

Semi-infinite linear programming approaches to semidefinite programming problems[‡]

Kartik Krishnan[‡] John E. Mitchell[§]

December 9, 2002

Abstract

Interior point methods, the traditional methods for the *SDP*, are fairly limited in the size of problems they can handle. This paper deals with an *LP* approach to overcome some of these shortcomings. We begin with a semi-infinite linear programming formulation of the *SDP* and discuss the issue of its discretization in some detail. We further show that a lemma due to Pataki on the geometry of the *SDP*, implies that no more than $O(\sqrt{k})$ (where k is the number of constraints in the *SDP*) linear constraints are required. To generate these constraints we employ the spectral bundle approach due to Helmberg and Rendl. This scheme recasts any *SDP* with a bounded primal feasible set as an eigenvalue optimization problem. These are convex nonsmooth problems that can be tackled by bundle methods for nondifferentiable optimization. Finally we present the rationale for using the columns of the bundle P maintained by the spectral bundle approach, as our linear constraints. We present numerical experiments that demonstrate the efficiency of the *LP* approach on two combinatorial examples, namely the max cut and min bisection problems.

*This work was supported in part by NSF grant numbers CCR-9901822

[†]To appear in *Novel Approaches to Hard Discrete Optimization*, (P.M. Pardalos, and H. Wolkowicz editors), Fields Institute Communication Series, American Mathematical Society, 2003, pp. 121-140.

[‡]Department of Mathematical Sciences, Rensselaer Polytechnic Institute, 110 8th Street, Troy, New York, 12180 (kartik@caam.rice.edu).

[§]Department of Mathematical Sciences, Rensselaer Polytechnic Institute, 110 8th Street, Troy, New York, 12180 (mitchj@rpi.edu).

The *LP* approach potentially allows one to approximately solve large scale semidefinite programs using state of the art linear solvers. Moreover one can incorporate these linear programs in a branch and cut approach for solving large scale integer programs.

Keywords: Semidefinite Programming, Linear Programming, Semi-infinite Programming, Spectral Bundle, Eigenvalue Optimization, Combinatorial Optimization.

AMS subject classifications: Primary, 65F15, 65K05; Secondary, 52A41, 90C05, 90C06, 90C22, 90C27, 90C34, 90C57

1 Introduction

Semidefinite programming (*SDP*) is one of the most exciting and active research areas in optimization recently. This tremendous activity was spurred by the discovery of important applications in combinatorial optimization, control theory, the development of efficient interior point algorithms for solving *SDP* problems, and the depth and elegance of the underlying optimization theory. Excellent survey articles for *SDP* include Vandenberghe and Boyd [43], the *SDP* handbook edited by Wolkowicz et al [45], Helmberg [20] and Todd [42].

Since the seminal work of Alizadeh [1] and Nesterov and Nemirovskii [37], the study of interior point methods has dominated algorithmic research in semidefinite programming. However, for practical applications with many constraints k , the number of arithmetic operations per iteration is often too high. The main computational task here, is the factorization of a dense *Schur complement* matrix M of size k , in computing the search direction. Moreover this matrix is to be recomputed in each iteration, which is the most expensive operation in each iteration. For most problems, the constraint matrices have a special structure, which can be exploited to speed up the computation of this matrix. In particular in combinatorial applications, these constraints often have a rank one structure. Benson, Ye and Zhang [4] have proposed a dual scaling algorithm that exploits this rank one feature, and the sparsity in the dual slack matrix. However even in their approach the matrix M is dense, and the necessity to store and factorize this matrix limits the applicability of

these methods to problems with about 3000 constraints on a well equipped work station.

Consider the semidefinite programming problem

$$\begin{aligned} \min \quad & C \bullet X \\ \text{subject to} \quad & \mathcal{A}(X) = b \quad (SDP) \\ & X \succeq 0, \end{aligned}$$

with dual

$$\begin{aligned} \max \quad & b^T y \\ \text{subject to} \quad & \mathcal{A}^T y + S = C \quad (SDD) \\ & S \succeq 0 \end{aligned}$$

where $X, S \in \mathcal{S}^n$, the space of real symmetric $n \times n$ matrices. We define

$$C \bullet X = \text{Trace}(C^T X) = \sum_{i,j=1}^n C_{ij} X_{ij}$$

where $\mathcal{A} : \mathcal{S}^n \rightarrow \mathbb{R}^k$ and $\mathcal{A}^T : \mathbb{R}^k \rightarrow \mathcal{S}^n$ are of the form

$$\mathcal{A}(X) = \begin{bmatrix} A_1 \bullet X \\ \vdots \\ A_k \bullet X \end{bmatrix} \quad \text{and} \quad \mathcal{A}^T y = \sum_{i=1}^k y_i A_i$$

with $A_i \in \mathcal{S}^n, i = 1, \dots, k$. We assume that A_1, \dots, A_k are linearly independent in \mathcal{S}^n . $C \in \mathcal{S}^n$ is the cost matrix, $b \in \mathbb{R}^k$ the RHS vector. The matrix $X \in \mathcal{S}^n$ is constrained to be positive semidefinite (*psd*) expressed as $X \succeq 0$. This is equivalent to requiring that $d^T X d \geq 0, \forall d$. On the other hand $X \succ 0$ denotes a positive definite (*pd*) matrix, i.e. $d^T X d > 0, \forall d \neq 0$. \mathcal{S}_+^n and \mathcal{S}_{++}^n denote the space of symmetric *psd* and *pd* matrices respectively. Also $\text{diag}(X)$ is a vector whose components are the diagonal elements of X , and $\text{Diag}(d)$ is a diagonal matrix, with the components of d . In the succeeding sections we use $\text{Trace}(X)$ and $\text{tr}(X)$ interchangeably, to denote the trace of the symmetric matrix X . $\lambda_{\min}(M)$ denotes the minimum eigenvalue of the matrix $M \in \mathcal{S}^n$. An excellent reference for these linear algebra preliminaries is Horn and Johnson [26].

Assumption 1 *Both (SDP) and (SDD) have strictly feasible points, namely the sets $\{X \in \mathcal{S}^n : \mathcal{A}(X) = b, X \succ 0\}$ and $\{(y, S) \in \mathbb{R}^k \times \mathcal{S}^n : \mathcal{A}^T y + S = C, S \succ 0\}$ are nonempty.*

This assumption guarantees that both (SDP) and (SDD) attain their optimal solutions X^* and (y^*, S^*) , and their optimal values are equal, i.e. $C \bullet X^* = b^T y^*$. Thus the duality gap $X^* S^* = 0$ at optimality.

Assumption 2

$$\mathcal{A}(X) = b \text{ implies } \text{tr}X = a \tag{1}$$

for some constant $a \geq 0$.

It can be shown that any (SDP) with a bounded feasible set can be rewritten to satisfy Assumption 2. Later we shall see that this enables us to rewrite (SDD) as an eigenvalue optimization problem. A large class of semidefinite programs, in particular several important relaxations of combinatorial optimization problems, can be formulated to satisfy this assumption, such as max cut, Lovasz theta, semidefinite relaxations of box constrained quadratic programs etc.

A number of approaches for large scale SDP 's have been developed recently. A central theme in most of these schemes is a lemma due to Pataki [39] on the rank of the optimal X matrices. The idea here is to exclusively deal with the set of optimal matrices (a subset of the set of feasible matrices), thereby working in a lower dimensional space. This allows us to handle SDP 's that are inaccessible to interior point methods due to their size. We discuss this issue in section 3. Such approaches include the *spectral bundle* approach due to Helmberg and Rendl [21], a nonsmooth optimization technique applicable to eigenvalue optimization problems. We present an overview of this scheme in section 4. Other large scale methods include Burer et al [10, 11, 12], who formulate (SDP) as nonconvex programming problems using low rank factorizations of the primal matrix X , and Vanderbei and Benson [44]. Finally we must mention that Burer et al [13, 14] have come up with attractive heuristics for max cut and maximum stable set problems, where they solve (SDP) with an additional restriction on the rank of the primal matrix X .

The primary objective in this paper is to develop an LP approach to solving (SDP). The aim is to utilize some of the large scale SDP approaches within this LP framework. This potentially allows us to approximately solve large-scale semidefinite programs using state of the art linear solvers. Moreover one could incorporate these linear programs in a branch and cut approach to solving large integer programs. This should overcome some of the

difficulties involved in branch and cut *SDP* approaches, i.e. the cost of solving the *SDP* relaxations and the issue of utilizing the solution of the parent *SDP* problem to construct an appropriate solution for the child problem (restart).

Ben Tal et al [6] have come up with an intriguing polyhedral approximation to the second order cone. This is subsequently improved in Glineur [18], who also discusses various computational results. However we are not aware of similar results for the cone of *SDP* matrices.

This paper is organised as follows. We present a semi-infinite linear programming formulation for the *SDP* in section 2. In particular we discuss which of the two semi-infinite formulations, we wish to use (primal or dual) and the issue of discretization of this semi-infinite formulation in some detail. Section 3 presents some results on the geometry of *SDP*. In particular we present Pataki's lemma on the rank of extreme matrices in (*SDP*) and its consequences on the discretization of semi-infinite *LP*'s and conclude with the appropriate constraints needed in the *LP* relaxations. Section 4 describes the spectral bundle method due to Helmberg and Rendl [21], a fast algorithmic procedure to generate these constraints. Section 5 describes the rationale for using the columns of P as linear constraints and introduces the max cut and the min bisection problems, two combinatorial problems on which we wish to test the *LP* approach. Section 6 describes some computational results and we conclude with some observations and acknowledgements in sections 7 and 8 respectively.

Note that the convex constraint $X \succeq 0$ is equivalent to

$$d^T X d = d d^T \bullet X \geq 0 \quad \forall d \in \mathbb{R}^n \quad (2)$$

These constraints are linear inequalities in the matrix variable X , but there is an *infinite* number of them. Thus SDP is a *semi-infinite* linear programming problem in $\mathbb{R}^{\frac{n(n+1)}{2}}$. The term semi-infinite programming derives from the fact that the LP has finitely many variables, with an infinite number of constraints. The survey paper by Hettich and Kortanek [25] discusses theory, algorithms, and applications of semi-infinite programming.

Since $d^T X d \geq 0$ can be rewritten as $\text{tr}(d d^T X) \geq 0$, the definition of positive semidefiniteness immediately gives the following :

Corollary 1 *The symmetric $n \times n$ matrix S is positive semidefinite if and only if $S \bullet M \geq 0$ for all symmetric rank one matrices M .*

2 A semi-infinite linear programming formulation

The only nonlinearity in (SDP) and (SDD) is the requirement that the matrices X and S need to be positive semidefinite. As we have seen in the previous section these convex constraints are equivalent to an infinite number of linear constraints, giving rise to semi-infinite linear programs. We now consider two semi-infinite linear programs $(PSIP)$ and $(DSIP)$ for (SDP) and (SDD) respectively. These formulations follow directly from Corollary 1.

$$\begin{aligned} \min \quad & C \bullet X \\ \text{subject to} \quad & \mathcal{A}(X) = b & (PSIP) \\ & d^T X d \geq 0 \quad \forall d \in B \end{aligned}$$

$$\begin{aligned} \max \quad & b^T y \\ \text{subject to} \quad & \mathcal{A}^T y + S = C & (DSIP) \\ & d^T S d \geq 0 \quad \forall d \in B \end{aligned}$$

Here B is a compact set, typically $\{d : \|d\|_2 \leq 1\}$ or $\{d : \|d\|_{\text{inf}} \leq 1\}$. These are commonly used in *trust region methods*. (Conn et al [15]). A few remarks are now in order.

1. Since X is $n \times n$ and symmetric, $(PSIP)$ is a semi-infinite linear program in $\binom{n+1}{2} = \frac{n(n+1)}{2} = O(n^2)$ variables.
2. There are k variables in the semi-infinite formulation $(DSIP)$. We have $k \leq \binom{n+1}{2}$ (since the matrices $A_i, i = 1, \dots, k$ are linearly independent).
3. It is more efficient to deal with the dual semi-infinite formulation, since we are dealing with smaller LP 's.

We shall henceforth refer to $(DSIP)$ as (LDD) . The dual (LDP) then has the following form.

$$\begin{aligned} \min \quad & \int_B (d^T C d) dx \\ \text{subject to} \quad & \int_B (d^T A_i d) dx = b_i \quad i = 1, \dots, k & (LDP) \\ & x \geq 0 \end{aligned}$$

We have permitted an infinite number of vectors $z \in \mathbb{R}^k$ whose i th component is given by $d^T A_i d, i = 1, \dots, k$ in the representation of $b \in \mathbb{R}^k$. However

the *reduction* theorem stated below, indicates that at most k vectors z_i are required in the representation of b . For a constructive proof see Glashoff and Gustafson [17].

Theorem 1 *Let the vector $z \in \mathbb{R}^k$ be a nonnegative linear combination of the m vectors $z_1, \dots, z_m \in \mathbb{R}^k$, i.e.*

$$z = \sum_{i=1}^m x_i z_i, \quad x_i \geq 0, \quad i = 1, \dots, m$$

Then z admits a representation

$$z = \sum_{i=1}^m \bar{x}_i z_i, \quad \bar{x}_i \geq 0, \quad i = 1, \dots, m$$

such that at most k of the numbers \bar{x}_i are nonzero and such that the set of vectors $\{z_i | \bar{x}_i > 0\}$, is linearly independent.

We cannot immediately conclude from theorem 1 that we only need to consider feasible solutions $\{x_1, \dots, x_m\}$ with $m \leq k$, and that we can put $m = k$ from the start. It is quite possible that in the transition from (*LDD*) to (*LDR*) the value of the objective function is affected, i.e.

$$\int_B (d^T C d) dx < \sum_{j=1}^k (d_j^T C d_j) x_j$$

Thus we should apply the reduction theorem on $k + 1$ equations with the $(k + 1)$ th constraint being

$$\int_B (d^T C d) dx = b_0$$

We obtain the important result that $m = k + 1$ points d_i are *enough*. However if we know that (*SDP*) has a solution (Assumption 1), then we can put $m = k$ from the outset (Glashoff and Gustafson [17]). Thus an optimal solution to (*LDP*) has a finite support i.e. there are only a finite number of components m of x that are nonzero.

We give conditions ensuring that there exists a discretization of (*LDD*) with the same optimal value.

Theorem 2 *Suppose that the optimal value of the linear semi-infinite programming problem (*LDD*) is finite. If objective values of (*LDD*) and (*LDP*) are the same, i.e. there is no duality gap, and the dual problem (*LDP*) has an optimum solution. Then (*LDD*) has a finite discretization (*LDR*) with the same optimal value.*

Proof: We have shown that an optimal solution to (LDP) has a finite support. Now let (LPR) and (LDR) be the discretizations of (LDP) and (LDD) , respectively, corresponding to an optimal solution of (LDP) with a finite support. Thus we have $\text{val}(LDP) = \text{val}(LPR) = \text{val}(LDR)$, the latter equality following from the strong duality theorem in finite linear programming. Since the feasible set of (LDR) includes the feasible set of (LDD) , $\text{val}(LDD) \geq \text{val}(LDR)$. Thus

$$\text{val}(LDD) \geq \text{val}(LDR) = \text{val}(LPR) = \text{val}(LDP)$$

which together with the assumption $\text{val}(LDD) = \text{val}(LDP)$ imply that $\text{val}(LDD) = \text{val}(LDR)$. \square

In the presence of a duality gap we can show that

$$\text{val}(LDD) > \text{val}(LDR) = \text{val}(LPR) = \text{val}(LDP)$$

In this case $\text{val}(LDR)$ is a *subvalue* for (LDD) (see Anderson and Nash [3]). Moreover the duality gap can be as large as possible. See Anderson and Nash [3], Glashoff and Gustafson [17] and Bonnans and Shapiro [7] for examples of duality gaps in linear semi-infinite programming.

We now establish a bound on the number of constraints m in (LDR) .

Theorem 3 *Suppose (LDD) is consistent and that there exists a finite discretization of (LDD) with the same optimal value. Then there exists a discretization (LDR) such that $\text{val}(LDD) = \text{val}(LDR)$ and $m \leq k$.*

Proof: Follows from Theorem 1. \square

See Bonnans and Shapiro [7] for a proof based on Helly's theorem. Moreover if we were to solve (LDR) using a simplex scheme, then not more than k constraints would be binding at optimality. The following results are only special cases for general results on the discretization of convex semi-infinite programs (Borwein [8], Ben Tal et al [5], Bonnans and Shapiro [7] and Hettich and Kortanek [25]).

We discuss the finite linear programs (LDR) and (LPR) and some of their properties below. Given a finite set of vectors $\{d_i, i = 1, \dots, m\}$, we obtain the relaxation

$$\begin{aligned} \max \quad & b^T y \\ \text{subject to} \quad & d_i d_i^T \bullet \mathcal{A}^T y \leq d_i d_i^T \bullet C \quad \text{for } i = 1, \dots, m. \end{aligned} \quad (LDR)$$

We now derive the linear programming dual to (LDR) . We have

$$\begin{aligned} d_i d_i^T \bullet \mathcal{A}^T y &= d_i d_i^T \bullet \left(\sum_{j=1}^k y_j A_j \right) \\ &= \sum_{j=1}^k y_j d_i^T A_j d_i. \end{aligned}$$

Thus, the constraints of (LDR) can be written as

$$\sum_{j=1}^k y_j d_i^T A_j d_i \leq d_i^T C d_i \quad \text{for } i = 1, \dots, m.$$

It follows that the dual problem is

$$\begin{aligned} \min \quad & \sum_{i=1}^m d_i^T C d_i x_i \\ \text{subject to} \quad & \sum_{i=1}^m d_i^T A_j d_i x_i = b_j \quad \text{for } j = 1, \dots, k \\ & x \geq 0. \end{aligned}$$

This can be rewritten as

$$\begin{aligned} \min \quad & C \bullet \left(\sum_{i=1}^m x_i d_i d_i^T \right) \\ \text{subject to} \quad & \mathcal{A} \left(\sum_{i=1}^m x_i d_i d_i^T \right) = b \quad (LPR) \\ & x \geq 0. \end{aligned}$$

Lemma 1 *Any feasible solution x to (LPR) will give a feasible solution X to (SDP) .*

Proof: This lemma follows directly from the fact that (LPR) is a constrained version of (SDP) . However we present a formal proof. Set $X = \sum_{i=1}^m x_i d_i d_i^T$. From (LPR) it is clear that this X satisfies $\mathcal{A}X = b$. Moreover X is psd. To see this

$$\begin{aligned} d^T X d &= d^T \left(\sum_{i=1}^m x_i d_i d_i^T \right) d = \sum_{i=1}^m x_i (d_i^T d)^2 \\ &\geq 0 \quad \forall d \end{aligned}$$

where the last inequality follows from the fact that $x \geq 0$. Moreover this X also satisfies $\text{tr}(X) = a$. \square

Thus the optimal value to (LPR) gives an upper bound on the optimal value of (SDP) . Moreover lemma 1 suggests a way of recovering a feasible primal X matrix, which is extremely important if we are solving the underlying integer program. For more details refer to section 5.2 on the max cut problem.

3 Geometry of the SDP

In this section we study the geometry of the (*SDP*) problem. In particular we begin with Pataki's lemma [39] which gives a bound on the rank of optimal X matrices. This has immediate implications on the size of the *LP* relaxation (*LDR*). We conclude this section with the constraints, we need in this *LP* relaxation.

We have strong duality $XS = 0$ at optimality (Assumption 1). This implies that X and S commute at optimality, and thus share a common set of eigenvectors. To emphasize this point, we present the following lemma 2 (Alizadeh et al [2]).

Lemma 2 *Let X and (y, S) be primal and dual feasible respectively. Then they are optimal if and only if there exists $\bar{P} \in \mathbb{R}^{n \times n}$, with $\bar{P}^T \bar{P} = I$, such that*

$$\begin{aligned} X &= \bar{P} \text{Diag}(\lambda_1, \dots, \lambda_n) \bar{P}^T \\ S &= \bar{P} \text{Diag}(\omega_1, \dots, \omega_n) \bar{P}^T \\ \lambda_i \omega_i &= 0 \quad , i = 1, \dots, n \end{aligned}$$

Here \bar{P} is an orthogonal matrix containing the eigenvectors of X and S . Moreover the complementarity condition suggests that if X is of rank r and S is of rank s , we have $r + s \leq n$. We can replace this inequality by an equality if we impose *nondegeneracy* assumptions on (*SDP*) and (*SDD*). For more on nondegeneracy in the context of (*SDP*) refer to Alizadeh et al [2].

The following Theorem 4, due to Pataki [39] gives an upper bound on the rank r , of optimal X matrices.

Theorem 4 *There exists an optimal solution X^* with rank r satisfying the inequality $\frac{r(r+1)}{2} \leq k$. Here k is the number of constraints in (*SDP*).*

Theorem 4 suggests that there is an optimal matrix X that satisfies the upper bound (whose rank is around $O(\sqrt{k})$). A similar result is established in Alizadeh et al [2] under a nondegeneracy assumption. However, without this assumption the bound need not hold for all solutions.

Thus the optimal X can be expressed as $X = P\Lambda P^T$. Here $\Lambda \in \mathcal{S}^r$ is a diagonal matrix containing the r nonzero eigenvalues of X , and P is an orthonormal matrix satisfying $P^T P = I_r$, and containing the eigenvectors,

corresponding to these r eigenvalues. Moreover to preserve the set of optimal solutions, r should be at least $\frac{\sqrt{1+8k}-1}{2}$.

From Lemma 2, it is clear that the columns in P belong to the null space of the optimal S , i.e. we have $p_i^T S p_i = 0$, $i = 1, \dots, r$.

In the context of solving an (SDP) as an (LP) , Pataki's theorem suggests that there is an relaxation (LDR) which exactly captures the (SDP) objective value, and has no more than $O(\sqrt{k})$ constraints. This is a tightening of Theorem 3, which relies solely on linear semi-infinite programming theory. As a result not more than $O(\sqrt{k})$ variables in (LPR) would be nonzero. We express this fact in corollary 2 below.

Corollary 2 *Any (LPR) with $val(LPR) = val(SDP)$ must be degenerate.*

Theorem 5 *(SDP) is equivalent to (LDR) with $d_i = p_i$, $i = 1, \dots, r$.*

Proof: The optimal value to (LPR) gives an upper bound on the optimal value of (SDP) . Thus an optimal solution to (SDP) is also optimal in (LPR) , provided it is feasible in (LPR) . The optimal solution to (SDP) is given by $X = P\Lambda P^T = \sum_{i=1}^r \lambda_i p_i p_i^T$, where $\lambda_i > 0$, $i = 1, \dots, r$, and p_i , $i = 1, \dots, r$ are the corresponding eigenvectors. This is clearly feasible in (LPR) . This corresponds to (LDR) with $d_i = p_i$, $i = 1, \dots, r$. \square

We must mention at this stage that we could in practice solve (SDP) using an interior point scheme, and utilize the P corresponding to the strictly positive eigenvalues of X as d . This would give an (LDR) which attains the (SDP) optimal value. However as we have seen in the previous section interior point schemes are fairly limited in the size of problems they can handle. Besides interior point schemes do not exploit Pataki's lemma (Theorem 4).

The spectral bundle method discussed in the next section, provides a way to estimate these vectors quickly.

4 The spectral bundle method

The spectral bundle method is due to Helmberg and Rendl [21]. Other references include Helmberg et al [20, 22, 23] and Oustry [38]. In this section we give a detailed description of the spectral bundle scheme.

Since our original (SDP) is a minimization problem, we will be dealing in this section, with the minimum eigenvalue function, a concave function.

However we shall be using terms like subgradients, subdifferential etc, usually associated with convex functions. These terms should be understood to be the corresponding analogues for a concave function.

Consider the eigenvalue optimization problem (3).

$$\max_y \quad a\lambda_{\min}(C - \mathcal{A}^T y) + b^T y \quad (3)$$

Problems of this form are equivalent to the dual of semidefinite programs (*SDP*), whose primal feasible set has a constant trace, i.e. $\text{Trace}(X) = a$ for all $X \in \{X \succeq 0 : \mathcal{A}(X) = b\}$. This can be easily verified as follows. From the variational characterization of the minimum eigenvalue function, we have

$$\lambda_{\min}(C - \mathcal{A}^T y) = \min_{X: \text{tr} X = 1, X \succeq 0} (C - \mathcal{A}^T y) \bullet X \quad (4)$$

Thus (3) is equivalent to taking the Lagrangian dual of (*SDP*) with y being the vector of dual variables corresponding to $\mathcal{A}(X) = b$, and observing that $a\lambda_{\min}(C - \mathcal{A}^T y) = \min_{X: \text{tr} X = a, X \succeq 0} (C - \mathcal{A}^T y) \bullet X$. We can rewrite (3) strictly as an eigenvalue optimization problem by incorporating the linear term $b^T y$ into the eigenvalue function (each A_i is now replaced with $(A_i - b_i I)$). It can be shown that any SDP with a bounded feasible set can be written in this way, i.e. as an eigenvalue optimization problem. The minimum eigenvalue function $\lambda_{\min}(\cdot)$ is a nonsmooth concave function (the concavity follows from the variational characterization). In fact this function is nonsmooth precisely at those points where the minimum eigenvalue has a multiplicity greater than one. A general scheme to minimize such functions is the *bundle* scheme (Kiwiel [30, 31, 32], Lemarechal [35] and Schramm et al [41] and the books by Urruty and Lemarechal [27, 28] (especially Chapter XV(3)). An excellent survey on eigenvalue optimization appears in Lewis and Overton [36].

To motivate the bundle approach, we begin with some preliminaries for the minimum eigenvalue function.

The subdifferential for the minimum eigenvalue function, i.e. the collection of subgradients has the following representation.

$$\partial\lambda_{\min}(X) = \{PVP^T : V \in \mathcal{S}^r, \text{tr}(V) = 1, V \succeq 0\} \quad (5)$$

Here P is an orthogonal matrix, containing all the eigenvectors corresponding to the minimum eigenvalue. The expression (5) implies that the subdifferential is given by the convex hull of rank one matrices pp^T , where p is

an normalised eigenvector corresponding to the minimum eigenvalue (these eigenvectors are orthogonalized with respect to each other so that P is an orthogonal matrix satisfying $P^T P = I_r$). Also, r here is the multiplicity of the minimum eigenvalue function. An upper bound on r is given by Pataki's lemma (Theorem 4). We expect the value of r to increase as we approach optimality. This is because we are maximizing the minimum eigenvalue and the eigenvalues tend to cluster together. Once we have the subdifferential we can write the directional derivative $\lambda'_{min}(y, D)$ as follows :

$$\lambda'_{min}(y, D) = \lambda_{min}(P^T D P) \quad (6)$$

Here $D = -A^T d = -\sum_{i=1}^k d_i A_i$ where d is the current search direction. P is the orthogonal matrix containing the eigenvectors corresponding to the minimum eigenvalue function. The expression follows immediately from the fact that the directional derivative is the support function for the subdifferential (Urruty and Lemarechal [27] and Oustry [38]).

We can design a *steepest descent* like scheme for maximizing the minimum eigenvalue function as follows :- The search direction d involves solving the following subproblem

$$\max_{d \in \mathbb{R}^k, \|d\| \leq 1} \lambda_{min}(P^T D P) = \min_{W \in \partial \lambda_{min}(X)} \frac{1}{2u} \|b - \mathcal{A}(W)\|^2 \quad (7)$$

The constraint $\|d\| \leq 1$ in (7) ensures that solution to the max-min problem on the left is bounded. We can interchange the max and the min using strong duality to get the problem on the right. Note that we have introduced the constraint $\|d\| \leq 1$ into the objective function by means of Lagrangian weight parameter u (the term $\frac{u\|d\|^2}{2}$ penalises us from going too far from the current point). This gives a quadratic semidefinite programming problem (*QSDP*). A few points are now in order :

1. At optimality when the search direction $d = 0$, we get primal feasibility $\mathcal{A}(X) = b$.
2. Solving (*QSDP*) with $X = P V P^T \in \partial \lambda_{min}(X)$ amounts to relaxing $S \succeq 0$ to $P^T S P \succeq 0$.
3. This amounts to the solution of the following eigenvalue problem (8) over a *relaxation of the dual feasible region*.

$$\max_y \quad a \lambda_{min}(P^t (C - \mathcal{A}^T y) P) + b^T y \quad (8)$$

The spectral bundle returns the objective value of a feasible point in (*SDD*). Hence the scheme provides a lower bound on the optimal (*SDP*) objective value.

4. For the convergence of the scheme, we actually need to consider the entire ϵ subdifferential at each iteration which consists of all eigenvectors corresponding to eigenvalues which are within an ϵ of the minimum eigenvalue.

Computing the entire ϵ subdifferential at each iteration is difficult, and this is where the bundle idea comes in handy (Helmberg et al [21]). Instead of computing the entire subdifferential, we consider an arbitrary subgradient from the subdifferential and utilize the important subgradients from the previous iterations. Helmberg and Rendl [21] consider the following subset (9) of the subdifferential.

$$\begin{aligned} \hat{\mathcal{W}} &\subset \partial\lambda_{\min}(S) \\ &= \{W : W = \alpha\bar{W} + PVP^T, \alpha + \text{tr}(V) = 1, \alpha \geq 0, V \succeq 0\} \end{aligned} \quad (9)$$

Here \bar{W} is known as the *aggregate matrix* and contains the less important subgradient information, while P is known as the *bundle* and contains the important subgradient information. We are now ready to describe the key ideas in the spectral bundle approach.

1. Consider a subset $X = PVP^T$ of the feasible X matrices. We are exploiting Pataki's lemma [39] by considering only the subset of all optimal X matrices, which allows us to operate in a lower dimensional space.
2. Use this X to improve y . This involves solving the following direction finding subproblem

$$\min_{X \in \hat{\mathcal{W}}} \frac{1}{2u} \|b - \mathcal{A}(X)\|^2 \quad (QSDP)$$

We are now optimizing over all X belonging to the set $\hat{\mathcal{W}}$ instead of the entire subdifferential. (*QSDP*) is a quadratic SDP in $\binom{r+1}{2} + 1$ variables. Moreover the aggregate matrix \bar{W} gives us the flexibility of using fewer columns (than the r given by Pataki's lemma) in P . This is important, for we want to keep r small so that (*QSDP*) can be solved quickly.

3. Use the value of y in turn to improve X . This is done by updating the bundle P in each iteration. P attempts to recreate $\partial\lambda_{\min}(S)$ at each iteration. If P is a bad approximation to the subdifferential at a point, we remain at that point and update P with more subgradient information. This is termed as a *null step* in the spectral bundle scheme.

We have omitted a lot of details here; on how P and \bar{W} are updated in each iteration, the ideas behind the convergence of the scheme etc. These are not very important for the discussion in the next section. For more details refer to Helmberg et al [20, 21].

The spectral bundle scheme has very good global convergence properties. However it is only a *first order* scheme, since we are carrying out a first order approximation of the minimum eigenvalue function. A second order bundle method which converges globally and which enjoys asymptotically a quadratic rate of convergence was developed by Oustry [38]. Helmberg and Kiwiel [22] also extend the spectral bundle approach to problems with bounds.

5 A set of linear constraints

A set of linear constraints for (*LDR*) can be derived from the bundle information used by the spectral bundle method. We propose using the columns of P as the vectors $\{d_j\}, j = 1, \dots, r$ in (*LDR*). Since the number of vectors r in the bundle P is $O(\sqrt{k})$ and we need at least k constraints to guarantee a basic feasible solution, we need to look for other constraints as well. We shall henceforth label these constraints as *box* constraints. Note that the columns of P are dense, leading to a dense linear programming formulation (*LDR*). We try to compensate for these dense constraints, by choosing d for our box constraints that are sparse.

This section is organized as follows. The rationale for using the columns of P as d is discussed in section 5.1. We illustrate the *LP* procedure on the max cut problem in section 5.2, and the min bisection problem in section 5.3.

5.1 Rationale for using the columns of P

We are now ready to present the primary result in the paper.

Theorem 6 *If the spectral bundle terminates, and the number of columns r in P satisfies Pataki's lemma, then these columns are candidates for the cutting planes d_i .*

Proof: At the outset we shall assume that the bundle scheme terminates in a finite number of steps. This will happen when $d = 0$ in $(QSDP)$. We then have primal feasibility, i.e. $\mathcal{A}(X) = b$, since $0 \in \partial\lambda_{min}(S)$. Thus $(QSDP)$ is now (SDP) for some $X = \alpha\bar{W} + PVP^T$. Since P contains the important subgradient information we expect $\text{tr}(X) \approx \text{tr}(V)$. Moreover if P contains at least r columns, where r is given by Pataki's theorem 4, then α would be zero (since we have not more than r nonzero eigenvalues at optimality). Thus PVP^T is essentially a spectral decomposition of X , with P containing the eigenvectors corresponding to the nonzero eigenvalues of X . Since we have strong duality (Assumption 1), the vectors in P provide a basis for the null space of S . From theorem 5 we find the columns of P ensure that $\text{val}(LDR) = \text{val}(SDP) = \text{val}(SDD)$. \square

Typically the bundle method converges only in the limit to the optimal solution. When the method terminates, the bundle P should contain a good approximation for the ϵ subdifferential of $\lambda_{min}(S) = 0$.

$$\begin{aligned} \lambda_{min}(S^*) &\leq p_i p_i^T \bullet S^* + \epsilon \\ &= 0 \end{aligned}$$

We have a \leq inequality since $\lambda_{min}(S)$ is a concave function. This is the best approximation to $\lambda_{min}(S)$ in a first order sense, since any other d would satisfy $dd^T \bullet S^* > 0$ (strict inequality). Since these constraints are nearly binding, if we employ a simplex scheme to solve (LDR) we would expect these constraints to be active at the optimal solution (among others since k constraints would be binding at optimality) and by complementary slackness the corresponding dual variables x will be nonzero in (LPR) (again we expect them to be the larger ones in the nonzero x). Thus the columns of P are *good* candidates for the cutting planes d_i .

5.2 The Max Cut problem

A semidefinite programming relaxation of the max cut problem is Goemans and Williamson [19]. The SDP solution followed by a randomised rounding leads to an 0.878 approximation algorithm for the max cut problem.

$$\begin{aligned} & \max && \frac{L}{4} \bullet X \\ & \text{subject to} && \text{diag}(X) = e \\ & && X \succeq 0, \end{aligned} \tag{10}$$

with dual

$$\begin{aligned} & \min && e^T y \\ & \text{subject to} && -\text{Diag}(y) + S = -\frac{L}{4} \\ & && S \succeq 0 \end{aligned} \tag{11}$$

Here $L = \text{Diag}(Ae) - A$ is the Laplacian matrix of the graph, where A is the weighted adjacency matrix with $A_{ii} = 0, \forall i$ and $A_{ij} = w_{ij}, \forall \{i, j\} \in E$. Thus the Laplacian matrix is

$$\begin{aligned} L_{ii} &= \sum_j w_{ij} \quad \forall i \\ L_{ij} &= -w_{ij} \quad i \neq j \end{aligned}$$

Note that the a in $\text{tr}X = a$ is trivially n , the number of nodes in the graph.

Since S is psd, we have $d^T S d = d^T (\text{Diag}(y) - \frac{L}{4}) d \geq 0, \forall d$. In particular we propose to use the following d for the max cut problem.

MC1 Setting $d = e_i, i = 1 \dots n$, where e_i is the i th standard basis vector for R^n . In particular e_i has a one in the i th position and zeros elsewhere. This generates the constraint $y_i \geq \frac{L_{ii}}{4}, i = 1 \dots n$.

MC2 Setting $d = (e_i + e_j)$ and $(e_i - e_j), \forall \{i, j\} \in E$, gives rise to the constraints $y_i + y_j \geq \frac{L_{ii}}{4} + \frac{L_{jj}}{4} + \frac{L_{ij}}{2}$ and $y_i + y_j \geq \frac{L_{ii}}{4} + \frac{L_{jj}}{4} - \frac{L_{ij}}{2}$ respectively. Together these give $y_i + y_j \geq \frac{L_{ii}}{4} + \frac{L_{jj}}{4} + |\frac{L_{ij}}{2}|$.

MC3 The constraints in the bundle namely the columns $p_i, i = 1, \dots, r$ of the matrix P .

We consider an LP relaxation, LP containing MC1 and MC3. To obtain tighter relaxations we can consider MC2 in addition to MC1 and MC3.

The LP relaxation LP is summarised below

$$\begin{aligned} & \min && e^T y \\ & \text{subject to} && y_i \geq \frac{L_{ii}}{4} \quad \forall i = 1, \dots, n \\ & && \sum_{i=1}^n p_{ji}^2 y_i \geq p_j^T \frac{L}{4} p_j \quad \forall j = 1, \dots, r \end{aligned} \tag{12}$$

with dual

$$\begin{aligned} \max \quad & \sum_{i=1}^n \frac{L_{ii}}{4} x_i + \sum_{j=1}^r w_j \frac{p_j^T L p_j}{4} \\ \text{subject to} \quad & \begin{bmatrix} 1 & & & & \\ & \ddots & & & \\ & & p_1^2 & \dots & p_r^2 \\ & & 1 & \downarrow & \downarrow \end{bmatrix} \begin{bmatrix} x \\ w \end{bmatrix} = e \\ & \begin{bmatrix} x \\ w \end{bmatrix} \geq 0 \end{aligned} \quad (13)$$

Here p_{ji} refers to the j th component of vector p_i and p_i^2 , $i = 1, \dots, r$ are vectors obtained by squaring all the components of p_i , $i = 1, \dots, r$. (12) has $n+r$ constraints in all. Note that $x \in \mathbb{R}^n$ and $w \in \mathbb{R}^r$ are the dual variables corresponding to $y \geq \text{diag} \frac{L}{4}$ and the bundle constraints respectively. To get a solution X to SDP , set $X = \text{Diag}(x) + \sum_{j=1}^r w_j p_j p_j^T$. This matrix X is positive semidefinite since $x \geq 0$ and $w \geq 0$. Moreover

$$\frac{L}{4} \bullet X = \frac{L}{4} \bullet (\text{Diag}(x) + \sum_{j=1}^r w_j p_j p_j^T) = \sum_{i=1}^n \frac{L_{ii}}{4} x_i + \sum_{j=1}^r w_j \frac{p_j^T L p_j}{4}$$

This is precisely the objective value in (13). We have thus generated the X which could be used in the Goemans and Williamson rounding procedure [19] to generate an approximate solution to the max cut problem.

Using the Goemans and Williamson [19] (GW) rounding procedure on the X generated by solving the relaxation LP , we can generate a cut that is at least 0.878 times the LP objective value. We cannot guarantee that the objective value of relaxation LP is an upper bound on the maximum cut value. However, in practice the LP objective is within 1% of the spectral bundle objective value, which incidentally is an upper bound on the optimal SDP value. Thus we have some performance guarantee on the cut produced by solving the LP relaxation LP followed by the GW randomized rounding procedure.

5.3 The Min Bisection problem

The semidefinite programming relaxation for the min bisection problem was independently proposed by Frieze and Jerrum [16] and Ye [46]. The relaxation is

$$\begin{aligned}
& \min && \frac{L}{4} \bullet X \\
& \text{subject to} && \text{diag}(X) = e \\
& && ee^T \bullet X = 0 \\
& && X \succeq 0,
\end{aligned} \tag{14}$$

with dual

$$\begin{aligned}
& \min && e^T y \\
& \text{subject to} && -y_0(ee^T) - \text{Diag}(y) + S = \frac{L}{4} \\
& && S \succeq 0
\end{aligned} \tag{15}$$

We must note that Frieze and Jerrum [16] had the equipartition constraint as $ee^T \bullet X \leq 0$. But since the optimal X is psd, we must have $e^T X e = ee^T \bullet X \geq 0$ at optimality, which is equivalent to $ee^T \bullet X = 0$.

Here L refers to the Laplacian matrix of the graph. y_0 is the dual variable corresponding to the constraint $ee^T \bullet X = 0$. To get the signs right, we need to take the negative of the objective value of (SDD) to get the optimal solution to the min bisection problem. Again $a = n$.

Note that the primal (14) does not have any Slater point (strictly feasible point) due to the equipartition constraint $ee^T \bullet X = 0$. Thus (SDD) need not attain its optimal solution. (There is no duality gap however since the dual (15) has a Slater point). Moreover (15) has an unbounded optimal face. To observe this set $y_0 \rightarrow \infty$ in (15). Doing so keeps S psd, but since y_0 does not appear in the objective function, this value remains unchanged.

We must emphasize here, that the absence of a primal Slater point affects our LP relaxations (since we are dealing with (LDD)), and we need not have a discretization, if (LDD) does not attain its optimal solution (Theorem 2).

Since $S = y_0 ee^T + \text{Diag}(y) + \frac{L}{4}$ is psd, we require $d^T S d = d^T (y_0 ee^T + \text{Diag}(y) + \frac{L}{4}) d \geq 0, \forall d$.

In particular we propose to use the following d for the min bisection problem.

MB1 Setting $d = e_i, i = 1, \dots, n$ gives $y_0 + y_i \geq -\frac{L_{ii}}{4}, i = 1, \dots, n$.

MB2 Setting $d = e$, where e is the all ones vector gives $ny_0 + \sum_{i=1}^n y_i \geq 0$, since $Le = 0$.

MB3 The constraints in the bundle namely the columns $p_i, i = 1, \dots, r$. This gives $y_0(p_i^T e)^2 + \sum_{j=1}^n p_{ji}^2 y_j \geq -p_i^T \frac{L}{4} p_i, i = 1, \dots, r$.

MB4 Since the *SDP* has an unbounded feasible set, we impose an upper bound on y_0 say u .

The resulting *LP* is

$$\begin{aligned}
& \min && e^T y \\
& \text{subject to} && y_0 + y_i \geq -\frac{L_{ii}}{4} \quad i = 1, \dots, n \\
& && ny_0 + \sum_{i=1}^n y_i \geq 0 \\
& && (p_i^T e)^2 y_0 + \sum_{j=1}^n p_{ji}^2 y_i \geq -p_i^T \frac{L}{4} p_i \quad i = 1, \dots, r \\
& && y_0 \leq u
\end{aligned} \tag{16}$$

Here p_{ji} refers to the j th component of the vector p_i . The *LP* (16) has $n + 1 + r$ constraints in all (excluding the upper bound).

If we set the upper bound u , we are in essence solving the following pairs of *SDP*.

$$\begin{aligned}
& \min && \begin{bmatrix} \frac{L}{4} & 0 \\ 0 & u \end{bmatrix} \bullet \begin{bmatrix} X & 0 \\ 0 & x_s \end{bmatrix} \\
& \text{subject to} && \text{diag}(X) = e \\
& && ee^T \bullet X = x_s \\
& && \begin{bmatrix} X & 0 \\ 0 & x_s \end{bmatrix} \succeq 0
\end{aligned} \tag{17}$$

with dual

$$\begin{aligned}
& \max && 0y_0 + e^T y \\
& \text{subject to} && \text{Diag}(y) + ee^T y_0 + S = \frac{L}{4} \\
& && S \succeq 0 \\
& && y_0 \leq u
\end{aligned} \tag{18}$$

Here x_s is the dual variable corresponding to the upper bound constraint $y_0 \leq u$. Also we have $ee^T \bullet X = x_s$ in (17). Similarly the dual variable corresponding to this upper bound constraint in (16) should provide an estimate for $ee^T \bullet X$. Note that the (17) has a Slater point and hence the dual (18) attains its optimal solution.

6 Computational results

In this section we test the linear programming approach on the max cut and min bisection problems. The instances are taken from the 7th DIMACS Implementation Challenge [40] and Borchers' SDPLIB [9]. The bundle constraints are computed using Helmbert's spectral bundle code *SBmethod, Version 1.1* [24] available at <http://www.zib.de/helmbert/index.html>. *CPLEX 6.5* [29] is employed in solving the LP relaxations. All tests are executed on a *Sun Ultra 5.6, 440MHz* machine with 128 MB of memory.

We utilize the default bundle parameters which are :

1. The relative tolerance (*-te*). The default value is $1e - 5$.
2. The size of the bundle i.e. the number of columns in P . This in turn is controlled by
 - (a) The maximum number of vectors kept n_k (*-mk*). The default value is 20.
 - (b) The maximum number of vectors added (*-ma*). The default value is 5.
 - (c) The minimum number of vectors added n_{min} (*-mik*). The default value is 5.

The columns in the tables represent

n Number of nodes in the graph.

k Number of *SDP* constraints.

m Number of edges in the graph.

r Bundle size, the number of columns in P .

% Error $|\frac{SDP-LP}{SDP} \times 100|$.

SDP The objective value of *SDP*.

LP The value of *LP* relaxation.

m1 The number of constraints in the *LP* relaxation.

| Name | n | m | r | SDP | LP | % Error |
|-----------------------------|------|-------|----|----------|----------|---------|
| toruspm-8-50 ¹ | 512 | 1536 | 16 | 527.81 | 525.91 | 0.36 |
| toruspm3-15-50 ¹ | 3375 | 10125 | 21 | 3.47e+03 | 3.43e+03 | 1.15 |
| torusg3-8 ¹ | 512 | 1536 | 13 | 4.57e+07 | 4.54e+07 | 0.66 |
| torusg3-15 ¹ | 3375 | 10125 | 18 | 3.13e+08 | 3.10e+08 | 0.96 |
| mcp100 ² | 100 | 269 | 10 | 226.16 | 225.75 | 0.18 |
| mcp124-1 ² | 124 | 149 | 20 | 141.99 | 141.06 | 0.65 |
| mcp124-2 ² | 124 | 318 | 11 | 269.88 | 268.97 | 0.34 |
| mcp124-3 ² | 124 | 620 | 11 | 467.75 | 467.37 | 0.08 |
| mcp124-4 ² | 124 | 1271 | 10 | 864.41 | 863.72 | 0.08 |
| mcp250-1 ² | 250 | 331 | 10 | 317.26 | 317.18 | 0.03 |
| mcp250-2 ² | 250 | 612 | 14 | 531.93 | 531.18 | 0.14 |
| mcp250-3 ² | 250 | 1283 | 13 | 981.17 | 980.32 | 0.09 |
| mcp250-4 ² | 250 | 2421 | 13 | 1681.96 | 1679.70 | 0.13 |
| mcp500-1 ² | 500 | 625 | 10 | 598.15 | 594.12 | 0.67 |
| mcp500-2 ² | 500 | 1223 | 13 | 1070.06 | 1069.90 | 0.02 |
| mcp500-3 ² | 500 | 2355 | 14 | 1847.97 | 1843.20 | 0.26 |
| mcp500-4 ² | 500 | 5120 | 15 | 3566.74 | 3559.80 | 0.19 |
| maxG11 ² | 800 | 1600 | 11 | 629.16 | 625.99 | 0.50 |
| maxG32 ² | 2000 | 4000 | 14 | 1567.64 | 1557.91 | 0.62 |
| maxG51 ² | 1000 | 5909 | 19 | 4003.81 | 3988.30 | 0.39 |

Table 1: Max Cut Test Results

In table 1 we compare the *SDP* objective value with the value of the *LP* relaxation. It is seen that the *LP* relaxation provides a fairly good approximation to the *SDP* objective value, with the %error within a % of the *SDP* objective value.

| Name | SDP | | LP ml |
|----------------|------|------|----------|
| | k | n | |
| toruspm-8-50 | 512 | 512 | 528 |
| toruspm3-15-50 | 3375 | 3375 | 3396 |
| torusg3-8 | 512 | 512 | 525 |
| torusg3-15 | 3375 | 3375 | 3393 |
| mcp100 | 100 | 100 | 110 |
| mcp124-1 | 124 | 124 | 144 |
| mcp124-2 | 124 | 124 | 135 |
| mcp124-3 | 124 | 124 | 135 |
| mcp124-4 | 124 | 124 | 134 |
| mcp250-1 | 250 | 250 | 260 |
| mcp250-2 | 250 | 250 | 264 |
| mcp250-3 | 250 | 250 | 263 |
| mcp250-4 | 250 | 250 | 263 |
| mcp500-1 | 500 | 500 | 510 |
| mcp500-2 | 500 | 500 | 513 |
| mcp500-3 | 500 | 500 | 514 |
| mcp500-4 | 500 | 500 | 515 |
| maxG11 | 800 | 800 | 811 |
| maxG32 | 2000 | 2000 | 2014 |
| maxG51 | 1000 | 1000 | 1019 |
| maxG55 | 5000 | 5000 | 5025 |
| maxG60 | 7000 | 7000 | 7025 |

Table 2: Sizes of the max cut relaxations

We list the sizes of the *LP* relaxation in table 2. Note that $k = n$, for the max cut problem. Thus the *LP* relaxation has $(n + \sqrt{n}) = O(n)$ constraints.

¹DIMACS [40]

²SDPLIB [9]

³Run out of memory

| Name | m | n | r | SDP | LP | m1 | % Error | $ ee^T \bullet X $ |
|------------------------|--------|-------|----|---------|-----------------|-----|---------|--------------------|
| bm1 ¹ | 4711 | 882 | 10 | 23.44 | 24.99 | 893 | 6.59 | 0.06 |
| gpp100 ² | 264 | 100 | 10 | 44.94 | 45.85 | 111 | 0.44 | 0.00 |
| gpp124-1 ² | 149 | 124 | 10 | 7.34 | 7.35 | 135 | 0.01 | 0.00 |
| gpp124-2 ² | 318 | 124 | 10 | 46.86 | 47.52 | 135 | 0.55 | 0.00 |
| gpp124-3 ³ | 620 | 124 | 11 | 153.01 | 153.59 | 136 | 0.16 | 0.00 |
| gpp124-4 ² | 1271 | 124 | 11 | 418.99 | 420.35 | 261 | 0.14 | 0.01 |
| gpp250-1 ² | 331 | 250 | 10 | 15.45 | 15.45 | 261 | 0.88 | 0.00 |
| gpp250-2 ² | 612 | 250 | 12 | 81.87 | 82.22 | 263 | 0.26 | 0.00 |
| gpp250-3 ² | 1283 | 250 | 13 | 303.50 | 305.99 | 264 | 0.56 | 0.00 |
| gpp250-4 ² | 2421 | 250 | 13 | 747.30 | 751.72 | 264 | 0.43 | 0.00 |
| gpp500-1 ² | 625 | 500 | 11 | 25.30 | 26.83 | 512 | 3.62 | 0.00 |
| gpp500-2 ² | 1223 | 500 | 13 | 156.06 | 158.75 | 514 | 1.06 | 0.00 |
| gpp500-3 ² | 2355 | 500 | 15 | 513.02 | 520.21 | 516 | 0.95 | 0.01 |
| gpp500-4 ² | 5120 | 500 | 15 | 1567.02 | 1578.94 | 516 | 0.57 | 0.00 |
| biomedP ¹ | 629839 | 6514 | - | 33.60 | MM ³ | - | - | - |
| industry2 ¹ | 798219 | 12637 | - | 65.61 | MM ³ | - | - | - |

Table 3: Min Bisection Test Results

In table 3 we compare the *SDP* objective value with the value of the *LP* relaxation (16). Here $u = 1$ is the upper bound on the variable y_0 . It is seen that the *LP* relaxation provides a good approximation to the *SDP* objective value. Moreover the dual variable x_s provides an estimate for $|ee^T \bullet X|$. This value is well below 0.1 for all the reported instances. A typical *LP* relaxation has $n + 1 + \sqrt{n} = O(n)$ constraints.

7 Conclusions

In this paper we have described *LP* approaches to solving *SDP* problems. The *LP* approach uses the columns of the columns of P , in the spectral bundle approach developed by Helmberg and Rendl [21] as constraints in the *SDP*. The number of these constraints is bounded by the square root of the number of constraints in the *SDP*. The resulting *LP*'s can be solved quickly and provide reasonably accurate solutions. The key idea in using these constraints is that they provide a good approximation to the null space of S at optimality. It must emphasized that main computational task in this bundle *LP* approach is in estimating the columns of the bundle P . Solving the resulting *LP*'s is relatively trivial.

We can look at the LP approach presented in this chapter also as a way of generating feasible primal iterates for the spectral bundle method. When the spectral bundle terminates, the primal matrix X is not as yet primal feasible, namely $\mathcal{A}(X) \approx b$ (in fact equality is attained only in the limit). It is important to get hold of a primal feasible X in an SDP cutting plane approach for combinatorial optimization problems.

The LP relaxation we solve is a relaxation to the dual SDP (*SDD*). The objective value returned by the spectral bundle scheme is for a feasible y , where $S = C - \mathcal{A}^T y$ is psd. Thus when (*SDP*) is a minimization problem, the two schemes give upper and lower bounds respectively on the SDP objective value. Thus our approach provides another termination criteria for the bundle approach (close to optimality). For the maxcut problem cannot guarantee that our *LP* relaxation is an upper bound on the max cut value. However in all cases the *LP* relaxation is within a small percentage of the *SDP* objective value, a guaranteed upper bound on the *SDP* objective value.

We could, in practice strengthen the *LP* relaxations by computing a spectral decomposition of the optimal $X = \alpha \bar{W} + PV P^T = \bar{P} \bar{V} \bar{P}^T$, and use the columns of \bar{P} (which correspond to the nonzero eigenvalues in \bar{V}) as our

linear constraints. But we found that this does not substantially improve the LP relaxations for the max cut and min bisection problems, and hence we have chosen not to mention these in our computational results. However this choice may be important for (SDP) with a large number of constraints such as k equipartition and Lovasz theta problems (since we attempt to solve these problems with a small number of columns in P). As a result the aggregate matrix \bar{W} , which contains the remaining subgradient information, becomes important.

We have successfully tested the bundle LP approach on max cut, min bisection and box constrained QP 's. However it must be emphasized that we have not had much success with SDP 's with a large number of constraints such as Lovasz theta problems (Krishnan and Mitchell [33]). Since the number of constraints in the bundle is $O(\sqrt{k})$, larger bundle sizes are necessary. Moreover it might be interesting to consider a *second order* bundle scheme (Oustry [38]) to obtain quick solutions with smaller bundle sizes.

Another interesting topic for future research is to incorporate the LP approach within a cutting plane or a branch and bound approach to solving integer programs. However this approach is not entirely straightforward. For instance for the max cut problem, it is not possible to guarantee *a priori* that the LP feasible region will contain the entire max cut polytope. This will lead to infeasibility during the course of a cutting plane approach. One way to preserve feasibility is to choose *box* constraints that correspond to cuts in the max cut problem.

We have also been working on cutting plane approaches (Krishnan and Mitchell [33], [34]) to produce a sequence of LP 's whose solutions converge to the SDP objective value. These cutting plane approaches arise naturally out of *exchange* schemes developed for semi-infinite programming (Hettich and Kortanek [25]). To make such a cutting plane algorithm competitive with the bundle LP approach several refinements are required. The cutting plane approach uses an interior point algorithm to solve the LP relaxations approximately, because this results in better constraints than a simplex cutting plane method. However the bundle LP approach is superior to these cutting plane approaches, since not only do we get better objective values, but the resulting relaxations are smaller as well. One way to explain this, is that Theorem 4 due to Pataki [39], which suggests that there exists a degenerate LP discretization that exactly captures the SDP objective value. The cutting plane approaches rely on *nondegeneracy* for convergence, besides nondegeneracy is also a *generic* property for the LP , i.e. it occurs almost

everywhere (Alizadeh et al [2]).

To conclude it is felt that a beginning is made to solve an *SDP* with a bounded feasible set as an *LP*. We provide empirical evidence that only a few constraints, bounded by the square root of the number of constraints in the *SDP* are typically required. This potentially allows us to approximately solve large scale *SDP*'s using the state of the art linear solvers that are readily available.

8 Acknowledgements

The authors wish to thank Christoph Helmberg for making his spectral bundle code available to us and for useful discussions.

References

- [1] F. ALIZADEH, *Interior Point Methods in Semidefinite Programming with applications to Combinatorial Optimization*, SIAM Journal on Optimization, 5 (1995), pp. 13-51.
- [2] F. ALIZADEH, J.-P.A. HAEBERLY AND M.L. OVERTON, *Complementarity and Nondegeneracy in Semidefinite Programming*, Mathematical Programming, 77(1997), pp. 111-128.
- [3] E.J. ANDERSON AND P. NASH, *Linear Programming in Infinite-Dimensional Spaces*, Wiley, New York, 1987.
- [4] S.J. BENSON, Y. YE AND X. ZHANG, *Solving Large-Scale Semidefinite Programs for Combinatorial Optimization*, SIAM Journal on Optimization, 10 (2000), pp. 443-461.
- [5] A. BEN-TAL, A. BEN-ISREAL, AND E. ROSINGER, *A Helly type theorem and semi-infinite programming*. In C.V. Coffman and G.J. Fix, editors, *Constructive Approaches to Mathematical models*, pages 127-135. Academic Press, New York 1979.
- [6] A. BEN-TAL AND A. NEMIROVSKI, *On polyhedral approximations of the second order cone*, Technical report, Minerva Optimization Center, Faculty of Industrial Engineering and Management, Technion, Haifa, Isreal, 1998.

- [7] J.F. BONNANS AND A. SHAPIRO, *Perturbation Analysis of Optimization Problems*, Springer-Verlag New York, 2001.
- [8] J.M. BORWEIN, *Direct theorems in semi-infinite convex programming*, Mathematical Programming, 21 (1981), pp. 301-318
- [9] B. BORCHERS, *SDPLIB 1.2, A Library for Semidefinite Programming Test Problems*, Optimization Methods and Software 11(1) (1999), pp. 683-690.
- [10] S. BURER AND R.D.C. MONTEIRO, *A nonlinear programming algorithm for solving semidefinite programs via low-rank factorization*, Working paper, School of ISyE, Georgia Tech., Atlanta, GA, March 2001.
- [11] S. BURER, R.D.C. MONTEIRO AND Y. ZHANG, *Solving Semidefinite Programs via Nonlinear Programming. Part I : Transformations and Derivatives*, Working paper, School of ISyE, Georgia Tech., Atlanta, GA, September 1999.
- [12] S. BURER, R.D.C. MONTEIRO AND Y. ZHANG, *Solving Semidefinite Programs via Nonlinear Programming. Part II : Interior point Methods for a Subclass of SDP's*, Working paper, School of ISyE, Georgia Tech., Atlanta, GA, October 1999.
- [13] S. BURER, R.D.C. MONTEIRO AND Y. ZHANG, *Rank-Two Relaxation Heuristics for Max-Cut and Other Binary Quadratic Programs*, SIAM Journal on Optimization, 12(2001), pp. 503-521.
- [14] S. BURER, R.D.C. MONTEIRO AND Y. ZHANG, *Maximum stable set formulations and heuristics based on continuous optimization*, Working paper, School of ISyE, Georgia Tech., Atlanta, GA, December 2000.
- [15] A.R. CONN, N.I.M. GOULD AND P.L. TOINT, *Trust-Region Methods*, MPS-SIAM Series on Optimization, 2000.
- [16] A. FRIEZE AND M.R. JERRUM, *Improved approximation algorithms for Max k-cut and Max Bisection*, Algorithmica 18, pp. 61-77.
- [17] K. GLASHOFF AND S.A. GUSTAFSON, *Linear Optimization and Approximation*, Springer Verlag, New York, 1983.

- [18] F. GLINEUR, *Computational experiments with a linear approximation of second order cone optimization*, Technical Report, Faculte Polytechnique de Mons, Belgium, November 2000.
- [19] M. GOEMANS AND D.P. WILLIAMSON, *Improved approximation algorithms for max cut and satisfiability problems using semidefinite programming*, J. A.C.M 42 (1995), pp. 1115-1145.
- [20] C. HELMBERG, *Semidefinite Programming for Combinatorial Optimization*, ZIB-Report ZR-00-34, Konrad-Zuse-Zentrum Berlin, October 2000.
- [21] C. HELMBERG AND F. RENDL, *A Spectral Bundle Method for Semidefinite Programming*, SIAM Journal on Optimization, 10 (2000), pp. 673-696.
- [22] C. HELMBERG AND K.C. KIWIEL, *A Spectral Bundle Method with bounds*, ZIB Preprint SC-99-37, Konrad-Zuse-Zentrum Berlin, December 1999.
- [23] C. HELMBERG AND F. OUSTRY, *Bundle methods to minimize the maximum eigenvalue function*. In : H. Wolkowicz, R. Saigal, and L. Vandenberghe, eds., *Handbook of Semidefinite Programming*, Kluwer Academic Publishers, Boston-Dordrecht-London, 2000, pp. 307-337.
- [24] C. HELMBERG, *SBmethod : A C++ Implementation of the Spectral Bundle Method*, ZIB-Report ZR-00-35, Konrad -Zuse-Zentrum Berlin, October 2000.
- [25] R. HETTICH AND K.O. KORTANEK, *Semi-Infinite Programming : Theory, Methods, and Applications*, SIAM Review 35(3) (1993), pp. 380-429.
- [26] R. HORN AND C. JOHNSON, *Matrix Analysis*, Cambridge University Press, 1990.
- [27] J.B. HIRIART URRUTY AND C. LEMARECHAL, *Convex Analysis and Minimization Algorithms I*, Springer Verlag, Berlin, Heidelberg, 1993.
- [28] J.B. HIRIART URRUTY AND C. LEMARECHAL, *Convex Analysis and Minimization Algorithms II*, Springer Verlag, Berlin Heidelberg, 1993.

- [29] ILOG CPLEX 6.5, *ILOG Inc.*, 1080 Linda Vista Avenue, Mountain View, CA 94043.
- [30] K.C. KIWIEL, *An aggregate subgradient method for nonsmooth convex minimization*, *Mathematical Programming* 27(1983), pp. 320-341.
- [31] K.C. KIWIEL, *Methods of Descent for Nondifferentiable Optimization*, *Lecture Notes in Mathematics*, vol. 1133, Springer, Berlin, Heidelberg (1985).
- [32] K.C. KIWIEL, *Proximity Control in bundle methods for convex nondifferentiable minimization*, *Mathematical Programming* 46 (1990), pp. 105-122.
- [33] K. KRISHNAN AND J.E. MITCHELL, *A linear programming approach to semidefinite programming problems*, Technical report, Dept. of Mathematical Sciences, RPI, Troy, NY, 12180, April 2001.
- [34] K. KRISHNAN AND J.E. MITCHELL, *A cutting plane LP approach to solving semidefinite programming problems*, Dept. of Mathematical Sciences, RPI, Troy, NY, in preparation.
- [35] C. LEMARECHAL, *Nondifferentiable Optimization*. In : G.L. Nemhauser, A.H.G. Rinnooy Kan, and M.J. Todd, eds., *Optimization*, North-Holland, Amsterdam-New York-Oxford-Tokyo, 1989, pp. 529-572.
- [36] A.S. LEWIS AND M.L. OVERTON, *Eigenvalue Optimization*, *Acta Numerica* 5(1996), pp. 149-190.
- [37] Y. NESTEROV AND A. NEMIROVSKII, *Interior Point Polynomial Algorithms in Convex Programming*, in *SIAM Studies in Applied Mathematics*, Philadelphia, PA, 1993.
- [38] F. OUSTRY, *A second order bundle method to minimize the maximum eigenvalue function*, *Mathematical Programming* 89 (2000), pp. 1-33.
- [39] G. PATAKI, *On the Rank of Extreme Matrices in Semidefinite Programs and the Multiplicity of Optimal Eigenvalues*, *Mathematics of Operations Research* 23(2), pp. 339-358.

- [40] G. PATAKI AND S. SCHMIETA, *The DIMACS library of semidefinite -quadratic-linear programs*, Computational Optimization Research, Columbia University, NY, November 1999.
- [41] H. SCHRAMM AND J. ZOWE, *A version of the bundle idea for minimizing a nonsmooth function : Conceptual idea, convergence analysis, numerical results*, SIAM Journal on Optimization 2 (1992), pp. 121-152.
- [42] M.J. TODD, *Semidefinite Optimization*, Acta Numerica, 10(2001), pp. 515-560.
- [43] L. VANDENBERGHE AND S. BOYD, *Semidefinite Programming*, SIAM Review, 38(1996), pp. 49-95.
- [44] R. VANDERBEI AND H. BENSON, *On Formulating Semidefinite Programming Problems as Smooth Convex Nonlinear Optimization Problems*, ORFE-99-01, Princeton University, NJ, January 2000.
- [45] H. WOLKOWICZ, R. SAIGAL AND L. VANDENBERGHE, *Handbook on Semidefinite Programming*, Kluwer Academic Publishers, 2000.
- [46] Y. YE, *A .699 Approximation Algorithm for Max Bisection*, Mathematical Programming, 90(1) (2001), pp. 101-111.