}$$

where the feasible set $\mathcal{F}_{nlp}$ is the open convex set

$$\mathcal{F}_{nlp} \equiv \mathcal{L}^{\mathcal{I}}_{++} \times \Re^m = \Psi^{-1}(\mathrm{int}\,(\mathcal{F}_{\mathrm{sdp}})). \tag{14}$$

Theorem 1. *If the function g is continuous and attains its minimum in $\mathcal{F}_{sdp}$, then*

$$\min\{g(Z^{\mathcal{I}}, y) : (Z^{\mathcal{I}}, y) \in \mathcal{F}_{sdp}\} = \inf\{f(L^{\mathcal{I}}, y) : (L^{\mathcal{I}}, y) \in \mathcal{F}_{nlp}\},$$

i.e., the optimal value of (NLSDP) equals the optimal value of (NLP).

Proof. Since Ψ is a homeomorphism between $\mathcal{F}_{nlp}$ and $\mathrm{int}\,(\mathcal{F}_{\mathrm{sdp}})$, the continuity of g implies

$$\begin{aligned}\min\{g(Z^{\mathcal{I}}, y) : (Z^{\mathcal{I}}, y) \in \mathcal{F}_{sdp}\} = \inf\{g(Z^{\mathcal{I}}, y) : (Z^{\mathcal{I}}, y) \in \mathrm{int}\,(\mathcal{F}_{\mathrm{sdp}})\} = \\ \inf\{g(\Psi(L^{\mathcal{I}}, y)) : (L^{\mathcal{I}}, y) \in \Psi^{-1}(\mathrm{int}\,(\mathcal{F}_{\mathrm{sdp}}))\} = \inf\{f(L^{\mathcal{I}}, y) : (L^{\mathcal{I}}, y) \in \mathcal{F}_{nlp}\}.\end{aligned}$$

4. The Derivative Formulas

In this section, we develop formulas for the computation of the first derivative of the objective function f of (NLP). In particular, we provide expressions for the first derivative of f in terms of the first derivatives of g and H. Also involved in the expressions for the gradient of f is a certain "dual estimate" that will play a fundamental role in the analysis of Section 6.

We assume throughout this subsection that g is differentiable on $\mathcal{F}_{sdp}$ and that H is differentiable on $\Re^m$. Given $(L^{\mathcal{I}}, y) \in \mathcal{L}^{\mathcal{I}} \times \Re^m$, we let $\nabla_{L^{\mathcal{I}}} f(L^{\mathcal{I}}, y)$ denote the matrix in $\mathcal{L}^{\mathcal{I}}$ whose (i,j)-entry is the partial derivative $(\partial f/\partial L^{\mathcal{I}}_{ij})(L^{\mathcal{I}}, y)$ for every $(i,j) \in \mathcal{I}$, and for a given $(Z^{\mathcal{I}}, y) \in \mathcal{S}^{\mathcal{I}} \times \Re^m$, we let $\nabla_{Z^{\mathcal{I}}} g(Z^{\mathcal{I}}, y)$ denote the matrix in $\mathcal{S}^{\mathcal{I}}$ whose (i,j)-entry is $(\partial g/\partial Z^{\mathcal{I}}_{ij})(Z^{\mathcal{I}}, y)$ for every $(i,j) \in \mathcal{I}$.

The following lemma establishes the existence and uniqueness of a matrix which plays an important role in the derivation of the derivatives of the function f.

Lemma 3. *For any $L \in \mathcal{L}^n_{++}$ and $D \in \mathcal{S}^{\mathcal{I}}$, there exists a unique matrix $X \in \mathcal{S}^n$ satisfying*

$$\pi_{\mathcal{I}}(X) = D, \tag{15}$$

$$\pi^l_{\mathcal{J}}(XL) = 0. \tag{16}$$

Proof. The existence and uniqueness of X follow from the following constructive argument. First, note that: (i) (15) and (16) form a square linear system of equations for $X \in \mathcal{S}^n$; (ii) $L_{jj} > 0$ for all j and $L_{kj} = 0$ for all $k < j$ since $L \in \mathcal{L}^n_{++}$; and (iii) $i > j$ for all $(i,j) \in \mathcal{J}$ since $\mathcal{J}$ contains no diagonal entries. It is already known from (15) that $X_{ij} = X_{ji} = D_{ij}$ for all $(i,j) \in \mathcal{I}$, and from (16) we immediately see that

$$X_{ij} = -\frac{1}{L_{jj}} \sum_{k>j}^{n} X_{ik} L_{kj}, \quad (i,j) \in \mathcal{J}. \tag{17}$$

Using this last relation recursively in a "bottom-to-top and right-to-left" order (or some other workable order) so as to guarantee that all the $X_{ik} = X_{ki}$ terms in the right-hand side have already been determined before their appearance in the sum, we conclude that the entries $X_{ij} = X_{ji}$ for all $(i,j) \in \mathcal{J}$ are uniquely determined.

The above lemma allows us to introduce an important quantity associated with a given pair $(L^{\mathcal{I}}, y) \in \mathcal{L}^{\mathcal{I}}_{++} \times \Re^m$.

Definition 3. *For any $(L^{\mathcal{I}}, y) \in \mathcal{L}^{\mathcal{I}}_{++} \times \Re^m$, the dual estimate $X = X(L^{\mathcal{I}}, y)$ is the unique matrix $X \in \mathcal{S}^n$ that satisfies the equations (15) and (16) with $D = \nabla_{Z^{\mathcal{I}}} g(Z^{\mathcal{I}}(L^{\mathcal{I}}, y), y)$.*

The term "dual estimate" is motivated by the development in Section 6 where we will see that the matrix X plays the role of a dual variable. The matrix X also plays an important role in expressing the derivatives of the function f.

Theorem 2. *Let $(L^{\mathcal{I}}, y) \in \mathcal{L}^{\mathcal{I}}_{++} \times \Re^m$, $L \equiv L^{\mathcal{I}} + L^{\mathcal{J}}(L^{\mathcal{I}}, y)$, $Z^{\mathcal{I}} \equiv Z^{\mathcal{I}}(L^{\mathcal{I}}, y)$ and $X \equiv X(L^{\mathcal{I}}, y)$. Then*

$$\nabla_{L^{\mathcal{I}}} f(L^{\mathcal{I}}, y) = 2\, \pi^l_{\mathcal{I}}(XL), \tag{18}$$

$$\nabla_y f(L^{\mathcal{I}}, y) = \nabla_y g(Z^{\mathcal{I}}, y) - (H'_y)^*(X), \tag{19}$$

where, given $y \in \Re^m$, $(H'_y)^* : \mathcal{S}^n \to \Re^m$ is the adjoint of the derivative operator $H'_y \equiv H'(y) : \Re^m \to \mathcal{S}^n$ of the map H at y, i.e., $(H'_y)^*$ is the linear operator defined for every $M \in \mathcal{S}^n$ by

$$\left[(H'_y)^*(M)\right]_k = \frac{\partial H}{\partial y_k}(y) \bullet M, \quad k = 1, 2, \ldots, m. \tag{20}$$

Proof. To prove (18), it is sufficient to show that $(\partial f / \partial L^{\mathcal{I}}_{ij})(L^{\mathcal{I}}, y) = 2(XL)_{ij}$ for all $(i,j) \in \mathcal{I}$. Differentiating (12) with respect to $L^{\mathcal{I}}_{ij}$ for a fixed index pair $(i,j) \in \mathcal{I}$, we obtain

$$\frac{\partial f}{\partial L^{\mathcal{I}}_{ij}}(L^{\mathcal{I}}, y) = \nabla_{Z^{\mathcal{I}}} g(Z^{\mathcal{I}}, y) \bullet \frac{\partial Z^{\mathcal{I}}}{\partial L^{\mathcal{I}}_{ij}}(L^{\mathcal{I}}, y) = X \bullet \frac{\partial Z^{\mathcal{I}}}{\partial L^{\mathcal{I}}_{ij}}(L^{\mathcal{I}}, y), \tag{21}$$

where $Z^{\mathcal{I}} \equiv Z^{\mathcal{I}}(L^{\mathcal{I}}, y)$. Here, the last equality follows from the fact that: (1) X satisfies (15) with $D = \nabla_{Z^{\mathcal{I}}} g(Z^{\mathcal{I}}, y)$, implying that $X - \nabla_{Z^{\mathcal{I}}} g(Z^{\mathcal{I}}, y) \in \mathcal{S}^{\mathcal{J}}$; (2) $(\partial Z^{\mathcal{I}} / \partial L^{\mathcal{I}}_{ij})(L^{\mathcal{I}}, y) \in \mathcal{S}^{\mathcal{I}}$; and (3) $\mathcal{S}^{\mathcal{I}}$

and $\mathcal{S}^{\mathcal{J}}$ are orthogonal subspaces in $\mathcal{S}^n$ with respect to the Frobenius inner product (denoted by the symbol $\bullet$). Differentiating (11) with respect to $L^{\mathcal{I}}_{ij}$, we obtain

$$\frac{\partial Z^{\mathcal{I}}}{\partial L^{\mathcal{I}}_{ij}}(L^{\mathcal{I}}, y) = L\left(\frac{\partial L^{\mathcal{J}}}{\partial L^{\mathcal{I}}_{ij}}(L^{\mathcal{I}}, y) + E^{ij}\right)^T + \left(\frac{\partial L^{\mathcal{J}}}{\partial L^{\mathcal{I}}_{ij}}(L^{\mathcal{I}}, y) + E^{ij}\right)L^T.$$

Taking inner product of both sides of this equation with X and using the fact that X is symmetric, we obtain

$$X \bullet \frac{\partial Z^{\mathcal{I}}}{\partial L^{\mathcal{I}}_{ij}}(L^{\mathcal{I}}, y) = 2(XL) \bullet \left(\frac{\partial L^{\mathcal{J}}}{\partial L^{\mathcal{I}}_{ij}}(L^{\mathcal{I}}, y) + E^{ij}\right) = 2(XL) \bullet E^{ij} = 2(XL)_{ij}, \tag{22}$$

where the second equality follows from (16) and the fact that $(\partial L^{\mathcal{J}}/\partial L^{\mathcal{I}}_{ij})(L^{\mathcal{I}}, y) \in \mathcal{L}^{\mathcal{J}}$. Combining (21) and (22), we conclude that (18) holds.

We will now briefly outline the proof of (19), which is very similar to the one above. Differentiating (12) with respect to y_k for a fixed $k \in \{1, \ldots, m\}$, we obtain

$$\begin{aligned}\frac{\partial f}{\partial y_k}(L^{\mathcal{I}}, y) &= \nabla_{Z^{\mathcal{I}}} g(Z^{\mathcal{I}}, y) \bullet \frac{\partial Z^{\mathcal{I}}}{\partial y_k}(L^{\mathcal{I}}, y) + \frac{\partial g}{\partial y_k}(Z^{\mathcal{I}}, y) \\ &= X \bullet \frac{\partial Z^{\mathcal{I}}}{\partial y_k}(L^{\mathcal{I}}, y) + \frac{\partial g}{\partial y_k}(Z^{\mathcal{I}}, y),\end{aligned} \tag{23}$$

where the second equality follows by arguments similar to the ones above. Differentiating (11) with respect to y_k, we obtain

$$\frac{\partial Z^{\mathcal{I}}}{\partial y_k}(L^{\mathcal{I}}, y) = L\left(\frac{\partial L^{\mathcal{J}}}{\partial y_k}(L^{\mathcal{I}}, y)\right)^T + \left(\frac{\partial L^{\mathcal{J}}}{\partial y_k}(L^{\mathcal{I}}, y)\right)L^T - \frac{\partial H}{\partial y_k}(y).$$

Taking inner product of both sides of this equation with X and using arguments similar to the ones above, we conclude that

$$X \bullet \frac{\partial Z^{\mathcal{I}}}{\partial y_k}(L^{\mathcal{I}}, y) = -\frac{\partial H}{\partial y_k}(y) \bullet X.$$

Relation (19) now follows from (23) and the last identity.

5. Complexity of the Function and Derivative Evaluations

In this section, we calculate the computational time and space complexities of evaluating f at any $(L^{\mathcal{I}}, y) \in \mathcal{L}^{\mathcal{I}}_{++} \times \Re^m$ in both the dense and sparse cases. We also obtain, under the mild assumption that certain important matrices are readily available, an expression of the complexities of computing the gradient of f at any $(L^{\mathcal{I}}, y)$ in the dense and sparse cases.

We adopt the same convention as in Golub and Van Loan [8] for counting flops, that is, a flop is a floating point operation (e.g., the inner product of two n-vectors involves $2n$ flops).

5.1. The dense case

The efficient computation of $f(L^{\mathcal{I}}, y)$ for any $(L^{\mathcal{I}}, y) \in \mathcal{L}^{\mathcal{I}}_{++} \times \Re^m$ can be arranged in the following steps:

(1) compute $H(y)$;
(2) use $H(y)$ to compute $L^{\mathcal{J}} = L^{\mathcal{J}}(L^{\mathcal{I}}, y)$;
(3) use $H(y)$ and $L^{\mathcal{J}}$ to compute $Z^{\mathcal{I}} = Z^{\mathcal{I}}(L^{\mathcal{I}}, y)$;
(4) compute $g(Z^{\mathcal{I}}, y)$, which equals $f(L^{\mathcal{I}}, y)$.

We first consider the number of flops required to complete steps (1) through (4), and for this, we introduce two technical definitions. For any function $\xi : V \to W$ between two sets V and W, and for each $v \in V$, we denote by $\mathrm{fps}(\xi(v))$ the number of flops required to compute $\xi(v) \in W$. Moreover, we define $\mathrm{fps}(\xi) \equiv \sup\{\mathrm{fps}(\xi(v)) : v \in V\}$. We assume that both $\mathrm{fps}(g)$ and $\mathrm{fps}(H)$ are finite.

Proposition 3. *For each $(L^{\mathcal{I}}, y) \in \mathcal{L}^{\mathcal{I}}_{++} \times \Re^m$, the computation of $f(L^{\mathcal{I}}, y)$ takes at most $\mathcal{O}((|\mathcal{I}| + |\mathcal{J}|)n + \mathrm{fps}(g) + \mathrm{fps}(H)) = \mathcal{O}(n^3 + \mathrm{fps}(g) + \mathrm{fps}(H))$ flops.*

Proof. Steps (1) and (4) together take no more than $\mathrm{fps}(H)+\mathrm{fps}(g)$ flops. A straightforward analysis of the proof of Lemma 1 shows that step (2) takes at most $\mathcal{O}(|\mathcal{J}|n)$ flops, and it is easy to see that step (3) takes at most $\mathcal{O}(|\mathcal{I}|n)$ flops. Hence, the computation of $f(L^{\mathcal{I}}, y)$ takes at most $\mathcal{O}((|\mathcal{I}| + |\mathcal{J}|)n+\mathrm{fps}(g)+\mathrm{fps}(H))$ flops, and the equality of the proposition follows from the fact that $|\mathcal{I}|+|\mathcal{J}| = n(n + 1)/2$.

To consider the space requirements of the evaluation of f, let ξ be as above. Then for all $v \in V$, we denote by $\mathrm{spc}(\xi(v))$ the space required to store v and to compute and store $\xi(v) \in W$. We also define $\mathrm{spc}(\xi) \equiv \sup\{\mathrm{spc}(\xi(v)) : v \in V\}$. As with the function fps, we assume that $\mathrm{spc}(g)$ and $\mathrm{spc}(H)$ are finite.

Proposition 4. *For each $(L^{\mathcal{I}}, y) \in \mathcal{L}^{\mathcal{I}}_{++} \times \Re^m$, the computation of $f(L^{\mathcal{I}}, y)$ requires at most $\mathcal{O}(|\mathcal{I}| + |\mathcal{J}| + spc(g) + spc(H)) = \mathcal{O}(n^2 + spc(g) + spc(H))$ space.*

Proof. Steps (1) and (4) together require no more than $\mathrm{spc}(H) + \mathrm{spc}(g)$ space, and assuming that $L^{\mathcal{I}}$ and $H(z)$ are easily accessible in the computer's memory, step (2) takes no more than $|\mathcal{J}|$ units of additional space. Similarly, $|\mathcal{I}|$ space is needed for step (3). Hence, the result of the proposition follows from the equality $|\mathcal{I}| + |\mathcal{J}| = n(n + 1)/2$.

We remark that, in the dense case, $\mathrm{spc}(H)$ is at least $\mathcal{O}(n^2)$. Hence, the additional space requirements incurred due to the transformation from (NLSDP) to (NLP) are not significantly greater than the requirements for evaluating g.

We now turn to the computation of the gradient of f at any $(L^{\mathcal{I}}, y) \in \mathcal{L}^{\mathcal{I}}_{++} \times \Re^m$, and for this, we assume that $L \equiv L^{\mathcal{I}} + L^{\mathcal{J}}(L^{\mathcal{I}}, y)$ and $Z^{\mathcal{I}} \equiv Z^{\mathcal{I}}(L^{\mathcal{I}}, y)$ have already been computed and stored. To simplify notation, throughout this section we let $\nabla g = \nabla g(Z^{\mathcal{I}}, y)$, $\nabla_y g = \nabla_y g(Z^{\mathcal{I}}, y)$, $\nabla_{Z^{\mathcal{I}}} g = \nabla_{Z^{\mathcal{I}}} g(Z^{\mathcal{I}}, y)$. Using Theorem 2, we can arrange the computation of the gradient of f at $(L^{\mathcal{I}}, y) \in \mathcal{L}^{\mathcal{I}}_{++} \times \Re^m$ as follows:

(1) use $Z^{\mathcal{I}}$ and y to compute $\nabla_{Z^{\mathcal{I}}}\, g(Z^{\mathcal{I}}, y)$, $\nabla_y\, g(Z^{\mathcal{I}}, y)$ and $(\partial H/\partial y_k)(y)$ for $k = 1, \ldots, m$;
(2) let $\pi_{\mathcal{I}}(X) = \nabla_{Z^{\mathcal{I}}}\, g(Z^{\mathcal{I}}, y)$, and then use L to compute $\pi_{\mathcal{J}}(X)$ via (17);
(3) use X and L to compute $\nabla_{L^{\mathcal{I}}}\, f(L^{\mathcal{I}}, y) = 2\pi^l_{\mathcal{I}}(XL)$;
(4) use $\nabla_y\, g(Z^{\mathcal{I}}, y)$, $(H'_y)^*$, and X to compute $\nabla_y\, f(L^{\mathcal{I}}, y) = \nabla_y\, g(Z^{\mathcal{I}}, y) - (H'_y)^*(X)$,

where the operation $(H'_y)^*(X)$ is defined in (20).

Define

$$\mathrm{fps}\left((H')^*\right) \equiv \sum_{k=1}^{m} \mathrm{fps}\left(\frac{\partial H}{\partial y_k}\right)$$

and also define $\mathrm{spc}((H')^*)$ similarly; we again assume that both quantities are finite. In addition, define

$$\mathrm{nz}((H')^*) = \sup\left\{\mathrm{nz}((H'_y)^*) : y \in \Re^m\right\}$$

where

$$\mathrm{nz}((H'_y)^*) \equiv \max\left\{\sum_{k=1}^{m} \mathrm{nz}\left(\frac{\partial H(y)}{\partial y_k}\right) : y \in \Re^m\right\},$$

and, for each $M \in \mathcal{S}^n$, $\mathrm{nz}(M)$ is the number of nonzero entries in the lower triangular part of M. Note that, although the definition of $\mathrm{nz}(\cdot)$ can detect sparsity, $\mathrm{nz}((H'_y)^*) = \mathcal{O}(mn^2)$ for any $y \in \Re^m$ in the dense case. We have the following proposition.

Proposition 5. *Let $(L^{\mathcal{I}}, y) \in \mathcal{L}^{\mathcal{I}}_{++} \times \Re^m$ be given, and suppose that $L^{\mathcal{I}} + L^{\mathcal{J}}(L^{\mathcal{I}}, y)$ and $Z^{\mathcal{I}}(L^{\mathcal{I}}, y)$ have already been computed and stored. Then, the number of flops required for the computation of $\nabla f(L^{\mathcal{I}}, y)$ is at most*

$$\begin{aligned}&\mathcal{O}\left((|\mathcal{I}| + |\mathcal{J}|)n + nz((H')^*) + \mathrm{fps}(\nabla g) + \mathrm{fps}((H')^*)\right)\\ &= \mathcal{O}(n^3 + mn^2 + \mathrm{fps}(\nabla g) + \mathrm{fps}((H')^*)),\end{aligned}$$

and the amount of space required is at most

$$\mathcal{O}\left(|\mathcal{I}| + |\mathcal{J}| + spc(\nabla g) + spc((H')^*)\right) = \mathcal{O}(n^2 + spc(\nabla g) + spc((H')^*))$$

Proof. We first address the number of flops required to compute the gradient. Step (1) clearly takes no more than $\mathrm{fps}(\nabla g) + \mathrm{fps}((H')^*)$ flops, and it is easy to see using the recursive formula for X described in (17) that step (2) takes at most $\mathcal{O}(|\mathcal{J}|n)$ flops. Step (3) requires $\mathcal{O}(|\mathcal{I}|n)$ flops, and step (4) takes at most $\mathrm{nz}((H')^*)$ flops, as can be seen from Theorem 2. Noting that $|\mathcal{I}| + |\mathcal{J}| = n(n+1)/2$ and that, in the dense case, $\mathrm{nz}((H')^*) = \mathcal{O}(mn^2)$, we obtain the first result of the proposition.

We now establish the space requirements of the gradient computation. Step (1) requires $\mathrm{spc}(\nabla g) + \mathrm{spc}((H')^*)$ space, and steps (2) and (3) need $|\mathcal{I}| + |\mathcal{J}|$ units of additional space. Finally, step (4) takes no additional space, since the computation of $\nabla_y f(L^{\mathcal{I}}, y)$ simply involves m matrix dot products involving matrices already computed and stored and since the contents of $\nabla_y g(Z^{\mathcal{I}}, y)$ can clearly be overwritten with $\nabla_y f(L^{\mathcal{I}}, y)$. Noting once again that $|\mathcal{I}| + |\mathcal{J}| = n(n+1)/2$, we have the second result of the proposition.

5.2. The sparse case

We first note that the sparsity characteristics of the feasible set $\mathcal{F}_{sdp}$ of (NLSDP) are directly derived from the sparsity characteristics of $H(\Re^m) \subseteq \mathcal{S}^n$, that is, the image of $\Re^m$ under the map H. These sparsity characteristics, which we will loosely refer to as the sparsity of H, have obvious consequences for the values fps(H) and fps$((H')^*)$ introduced in the previous subsection, but the sparsity of H also plays a significant role in determining the sparsity characteristics of $L^{\mathcal{J}}(L^{\mathcal{I}}, y)$. In what follows, we relate the sparsity of H to the time and space complexities of the function and gradient evaluations.

Via the Cholesky factorization , it is well-known that, in general, the greater the sparsity of $H(y)$ is, the greater the sparsity of $L^{\mathcal{J}}(L^{\mathcal{I}}, y)$ will be. For each $(L^{\mathcal{I}}, y) \in \mathcal{L}^{\mathcal{I}}_{++} \times \Re^m$, define $\mathcal{Z}(L^{\mathcal{I}}, y) \subseteq \mathcal{J}$ to be the index subset of $\mathcal{J}$ of zero entries of $L^{\mathcal{J}}(L^{\mathcal{I}}, y)$, i.e.,

$$\mathcal{Z}(L^{\mathcal{I}}, y) \equiv \{(i,j) \in \mathcal{J} : [L^{\mathcal{J}}(L^{\mathcal{I}}, y)]_{ij} = 0\},$$

and define

$$\begin{aligned} &\mathcal{Z} \equiv \cap\{\mathcal{Z}(L^{\mathcal{I}}, y) : (L^{\mathcal{I}}, y) \in \mathcal{L}^{\mathcal{I}}_{++} \times \Re^m\}, \\ &\widehat{\mathcal{J}} \equiv \mathcal{J} \setminus \mathcal{Z}. \end{aligned}$$

The following lemma is an immediate consequence of Lemma 3.

Lemma 4. *Let $(L^{\mathcal{I}}, y) \in \mathcal{L}^{\mathcal{I}}_{++} \times \Re^m$ be given and define $L \equiv L^{\mathcal{I}} + L^{\mathcal{J}}(L^{\mathcal{I}}, y)$ and $Z^{\mathcal{I}} \equiv Z^{\mathcal{I}}(L^{\mathcal{I}}, y)$. Then, there exists a unique matrix $\widehat{X} \equiv \widehat{X}(L^I, y)$ in $\mathcal{S}^{\mathcal{I} \cup \widehat{\mathcal{J}}}$ satisfying*

$$\begin{aligned} \pi_{\mathcal{I}}(\widehat{X}) &= \nabla_{Z^{\mathcal{I}}}\, g(Z^{\mathcal{I}}, y), \\ \pi^l_{\widehat{\mathcal{J}}}(\widehat{X} L) &= 0. \end{aligned}$$

Proof. By Lemma 3, there exists a unique matrix $\widehat{X}$ satisfying $\pi_{\mathcal{I} \cup \mathcal{Z}}(\widehat{X}) = \nabla_{Z^{\mathcal{I}}}\, g(Z^{\mathcal{I}}, y)$ and $\pi^l_{\widehat{\mathcal{J}}}(\widehat{X} L) = 0$. Clearly, this matrix also satisfies the conditions of the lemma.

We refer to the matrix $\widehat{X}(L^I, y)$ introduced in the above lemma as the sparse dual estimate at (L^I, y). The following sparse analogue of Theorem 2 gives a representation of the gradient of f in terms of the sparse dual estimate.

Theorem 3. *Let $(L^{\mathcal{I}}, y) \in \mathcal{L}^{\mathcal{I}}_{++} \times \Re^m$ be given and define $L \equiv L^{\mathcal{I}} + L^{\mathcal{J}}(L^{\mathcal{I}}, y)$, $Z^{\mathcal{I}} \equiv Z^{\mathcal{I}}(L^{\mathcal{I}}, y)$ and $\widehat{X} \equiv \widehat{X}(L^{\mathcal{I}}, y)$. Then*

$$\begin{aligned} \nabla_{L^{\mathcal{I}}} f(L^{\mathcal{I}}, y) &= 2\, \pi^l_{\mathcal{I}}(\widehat{X} L), \\ \nabla_y f(L^{\mathcal{I}}, y) &= \nabla_y\, g(Z^{\mathcal{I}}, y) - (H'_y)^*(\widehat{X}). \end{aligned}$$

Proof. Using the fact that $(\partial L^{\mathcal{J}} / \partial L^{\mathcal{I}}_{ij})(L^{\mathcal{I}}, y) \in \mathcal{L}^{\widehat{\mathcal{J}}}$ for all $(i,j) \in \mathcal{I}$ and $(\partial L^{\mathcal{J}} / \partial y_k)(L^{\mathcal{I}}, y) \in \mathcal{L}^{\widehat{\mathcal{J}}}$ for all $k = 1, \ldots, m$, it is easy to verify that the arguments used in the proof of Theorem 2 hold if we replace X by $\widehat{X}$.

Two comments regarding the relationship between $X(L^I, y)$ and $\widehat{X}(L^{\mathcal{I}}, y)$ are in order. First of all, these two functions are not equal. The defining equations for $X(L^I, y)$ ensure that $X(L^I, y)$ is, in general, a dense symmetric matrix, while the equations in Lemma 4 guarantee that $\widehat{X}(L^{\mathcal{I}}, y)$ has the same sparsity pattern as $L \equiv L^{\mathcal{I}} + L^{\mathcal{J}}(L^{\mathcal{I}}, y)$. Secondly, it is not difficult to see that, like $X(L^{\mathcal{I}}, y)$, $\widehat{X}(L^{\mathcal{I}}, y)$ can be computed by a recursive formula similar to (17), but for $\widehat{X}$, the computation can be done in $\mathcal{O}((|\mathcal{I}| + |\widehat{\mathcal{J}}|)n)$ flops and in $\mathcal{O}(|\mathcal{I}| + |\widehat{\mathcal{J}}|)$ space.

We are now prepared to state the time and space complexities for the function and gradient evaluations in the sparse case.

Proposition 6. *For each $(L^{\mathcal{I}}, y) \in \mathcal{L}^{\mathcal{I}}_{++} \times \Re^m$, assume that $L^{\mathcal{I}} + L^{\mathcal{J}}(L^{\mathcal{I}}, y)$ and $Z^{\mathcal{I}}(L^{\mathcal{I}}, y)$ have already been computed. Then the numbers of flops required for the computation of $f(L^{\mathcal{I}}, y)$ and $\nabla f(L^{\mathcal{I}}, y)$ are, respectively, at most*

$$\mathcal{O}((|\mathcal{I}| + |\widehat{\mathcal{J}}|)n + \mathrm{fps}(g) + \mathrm{fps}(H)), \quad \textit{and}$$
$$\mathcal{O}((|\mathcal{I}| + |\widehat{\mathcal{J}}|)n + nz((H')^*) + \mathrm{fps}(\nabla g) + \mathrm{fps}((H')^*)).$$

In addition, the amount of space required for these operations is, respectively, at most

$$\mathcal{O}(|\mathcal{I}| + |\widehat{\mathcal{J}}| + spc(g) + spc(H)), \quad \textit{and}$$
$$\mathcal{O}(|\mathcal{I}| + |\widehat{\mathcal{J}}| + spc(\nabla g) + spc((H')^*)).$$

Proof. In the previous subsection, we showed that the evaluations of the function and gradient took $\mathcal{O}((|\mathcal{I}| + |\mathcal{J}|)n + \mathrm{fps}(g) + \mathrm{fps}(H))$ and $\mathcal{O}((|\mathcal{I}| + |\mathcal{J}|)n + \mathrm{nz}((H')^*) + \mathrm{fps}(\nabla g) + \mathrm{fps}((H')^*)$ flops, respectively, and that they required $\mathcal{O}(|\mathcal{I}| + |\mathcal{J}| + \mathrm{spc}(g) + \mathrm{spc}(H))$ and $\mathcal{O}(|\mathcal{I}| + |\mathcal{J}| + \mathrm{spc}(\nabla g) + \mathrm{spc}((H')^*))$, respectively. In this subsection, the same arguments are valid, except that $|\mathcal{J}|$ is replaced by $|\widehat{\mathcal{J}}|$. Hence, the proposition follows.

We remark that, in many special cases (e.g., the case of the linear SDP), the set $\mathcal{Z}$ defined above can be determined in polynomial time using the well-known symbolic factorization procedure. In addition, a reordering procedure can be performed in order to reformulate (NLSDP) so that the resultant size of $\mathcal{Z}$ is usually larger than the original size of $\mathcal{Z}$.

5.3. Complexity for the linear SDP

In the case of the linear SDP (Example 1), where $g(Z^{\mathcal{I}}, y) = B^{\mathcal{I}} \bullet Z^{\mathcal{I}} + b^T y$ and $H(y) = \sum_{k=1}^m y_k A_k - C$, we clearly have $\mathrm{fps}(\nabla g) = \mathrm{fps}((H')^*) = 0$ due to the linearity of g and H. Moreover, $\mathrm{fps}(g) = \mathcal{O}(|\mathcal{I}| + m)$, $\mathrm{fps}(H) = \mathcal{O}(\sum_{k=1}^m \mathrm{nz}(A_k) + \mathrm{nz}(C))$ and $\mathrm{nz}((H')^*) = \mathcal{O}(\sum_{k=1}^m \mathrm{nz}(A_k))$. Notice the only difference between $\mathrm{fps}(H)$ and $\mathrm{nz}((H')^*)$ is the appearance of the $\mathrm{nz}(C)$ in $\mathrm{fps}(H)$. Using these facts and the inequalities $n \le |\mathcal{I}|$ and $m \le O(n^2)$, Proposition 6 implies that the time complexities for the gradient and function evaluations are, respectively,

$$\mathcal{O}\left(n\,\mathrm{nz}(L) + \sum_{k=1}^m \mathrm{nz}(A_k) + \mathrm{nz}(C)\right),$$
$$\mathcal{O}\left(n\,\mathrm{nz}(L) + \sum_{k=1}^m \mathrm{nz}(A_k)\right),$$

where $\mathrm{nz}(L) = |\mathcal{I}| + |\widehat{\mathcal{J}}|$ denotes the number of nonzeros in the lower Cholesky factor L.

In Table 1, we specialize the complexities above to the three cases: (i) both L and the collection $\{A_1, \ldots, A_m, C\}$ are sparse; more specifically, we assume that $\sum_{k=1}^m \mathrm{nz}(A_k) + \mathrm{nz}(C) \le \mathcal{O}(n^2)$; (ii) L is dense but $\{A_1, \ldots, A_m, C\}$ is relatively sparse whenever $m \gg n$; more specifically, we assume that $\sum_{k=1}^m \mathrm{nz}(A_k) + \mathrm{nz}(C) \le \mathcal{O}(n^3)$; (iii) both L and $\{A_1, \ldots, A_m, C\}$ are dense. (Note that the fourth case never occurs, i.e., if $\{A_1, \ldots, A_m, C\}$ is dense, then L cannot be generally sparse.)

Table 1. Number of Flops Required for f and ∇f Evaluations

Sparse Case	Mixed Case	Dense Case
$\mathcal{O}(n\,\mathrm{nz}(L))$	$\mathcal{O}(n^3)$	$\mathcal{O}((n+m)n^2)$

The space requirements for evaluating f and ∇f can be considered similarly. Noting that $\mathrm{spc}(\nabla g) = \mathrm{spc}((H')^*) = 0$ and that $\mathrm{spc}(g) = \mathcal{O}(|\mathcal{I}| + m)$, Proposition 6 shows that the space complexities for the gradient and function evaluations are, respectively,

$$\mathcal{O}(\mathrm{nz}(L) + \mathrm{spc}(H)),$$
$$\mathcal{O}(\mathrm{nz}(L)).$$

If we make the reasonable assumption that the matrices $\{A_1, \ldots, A_m, C\}$ are stored in memory for use throughout all iterations, then it is easy to see that H can be computed in the space of the union of the nonzeros of the matrices $\{A_1, \ldots, A_m, C\}$, i.e.,

$$\mathrm{spc}(H) = \mathcal{O}\left(\mathrm{nz}\left(\sum_{k=1}^m |A_k| + |C|\right)\right),$$

where the absolute value of a matrix is taken element-wise. It is clear from the “fill-in” of the Cholesky factorization, however, that $\mathrm{nz}(L)$ is an upper-bound on $\mathrm{nz}\left(\sum_{k=1}^m |A_k| + |C|\right)$, and so we can further refine the above space complexity results. The resulting complexities for the sparse and dense cases are given in the Table 2.

Table 2. Space Required for f and ∇f Evaluations

Sparse Case	Dense Case
$\mathcal{O}(\mathrm{nz}(L))$	$\mathcal{O}(n^2)$

6. A First-Order Algorithm for the Linear SDP

Although the transformation proposed in Section 3 is applicable to both linear and nonlinear SDP problems in the form of (2), it is considerably more difficult to construct theoretically sound algorithms for nonlinear problems than linear ones. In this section, we develop a globally convergent, first-order algorithm for the special case of problem (2) in which g is linear, H is affine and $\mathcal{I} = \mathcal{D}$ (the diagonal).

This section is divided into four subsections. In the first subsection, we state the linear SDP problem to be solved by our algorithm, specialize the results developed in Sections 3 and 4 to this linear SDP, and give several properties of the dual estimate matrix X defined in Definition 3. In the second subsection, we state sufficient conditions for a sequence of points in the feasible region of the transformed problem to be bounded and to approach optimality. In the third subsection, we introduce the notion of the central path with respect to the transformed problem and show its equivalence to the central path for the original linear SDP problem. In the fourth subsection, we state a first-order, interior-point, log-barrier algorithm for solving the transformed problem.

6.1. The linear SDP problem and its transformed NLP problem

Consider the following linear SDP, specialized from Example 1:

$$\min \left\{ B^{\mathcal{D}} \bullet Z^{\mathcal{D}} + b^T y : Z^{\mathcal{D}} + \sum_{k=1}^{m} y_k A_k - C \succeq 0;\ (Z^{\mathcal{D}}, y) \in \mathcal{S}^{\mathcal{D}} \times \Re^m \right\},$$

where the data for the problem consist of $C, A_1, \ldots, A_m \in \mathcal{S}^n$, $B^{\mathcal{D}} \in \mathcal{S}^{\mathcal{D}}$, and $b \in \Re^m$. Note that $\mathcal{I} = \mathcal{D}$, the set of diagonal entries, and so the complementary set $\mathcal{J}$ consists of the strictly lower entries.

We wish to simplify the presentation of the above linear SDP, and so we introduce a vector variable $z \in \Re^n$ and express $Z^{\mathcal{D}} = \mathrm{Diag}(z)$, where $\mathrm{Diag} : \Re^n \to \mathcal{S}^D$ is the linear operator that places its argument on the diagonal of a diagonal matrix. Letting $d \in \Re^n$ be the unique vector satisfying $B^{\mathcal{D}} = \mathrm{Diag}(d)$, we also see that $B^{\mathcal{D}} \bullet Z^{\mathcal{D}}$ can be rewritten as $d^T z$. Now letting the linear operator $\mathcal{A}^* : \Re^m \to \mathcal{S}^n$ be defined by $\mathcal{A}^*(y) = \sum_{k=1}^m y_k A_k$ and introducing an auxiliary variable $S \in \mathcal{S}^n$, the above SDP can be rewritten as

$$(P) \qquad \min \left\{ d^T z + b^T y : \mathrm{Diag}(z) + \mathcal{A}^*(y) - C = S; S \succeq 0;\ (z, y) \in \Re^n \times \Re^m \right\}.$$

The dual SDP associated with (P) is the problem

$$(D) \qquad \max \left\{ C \bullet X : \mathrm{diag}(X) = d;\ \mathcal{A}(X) = b;\ X \succeq 0 \right\},$$

where $X \in \mathcal{S}^n$ is the optimization variable, $\mathrm{diag} : \mathcal{S}^{\mathcal{D}} \to \Re^n$ is the adjoint of Diag that extracts the diagonal of a diagonal matrix, and $\mathcal{A} : \mathcal{S}^n \to \Re^m$ is the adjoint of $\mathcal{A}^*$ whose k-th component function evaluates $A_k \bullet X$ for all $X \in \mathcal{S}^n$.

We remark that: (i) the above class of SDP problems (P) and (D) contains the SDP relaxations of many binary combinatorial optimization problems (see Section 2); and (ii) the above designation of primal and dual problems is opposite to the usual one where (D) is usually said to be the primal while (P) is said to be the dual.

We denote by $\mathcal{F}^0(P)$ and $\mathcal{F}^0(D)$ the sets of interior feasible solutions for problems (P) and (D), respectively, i.e.,

$$\mathcal{F}^0(P) \equiv \{(z, y, S) \in \Re^n \times \Re^m \times \mathcal{S}^n_{++} : \mathrm{Diag}(z) + \mathcal{A}^*(y) - C = S\},$$
$$\mathcal{F}^0(D) \equiv \{X \in \mathcal{S}^n_{++} : \mathrm{diag}(X) = d; \mathcal{A}(X) = b\}.$$

Note that $\mathcal{F}^0(P) \neq \emptyset$. We also note that the term $\mathrm{Diag}(z)$ found in the statement of problem (P) can be rewritten as $\sum_{i=1}^n z_i(e_i e_i^T)$, where $e_i \in \Re^n$ is the i-th column of the identity matrix of size n. We will make the following assumptions throughout the rest of the paper.

Assumption 1: $\mathcal{F}^0(D) \neq \emptyset$, i.e., (D) has a feasible solution that is positive definite.

Assumption 2: The matrices $\{e_i e_i^T\}_{i=1}^n \cup \{A_k\}_{k=1}^m$ are linearly independent.

We note that Assumption 1 implies in particular that $d \in \Re^n_{++}$.

In the remainder of the paper, we will replace the variable $L^{\mathcal{D}}$ with a vector variable $w \in \Re^n$ that satisfies $L^{\mathcal{D}} = \mathrm{Diag}(w)$. This will help to simplify notation, and it will be straightforward to translate ideas that are expressed in terms of $L^{\mathcal{D}}$ into ideas in terms of w. For example, the condition that $L^{\mathcal{D}} \in \mathcal{L}^{\mathcal{D}}_{++}$ becomes $w \in \Re^n_{++}$ when written in terms of w.

The following theorem specializes the results of Section 3 to the context of the above pair of primal-dual problems.

Theorem 4. *The following statements hold:*

(a) for each $(w, y) \in \Re^n_{++} \times \Re^m$, there exists a unique $(L^{\mathcal{J}}, z) \in \mathcal{L}^{\mathcal{J}} \times \Re^n$ such that

$$Diag(z) + \mathcal{A}^*(y) - C = \left(Diag(w) + L^{\mathcal{J}}\right)\left(Diag(w) + L^{\mathcal{J}}\right)^T; \tag{24}$$

(b) the functions $L^{\mathcal{J}}(w, y)$ and $z(w, y)$ defined according to (24) are each infinitely differentiable on their domain $\Re^n_{++} \times \Re^m$;

(c) the sets $\Re^n_{++} \times \Re^m$ and $\mathcal{F}^0(P)$ are in bijective correspondence according to the assignment $(w, y) \mapsto (z, y, S)$ where $z = z(w, y)$ and $S = LL^T$ for $L \equiv Diag(w) + L^{\mathcal{J}}(w, y)$.

As a consequence of Theorem 4, the problem obtained from (P) by restricting the feasible region to the set $\mathcal{F}^0(P)$ can be recast as the nonlinear program

$$(NLP) \qquad \inf\left\{f(w, y) : (w, y) \in \Re^n_{++} \times \Re^m\right\}.$$

where $f : \Re^n_{++} \times \Re^m \to \Re$ is given by

$$f(w, y) = d^T z(w, y) + b^T y, \quad \forall (w, y) \in \Re^n_{++} \times \Re^m.$$

It follows from Theorem 1 that problems (P) and (NLP) have the same optimal values. However, (NLP) has an open feasible set and in general does not have an optimal solution. In fact, it can be easily seen that if $(b,d) \neq 0$ then all optimal solutions of (P) lie in the boundary of $\mathcal{F}^0(P)$, and in this case (NLP) does not have an optimal solution.

Associated with a point $(w,y) \in \Re^n_{++} \times \Re^m$, define

$$L(w,y) \equiv \mathrm{Diag}(w) + L^{\mathcal{J}}(w,y) \in \mathcal{L}^n, \tag{25}$$

$$S(w,y) \equiv L(w,y)L(w,y)^T \in \mathcal{S}^n_{++}, \tag{26}$$

The following result is an immediate consequence of Lemma 3 and Theorem 2.

Proposition 7. *Let $(w,y) \in \Re^n_{++} \times \Re^m$ be given and define $L \equiv L(w,y)$. Then, the system of linear equations*

$$diag(X) = d, \quad (XL)_{ij} = 0,\ i > j, \quad X \in \mathcal{S}^n \tag{27}$$

has a unique solution in $\mathcal{S}^n$, which we will denote by $X(w,y)$. Moreover, the matrix $X \equiv X(w,y)$ satisfies

(a) $\nabla_w f(w,y) = 2\ diag(XL)$;
(b) $\nabla_y f(w,y) = b - \mathcal{A}(X)$.

In the following lemma, we establish several important properties of the matrix function $X(w,y)$.

Lemma 5. *Let $(w,y) \in \Re^n_{++} \times \Re^m$ be given and define $L \equiv L(w,y)$, $S \equiv S(w,y)$, $X \equiv X(w,y)$, and $\nabla_w f \equiv \nabla_w f(w,y)$. Then,*

(a) XL is upper triangular, or equivalently, L^TXL is diagonal;
(b) $X \succeq 0$ if and only if $\nabla_w f \geq 0$; in addition, $X \succ 0$ if and only if $\nabla_w f > 0$;
(c) $w \circ \nabla_w f(w,y) = 2\ diag(L^TXL)$, hence $w^T\nabla_w f(w,y) = 2\,\mathrm{tr}(L^TXL) = 2\,X \bullet S$, where "$\circ$" denotes the Hadamard (element-wise) product.

Proof. The upper triangularity of XL follows directly from (27). Since L^T and XL are both upper triangular, so is the product L^TXL which is also symmetric. Hence L^TXL must be diagonal. On the other hand, if L^TXL is diagonal, say it equals D, then $XL = L^{-T}D$ is upper triangular. So, a) follows.

To prove the first part of (b), we note that the nonsingularity of L implies that $X \succeq 0$ if and only if $L^TXL \succeq 0$, but since L^TXL is diagonal by (a), $L^TXL \succeq 0$ if and only if $\mathrm{diag}(L^TXL) \geq 0$. Given that both L^T and XL are upper triangular matrices, it is easy to see that $\mathrm{diag}(L^TXL)$ is the Hadamard product of $\mathrm{diag}(L^T)$ and $\mathrm{diag}(XL)$. Since $\mathrm{diag}(L^T) > 0$, it follows that $\mathrm{diag}(L^TXL) \geq 0$ if and only if $\mathrm{diag}(XL) \geq 0$. The first statement of (b) now follows from the sequence of implications just derived and the fact that $\nabla_w f = 2\,\mathrm{diag}(XL)$ by Proposition 7(a).

The second part of (b) can be proved by an argument similar to the one given in the previous paragraph; we need only replace the inequalities by strict inequalities.

Statement (c) follows from (25), Proposition 7(a), and the simple observation that the diagonal of L^TXL is the Hadamard product of $\mathrm{diag}(L^T) = w$ and $\mathrm{diag}(XL)$ since both L^T and XL are upper triangular.

The following theorem establishes that the matrix $X(w,y)$ plays the role of a (possibly infeasible) dual estimate for any $(w,y) \in \Re^n_{++} \times \Re^m$.

Theorem 5. *Let $(w,y) \in \Re^n_{++} \times \Re^m$, and define $L \equiv L(w,y)$, $S \equiv S(w,y)$, $X \equiv X(w,y)$, $\nabla_w f \equiv \nabla_w f(w,y)$, and $\nabla_y f \equiv \nabla_y f(w,y)$. Then:*

(a) X is feasible for (D) if and only if $\nabla_w f \geq 0$ and $\nabla_y f = 0$;
(b) X is strictly feasible for (D) if and only if $\nabla_w f > 0$ and $\nabla_y f = 0$;
(c) $X \bullet S \leq \epsilon$ if and only if $w^T \nabla_w f \leq 2\,\epsilon$.

Proof. By the definition of X, we have $\mathrm{diag}(X) = d$ and $X \in \mathcal{S}^n$. The theorem is an easy consequence of Proposition 7(b) and Lemma 5(b)-(c).

For each $(w,y) \in \Re^n_{++} \times \Re^m$, Theorem 5 gives necessary and sufficient conditions for $X(w,y)$ to be a feasible or strictly feasible solution for (D). These conditions are based entirely on the gradient $\nabla f(w,y)$. Moreover, whenever $\nabla_w f(w,y) \geq 0$ at a given point $(w,y) \in \Re^n_{++} \times \Re^m$, the closeness to optimality of the point (w,y) can be easily measured by the magnitudes of $\nabla_y f(w,y)$ and $w^T \nabla_w f(w,y)$. Since $S(w,y)$ is positive definite and since $X(w,y)$ is positive semidefinite and nonzero whenever $\nabla f_w(w,y) \geq 0$, the duality gap $X(w,y) \bullet S(w,y)$ is positive whenever $\nabla f_w(w,y) \geq 0$, though it can converge to zero as (w,y) approaches the boundary of $\Re^n_{++} \times \Re^m$.

6.2. Boundedness and Optimality Results

Recall that our transformation is not defined on the boundary of $\Re^n_{++} \times \Re^m$ because it requires division by the components of w in order to evaluate the function $L^{\mathcal{J}}(w,y)$. Since some components of w will necessarily go to zero as optimality is approached, it is natural to ask whether (and where) the function $L^{\mathcal{J}}(w,y)$ and other related functions $L(w,y)$, $z(w,y)$, $S(w,y)$ and $X(w,y)$ will remain bounded as (w,y) approaches the boundary of $\Re^n_{++} \times \Re^m$. In this section, we provide some results on this boundedness issue. We also give sufficient conditions for the accumulation points of sequences $\{X(w^k,y^k)\}$ and $\{(z(w^k,y^k), y^k, S(w^k,y^k))\}$ to be optimal solutions of problems (P) and (D), respectively.

The following technical result is a slight variant of Lemma 2 of Monteiro and Pang [14].

Proposition 8. *Let a linear map $\tilde{\mathbf{A}} : \mathcal{S}^n \to \Re^p$ be given and consider the induced linear map $\mathbf{G} : \mathcal{S}^n \times \mathcal{S}^n \times \Re^p \to \mathcal{S}^n \times \Re^p$ defined by*

$$\mathbf{G}(X,S,y) \equiv \begin{pmatrix} \tilde{\mathbf{A}}^*(y) - S \\ \tilde{\mathbf{A}}(X) \end{pmatrix}, \quad \forall\,(X,S,y) \in \mathcal{S}^n \times \mathcal{S}^n \times \Re^p. \tag{28}$$

Then, for any constant $\Gamma \geq 0$ and any compact set $\mathcal{K} \subset \mathbf{G}(\mathcal{S}^n_{++} \times \mathcal{S}^n_{++} \times \Re^p)$, the set

$$\mathcal{F}(\Gamma,\mathcal{K}) \equiv \{(X,S) \in \mathcal{S}^n_+ \times \mathcal{S}^n_+ : X \bullet S \leq \Gamma,\ \mathbf{G}(X,S,y) \in \mathcal{K} \text{ for some } y \in \Re^p\}$$

is compact.

Proof. Let a constant $\Gamma \geq 0$ and a compact set $\mathcal{K} \subset \mathbf{G}(\mathcal{S}^n_{++} \times \mathcal{S}^n_{++} \times \Re^p)$ be given. Since $\mathcal{F}(\Gamma, \mathcal{K})$ is obviously closed, it suffices to show that this set is bounded. Assume for contradiction that there exists a sequence $\{(X^k, S^k)\} \subset \mathcal{F}(\Gamma, \mathcal{K})$ such that $\lim_{k\to\infty} \|(X^k, S^k)\| = \infty$. Then, $\{X^k \bullet S^k\}$ is bounded and $\{\mathbf{G}(X^k, S^k, y^k)\} \subset \mathcal{K}$ for some sequence $\{y^k\} \subset \Re^p$. Since $\mathcal{K}$ is compact, we may assume without loss of generality that for some $\mathbf{G}^\infty \in \mathcal{K} \subset \mathbf{G}(\mathcal{S}^n_{++} \times \mathcal{S}^n_{++} \times \Re^p)$, we have $\mathbf{G}^\infty = \lim_{k\to\infty} \mathbf{G}(X^k, S^k, y^k)$. Clearly, we have $\mathbf{G}^\infty = \mathbf{G}(X^\infty, S^\infty, y^\infty)$ for some $(X^\infty, S^\infty, y^\infty) \in \mathcal{S}^n_{++} \times \mathcal{S}^n_{++} \times R^p$. Since $X^\infty \succ 0$ and $S^\infty \succ 0$, there exists $\eta > 0$ such that the set

$$\mathcal{N}_\infty \equiv \{ (X, S, y) \in \mathcal{S}^n_{++} \times \mathcal{S}^n_{++} \times \Re^p : \eta^{-1} I \succ X \succ \eta I, \; \eta^{-1} I \succ S \succ \eta I \}$$

contains $(X^\infty, S^\infty, y^\infty)$. Since $\mathcal{N}_\infty$ is an open set and every linear map maps open sets into open sets relative to its image, we conclude that $\mathbf{G}(\mathcal{N}_\infty)$ is open relative to the image of $\mathbf{G}$. Since $\mathbf{G}(X^\infty, S^\infty, y^\infty) \in \mathbf{G}(\mathcal{N}_\infty)$ and $\mathbf{G}^\infty = \lim_{k\to\infty} \mathbf{G}(X^k, S^k, y^k)$, we conclude that for all k sufficiently large, say $k \geq k_0$, we have $\mathbf{G}(X^k, S^k, y^k) \in \mathbf{G}(\mathcal{N}_\infty)$, and hence that $\mathbf{G}(X^k, S^k, y^k) = \mathbf{G}(\tilde{X}^k, \tilde{S}^k, \tilde{y}^k)$ for some $(\tilde{X}^k, \tilde{S}^k, \tilde{y}^k) \in \mathcal{N}_\infty$. Using the definition of $\mathbf{G}$, we immediately see that $(X^k - \tilde{X}^k) \bullet (S^k - \tilde{S}^k) = 0$ for all $k \geq k_0$. This equality together with the fact that $(\tilde{X}^k, \tilde{S}^k, \tilde{y}^k) \in \mathcal{N}_\infty$ implies

$$X^k \bullet S^k + \eta^{-2} n \geq X^k \bullet S^k + \tilde{X}^k \bullet \tilde{S}^k = X^k \bullet \tilde{S}^k + S^k \bullet \tilde{X}^k \geq \eta (X^k \bullet I + S^k \bullet I),$$

for every $k \geq k_0$. Since $\{X^k \bullet S^k\}$ is bounded and $\{(X^k, S^k)\} \subset \mathcal{S}^n_+ \times \mathcal{S}^n_+$, the above inequality implies that $\{X^k\}$ and $\{S^k\}$ are bounded. This contradicts the assumption that $\lim_{k\to\infty} \|(X^k, S^k)\| = \infty$, hence the result follows.

As a consequence of the above proposition, we obtain the following result which guarantees the boundedness of certain sets in the (w, y)-space, as well as some relevant functions defined on these sets.

Lemma 6. *There exists $\beta > 0$ such that for any $\Gamma > 0$ the set $\mathcal{B} \equiv \mathcal{B}(\Gamma, \beta)$ defined by*

$$\mathcal{B}(\Gamma, \beta) \equiv \left\{ (w, y) \in \Re^n_{++} \times \Re^m : \nabla_w f \geq 0, \; w^T \nabla_w f \leq 2\,\Gamma, \; \|\nabla_y f\| \leq \beta \right\},$$

is bounded, and the functions $z(\cdot,\cdot)$, $L(\cdot,\cdot)$, $X(\cdot,\cdot)$, $S(\cdot,\cdot)$ and $\nabla_w f(\cdot,\cdot)$ are all bounded on $\mathcal{B}(\Gamma, \beta)$.

Proof. Let $\tilde{\mathbf{A}} : \mathcal{S}^n \to \Re^{n+m}$ be the linear map defined by $\tilde{\mathbf{A}}(X) = (\mathrm{diag}(X), \mathcal{A}(X))$ for all $X \in \mathcal{S}^n$. Clearly, its adjoint $\tilde{\mathbf{A}}^* : \Re^n \times \Re^m \to \mathcal{S}^n$ is the map defined by $\tilde{\mathbf{A}}^*(z, y) = \mathrm{diag}(z) + \mathcal{A}^*(y)$ for all $(z, y) \in \Re^n \times \Re^m$. Assumption 2 implies that $\tilde{\mathbf{A}}^*$ is an injective map, and hence that $\tilde{\mathbf{A}}$ is an onto map. This implies that the linear map $\mathbf{G}$ defined in (28) is an onto, and hence open, map. Thus, the set $\mathbf{G}(\mathcal{S}^n_{++} \times \mathcal{S}^n_{++} \times \Re^p)$ is open for $p = n + m$. Since by Assumption 1 the point $(C, (d, b)) \in \mathbf{G}(\mathcal{S}^n_{++} \times \mathcal{S}^n_{++} \times \Re^p)$, there exists $\beta > 0$ such that the compact set $\mathcal{K} \equiv \{(C, (d, u)) \in \mathcal{S}^n \times \Re^{n+m} : \|u - b\| \leq \beta\}$ is contained in $\mathbf{G}(\mathcal{S}^n_{++} \times \mathcal{S}^n_{++} \times \Re^p)$. Then, by Proposition 8, the set $\mathcal{F}(\Gamma, \mathcal{K})$ is bounded for every $\Gamma \geq 0$. Now, using (27), Proposition 7(b) and Lemma 5(b)-(c), we easily see that $(w, y) \in \mathcal{B}(\Gamma, \beta)$ if and only if $(X(w, y), S(w, y)) \in \mathcal{F}(\Gamma, \mathcal{K})$. By Proposition 8, this implies that the functions $X(\cdot,\cdot)$ and $S(\cdot,\cdot)$ are bounded on $\mathcal{B}$. By (26), we have $\mathrm{tr}(S(w,y)) = \|L(w,y)\|_F^2$ for all $(w,y) \in \Re^n_{++} \times \Re^m$. Hence, the boundedness of $S(\cdot,\cdot)$ on $\mathcal{B}$ implies that of $L(\cdot,\cdot)$, and in turn that of $w \equiv \mathrm{diag}(L)$ on $\mathcal{B}$. Moreover, since $S(\cdot,\cdot)$ is bounded on $\mathcal{B}$, then so is $\tilde{\mathbf{A}}^*(z,y) = C - S$. By the injectivity of $\tilde{\mathbf{A}}^*$, we conclude that (y,z) is bounded on $\mathcal{B}$. Finally, $\nabla_w f(w,y)$ is bounded on $\mathcal{B}$ because of Proposition 7(a).

Given a sequence of points $\{(w^k,y^k)\}_{k\geq 0} \subset \Re^n_{++} \times \Re^m$, we define $L^k \equiv L(w^k,y^k)$, $X^k \equiv X(w^k,y^k)$, $S^k \equiv S(w^k,y^k)$, $z^k \equiv z(w^k,y^k)$, $\nabla_w f^k \equiv \nabla_w f(w^k,y^k)$ and $\nabla_y f^k \equiv \nabla_y f(w^k,y^k)$ for all $k \geq 0$. The following result gives sufficient conditions for the accumulation points of the sequences $\{X^k\}$ and $\{(z^k,y^k,S^k)\}$ to be optimal solutions of problems (D) and (P), respectively. This result will be used to establish the convergence of the sequence generated by the log-barrier algorithm of Section 6.3.

Theorem 6. *Let $\{(w^k,y^k)\}_{k\geq 0} \subset \Re^n_{++} \times \Re^m$ be a sequence of points such that $\nabla_w f^k \geq 0$ for all $k \geq 0$, the sequence $\{(w^k)^T \nabla_w f^k\}$ is bounded, and $\lim_{k\to\infty} \nabla_y f^k = 0$. Then,*

(a) the sequences $\{X^k\}$, $\{(z^k,y^k,S^k)\}$, $\{L^k\}$, $\{w^k\}$ and $\{\nabla_w f^k\}$ are all bounded;
(b) if in addition $\lim_{k\to\infty}(w^k)^T \nabla_w f^k = 0$, then any accumulation points of $\{X^k\}$ and $\{(z^k,y^k,S^k)\}$ are optimal solutions of (D) and (P), respectively.

Proof. Statement (a) follows immediately from Lemma 6. To prove (b), let X^∞ and $(z^\infty,y^\infty,S^\infty)$ be accumulation points of the sequences $\{X^k\}$ and $\{(z^k,y^k,S^k)\}$, respectively. The assumption of the lemma and Proposition 7(b) imply that

$$\lim_{k\to\infty} \mathcal{A}(X^k) - b = \lim_{k\to\infty} \nabla_y f^k = 0$$

and that $X^k \succeq 0$ and $\mathrm{diag}(X^k) = d$ for all $k \geq 0$. This clearly implies that $X^\infty \succeq 0$, $\mathcal{A}(X^\infty) = b$ and $\mathrm{diag}(X^\infty) = d$, that is, X^∞ is a feasible solution of (P). Since each (z^k,y^k,S^k) is a feasible solution of (P), it follows that $(z^\infty,y^\infty,S^\infty)$ is also a feasible solution of (P). Moreover, we have $(w^k)^T \nabla_w f^k = 2X^k \bullet S^k$, for all $k \geq 0$, from which it follows that $X^\infty \bullet S^\infty = 0$. We have thus shown that X^∞ and $(z^\infty,y^\infty,S^\infty)$ are optimal solutions of (D) and (P).

6.3. The Central Path for the Transformed Problem

It is well known that under a homeomorphic transformation ξ, any path in the domain of ξ is mapped into a path in the range of ξ, and vice versa. Furthermore, given any continuous function f from the range of ξ to $\Re$, the extremers of f in the range of ξ are mapped into corresponding extremers of the composite function $f(\xi(\cdot))$ in the domain of ξ. In particular, if f has a unique minimizer in the range of ξ, then this minimizer is mapped into the unique minimizer of $f(\xi(\cdot))$ in the domain of ξ.

In view of these observations, it is easy to see that under the transformation introduced in Section 3, the central path of a convex SDP problem is transformed into a new "central path" in the "orthants" of the transformed space. Furthermore, since the points on the original central path are the unique minimizers of a defining log-barrier function corresponding to different parameter values, the points on the transformed central path are therefore unique minimizers of the transformed log-barrier function corresponding to different parameter values. In general, however, it is possible that extraneous, non-extreme stationary points be introduced to the transformed log-barrier function by the nonlinear transformations applied. In this section, we show that at least for the linear SDP problem studied in this section, the transformed log-barrier functions do not have any such non-extreme stationary point.

Since the variable w of (NLP) is constrained to be positive, a natural problem to consider is the following log-barrier problem associated with (NLP), which depends on the choice of a constant $\nu > 0$:

$$(NLP_\nu) \qquad \min\left\{ f(w,y) - 2\nu \sum_{i=1}^{n} \log w_i : (w,y) \in \Re^n_{++} \times \Re^m \right\}.$$

(The reason for the factor 2 will become apparent shortly.) Not surprisingly, (NLP_ν) is nothing but the transformed form of the standard log-barrier problem for (P),

$$(P_\nu) \quad \min\left\{ d^T z + b^T y - \nu \log(\det S) : \mathrm{Diag}(z) + \mathcal{A}^*(y) - C = S;\ S \succ 0 \right\},$$

under the bijective transformation given in Theorem 4 between $\Re^n_{++} \times \Re^m$ and $\mathcal{F}^0(D)$. Indeed, (P_ν) is transformed into the nonlinear program

$$\min\left\{ f(w,y) - \nu \log(\det(S(w,y))) : (w,y) \in \Re^n_{++} \times \Re^m \right\},$$

which is exactly (NLP_ν) after the simplification

$$\log(\det S) = \log(\det(LL^T)) = 2\log(\det L) = 2\log\left(\prod_{i=1}^{n} w_i\right) = 2\sum_{i=1}^{n} \log w_i,$$

where $S \equiv S(w,y)$ and $L \equiv L(w,y)$ and the second inequality follows from the fact that the determinant of a triangular matrix is the product of its diagonal entries.

Recall that (P) always has interior feasible solutions. Hence, under Assumptions 1 and 2, (P_ν) and the corresponding standard log-barrier problem for (D),

$$(D_\nu) \qquad \max\left\{ C \bullet X + \nu \log(\det X) : \mathrm{diag}(X) = d;\ \mathcal{A}(X) = b;\ X \succ 0 \right\},$$

each have unique solutions (z_ν, y_ν, S_ν) and X_ν, respectively, such that $X_\nu S_\nu = \nu I$. One can ask whether (NLP_ν) has a unique stationary point, and if so, how this unique stationary point (i.e., global minimum) relates to X_ν and (z_ν, y_ν, S_ν). The following theorem establishes that (NLP_ν) does in fact have a unique stationary point (w_ν, y_ν) which is simply the inverse image of the point (z_ν, y_ν, S_ν) under the bijective correspondence given in Theorem 4(c).

Theorem 7. *For each $\nu > 0$, problem (NLP_ν) has a unique minimum point, which is also its unique stationary point. This minimum (w_ν, y_ν) is equal to the inverse image of the point (z_ν, y_ν, S_ν) under the bijective correspondence of Theorem 4(c). In particular, we have $z(w_\nu, y_\nu) = z_\nu$ and $S(w_\nu, y_\nu) = S_\nu$. Moreover, $X(w_\nu, y_\nu) = X_\nu$.*

Proof. Let (w, y) be a stationary point of (NLP_ν), and define $\nabla f \equiv \nabla f(w, y)$, $L \equiv L(w, y)$, $X \equiv X(w, y)$, $z \equiv z(w, y)$, and $S \equiv S(w, y)$. Since (w, y) is a stationary point of (NLP_ν), it satisfies the first-order optimality conditions of (NLP_ν)

$$\nabla_w f \circ w = 2\,\nu e, \qquad \nabla_y f = 0, \tag{29}$$

where $e \in \Re^n$ is the vector of all ones. Notice that (29), together with Theorem 5 and Proposition 7, implies that X is strictly feasible for (D).

Proposition 7(a) implies that the first equation of (29) is equivalent to $\mathrm{diag}(XL) \circ w = \nu e$, and the equality $w = \mathrm{diag}(L^T)$ implies that $\mathrm{diag}(XL) \circ \mathrm{diag}(L^T) = \nu e$, which in turn implies that $\mathrm{diag}(L^T XL) = \nu e$ since XL is upper triangular by Lemma 5. Since $L^T XL$ is diagonal by the same proposition, it follows that $L^T XL = \nu I$, and hence that $XS = \nu I$. This clearly implies that $X = X_\nu$ and $(z, y, S) = (z_\nu, y_\nu, S_\nu)$. This conclusion shows that (NLP_ν) has a unique stationary point satisfying all the conditions stated in the theorem. That this stationary point is also a global minimum follows from the fact that (z_ν, y_ν, S_ν) is a global minimum of (P_ν).

6.4. A globally convergent log-barrier algorithm

In this subsection, we introduce a straightforward barrier algorithm for solving (NLP). The convergence of the algorithm is a simple consequence of Theorem 6(b).

Let constants $\gamma_1 \in [0, 1)$, $\gamma_2 > 1$, and $\Gamma > 0$ be given, and for each $\nu > 0$, define

$$\mathcal{N}(\nu) \equiv \left\{(w, y) \in \Re^n_{++} \times \Re^m : 2\gamma_1 \nu\, e \leq w \circ \nabla_w f(w, y) \leq 2\gamma_2 \nu\, e, \|\nabla_y f(w, y)\| \leq \Gamma \nu \right\}, \tag{30}$$

where $e \in \Re^n$ is the vector of all ones. Note that each $(w, y) \in \mathcal{N}(\nu)$ satisfies $\nabla_w f(w, y) > 0$ and that the unique minimizer (w_ν, y_ν) of (NLP_ν) is in $\mathcal{N}(\nu)$, as can be seen from equation (29). We propose the following algorithm:

Log-Barrier Algorithm:

Let $\sigma \in (0, 1)$ and $\mu_0 > 0$ be given, and set $k = 0$.

For $k = 0, 1, 2, \ldots$

1. Use a minimization method to solve (NLP_{μ_k}) approximately, obtaining a point $(w^k, y^k) \in \mathcal{N}(\mu_k)$.
2. Set $\mu_{k+1} = \sigma \mu_k$, increment k by 1, and return to step 1.

End

We stress that since (NLP_{μ_k}) has a unique stationary point (w_ν, y_ν) for all $\mu_k > 0$ which is also the unique minimum, step 1 of the algorithm will succeed using any reasonable unconstrained minimization method. Specifically, any convergent, gradient-based method will eventually produce a point in the set $\mathcal{N}(\mu_k)$.

In view of (30), the algorithm clearly produces a sequence of points $\{(w^k, y^k)\}_{k \geq 0}$ that satisfies the hypotheses of Theorem 6(b). Hence, the log-barrier algorithm converges in the sense of Theorem 6.

In practice, a reasonable stopping criterion for the above barrier algorithm would be that for a sufficiently small barrier parameter value the gradient of the corresponding barrier function becomes sufficiently small. When a gradient-based method is applied, however, a very high accuracy should not be expected without an excessively long computation.

7. Final Remarks

Large-scale SDP problems are difficult to solve by conventional interior-point methods due to the necessity of solving large and mostly dense linear systems at every iteration. First-order algorithms based only on gradient information are attractive alternatives to overcome this difficulty. In this paper, we have proposed an approach that leads to first-order algorithms for solving a class of large-scale SDP problems via nonlinear programming techniques.

The proposed transformation is applicable to both linear and nonlinear SDP problems of the form given in (2). In this paper, we proposed a first-order log-barrier algorithm and studied its global convergence for certain linear SDP problems. A similar algorithm can also be easily constructed for nonlinear SDP problems that might be potentially useful, but whose global convergence might be a much more complicated issue.

Although SDP problems of the given form are formulated as nonlinear programs, a generic nonlinear optimization solver is unlikely to be effective in solving very large-scale SDP problems without fully exploiting the special structures of these problems.

We have recently implemented variants of the first-order barrier algorithm for several classes of linear SDP problems and found that the proposed approach is indeed a viable approach to solving some large-scale SDP problems from combinatorial optimization. The implementation details and numerical results are reported in [3]. Moreover, in [4] we have extended the application of the proposed transformation to linear SDP problems of a general form (where the diagonal of the primal matrix variable is not required to be fixed), and presented preliminary numerical results. In addition, we have proposed and analyzed a second-order potential reduction algorithm.

Acknowledgements. We thank two anonymous referees and an associate editor for their valuable comments and suggestions that have helped us to improve the paper.

References

1. S. Benson, Y. Ye, and X. Zhang. Solving large-scale sparse semidefinite programs for combinatorial optimization. SIAM Journal on Optimization, 10:443-461, 2000.
2. S. Burer and R. D. C. Monteiro. A projected gradient algorithm for solving the MAXCUT SDP relaxation. *Optimization Methods and Software*, 15:175-200, 2001.
3. S. Burer, R. D. C. Monteiro and Y. Zhang. A computational study of a gradient-based log-barrier algorithm for a class of large-scale SDPs. Manuscript. June, 2001. Submitted to *Mathematical Programming Series B*.
4. S. Burer, R. D. C. Monteiro, and Y. Zhang. Interior-point algorithms for semidefinite programming based on a nonlinear formulation. To appear in *Computational Optimization and Applications*.
5. C. Delorme and S. Poljak. Laplacian eigenvalues and the maximum cut problem. *Mathematical Programming*, 62:557-574, 1993.

6. K. Fujisawa, M. Kojima, and K. Nakata. Exploiting sparsity in primal-dual interior-point methods for semidefinite programming. *Mathematical Programming*, 79:235–253, 1997.
7. M. X. Goemans and D. P. Williamson. Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. *Journal of ACM*, pages 1115–1145, 42 (1995).
8. G. H. Golub and C. E. Van Loan. *Matrix Computations: Second Edition*. The Johns Hopkins University Press, Baltimore, MD 21211, 1989.
9. C. Helmberg and F. Rendl. A spectral bundle method for semidefinite programming. SIAM Journal on Optimization, 10:673-696, 2000.
10. S. Homer and M. Peinado. Design and performance of parallel and distributed approximation algorithms for maxcut. manuscript, Dept. of Computer Science and Center for Computational Science, Boston University, 111 Cummington Street, Boston, MA, 02215, USA, 1995.
11. M. Laurent, S. Poljak, and F. Rendl. Connections between semidefinite relaxations of the max-cut and stable set problems. *Mathematical Programming*, 77:225–246, 1997.
12. L. Lovász. On the Shannon Capacity of a graph. *IEEE Transactions of Information Theory*, IT-25(1):1–7, January 1979.
13. L. Lovász and A. Schrijver. Cones of matrices and set functions, and 0-1 optimization. *SIAM Journal on Optimization*, 1:166–190, 1991.
14. R. D. C. Monteiro and J. S. Pang. On two interior-point mappings for nonlinear semidefinite complementarity problems. *Mathematics of Operations Research*, 23:39–60, 1998.
15. Y. Nesterov. Semidefinite relaxation and nonconvex quadratic optimization. *Optimization Methods and Software*, 9:141–160, 1998.
16. S. Poljak and F. Rendl. Nonpolyhendral relaxations of graph-bisection problems. *SIAM Journal on Optimization*, 5:467-487, 1995.
17. S. Poljak, F. Rendl, and H. Wolkowicz. A recipe for semidefinite relaxation for 0-1 quadratic programming. *Journal of Global Optimization*, 7:51–73, 1995.
18. S. Poljak and Z. Tuza. Maximum cuts and largest bipartite subgraphs. Combinatorial Optimization. Editors: W. Cook, L. Lovász, and P. Seymour. DIMACS series in Discrete Mathematics and Theoretical Computer Science, Vol.20. American Mathematical Society. 1995.
19. N. Z. Shor. Quadratic optimization problems. *Soviet Journal of Computer and Systems Science*, 25:1–11, 1987. Originally published in Tekhnicheskaya Kibernetika, No. 1, 1987, pp. 128–139.