Mathematical Programming Computation
https://doi.org/10.1007/s12532-025-00275-1

FULL LENGTH PAPER

Instance-specific linear relaxations of semidefinite optimization problems

Daniel de Roux[1] · **Robert Carr**[2] · **R. Ravi**[1]

Received: 31 March 2023 / Accepted: 21 November 2024

Abstract

We introduce a generic technique to obtain linear relaxations of semidefinite programs with provable guarantees based on the commutativity of the constraint and the objective matrices. We study conditions under which the optimal value of the SDP and the proposed linear relaxation match, which we then relax to provide a flexible methodology to derive effective linear relaxations. We specialize these results to provide linear programs that approximate well-known semidefinite programs for the max cut problem proposed by Poljak and Rendl, and the Lovász theta number; we prove that the linear program proposed for max cut certifies a known eigenvalue bound for the maximum cut value and is in fact stronger. Our ideas can be used to warm-start algorithms that solve semidefinite programs by iterative polyhedral approximation of the feasible region. We verify this capability through multiple experiments on the max cut semidefinite program, the Lovász theta number and on three families of semidefinite programs obtained as convex relaxations of certain quadratically constrained quadratic problems.

Keywords Semidefinite programming · Linear relaxation

Mathematics Subject Classification 90C22 · 90C05 · 05C85

Daniel de Roux is now at Google, Mountain View, CA, USA.

✉ R. Ravi
ravi@andrew.cmu.edu

Daniel de Roux
dderoux@google.com

Robert Carr
bobcarr@swcp.com

[1] Tepper School of Business, Carnegie Mellon University, Pittsburgh, USA

[2] University of New Mexico, Albuquerque, USA

Published online: 06 March 2025

1 Introduction

Semidefinite optimization, i.e., the optimization of a linear function over the set of positive semidefinite matrices intersected with an affine subspace [71], is one of the most active research areas in convex mathematical optimization. The generic formulation for a semidefinite optimization problem (SDP) is

$$\begin{aligned} \min_{X \in \mathbb{S}^n} \ & \langle C, X \rangle \\ \text{s.t: } & \langle A_i, X \rangle = b_i, \ \forall i \in [r], \\ & X \succeq 0 \end{aligned} \tag{SDP}$$

where $\mathbb{S}^n$ denotes the set $n \times n$ symmetric matrices, $C \in \mathbb{S}^n$ is a symmetric (without loss of generality) cost matrix, $\langle \cdot, \cdot \rangle$ denotes the Frobenius inner product, $r \in \mathbb{N}$, $[r]$ denotes the set of integers $\{1, \ldots, r\}$ and a symmetric $n \times n$ matrix X is *positive semidefinite*, denoted $X \succeq 0$, if and only if $v^\top X v \geq 0$ for all $v \in \mathbb{R}^n$.

SDPs arise naturally in combinatorial optimization [4, 33, 46, 54], control theory [1, 37, 57], polynomial optimization [44, 57, 58], machine learning [22, 43] and are solvable in polynomial time up to an arbitrary accuracy via the theory of interior-point methods [53]. Nonetheless, it is well known that SDPs are challenging to solve in practice. Typical off-the-shelf solvers use interior-point methods, which require computation of large Hessian matrices (and their inverses) and are often intractable due to memory limitations. For an illustration, see [11, chapter 6.7], [50], and [12] where it is mentioned that state-of-the-art solver such as MOSEK [6] cannot solve semidefinite problems with a symmetric matrix X on more than 250 rows. Inspired by these practical limitations, researchers have proposed several ideas to solve large-scale semidefinite programs. Among them, we have i) exploiting structure of the problem (such as sparsity and symmetry), ii) producing low rank solutions, iii) algorithms based on augmented Lagrangians and the alternating direction method of multipliers, and iv) approaches that trade scalability with conservatism by iteratively finding inner and outer polyhedral approximations of the semidefinite problem. See [49] for a survey of all of these methods.

Although all of these techniques enjoy a rich literature, the algorithms of iv) are of special theoretical interest. The cornerstone of these methods is constructing inner and outer polyhedral approximations of the semidefinite cone in order to find a sequence of improving feasible solutions together with tighter bounds on the objective of the SDP, allowing one to trade off between scalability and conservatism.

Research on this class of algorithms is relevant due to its intimate connection with a fundamental question in convex geometry: can the positive semidefinite cone be approximated by polyhedra? Taking the perspective of the field of optimization, this question can be framed by asking if linear programs are strong enough to approximate semidefinite ones. These twin questions, relevant in the fields of optimization and convex geometry respectively, have given rise to a thriving body of research [2, 12, 18, 26].

Outer approximations have been the focus of substantial effort since the hardness of SDP comes from the semidefinite constraint and so one may drop it and and add linear constraints on X implied by $X \succeq 0$. In this case, (SDP) is relaxed to a linear program. A typical example is to add the constraints $X_{i,i} \geq 0,\ \forall i \in [n]$ and $X_{ii} + X_{jj} \pm 2X_{i,j} \geq 0,\ \forall i \in [n],\ \forall j \in [n]$ which are valid for any $X \succeq 0$. These relaxations tend to be weak and seldom used in practice [14, 17]. A well studied example of this phenomenon is the maximum cut problem and the theoretical hardness of approximating it with linear programs which we will discuss in depth in Sect. 3.

The previous approach can be improved using ideas of Kelley [40]. The strategy is to sequentially refine the linear relaxations by aggregation of cutting planes. More concretely, consider the linear relaxation of SDP given by

$$
\begin{aligned}
\min_{X \in \mathbb{S}^n} \ & \langle C, X \rangle \\
\text{s.t: } & \langle A_i, X \rangle = b_i,\ \ \forall i \in [r], \\
& v^\top X v \geq 0 \ \forall\ v \in \mathscr{S}
\end{aligned}
\tag{$L_{\mathscr{S}}$}
$$

where $\mathscr{S}$ is a finite subset of $\mathbb{R}^n$. Here, we simply insist that $v^\top X v \geq 0$ only for the elements v of the set $\mathscr{S}$. If a solution to this program is not positive semidefinite, we may update $\mathscr{S}$ iteratively. This results in the following algorithm:

Algorithm 1

Fix a finite set $\mathscr{S} \subseteq \mathbb{R}^n$. Drop the semidefinite constraint $X \succeq 0$ of program SDP and solve the resulting linear program $L_{\mathscr{S}}$ finding a minimizer X^*.
while X^* has a negative eigenvalue **do**
    Find a eigenvector v corresponding to the most negative eigenvalue of X^*. Add v to $\mathscr{S}$.
    Solve the updated linear program to find a new minimizer X^*.
end while
return X^*.

The specific implementations of this algorithm mainly differ in how one updates the set $\mathscr{S}$. In [2, 3],the authors use the extreme rays of the set of diagonally dominant matrices, which are then rotated by matrices obtained from a Cholesky decomposition of an optimal solution to the dual of ($L_{\mathscr{S}}$). They also propose an inner approximation of the positive semidefinite cone based on the so-called $DSOS_n$ and $SDSOS_{n,d}$ cones. In a different line of work [7, 24, 62, 66, 76] chose the elements v of $\mathscr{S}$ favoring sparsity, with the idea that the resulting linear programs will be easier to solve. Bundle methods, such as the *spectral bundle method* of Helmberg and Rendl [35] work with the dual of (SDP), under the further restriction that X has a constant trace. [42] presents a unifying framework for the latter and similar methods. In [12], the constraint $X \succeq 0$ is replaced for infinitely many constraints of the form $f(X, Y) \leq 0$ which must hold for every Y in some convex set $\mathscr{Y}$ and where f is a Lipschitz continuous function. The authors further argue that one should instead solve a second-order cone relaxation, adding the constraints

$$\left\| \begin{pmatrix} 2X_{i,j} \\ X_{i,i} - X_{j,j} \end{pmatrix} \right\|_2 \leq X_{i,i} + X_{j,j}, \ \forall i \in [n], \ \forall j \in [n],$$

which are valid for (SDP).

It is noteworthy that mostly all of these works discuss how to update $\mathscr{S}$, but seldom consider how to initialize it. Typically $\mathscr{S}$ is set to the standard basis of $\mathbb{R}^n$, resulting in the linear constraints $X_{ii} \geq 0, \ i \in \{1, \ldots, n\}$, which are implied by the constraint $X \succeq 0$. Interestingly, under mild conditions, there exists a finite set $\mathscr{S}$ that ensures that the optimal values of the SDP and the linear relaxation $L_{\mathscr{S}}$ match, supporting the approach of using Algorithm 1.

Observation 1 *Suppose that both (SDP) and its dual, given by the following semi-definite optimization program*

$$\begin{aligned} \max_{y \in \mathbb{R}^r} \ & b^\top y \\ s.t{:}\ & C - \sum_{i=1}^{r} y_i A_i \succeq 0 \end{aligned} \tag{DSDP}$$

are strictly feasible. Let y^* be an optimal solution to (DSDP). Let $v_1, \ldots, v_n$ be an orthonormal basis of $\mathbb{R}^n$ of eigenvectors of $C - \sum_{i=1}^m y_i^* A_i = S^*$ with $S^* = \sum_{i=1}^n \beta_i v v^\top$, and β_i the eigenvalues of S^*. Let $\mathscr{S}^* = \{v_1, \ldots, v_n\}$. Then, $L_{\mathscr{S}^*}$ is solvable, and its optimal value matches the optimal value of (SDP).

The proof of this observation is deferred to the Appendix A. Similar versions of Observation 1 can be found in [42] and [68]. In fact, [68] proves that if $\mathscr{S} = \{u_1, \ldots, u_l\}$ are the vectors generated by the spectral bundle method of [35] of Rendl et al. to solve DSDP, the objective value of $(L_{\mathscr{S}})$ matches that of (SDP), but this is hardly surprising: if we knew in advance the set of vectors $\mathscr{S}^*$ given by Observation 1, we could set $\mathscr{S} = \mathscr{S}^*$ and solve (SDP) as a linear program. More importantly, we emphasize that finding the sets $\mathscr{S}^*$ and $\{u_1, \ldots, u_l\}$ requires solving another comparable SDP, namely DSDP.

In this paper, we tackle the task of finding a better set $\mathscr{S}$ to initialize Algorithm 1 under certain computational restrictions by drawing inspiration from the question of when - if ever- one can avoid the iterative procedure suggested by Kelley and exactly solve the semidefinite program with a linear program. By “exactly solving" we mean finding a linear relaxation of the SDP whose optimal value equals that of the SDP.

Technically, Observation 1 indicates that the question of *exactly* solving an SDP with a linear problem is ill-posed if one does not restrict the set of *algorithms* one is allowed to use to process the instance. We can consider at least three possible approaches to amend this issue. First, restricting the access one has to the given instance. For example, say we are not shown a full SDP instance, but one is allowed to sample a small subset of the entries of the objective and constraints matrices. Second, to only have access to algorithms with at most a certain computational complexity, say matrix multiplication complexity. However, this would require fixing a concrete computational model and proving lower bounds for the complexity of the algorithms to be used, which are typically very hard to obtain. A third approach, which we take in this paper, is to fix an oracle $\mathscr{O}$, that we can query at most a constant number of times. Concretely, we will assume that we have at our disposal an oracle that can compute a eigenvector decomposition of a symmetric matrix, and that can solve linear programs of polynomial size. If the SDP can in fact be solved with such an oracle, we say it is *solvable under* $\mathscr{O}$.

1.1 Hardness of approximation of the max cut problem

The question of finding a good set $\mathscr{S}$ to initialize Algorithm 1 amounts to finding a linear approximations to a semidefinite programs together with a guarantee that the approximation is good. This line of research is motivated by the question of whether the maximum cut (max cut henceforth) problem can be approximated using a linear program by a factor strictly better than 2. This problem consists in finding a bipartition of the nodes of a given graph that maximizes the number of edges with one end in both parts. The results of Poljak, Rendl, Goemans and Williamson [33, 60] show that max cut can be approximated to within a factor of 1.13 by an SDP relaxation. Therefore, a linear approximation of factor at most 1.769 to that SDP would result in a linear approximation the the max cut problem with an approximation better than 2.[1] Such a result would be striking as the common belief is that max cut cannot be approximated within a factor better than 2 with a linear program in the restricted case that the feasible region of the program is independent of the graph and solely depends on the number of vertices [14, 17, 18, 41, 72]. In Sect. 3, we explore in detail the hardness of approximation results for max cut.

Drawing inspiration from the study of exact solvability of an SDP with an LP, we make the case that we can obtain "good starting" linear approximations for semidefinite programs if one is allowed to let $\mathscr{S}$ depend on the dual of the semidefinite program. The heart of the argument is that the obstructions mentioned for max cut emerge specifically when the polytopes being optimized are determined solely by the number of variables (node pairs for max cut) in a given instance. Hence, we propose to let $\mathscr{S}$ depend on the matrices C and $A_1, \ldots, A_r$ which determine the objective and the constraints of (SDP), and consequently on the feasible region of DSDP. Crucially, such formulations trivially avoid the results in [14] and [41]. We call linear approximations with such dependence "instance-specific". Notice that making some assumption on the algorithms that we can use to interact with the instance is essential here. To illustrate this point, imagine we wish to write a linear program to find the max cut value $mc(G)$ of a graph G. To do so, we can compute a max cut of the graph using brute force and then write an LP with a linear constraint insisting that the objective equals $mc(G)$.

1.2 Exact linear relaxations under $\mathscr{O}$

To find candidate sets $\mathscr{S}$ that guarantee that the linear program $L_{\mathscr{S}}$ is a strong relaxation of SDP we first explore sufficient conditions under which the SDP is solvable

[1] Since $1.77 \cdot 1.13 = 2$.

under the oracle $\mathscr{O}$. Although Observation 1 suggests an answer, such a set of vectors cannot, as far as we are aware, be obtained with the oracles we are considering. In Section 2, we present Theorems 1 and 2 which will provide solvability under $\mathscr{O}$ without requiring the solution of a semidefinite program. Our results are tied to the geometry of the dual feasible region of SDP, and a relevant case is when the dual feasible region is a polyhedron. If such is the case and an explicit description of it is available, then program DSDP can be solved as a linear program. Theorem 1 shows that under the same condition the *primal* SDP can be solved with a linear program as well. Unfortunately, this theorem is not very useful as it requires enumerating the vertices of the feasible region, which may grow exponentially. The polyhedral assumption has received attention from the literature in the context of *quadratically constrained quadratic problems* (QCQPs) [74], and perhaps more so a weakening of it: simultaneous diagonalizability.

Definition 1 A set of matrices $\{A_i\}_{i\in I} \subseteq \mathbb{R}^{n\times n}$ where I is some set of indices which may be infinite, is said to be simultaneously diagonalizable (SD) if there exists an invertible, orthogonal matrix $U \in \mathbb{R}^n$ such that every element of the set $\{U^\top A_i U\}_{i\in I}$ is a diagonal matrix. Note that $U^\top U = UU^\top = I_n$ as U is orthogonal.

It turns out that if the set of matrices defining the dual feasible region Γ of SDP is simultaneously diagonalizable, then Γ is a polyhedron [74].

A set of matrices $\{A_i\}_{i\in I} \subseteq \mathbb{R}^{n\times n}$ where I is some set of indices, is said to be simultaneously diagonalizable if there exists an invertible, orthogonal matrix $U \in \mathbb{R}^n$ such that every element of the set $\{U^\top A_i U\}_{i\in I}$ is a diagonal matrix.

$\{C, A_1, \ldots, A_r\}$ is simultaneously diagonalizable if there exists U orthogonal s.t.

$$U^\top CU,\ U^\top A_1 U, \ldots,\ U^\top A_r U$$

are all diagonal.

Observation 2 *Let Γ be a spectrahedron given by the representation $\Gamma = \{y \in \mathbb{R}^n : C - \sum_i^r A_i y_i) \succeq 0\}$. If the set of matrices $\{C, \{A_i\}_{i\in[r]}\}$ is simultaneously diagonalizable, then Γ is polyhedral.*

We prove this fact in Sect. 2, and point out that the given condition is sufficient but not necessary. Under this more stringent condition, we prove in Theorem 2 that $\mathscr{O}$ can be used to solve SDP.

It will typically not be the case that the dual feasible set Γ is polyhedral, and much less that the matrices $C, \{A_i\}_{i\in[r]}$ are simultaneously diagonalizable. In Sect. 2 we prove that this condition is equivalent to the simultaneous diagonalizability of matrices $C - \sum_i A_i p_i$ and $C - \sum_i A_i q_i$ for all p and q in $\mathbb{R}^r$. This characterization suggests that we only insist of the commutativity of the matrices $C - \sum_i A_i p_i$ and $C - \sum_i A_i q_i$ for some p and q. It turns out that this is the key idea to initialize the set $\mathscr{S}$ in Algorithm 1. In Sect. 2 we set the theoretical background of these considerations, and in the following sections we explore their applications to three families of semidefinite optimization problems: the max cut problem, The Lovász theta number and the more generic Shor SDP relaxation of quadratically constrained quadratic problems.

We stress that the intention of the presented approach is to further explore when an SDP can be solved with a linear program, and to improve on existing cutting plane approaches to solve SDPs (such as the conservative methods described in [49]). This family of methods is *not* the de-facto choice to solve large scale semidefinite programs, and very strong methods exist which can scale substantially such as [56, 75, 77, 79, 81]. Nevertheless, We point out that these methods might come with their own limitations and in settings where SDPs appear naturally, such as in the sum-of-squares hierarchy for polynomial optimization [78], or whenever optimal solutions to the SDPs are not low rank. In these regimes, polyhedral approximations might be a good alternative. In addition, developing stronger polyhedral approximations to SDPs has consequences in approaches to integer semidefinite programs, which has received attention recently [20, 30, 31, 38, 80] and in spatial branch-and-bound algorithms for non-convex quadratic problems.

1.3 Overview and outline of this paper

(a) In Sect. 2 we derive two sufficient conditions for solvability of an SDP under $\mathscr{O}$. These conditions are then weakened to produce a strategy to provide candidate starting sets $\mathscr{S}_0$ for outer polyhedral approximation algorithms to solve SDPs.
(b) In Sect. 3, we study the setting of finding a maximum cut of a graph G using the semidefinite relaxation of Poljak, Rendel, Goemans and Williamson [33, 60]. Even though the conditions for exact solvability are not met, we use the relaxed version to provide a linear program that certifies a spectral bound in contrast to previous linear relaxations for the maximum cut problem. We then derive a solvability result under $\mathscr{O}$, recovering and generalizing a theorem of Alon and Sudakov [5].
(c) In Sect. 4 we introduce linear relaxations of the Lovász theta number SDP and Shor's semidefinite relaxation for quadratically constrained quadratic programs. We recall as well our linear relaxation of max cut, and introduce a linear strengthening of the max cut SDP.
(d) In Sect. 5 we extensively test our methods empirically on random instances of the problems introduced in Sect. 4. We discuss solving times of the proposed programs.
(e) In Appendix A, we prove Observation 1 and provide an alternative proof of Lemma 7. In Appendix B,[2] we show the performance of our linear program in the case where the original SDP is itself a relaxation of an underlying optimization problem. We study the case of the max cut problem and the *sparse PCA* problem, where both the SDPs and our linear relaxations can be used to recover a solution to the underlying problem. We show that the quality of our linear programs is competitive with that of the SDPs. For max cut, we compare with results obtained by Mirka and Williamson in [51]. In Appendix C,[3] we include additional figures for the experiments in Sect. 5.

[2] Appendix B is included in the online companion of this paper.

[3] Appendix C is included in the online companion of this paper.

1.4 Notation

We denote the set of square, real, $n \times n$ symmetric matrices by $\mathbb{S}^n$. We denote the cardinality of a set I by $|I|$.

We denote by $e_1, \ldots, e_n$ the standard basis of $\mathbb{R}^n$ and the $n \times n$ identity matrix by I_n. For a symmetric matrix W we let $\lambda_1(W) \geq \lambda_2(W) \geq \cdots \geq \lambda_n(\mathrm{W})$ be its eigenvalues. When the matrix is clear from the context, we drop the terms in parentheses and simply write $\lambda_1 \geq \cdots \geq \lambda_n$. For $A \in \mathbb{S}^n$ we write $tr(A)$ for the trace of A: $tr(A) = \sum_{i=1}^n A_{ii}$ and write $\|A\|_F$ to denote the Frobenius norm of A: $\|A\|_F = \sqrt{\sum_{i=1}^n \sum_{j=1}^n A_{ij}^2}$. The ℓ_1 norm of A is given by $\|A\|_1 = \sum_{i,j} |A_{ij}|$. We denote by $\langle \cdot, \cdot \rangle$ the usual Frobenius inner product of two matrices in $\mathbb{S}^n$, recalling that for two matrices $A, B \in \mathbb{S}^n$, $\langle A, B \rangle = tr(A^T B) = tr(AB)$. We denote by $\vec{1}$ the vector of all ones in $\mathbb{R}^n$ and by J the matrix of all ones. If A is a matrix, we denote by $diag(A)$ the vector given be the diagonal of A. If u is a vector, $diag(u)$ denotes the matrix with u on its diagonal. We denote by $\mathscr{E}(A)$ an arbitrary orthonormal basis consisting of eigenvectors of A. In particular, if $A \in \mathbb{S}^n$ and $\mathscr{E}(A) = \{v_1, \ldots, v_n\}$ then we have $A = \sum_{i=1}^n \lambda_i v_i v_i^\top$ [39]. Finally, given a weighted graph G we denote, respectively, the value of the maximum cut of G, the adjacency matrix, the number of edges and the laplacian matrix by $mc(G)$, $W(G)$, m and $\mathscr{L}(G)$. If G is clear from the context, we drop the dependency on G and simply write mc, W, m and $\mathscr{L}$.

2 Instance-specific linear relaxations of semidefinite optimization problems

In this section we explore the question of exact solvability of semidefinite programs given access to an oracle $\mathscr{O}$, with the following properties:

- Given a set of simultaneously diagonalizable matrices $\{A_1, \ldots, A_r\}$, $\mathscr{O}$ can be called once to compute an orthogonal matrix U such that $U^\top A_i U$ are diagonal matrices for $i = 1, \ldots r$. For an implementation of such an oracle see [34].
- $\mathscr{O}$ can be called a constant number of times to find an optimal solution to a linear program of polynomial size in the bit representation of the information of the SDP, namely the objective and constraint matrices.

In case we can find the optimal value of program SDP by querying $\mathscr{O}$ at most a constant number of times, we say that the SDP is solvable under $\mathscr{O}$, and our intention is to derive sufficient conditions that guarantee solvability of the SDP. It is to be expected that such conditions are not applicable except in some rare cases. We posit that we can derive weakenings of them to provide a starting set $\mathscr{S}$ for Algorithm 1. Recall that a generic SDP is given by

$$\begin{aligned} \min_{X \in \mathbb{S}^n} \; & \langle C, X \rangle \\ \text{s.t: } & \langle A_i, X \rangle = b_i, \;\; \forall i \in \{1, \ldots, r\}, \\ & X \succeq 0. \end{aligned} \tag{SDP}$$

The dual of this program is:

$$\max_{y \in \mathbb{R}^n} \; b^\top y$$
$$\text{s.t: } C - \sum_{i=1}^{m} y_i A_i \succeq 0. \tag{DSDP}$$

Throughout this paper, we will assume "generic SDPs" and their duals are strictly feasible, and therefore strong duality holds. A spectrahedron Γ is the intersection of the cone of positive semidefinite matrices and an affine subspace. If we identify the affine subspace with $\mathbb{R}^r$ then we can write Γ as:

$$\Gamma = \{y \in \mathbb{R}^r : y_1 A_1 + \cdots + y_r A_r + A_{r+1} \succeq 0\}$$

where $A_1, \ldots, A_r, \; A_{r+1}$ are symmetric $n \times n$ matrices. In general, the map $\mathscr{A} : \mathbb{R}^r \to \mathbb{S}^n$ given by $\mathscr{A}(y) = y_1 A_1 + \cdots + y_r A_r + A_{r+1}$ is called an affine symmetric matrix map. Through duality, one can see that spectrahedrons are to semidefinite programs what polyhedra are to linear programs [73]. It is clear that whenever Γ is polytope and we have an explicit representation of it given by a system of linear equations $Ax \leq d$, then program DSDP reduces to a linear program. More interestingly perhaps is that the primal problem SDP can also be solved as a linear program, albeit on potentially an exponential number of constraints.

Theorem 1 *Consider a generic semidefinite optimization problem SDP, with dual given by DSDP. Suppose that the set*

$$\Gamma = C - \sum_{i=1}^{r} y_i A_i \succeq 0$$

is a polytope with extreme points $p_1 \ldots, p_k$, and define $\mathscr{S} := \bigcup_{i=1}^{k} \mathscr{E}(C - \mathscr{A}(p_k))$. Then, $L_{\mathscr{S}}$ is a linear program and solves SDP.

Proof The maximum value of the function $b^\top y$ over Γ is achieved at some vertex p of Γ. By strong duality and the solvability of DSDP, there exists some $X^* \succeq 0$ which solves program SDP. In particular, program $L_{\mathscr{S}}$ with $\mathscr{S} := \bigcup_{i=1}^{k} \mathscr{E}(C - \mathscr{A}(p_k))$ where $p_1, \ldots, p_k$ are the vertices of Γ is feasible. Let $\hat{X}$ be an optimal solution to this program. Let $\{v_1 \ldots v_n\} \subseteq \mathscr{S}$ be an orthonormal eigenbasis for the matrix $C - \mathscr{A}(p)$. Since this matrix is positive semidefinite, we can write $C - \mathscr{A}(p) = \sum_{i=1}^{n} \beta_i v_i v_i^\top$ where the $\beta_i, i \in [n]$ are the (non-negative) eigenvalues of $C - \mathscr{A}(p)$. By feasibility of $\hat{X}$, $v_i^\top \hat{X} v_i \geq 0$ for all $i \in [n]$. Multiplying each term by $\beta_i \geq 0$ we derive

$$\sum_{i=1}^{n} \beta_i \left\langle \hat{X}, v_i v_i^T \right\rangle = \left\langle \hat{X}, \sum_{i=1}^{n} \beta_i v_i v_i^T \right\rangle = \left\langle \hat{X}, C - \mathscr{A}(p) \right\rangle \geq 0.$$

This implies that $\left\langle \hat{X}, C \right\rangle \geq \left\langle \hat{X}, \mathscr{A}(p) \right\rangle$.

To conclude, recall that for $j \in [r]$, $\langle X, A_j \rangle = b_j$ giving the inequality $\left\langle \hat{X}, C \right\rangle \geq b^\top p$. Again by strong duality and since the LP is a relaxation of the SDP, we have $b^\top p = \langle C, X^* \rangle \geq \left\langle C, \hat{X} \right\rangle$ yielding the desired equality $b^\top p = \langle C, X^* \rangle$. □

In [63] is its shown that deciding if a spectrahedron is a polyhedron is in co-NP, and an algorithm for deciding polyhedrality is given. [13] generalizes and improves the previous results. The algorithm presented in the latter paper runs in exponential time, as it requires enumerating the vertices of a certain polyhedron. Even if we knew that Γ is polyhedral, we do not have exact solvability under $\mathscr{O}$, as the previous problem has an exponential number of constraints. A particular case in which Γ is polyhedral and that has received attention in the literature is whenever the matrices C and $A_i,\ i \in [r]$ are simultaneously diagonalizable. This is the content of observation 2, which we now prove.

Proof (Proof of Observation 2. Also see [74], Lemma 9)

Let U be a matrix that simultaneously diagonalizes matrices C and $A_i,\ i \in [r]$ i.e. the matrices $C' = U^\top C U$ and $A_i' = U^\top A_i U$ are all diagonal. By Silvester's law of inertia [39], we have that $C - \mathscr{A}(y) \succeq 0$ if and only if $U^\top [C - \mathscr{A}(y)]\, U \succeq 0$ if and only if $C' - \sum_{i=1}^r y_i A_i' \succeq 0$. Hence, we have

$$\Gamma = \{y \in \mathbb{R}^r : C' - \sum_{i=1}^r y_i A_i' \succeq 0\}$$

which is a polyhedral set since all matrices involved are diagonal. □

For a clear exposition of the implications of this observation to QCQPs see [74] and the references therein. In addition, the authors show that the region Γ might be polyhedral even if the matrices C and $\{A_i\}_{\ i \in [r]}$ are not simultaneously diagonalizable. Although the latter condition is much more stringent, it allows us to avoid the need to have the vertices of Γ given to us explicitly, as Theorem 1 requires.

Theorem 2 *Let SDP be a semidefinite program with dual DSDP. Suppose that the set of matrices $\{C, A_1, \ldots, A_r\}$ is simultaneously diagonalizable. Then, SDP is solvable under $\mathscr{O}$.*

Proof Let U be a orthogonal matrix that simultaneously diagonalizes C and A_i for each $i \in [r]$. Let $v_1, \ldots . v_n$ denote the columns of U and set $\mathscr{S} = \{v_1, \ldots, v_n\}$. Let p^* be a dual optimal solution with $C - \mathscr{A}(p^*) = S^*$ where S^* is positive semidefinite. Since U diagonalizes each $A_i,\ i \in [r]$ and C, it is clear that the matrix $U^\top [C - \mathscr{A}(p)]\, U$ is diagonal. In other words, the matrix $U^\top S^* U = D$ for some diagonal matrix D with non-negative entries. This means that we can express S^* as

$$S^* = \sum_{i=1}^n \beta_i^* v_i v_i^\top,\ \beta_i^* \in \mathbb{R}_+\ \forall\ i \in [n].$$

We turn our attention the linear relaxation of SDP defined by $\mathscr{S}$, defined in Sect. 1, which we recall is given by

$$\begin{aligned} \min_{X\in\mathbb{S}^n}\ & \langle C, X\rangle \\ \text{s.t: } & \langle A_i, X\rangle = b_i,\ \forall i \in [r], \\ & v^\top X v \geq 0\ \forall\ v \in \mathscr{S}. \end{aligned} \tag{$L_{\mathscr{S}}$}$$

This program is linear and is a relaxation of SDP as any feasible solution to it is feasible for $L_{\mathscr{S}}$. Its dual is given by

$$\begin{aligned} \max_{y,\in\mathbb{R}^n,\beta\in\mathbb{R}^n_+}\ & b^\top y \\ \text{s.t: } & C - \sum_{i=1}^{r} y_i A_i = \sum_{i=1}^{n} \beta_i v_i v_i^\top. \end{aligned} \tag{$DL_{\mathscr{S}}$}$$

Observe that this program is a strenghening of program DSDP, and that S^* is feasible for this program. Therefore, their optimal values must match, and in particular the optimal value of $DL_{\mathscr{S}}$ is finite. By strong duality of linear programs, $L_{\mathscr{S}}$ is solvable and its optimal value equals the optimal value of both DSDP and $DL_{\mathscr{S}}$. Again by our strong duality assumption of programs SDP and DSDP, program $L_{\mathscr{S}}$ solves SDP. □

A class of problems that has been extensively studied in the literature and where the hypothesis of our previous theorem applies are simultaneously-diagonalizable QCQPs. Recall that a QCQP is a problem of the form

$$\inf_{x\in\mathbb{R}^n} q_0(x) : q_i(x) \leq\ 0\ \forall\ i\ \in [r]. \tag{QCQP}$$

where $q_i(x) = x^\top A_i x + 2b_i^\top x + c_i$ with $A_i \in \mathbb{S}^n$, $b \in \mathbb{R}^n$ and $c_i \in \mathbb{R}$ for all $i \in \{0, \ldots, r\}$. QCQPs are NP-hard to solve in general but admit tractable convex relaxations. The SDP relaxation of a QCQP is given by the following semidefinite program [8, 67]:

$$\begin{aligned} \inf_{x\in\mathbb{R}^n,X\in\mathbb{S}^n}\ & \langle A_0, X\rangle + 2b_0^\top x + c_0 \\ \text{s.t }: & \langle A_i, X\rangle + 2b_i^\top x + c_i \leq\ 0\ \forall i \in \{1, \ldots, r\}, \\ & x \in \{0, 1\}^n, diag(X) = x, \\ & \begin{bmatrix} X & x \\ x^\top & 1 \end{bmatrix} \succeq 0. \end{aligned} \tag{1}$$

Whenever the A_i are simultaneously diagonalizable and we have access to a matrix U such that $A_i = U^\top D_i U$ for $i \in \{0, \ldots, r\}$, we can perform the change of variables $y = Ux$ and $\tilde{b_i} = Ub_i,\ i \in \{0, \ldots, r\}$ to obtain the a diagonalized version of the problem

$$\inf_{y\in\mathbb{R}^n} q_0(y) : q_i(y) \leq \ 0 \,\forall\, i \in [r] \tag{2}$$

However, we have $q_i(y) = a_i^\top y^2 + 2\tilde{b}_i^\top y + c_i$, $d_i \in \mathbb{R}^n$, $\tilde{b}_i \in \mathbb{R}^n$ and $c_i \in \mathbb{R}$ for each $i \in [0,\ldots,r]$. Here, $y^2 \in \mathbb{R}^n$ is the vector whose entries are the squared entries of the vector $y \in \mathbb{R}^n$. Ben-Tal and den Hertog [10] and Locatelli [45] study a certain second order cone relaxation of this problem, and show that the optimal value of that relaxation and that of the SDP relaxation match. Our results imply that in fact, given access to a matrix U that simultaneously diagonalizes the A_i, $i \in [0,\ldots,r]$ we can solve the SDP relaxation (1) with the linear program $L_{\mathscr{S}}$ where $\mathscr{S}$ is the set of columns of U.

Corollary 1 *Consider a quadratically constrained quadratic problem given as in QCQP and such that the matrices $\{A_i\}_{i\in\{0,\ldots,r\}}$ are simultaneously diagonalizable by an orthogonal matrix U. Let opt be the optimal value of relaxation (1) of QCQP. Let $\mathscr{S}$ be the set of columns of U. Then, the objective value z of the linear relaxation $SP_{\mathscr{S}}$ of (1) equals opt.*

Proof The proof is immediate from Theorem 2. □

2.1 Finding initial sets

As we have seen in Theorem 2, we know some vectors whose inclusion in $\mathscr{S}$ guarantees solvability under $\mathscr{O}$. The reason this worked was that we were able to produce a feasible solution to $DL_{\mathscr{S}}$ which matches the objective of an *optimal* solution to DSDP. Nonetheless, the previous argument still holds for a generic feasible solution to DSDP: any dual feasible solution will generate sets $\mathscr{S}$ that satisfy the corresponding dual bound.

Lemma 1 *Consider a generic SDP problem and let $\hat{y}$ be a feasible solution to the dual of the SDP with objective value $b^\top\hat{y}$. Let $\mathscr{S} = \mathscr{E}(C - \mathscr{A}(\hat{y}))$. Then, the objective value z^* of program $L_{\mathscr{S}}$ satisfies*

$$z^* \geq b^\top\hat{y}.$$

The proof of this lemma is very similar to that of Theorem 2. This result indicates that finding a good set $\mathscr{S}$ amounts to finding feasible solutions to the dual of SDP whose objective value is close to optimal. This task is akin to finding good feasible solutions to SDP, or at worse to solve a semidefinite feasibility problem, which in principle may be as hard as solving the original problem. However, the results of the previous subsection suggest a way to get around this issue by exploiting simultaneous diagonalizability. Under a weakening of this assumption, we will be able to construct solutions, which will be automatically feasible for the the DSDP. To begin, we give in Proposition 1 a characterization of simultaneous diagonalizability which we will then relax.

Lemma 2 *Let $\{A_i\}_{i\in I} \subseteq \mathbb{S}^n$ be a set of symmetric matrices. Then, there exists a basis of orthonormal vectors $\{u_1,\dots,u_n\}$ that simultaneously diagonalizes $\{A_i\}_{i\in I}$ if and only if A_i and A_j commute for every i and $j \in I$, i.e, $A_iA_j = A_jA_i\ \forall i,j \in I$.*

See [21] for a proof.

The set of matrices $\{A_1,\dots A_r\} \subseteq \mathbb{S}^n$ is simultaneously diagonalizable if and only if for every p and $q \in \mathbb{R}^r$ the matrices $\mathscr{A}(p) = \sum_{i=1}^r p_iA_i$ and $\mathscr{A}(q) = \sum_{i=1}^r q_iA_i$ commute.

Proposition 1 *The set of matrices $\{A_1,\dots A_r\} \subseteq \mathbb{S}^n$ is simultaneously diagonalizable if and only if for every p and $q \in \mathbb{R}^r$ the matrices $\mathscr{A}(p) = \sum_{i=1}^r p_iA_i$ and $\mathscr{A}(q) = \sum_{i=1}^r q_iA_i$ commute, and hence are simultaneously diagonalizable.*

Proof Necessity is trivial by having p and q range over the standard basis of $\mathbb{R}^r$ and Lemma 2. For sufficiency, Let U be an orthonormal matrix such that the matrices $U^\top A_iU = D_i$ are diagonal $\forall i \in [r]$. Given p and $q \in \mathbb{R}^r$ we have:

$$U^\top \mathscr{A}(p)U = U^\top\left(\sum_{i=1}^r p_iA_i\right)U = \sum_{i=1}^r p_iD_i.$$

Similarly we have $U^\top\mathscr{A}(q)U = \sum_{i=1}^m q_iD_i$. Since diagonal matrices commute we have

$$\left(\sum_{i=1}^r p_iD_i\right)\left(\sum_{i=1}^r q_iD_i\right) = \left(\sum_{i=1}^r q_iD_i\right)\left(\sum_{i=1}^r p_iD_i\right).$$

Given that $U^\top U = I$, pre-and post-multiplying by U and $U^\top$ respectively gives:

$$U\left(\sum_{i=1}^r p_iD_i\right)U^\top U\left(\sum_{i=1}^r q_iD_i\right)U^\top = U\left(\sum_{i=1}^r q_iD_i\right)U^\top U\left(\sum_{i=1}^r p_iD_i\right)U^\top$$

and finally

$$\mathscr{A}(p)\mathscr{A}(q) = \mathscr{A}(q)\mathscr{A}(p).$$

Since these matrices commute, they are simultaneously diagonalizable. □

Given that commutativity of the set $\{C, A_1,\dots,A_r\}$ will typically not hold, we relax the equivalent condition given by the previous lemma to require that commutativity holds only for special class of p's and q's. In particular we will set $p = e_{r+1}$ and q such that for some subset $J \subseteq [r]$ we have $\sum_{j\in J} q_jA_j = I_n$. The idea is that if we have a point $y \in \mathbb{R}^n$, not necessarily dual feasible for which the matrices C and $\mathscr{A}(y)$ commute, then taking $\mathscr{S}$ to be the columns of a matrix that diagonalizes them will yield a linear program with objective value as good as the best dual feasible solution that lies on the set

$$\{A \in \mathbb{S}^n : \exists\, x, t \in \mathbb{R} : A = tI_n + x\sum_{j\in[r]\setminus J} q_jA_j\}.$$

Theorem 3 *Consider a generic semidefinite optimization problem of the form SDP, with dual DSDP. Suppose that there exists vectors $q^1, q^2 \in \mathbb{R}^r$ whose support is disjoint such that $\sum_{j=1}^r q_j^1 A_j = I_n$ and such that the matrices C and $\sum_{j=1}^r q_j^2 A_j$ commute and therefore are simultaneously diagonalizable by some orthogonal matrix U. Let $\mathscr{S}$ to be the set of columns of such an U. Then, the optimal value z of program $L_{\mathscr{S}}$ satisfies the bound*

$$\left(\sum_{j=1}^r b_j q_j^2\right) x + \left(\sum_{j=1}^r b_j q_j^1\right) t \leq z$$

for any x and t such that the matrix $C - x\left(\sum_{j=1}^r q_j A_j\right) + tI_n$ is positive semidefinite.

Proof Let U be a matrix that simultaneously diagonalizes C and $\mathscr{A}(q^2) = \sum_{j=1}^r q_j^2 A_j$. Let z be the optimal value of program $L_{\mathscr{S}}$ where $\mathscr{S}$ is the set of columns $v_1, \ldots, v_n$ of U. Recall that the dual of this program is given by

$$\begin{aligned} \max_{y, \in \mathbb{R}^n, \beta \in \mathbb{R}^n_+} \quad & b^\top y \\ \text{s.t: } C - \sum_{i=1}^r y_i A_i &= \sum_{i=1}^n \beta_i v_i v_i^\top . \end{aligned} \tag{$DL_{\mathscr{S}}$}$$

Since U diagonalizes C, any column v of U is an eigenvector of C with some corresponding eigenvalue λ, and the same holds for $\mathscr{A}(q)$ with some eigenvalue γ. Hence, v is a eigenvector of $C - x\mathscr{A}(q) + tI_n$ with corresponding eigenvalue $\lambda - x\gamma + t$. Since we are looking for x and t values such that $C - x\mathscr{A}(q) + tI_n$ is psd, this gives rise to the equation $\lambda - x\gamma + t \geq 0$, and we have such one equation for every column of U. This system is always feasible as the t variable is free. Hence, there exists x^*, t^* for which the matrix $C - x^*\mathscr{A}(q) + t^* I_n$ is positive semidefinite. As U diagonalizes C, $\mathscr{A}(q)$ and I_n as $U^\top I_n U = U^\top U = I_n$, $C - x^*\mathscr{A}(q) + t^* I_n$ is diagonalizable by U and thus can be written as $\sum_i \eta_i v_i v_i^\top$ with $\eta_i \geq 0$ for $i \in [n]$. Thus, setting $y_j = x^* q_j^2$ if j belongs to the support of q^2 and $y_j = t^* q_j^1$ if j belongs to the support of q^1 (here recall that q^1 and q^2 have disjoint support) gives a feasible solution to program $DL_{\mathscr{S}}$ by setting $\eta_i = \beta_i$ for $i \in [n]$. The objective value of this solution is

$$\left(\sum_{j=1}^r b_j q_j^2\right) x^* + \left(\sum_{j=1}^r b_j q_j^1\right) t^*. \tag{3}$$

$\square$

We make a few observations about this theorem. First and foremost, we didn't require that the matrix $I_n + \sum_{j=1}^{r} q_j A_j$ is feasible for program DSDP. Second, notice that we have required that we can aggregate some of the A_j to form the identity matrix. Although this seems quite constraining, it is always the case that such a combination exists by our assumption that DSDP is strictly feasible, i.e if there exists $q \in \mathbb{R}^r$ such that $C - \mathscr{A}(q) \succ 0$. In principle, finding such q would require finding finding a point in the interior of the dual feasible region, which might be non-trivial. This suggests that our theorem is easier to apply in regimes where it is more directly "obvious " which combination of the A_j forms the identity. This is the case in the max cut problem, the Lovász theta number, the sparse PCA problem, the extended trust region SDP relaxation and many others. Finally, we observe that even though the bound given in Eq. 3 is the best bound we can *prove*, there might be other "hidden " dual feasible solutions that certify a better bound for $L_{\mathscr{S}}$.

Observation 3 *(Hidden basis property) Let $\hat{y}$ be a dual feasible solution for program DSDP with objective value $b^\top \hat{y}$. Suppose that $y \in \mathbb{R}^r$ is a point such that the matrices $C - \mathscr{A}(\hat{y})$ and $C - \mathscr{A}(y)$ (which is not necessarily PSD) share a basis of orthonormal eigenvectors. Let $\mathscr{S} = \mathscr{E}(C - \mathscr{A}(y))$ then, the optimal value z of program $L_{\mathscr{S}}$ satisfies*

$$b^\top \hat{y} \le z.$$

The proof of this observation is straightforward, but note that we have required $\mathscr{S}$ to be some eigenbasis of $C - \mathscr{A}(y)$ rather than the set of columns of some orthogonal matrix that simultaneously diagonalizes C and $\mathscr{A}(y)$. Clearly, if U diagonalizes both of those matrices it diagonalizes any linear combination of them. As we will see in the max cut experiments, Theorem 3 will certify a spectral bound, but the LP relaxation will actually have a better objective than the bound of Theorem 3 guarantees in practice.

2.2 Finding commuting matrices

To apply Theorem 3, we need first to find a combination of the constraints matrices which commutes with the objective matrix C of SDP. This can be accomplished using a linear program. Picking L to be an arbitrary linear function on y gives the program

$$\begin{aligned} &\min_{y \in \mathbb{R}^n} \; L(y) \\ &\text{s.t: } C\mathscr{A}(y) = \mathscr{A}(y)C. \end{aligned} \tag{4}$$

$$\begin{aligned} &\min_{y \in \mathbb{R}^n} \; L(y) \\ &\text{s.t: } C\left(\sum_i y_i A_i\right) = \left(\sum_i y_i A_i\right)C. \end{aligned} \tag{5}$$

To select L, we propose a function that trades off between the ℓ_1 norm of the matrix $C - \mathscr{A}(y)$ and the dual objective function $b^\top y$. The intention of the ℓ_1 term is to promote solutions where $C - \mathscr{A}(y)$ is sparse, rendering the computation of an eigenbasis easier. The term $-b^\top y$ encourages having solutions with good dual objective value. This yields the program

$$\begin{aligned} &\min_{y\in\mathbb{R}^n} \sum_{i,j} \left|[C-\mathscr{A}(y)]_{ij}\right| - b^\top y \\ &\text{s.t: } C\mathscr{A}(y) = \mathscr{A}(y)C. \end{aligned} \tag{CG}$$

Note that the null vector is always a feasible solution to this program. In Sect. 5 we experimentally test this idea.

3 Linear relaxations of the max cut semidefinite program

The question of exactly - or approximately - solving an SDP with a linear program finds one of its historical roots in the max cut problem, where in a given undirected graph, we seek a bipartition of the nodes to maximize the number of edges with one end in both parts. Since linear programming has been one of the main paradigms to tackle NP-hard combinatorial optimization problems through the relax-and-round paradigm, substantial efforts were dedicated to find a linear programming relaxation of the max cut problem. A graph with m edges has always a cut of size at least $\frac{1}{2}m$ and any cut can cut at most m edges, so it is trivial to provide an algorithm with integrality gap[4] 2. For example, a randomized algorithm picking vertices at random or a greedy algorithm will have this guarantee. The question was then if there exists a linear program that could have an approximation ratio better than 2.

The starting point of this line of research was perhaps the linear relaxation for max cut given by [9, 61]. Let $G = (V, E)$ be an undirected, simple graph on m edges and W its adjacency matrix. We define

$$\begin{aligned} \alpha(G) &:= \max\langle W, X\rangle \\ X_{ij} + X_{ik} + X_{kj} &\le 2\ \forall i, j, k \in V \\ X_{ij} - X_{ik} - X_{jk} &\le 0\ \forall i, j, k \in V \\ 0 \le X_{ij} &\le 1\ \forall i, j \in V. \end{aligned} \tag{6}$$

Here we use a binary variable X_{ij} for each pair of vertices $\{i, j\}$ to denote if the edge between them is cut. The first set of 'triangle' constraints specify that at most two edges can be picked in a cut from any triangle, while the second set rules out exactly one edge from any triangle from being selected in a cut. In [61], Poljak and Tuza prove that for sparse and dense versions of Erdős-Rényi random graphs, the integrality gaps of this LP tend to $2 - o(1)$ and $\frac{4}{3} - o(1)$ respectively. Here, $G_{n,p}$ denotes the class of random graphs on n nodes where every edge is included independently of others with probability p.

Theorem 4 *(Poljak, Tuza) [61] Let $mc(G)$ denote the size of the max cut of G.*

[4] In this paper, we employ the convention that the integrality gap is a number that is at least 1 and hence is the ratio of the value of the relaxation to the optimal value of the max cut.

– *(Sparse graphs). Let* $p(n)$ *be a function such that* $0 < p < 1$, $p(n) \cdot n \to \infty$ *and* $p \cdot n^{1-a} \to 0$ *for every* $a > 0$, *then the expected relative error* $\frac{\alpha(G_{n,p}) - mc(G_{n,p})}{mc(G_{n,p})}$ *tends to* 1 *as* $n \to \infty$ *with probability* $1 - o(1)$.

– *(Dense graphs). Let* $p(n)$ *be a function such that* $0 < p < 1$, $p(n) = \Omega\left(\sqrt{\frac{\log(n)}{n}}\right)$. *Then the expected relative error* $\frac{\alpha(G_{n,p}) - mc(G_{n,p})}{mc(G_{n,p})}$, *tends to* $\frac{1}{3}$ *as* $n \to \infty$ *with probability* $1 - o(1)$.

Such integrality gap lower bounds for the basic LP encouraged two distinct approaches to solve the problem. The first one focused on adding valid constraints to formulation (6), such as ”hypermetric", and ”gap" constraints. See [25, 55] for more details. Nonetheless, a long line of research culminated in showing that such direct strengthenings will fail to provide an approximation factor better than 2 [17, 18, 72]. In particular, Kothari et al. [41] prove that this problem - and more generally Constraint Satisfaction Problems - is resistant to this strategy by showing that extended linear formulations are as powerful as the Sherali-Adams hierarchy, which in turn requires an exponential number of rounds (in ε) to certify an integrality gap better than $2 - \varepsilon$. The second approach, perhaps much more influential, considered stronger optimization relaxations, such as the vector optimization relaxation of Poljak and Rendel [60], shown to be SDP-representable and providing an approximation ratio of ~ 1.13 in the seminal work of Goemans and Williamson [33]. Naturally, this leads to the question if linear programs can well approximate semidefinite ones. In [14] Braun et. al. show that in principle one needs an exponential number of a constraints in an LP to correctly approximate an SDP. These two combined results extinguish the hope that linear programming may be used to approximate max cut. Since the question of finding a good set $\mathscr{S}$ to initialize Algorithm 1 amounts to finding a linear approximation to a semidefinite program, these results suggest that no systematic procedure can generate a good set $\mathscr{S}$ as in particular they would provide an approach to obtain a low-gap linear programming approximation to the max cut problem. In this sense, we propose to use instance-specific information to avoid the hardness of approximation results, in particular by exploiting bounds relating the spectrum of the graph to the value of the max cut, resulting in linear relaxations with better approximation ratios.

For a graph $G = (V, E)$ we set $m = |E|$ and denote by W its adjacency matrix. Recall that the semidefinite relaxation for max cut due to of Poljak, Rendl, Goemans and Williamson [33, 60] is given by

$$\begin{aligned} &\frac{1}{2}m + \frac{1}{4}\max_{X}\ \langle -W, X\rangle \\ &\text{s.t: } X \succeq 0,\quad X_{ii} = 1,\ \forall\, i \in [n]. \end{aligned} \tag{GW}$$

with dual

$$\begin{aligned} &\frac{1}{2}m + \frac{1}{4}\min_{\gamma \in \mathbb{R}^n} \sum_{i=1}^{n} \gamma_i \\ &\text{s.t: } W + diag(\gamma) \succeq 0. \end{aligned} \tag{DGW}$$

It is known that strong duality holds for this pair of programs: both (GW) and (DGW) are solvable and their objectives coincide. Delorme and Poljak show [23] that the max cut value of G on n nodes is upper bounded by the quantity

$$\min_{u \in \mathbb{R}^n : \sum_i u_i = 0} \quad \frac{n}{4} \lambda_1(\mathscr{L}(G) + diag(u)).$$

It turns out that this program is equivalent to program DGW [33]. In their seminal work, Goemans and Williamson show that this program achieves an approximation ratio of roughly $\frac{1}{0.878} \sim 1.138$. Through this equivalence, one can show that the semidefinite program GW satisfies a series of eigenvalue bounds. For instance, one may take u such that $\sum_{i=1}^n u_i = 0$ and all of the diagonal entries of the matrix $\mathscr{L}(G) + diag(u)$ equal $\frac{2m}{n}$. This results in what is usually known as *the* eigenvalue bound for max cut due to Mohar and Poljak [52]

$$mc(G) \leq \frac{1}{2} m + \frac{n}{4} \lambda_1(-W) = \frac{1}{2} m - \frac{n}{4} \lambda_n(W) \tag{7}$$

To see the second equality, recall that for any matrix A, $\lambda_n = \lambda_1(-A))$. See [5, 52] for an elementary proofs of this inequality. As mentioned in [55], conventional wisdom is that LPs cannot certify even the eigenvalue bound, and we are not aware of a polynomially sized linear program that certifies this bound.

3.1 Instance-specific linear relaxations

The specialization of program $L_{\mathscr{S}}$ to the max cut problem results in a polynomially sized linear program that explicitly depends of the adjacency matrix W on G, allowing us to circumvent the theoretical limitations of linear relaxations described in the introduction of this section. Using Theorem 3 this LP will be shown to satisfy the eigenvalue bound (7) whenever $\mathscr{S}$ is chosen appropriately. Fixing $\mathscr{S} = \{v_1, \ldots, v_k\}$, program $L_{\mathscr{S}}$ specializes to a linear program which we denote by program $SP_{\mathscr{S}}$.

$$\begin{aligned} &\max_{X \in \mathbb{S}^n} \ \frac{1}{2} m + \frac{1}{4} \langle -W, X \rangle \\ &\text{s.t: } v^\top X v \geq 0 \ \forall \ v \in \mathscr{S}, \ X_{ii} = 1, \ \forall \ i \in [n], \ \|X\|_\infty \leq 1. \end{aligned} \tag{$SP_{\mathscr{S}}$}$$

In this program we have included the constraint $\|X\|_\infty \leq 1$. As the following observation shows, this is a valid constraint for GW. Adding it is useful because it will guarantee that the dual of $SP_{\mathscr{S}}$ is always feasible, regardless of G.

Observation 4 *Let X be feasible for program (GW). Then, it is feasible for program $SP_{\mathscr{S}}$ for any set $\mathscr{S} \subseteq \mathbb{R}^n$.*

Proof Let X be feasible for (GW). This means X is positive semidefinite, and that there exists a set of vectors $x_1, \ldots, x_n$ such that $X_{ij} = x_i^\top x_j$ for all $i, j \in [n]$. For each $i \in [n]$ we have $X_{ii} = 1$ and thus we see that $\|x_i\|_2 = 1$. It follows that each entry of the vectors x_i is bounded by 1 and therefore that X_{ij} is bounded by 1 for all i and j. The other two constraints of the linear program are clearly satisfied by X. $\square$

It will be also be useful to consider the following strenghening of program GW depending of $\mathscr{S} = \{v_1, \ldots, v_k\}$.

$$\begin{aligned} &\frac{1}{2}m + \frac{1}{4}\max_{\eta \in \mathbb{R}^k} \left\langle -W, \sum_{i=1}^{k} \eta_i v_i v_i^\top \right\rangle \\ &\text{s.t: } diag\left(\sum_{i=1}^{k} \eta_i v_i v_i^T\right) \leq 1,\ \eta_i \geq 0,\ \ v_i \in \mathscr{S}\ \forall i \in [k],\ k = |\mathscr{S}|. \end{aligned} \tag{$SD_{\mathscr{S}}$}$$

Here, and for the rest of the paper, we denote by $z_{SP_{\mathscr{S}}}, z_{GW}, z_{DGW}, z_{SD_{\mathscr{S}}}$ the optimal values of $SP_{\mathscr{S}}$, GW, DGW and $SD_{\mathscr{S}}$ ignoring the additive constant $\frac{1}{2}m$ and the multiplicative constant $\frac{1}{4}$, respectively. For illustration, we have:

$$z_{GW} = \max\ \langle -W, X\rangle\ \text{s.t:} \quad X \succeq 0,\ \ X_{ii} = 1,\ \forall\, i \in [n].$$

By duality, we get the following relationships between these optimal values

$$z_{SD_{\mathscr{S}}} \leq z_{GW} = z_{DGW} \leq z_{SP_{\mathscr{S}}}.$$

Observe that we may employ different sets $\mathscr{S}$ to define SP and SD and the above relations will continue to hold. As a sanity check, we first observe that program $SP_{\mathscr{S}}$ satisfies the trivial bound for max cut.

Lemma 3 *Let $\mathscr{S}$ be an arbitrary subset of $\mathbb{R}^n$. Then $z_{SP_{\mathscr{S}}}$ satisfies:*

$$z_{SP_{\mathscr{S}}} \leq 2m$$

and therefore $\frac{1}{2}m + \frac{1}{4}z_{SP_{\mathscr{S}}} \leq m$.

Proof Let $\mathscr{S} = \{v_1, \ldots, v_k\}$. The dual of program $SP_{\mathscr{S}}$ is given by

$$\begin{aligned} \min_{\lambda,\alpha,\delta,\beta,\Lambda}\ &\frac{1}{2}m - \frac{1}{4}\left[tr(\Lambda) - \sum_{i \neq j}\delta_{ij} - \sum_{i\neq j}\alpha_{ij}\right] \\ \text{s.t: } &W - \Lambda = \sum_{i=1}^{k} \beta_i v_i v_i^\top, \\ &\delta_{ij} \geq 0\ \forall i \neq j \in [n], \\ &\alpha_{ij} \geq 0\ \forall i \neq j \in [n], \\ &\lambda_i \in \mathbb{R}\ \forall i \in [n], \\ &\beta_i \geq 0\ \forall i \in [n], \\ \Lambda \in \mathbb{S}^n,\ \Lambda_{ij} = &\delta_{ij} - \alpha_{ij}\ \forall i \neq j \in [n],\ \Lambda_{ii} = \lambda_i\ \forall i \in [n]. \end{aligned} \tag{$DSP_{\mathscr{S}}$}$$

The proof of this fact is deferred to Appendix A. Letting $\beta_i = 0 \; \forall i \in [n]$, $\Lambda = W$ where $\delta_{ij} = 1, \alpha_{ij} = 0$ whenever $W_{ij} = 1$ and 0 otherwise, we obtain a feasible solution for the previous program with $tr(\Lambda) - \sum_{i \neq j} \delta_{ij} - \sum_{i \neq j} \alpha_{ij} = -2m$. $\square$

It can be checked that for an arbitrary graph G, the feasible region of program DGW, namely $\Gamma = \{\gamma \in \mathbb{R}^n : W + diag(\gamma) \succeq 0\}$ is not necessarily polyhedral. However, we can exploit Theorem 3 to derive a set $\mathscr{S}$ for the relaxation $SP_{\mathscr{S}}$ that has a good objective value. Although this statement can be proven directly by simply giving a judicious choice of $\mathscr{S}$, we derive the result in a way that explicitly uses the theorem.

Theorem 5 *Let G be a graph on n vertices and W its adjacency matrix. Let λ_n denote the smallest eigenvalue of W. Set $\mathscr{S} = \mathscr{E}(W)$. Then*

$$z_{SP_{\mathscr{S}}} \leq -n\lambda_n := \chi(G).$$

Proof For $i = 1, \ldots, n$ let the matrix A_i denote the matrix of all zeros but with a single 1 in its i-th diagonal entry. Hence, Γ can be expressed as:

$$\Gamma = \{\gamma \in \mathbb{R}^n : W + \sum_{i}^{n} \gamma_i A_i \succeq 0\}.$$

To apply Theorem 3, we express the identity as some combination of the A_i. concretely, we let $\hat{\gamma} = \vec{1}$ be the vector of all ones in $\mathbb{R}^n$ so that we have that $\sum_{i=1}^{n} \hat{\gamma}_i A_i = I_n$. By Theorem 3, it follows that if $\mathscr{S} = \mathscr{E}(W)$ then the optimal value $z_{SP_{\mathscr{S}}}$ of program $SP_{\mathscr{S}}$ satisfies

$$z_{SP_{\mathscr{S}}} \leq t \cdot n$$

for any t such that $W + tI$ is positive semidefinite. Observe that $W - \lambda_n I$ is positive semidefinite. In particular, we obtain

$$z_{SP_{\mathscr{S}}} \leq -n\lambda_n. \qquad \square$$

We provide an alternate direct proof of this result in Appendix A by directly setting $\mathscr{S} = \mathscr{E}(W)$ and using the dual of program $SP_{\mathscr{S}}$. Interestingly, this result allows us to show that the linear relaxation $SP_{\mathscr{S}}$ is strictly stronger than the linear formulation for max cut given in program (6), in the sense that it gives -in contrast to the previous LP- the correct value of max cut for the graphs considered in Theorem 4. Perhaps more interestingly, we show that for random $d-$regular graphs the linear program $SP_{\mathscr{S}}$ with $\mathscr{S} = \mathscr{E}(W)$ approximates max cut with an approximation factor of $1 + O(\frac{1}{\sqrt{d}})$. This result is quite striking as it is precisely for random $d-$ regular graphs (with $d \in O(1)$) that the hardness of approximation for max cut using the Sherali-Adams hierarchy was shown [17, 18, 41, 72]. These two claims are the content of the next two corollaries.

Corollary 2 *Let $G = G(n, p)$ be sampled according to the Erdős-Rényi model [29] where p is a function of n. Let $g(n)$ be a non-decreasing function of n. Then, the* *ratio* $\frac{\frac{1}{2}m+\frac{1}{4}\chi(G)}{\frac{1}{2}m}$ *is at most* $1+\sqrt{\frac{2}{g(n)}}$ *as long as np is at least* $\frac{g(n)}{2}\log(n)$*, with high probability. In particular, for all dense graphs of Theorem 4,* $np \geq g(n) \geq \sqrt{n}$ *and the quotient converges to* 1*. For sparse graphs of Theorem 4 where* $np = c\log(n)$*, the quotient is at most* $1+\sqrt{\frac{2}{c}}$*.*

Proof Let $p = p(n)$ and $G = G(n, p)$ be sampled according to the Erdős-Rényi model with $np \in \Omega(\frac{g(n)}{2}\log(n))$. Letting $\varepsilon = \frac{1}{n}$ and applying Theorem 1 of [19] we have that with probability at least $1-\frac{1}{n}$

$$-\lambda_n \leq \sqrt{4np\ln\left(\frac{2n}{\varepsilon}\right)}+p.$$

Recalling that the number of edges m of G is $\theta(n^2p)$ with high probability, a direct computation of the quantity $\frac{\frac{1}{2}m-\frac{1}{4}n\lambda_n}{\frac{1}{2}m}$ gives the result. □

Corollary 3 *Suppose that G is a d-regular graph with* $-\lambda_n \leq c\cdot\sqrt{d}$ *for some constant c and* $\mathscr{S} = \mathscr{E}(W)$*. Then, the following inequality holds:*

$$\frac{z_{SP_{\mathscr{S}}}}{z_{SD_{\mathscr{S}}}} \leq 1+\frac{c}{\sqrt{d}}.$$

Proof Recall that a d-regular graph has $m = \frac{nd}{2}$ edges. This gives $n = \frac{2m}{d}$. Suppose $-\lambda_n \leq c\cdot\sqrt{d}$. Then, by Theorem 5 and that $z_{GW} \geq 0$ for any graph G we get

$$\frac{\frac{1}{2}m+\frac{1}{4}z_{SP_{\mathscr{S}}}}{\frac{1}{2}m+\frac{1}{4}z_{GW}} \leq \frac{\frac{1}{2}m-\frac{1}{4}n\lambda_n}{\frac{1}{2}m} = \frac{\frac{1}{2}m-\frac{1}{4}\lambda_n\frac{2m}{d}}{\frac{1}{2}m} = 1-\frac{\lambda_n}{d} \leq 1+\frac{c}{\sqrt{d}}.$$

□

It is known that random $d-$regular graphs satisfy the hypothesis of the theorem [27, 28, 70], justifying our previous claim on the guarantees of our linear relaxation on random $d-$regular graphs. Another class of graphs which satisfies the hypothesis of the theorem are the Ramanujan expander graphs [48], where $c = 2$. We contrast this result with the fact that the relative error of $\alpha(G)$- defined above in LP(6)- relative to the max cut of G tends to 1 for Ramanujan graphs [61].

To the best of our knowledge, this is the first linear relaxation of max cut with these two guarantees.

3.2 Hidden basis property and stronger guarantees

In the previous subsection, we considered a bound given by Theorem 3 using the fact that $-\lambda_n\vec{1}$ is feasible for program DGW. However, it might very well be the case that there are other "hidden" dual feasible solutions. Although we are not aware of any such solutions, it is illustrative to check whether or not program $SP_{\mathscr{S}}$ gives better solutions than the eigenvalue bound. This raises the question of the quality of our linear relaxation in the setup where the eigenvalue bounds fails to give an approximation factor better than 2 for the maximum cut value of a graph. Indeed, the eigenvalue bound is not powerful enough to provide an approximation factor better than $2-\varepsilon > 0$ for any given $\varepsilon > 0$ in general. As a matter of fact, for any given $\varepsilon > 0$, there exist a family of graphs whose maximum cut is bounded above by $\frac{1}{2}m + \varepsilon m$, but the eigenvalue bound cannot certify a bound better than $2-\varepsilon$. We give an example of such as class in our next definition, which is inspired by a remark in [55].

Definition 2 We say that a graph G is sampled from the class of random graphs $\mathscr{G}(n, d, l)$ if G has n vertices and two disjoint components G_1 and G_2 where G_1 is a random d-regular graph, G_2 is complete bipartite graph where each side of the bipartition has $\sqrt{n}$ nodes, and l random edges connect the G_1 and G_2.

Observe that the absolute value of most negative eigenvalue of the adjacency matrix of a graph sampled from $\mathscr{G}(n, d, l)$ is $\Omega(\sqrt{n})$ due to the bipartite component. If $d \in O(1)$ then the number of edges in G is linear in n and so is the maxcut of G. However, the eigenvalue bound is weak: it certifies that the maxcut size is at most $O(n^{1.5})$ (notice that this is worse even than the trivial upper bound of m). This class of graphs is suggested as an example in [55] as a class of graphs where the eigenvalue bound behaves poorly. However, our LP certifies a much better value, when $l = 0$, as the next observation shows:

Lemma 4 *Let G be a graph with two disconnected components G_1 and G_2, where $|V(G_1)| = n_1$, $|V(G_2)| = n_2$, λ^1 is the smallest eigenvalue of the adjacency matrix of graph G_1 and λ^2 is the smallest eigenvalue of the adjacency matrix of G_2. Let $\mathscr{S} = \mathscr{E}(W)$. Then, $SP_{\mathscr{S}}$ certifies:*

$$z_{SP_{\mathscr{S}}} \leq n_1\lambda^1 + n_2\lambda^2.$$

Proof The proof is basically the same as the proof of Theorem 5 by observing that the support of eigenvectors corresponding to disjoint components of a graph are disjoint. □

This result may seem artificial in the sense that G is a disconnected graph. However, we show through extensive experiments in Tables 1 and 2 in Sect. 5 that the quotient of the optimal value $SP_{\mathscr{S}}$ to the GW relaxation is significantly better than the quotient of $\chi(G)$ to the GW relaxation, even when edges are added between the components in these difficult examples for the eigenvalue bound.

3.3 Solvability of maxcut under $\mathscr{O}$

In the previous subsection, we have seen that we can derive a good starting set $\mathscr{S}$. In general, program $L_{\mathscr{S}}$ does not solve the max cut SDP. In this subsection we will show that whenever G is a *distance regular graph* then we have solvability of the max cut SDP under $\mathscr{O}$. The class of distance-regular graphs contains strongly regular graphs, which have been extensively studied for their algebraic, combinatorial and spectral properties [15, 69]. Famous graphs such as the Petersen graph belongs to this class. In what follows, we give a sufficient condition that ensures that the value of $SP_{\mathscr{S}}$ equals the optimal value of the GW relaxation, provided that $\mathscr{S}$ includes an orthonormal eigenbasis of W.

Definition 3 (Distance-regular graphs) For a graph G and u, v vertices in $V(G)$ define $G_j(u)$ to be the set of vertices of G at distance exactly j of u, i.e., the vertices $v \in V(G)$ such that the shortest path joining u and v has length j. We say G is distance regular if it is connected, d-regular for some d and there exists integers $c_i, b_i,\ i \in \mathbb{N}$ such that for any two vertices u, v at distance $i = d(u, v)$ there are precisely c_i neighbours of v in $G_{i+1}(u)$ and b_i neighbours of v in $G_{i-1}(u)$.

Examples of such graphs are all strongly regular graphs, Hamming graphs, complete graphs, cycles, and odd graphs (such as the Petersen graph) [15]. The next theorem will allow us to prove that our linear relaxations are tight for this class of graphs.

Theorem 6 *Let G be a graph and W its adjacency matrix. Let $\mathscr{S} = \mathscr{E}(W)$ and W_n be the eigenspace of W corresponding to λ_n. Suppose the dimension of W_n is k with $n > k \geq 1$. Suppose there exists an orthonormal basis $\mathscr{U} = \{u_1, \ldots, u_k\}$ of W_n such that the matrix A with rows $u_1, \ldots, u_k$ has columns with constant 2- norm, i.e. there exists some $c \in \mathbb{R}^+$ such that $\|A_j\|_2 = c\ \forall j \in [n]$ where A_j denotes the j-th column of A. Then, $z_{SD_{\mathscr{S}}}$ equals $-n\lambda_n$ and in particular*

$$z_{SP_{\mathscr{S}}} = z_{GW} = z_{SD_{\mathscr{S}}}.$$

Proof The proof requires two steps. First, we show that if such basis $\mathscr{U}$ exists and we let $\mathscr{S} = \mathscr{U}$ then the theorem holds. Second, we show that we may set $\mathscr{S}$ to be an arbitrary orthonormal basis of W_n. This is necessary since the dimension of $W_n \geq 2$ and hence orthonormal bases are not unique. This might break the theorem if we choose any other orthonormal basis for $\mathscr{S}$ instead of $\mathscr{U}$. We begin with the first step. Notice that $c = \sqrt{\frac{k}{n}}$. Indeed, since the u_i are unitary vectors we have that for all $i \in [k]$ $\sum_{j=1}^n A_{i,j}^2 = 1$. Summing over i gives $\sum_{i=1}^k \sum_{j=1}^n A_{i,j}^2 = k$. By our assumption of constant sum of the column vectors, we get $\sum_{i=1}^k A_{i,j}^2 = c^2\ \forall j \in [n]$. Summing over j gives $\sum_{j=1}^n \sum_{i=1}^k A_{i,j}^2 = nc^2$ and we get $k = nc^2$. Let $B = \sqrt{\frac{n}{k}} A^\top$ and $Y = BB^\top$. Let v_i denote the ith row of B, and recall that v_i has norm $\sqrt{\frac{k}{n}}$. This implies that $Y_{ii} = v_i \cdot v_i = \frac{n}{k} \cdot \frac{k}{n} = 1$. Finally, observe that $Y = \frac{n}{k} \sum_i^k u_i (u_i)^\top$. It follows that Y is feasible for $SD_{\mathscr{S}}$ with $\mathscr{S} = \mathscr{U}$. This solution has an objective value

$$z_{SD_{\mathscr{S}}} \geq \left\langle -W, B^T B \right\rangle \geq \left\langle -W, \frac{n}{k} \sum_{i=1}^k u_i u_i^T \right\rangle = -n\lambda_n.$$

For the second part, we show that we can take $\mathscr{S}$ to be any arbitrary orthonormal basis of W_n. Notice that the only fact that we used from $\mathscr{U}$ is that the matrix A formed by stacking the vectors u_i as rows has constant column norm. Therefore, it suffices to show that any matrix A' formed in the same way from an arbitrary basis $\mathscr{U}'$ has this same property. Hence, let $\mathscr{U}' = \{w_1, \ldots, w_k\}$ be an arbitrary basis of W_n and suppose that the basis $\mathscr{U}$ exists.

Since the vectors$\{u_1, \ldots, u_k\}$ are an orthonormal basis of W_n which is a lineal subspace of $\mathbb{R}^n$, we can extend this set of vectors to a full orthonormal basis $\{u_1, \ldots, u_k, u_{k+1}, \ldots u_n\}$ of $\mathbb{R}^n$. Further, observe that $\sum_{i=1}^n u_i(u_i)^\top = I_n$ where I_n is the $n \times n$ identity matrix. To see this, let $v = r_1 u_1 + \cdots + r_n u_n \in \mathbb{R}^n$ be an arbitrary vector expressed in the u_i, $i \in [n]$ basis. We have

$$\left(\sum_{i=1}^n u_i u_i^\top\right) v = \sum_{i=1}^n \langle u_i, v\rangle u_i = \sum_{i=1}^n r_i u_i = v. \tag{8}$$

We derive that $\sum_{i=1}^n u_i u_i^\top$ equals the identity matrix. Notice that this equation remains true if we replace the u_i for any arbitrary orthonormal basis of $\mathbb{R}^n$. Since the diagonal entries of $A^\top A$ equal $\frac{k}{n} = c^2$ we see that the diagonal entries of $\sum_{i=k+1}^n u_i u_i^\top$ equal $1 - c^2$. Finally it follows that $\{w_1, \ldots, w_k, u_{k+1}, \ldots, u_n\}$ is as well a basis for $\mathbb{R}^n$ and thus by Equation (8) we have $\sum_{i=1}^k w_i w_i^\top + \sum_{i=k+1}^n u_i u_i^\top = I_n$. This shows that every diagonal entry of the matrix $\sum_{i=1}^k w_i w_i^\top$ must equal c, and hence the matrix A' formed by stacking the vectors w_i as rows has constant column norm. The conclusion of the theorem follows from the inequality $z_{SD_{\mathscr{S}}} \le z_{GW} \le z_{SP_{\mathscr{S}}} \le -n\lambda_n$. □

Alon and Sudakov proved something similar to the first part of our proof in [5]. In the paper, the authors prove that $z_{GW} = \frac{1}{2}m - \frac{1}{4}n\lambda_n$ under the hypothesis that there exists a feasible solution $Y = B^\top B$ for the (GW) relaxation such that the columns of B are unitary vectors $v_1, \ldots, v_n$ and its rows $u_1, \ldots u_k$, $1 \le k \le n$ are eigenvectors of W corresponding to λ_n. We conclude this section with the corollary for distance-regular graphs.

Corollary 4 *Let G be a distance-regular graph. Let $\mathscr{S} = \mathscr{E}(W)$. Then*

$$z_{SP_{\mathscr{S}}} = z_{GW} = z_{SD_{\mathscr{S}}}.$$

Proof The results follows from the following theorem. It states that the eigenspaces of distance regular graphs satisfy the hypothesis of Theorem 6. □

Theorem 7 *([15], Theorem 4.1.4) Let G be a distance regular graph and λ an eigenvalue of G. Then, there exists a symmetric matrix whose columns span the eigenspace corresponding to λ and that have a constant norm.*

4 Applications to semidefinite programs

To verify the applicability of the ideas presented, we consider three families of semidefinite optimization problems, each illustrating an aspect of our work. The first problem considered is the semidefinite relaxation of the maxcut problem which we presented in Sect. 3. We present our experimental results in Sect. 5.

4.1 Maximum cut

The max cut problem is a prime example of how our methodology can be applied as it is a hard combinatorial problem that linear programs fail to approximate. We will test our ideas using two linear programs, already introduced in Sect. 3.

$$\begin{aligned} &\max_{X \in \mathbb{S}^n} \frac{1}{2}m + \frac{1}{4}\langle -W, X\rangle \\ &\text{s.t: } v^\top X v \geq 0 \; \forall \, v \in \mathscr{S}, \, X_{ii} = 1, \;\forall \, i \in [n], \, \|X\|_\infty \leq 1. \end{aligned} \tag{$SP_{\mathscr{S}}$}$$

By the results of Sect. 3, we know that as $n \to +\infty$, the optimal value of this program will converge to the optimal value of max cut for Erdős-Rényi graphs and random d-regular graphs whenever $\mathscr{S}$ contains a basis of eigenvector of the matrix W. We test the quality of the linear relaxation on such graphs, as well as on graphs of the family $\mathscr{G}(n, l, k)$ which was introduced in Sect. 3. This family was designed to have a trivial eigenvalue bound. In Appendix B[5] we include as well experiments on the quality of relaxations on 16 graphs taken from TSPLIB [64] and 14 graphs from the network repository [65]. Furthermore, we consider program

$$\begin{aligned} &\frac{1}{2}m + \frac{1}{4}\max_{\eta \in \mathbb{R}^k} \left\langle -W, \sum_{i=1}^{k} \eta_i x_i x_i^\top \right\rangle \\ &\text{s.t: } diag\left(\sum_{i=1}^{k} \eta_i x_i x_i^T\right) \leq 1, \, \eta_i \geq 0, \; x_i \in \mathscr{S} \;\forall i \in [k], \; k = |\mathscr{S}|. \end{aligned} \tag{$SD_{\mathscr{S}}$}$$

This program is useful as we can obtain graph cuts from its solution using the rounding procedure of Goemans and Williamson [33]. Since the focus of this paper is comparing the optimal value of the different linear relaxations versus the optimal value of the SDPs, we defer results on rounded solutions to Appendix B.

4.2 Lovász theta number

The second problem we consider is the Lovász theta number $\vartheta(G)$ introduced by Lovász in the seminal paper [46] as a convex relaxation for the stability number of a graph G. ϑ can be computed in polynomial time using a semidefinite program. Since $\vartheta(\bar{G})$ -where $\bar{G}$ is the complement of G- is lower and upper bounded resp. by the clique number and the chromatic number of G, it allows one to compute those numbers in polynomial time for graphs for which these two quantities coincide e.g., perfect graphs.

5 Appendix B is included in the online companion of this paper.

$\vartheta(G)$ can be computed by the following semidefinite optimization program:

$$\begin{aligned} &\max_{S\in\mathbb{R}^n} \langle J, X\rangle \\ \text{s.t: } &tr(X)=1,\ X_{i,j}=0\ \forall(i,j)\in E, \\ &X\succeq 0. \end{aligned} \tag{Tn}$$

This problem is related to our setup, as it is known that the feasible region of the dual program is polyhedral whenever the considered graph is perfect. This striking results coincides with the fact that it is precisely for these graphs where the theta number coincides with the *independence number* of the graph.

We apply the ideas developed in Sect. 2 on two families of graphs. The first class is that of *regular* graphs. Notice that the constraints $X_{i,j}=0\ \forall(i,j)\in E$ can be expressed as $\langle X, A^{ij}\rangle=0$ where A^{ij} is matrix of all zeros except it has a 1 in its ij,ji entries whenever G contains edge ij. It is clear that if A is the adjacency matrix of G, we have $A=\sum_{ij,\in E} A^{ij}$. Regular graphs are interesting in our setting as it is easy to check that whenever G is regular graphs, A its adjacency matrix, and J the matrix of all ones, we have $JA=AJ$. The second class of graphs we consider are Erdős-Rényi random graphs, which are typically not regular and it is not obvious how to combine the A^{ij} to obtain a matrix that commutes with J. We will use program (CG) to find such matrices.

Given a finite set $\mathscr{S}$, we obtain the linear relaxation of program (Tn):

$$\begin{aligned} &\max_{X\in\mathbb{S}^n} \langle J, X\rangle \\ \text{s.t: } &tr(X)=1,\ X_{i,j}=0\ \forall(i,j)\in E, \\ &v^\top Xv\geq 0\ \forall\ v\in\mathscr{S} \end{aligned} \tag{LTn}$$

In Sect. 5, we compare the objective value of programs (Tn) and (LTn), on Erdős-Rényi random graphs and $d-$regular graphs. Interestingly, this problem is much more resistant to the the cut generation strategy for solving the corresponding SDP proposed in Algorithm 1. As we will see, generating cuts through the separation oracle of the semidefinite cone fails completely on both Erdős-Rényi graphs and $d-$regular graphs. On the contrary, setting $\mathscr{S}$ to be the columns of a matrix that simultaneously diagonalizes J and A -where A is the adjacency matrix of G in the case of regular graphs or a matrix given by program (CG) in the case of Erdős-Rényi graphs - performs significantly better.

In our discussion on the max cut problem we showed that there is a eigenvalue bound for the max cut value that every graph satisfies, and one might wonder if there such a bound for the theta number. This is indeed the case, albeit only for regular graphs.

Remark 1 Let G be a $d-regular$ graph with n vertices. Let W be the adjacency matrix of G with largest eigenvalue λ_1 and smallest eigenvalue λ_n, then the Lovász theta number $\vartheta(G)$ satisfies

$$\vartheta(G) \leq \frac{-n\lambda_n}{\lambda_1 - \lambda_n} \tag{9}$$

For a proof of this result, see [46]. We conjecture that the objective value of the linear program (LTn) is also upper bound by $\frac{-n\lambda_n}{\lambda_1-\lambda_n}$ as this was the case in all the experiments we performed for $d-$regular graphs.

4.3 QCQPs

We consider more general SDPs obtained as the Shor relaxation [67] of certain QCQPs to test the proposed methodology in three different settings, each highlighting an interesting point. General QCQPs were introduced in Sect. 2, but in this section and Sect. 5 we will consider a more specialized version of them, following [8], of the form

$$\begin{aligned} \inf_{x\in\mathbb{R}^n} \; & x^\top A_0 x + b_0^\top x + c_0 \\ \text{s.t: } & x^\top A_i x + b_i^\top x \leq b_i \;\forall i \in [r], \\ & Dx = d, \\ & l \leq x \leq u, \end{aligned} \tag{10}$$

where r denotes the number of quadratic constraints and is at least 1. $A_i,\ i = \{0, \ldots, r\}$ are symmetric matrices, not necessarily PSD, $b_i,\ i = \{0, \ldots r\}$ are vectors in $\mathbb{R}^n$, D is a $q \times n$ real matrix and $d \in \mathbb{R}^q$. l and u are vectors in $\mathbb{R}^n$ and we assume that $-\infty < l \leq u < +\infty$ so that the bounding boxes are non-empty and bounded. If the bounding boxes are of the form $[l, u]^n$ we can do a linear change of variables so that $x \in [0, 1]^n$. Such problems admit the following SDP relaxation:

$$\begin{aligned} \inf_{x\in\mathbb{R}^n, X\in\mathbb{S}^n} \; & \langle A_0, X\rangle + b_0^\top x + c_0 \\ \text{s.t: } & \langle A_i, X\rangle + b_i^\top x \leq c_i \;\forall i \in [r], \\ & Dx = d, \\ & 0 \leq x_i \leq 1 \;\forall i \in [n], \\ & 0 \leq X_{i,j} \leq 1 \;\forall i, j \in [n], \\ & \begin{bmatrix} X & x \\ x^\top & 1 \end{bmatrix} \succeq 0. \end{aligned} \tag{11}$$

By letting $\hat{A}_i := \begin{bmatrix} A_i & b_i \\ b_i^\top & c_i \end{bmatrix}, i \in \{0, 1, \ldots, m\}$, $\hat{X} := \begin{bmatrix} X & x \\ x^\top & 1 \end{bmatrix}$ and by $\hat{X}_{n+1}$ the $n+1$'th column of $\hat{X}$ we can write the previous problem in the SDP form

$$
\begin{aligned}
\inf_{\hat{X}\in\mathbb{S}^{n+1}} \; & \left\langle \hat{A}_0, \hat{X} \right\rangle \\
\text{s.t } : & \left\langle \hat{A}_i, \hat{X} \right\rangle \le \; 0 \; \forall i \in [r], \\
& D\hat{X}_{n+1} = d, \\
& 0 \le \hat{X}_{i,j} \le 1 \; \forall i, j \in [n+1], \\
& X_{n+1,n+1} = 1, \\
& \hat{X} \succeq 0.
\end{aligned}
\tag{$QSDP$}
$$

In Sect. 5 we test our methodology on random QCQPs using instances generated as in [8].

4.4 Quadratic knapsack problem

An interesting point arises whenever the quadratic forms determining the objective and the constraints do not have linear and constant terms, i.e. $b_i = c_i = 0 \; \forall i \in \{0, \dots, r\}$. In that case, our methodology takes $\mathscr{S} = \{v_1, \dots, v_{n+1}\}$ to be the eigenvectors of a matrix in $\mathbb{S}^{n+1}$ whose $n+1$'th row and column are 0. Hence, the constraints $v^\top \hat{X} v \ge 0$ in program $QSDP$ essentially ignore the last row and column of $\hat{X}$ and amount to the constraints $u_i^\top X u_i \ge 0$ where $u_1, \dots, u_n$ are a basis of eigenvectors of an aggregation of the A_i, $i \in [r]$. This is a weaker constraint than what we actually want, which is $u_i^\top \left(X - xx^\top\right) u_i \ge 0, \; i \in [n]$.

There are a few approaches we can consider to deal with this issue. For instance, we could choose to overlook it entirely and proceed by relaxing $QSDP$ to an LP, ignoring that the b_i are 0. Alternatively, if we have a linear constraint $b_i^\top x = c$, we may set $\hat{A}_0 = \begin{bmatrix} A_0 & b_i \\ b_i^\top & -2c \end{bmatrix}$ which shifts the objective by a constant. Finally, and perhaps more interestingly, we may use the constraints $u_i^\top \left(X - xx^\top\right) u_i \ge 0, \; i \in [n]$ directly, which can be equivalently rewritten as:

$$
u_i^\top X u_i \ge u_i^\top \left(xx^\top\right) u_i^\top = (u_i^\top x)^2 \; \forall i \in [n]. \tag{12}
$$

These are *second order cone constraints* which result in a second order cone relaxation of program QCQP depending on a set $\mathscr{S}$ of vectors u in $\mathbb{R}^n$. Such a program is both a relaxation of $QSDP$, and a strenghening of the linear relaxation that changes the constraint $\hat{X} \succeq 0$ for $u^\top X u \ge 0$ with $u \in \mathscr{S}$, for any finite set $\mathscr{S}$.

We test these different possibilities in Sect. 5 on instances of the *Quadratic Knapsack problem* [59] which is a QCQP of the form

$$
\begin{aligned}
& \max_{x\in\mathbb{R}^n} x^\top A_0 x \\
& s.t : \sum_{j=1}^{k} w_j x_j \le C, \; x \in \{0,1\}^n
\end{aligned}
\tag{QKP}
$$

where $w \in \mathbb{R}^n$, $A_0 \in \mathbb{S}^n$, $C \in \mathbb{R}_+$. It has been noted in the literature that the usual Shor semidefinite relaxation of this program is not very strong [36, 59] and one may add certain valid inequalities which result in the following tighter SDP:

$$
\begin{aligned}
&\max_{X \in \mathbb{S}^n} \langle A, X \rangle \\
s.t: &\sum_{j=1}^{n} w_j X_{ij} - C X_{ii} \leq 0 \, \forall i \in [n], \\
&X - diag(X) diag(X)^\top \succeq 0.
\end{aligned} \tag{QKPSDP}
$$

Using the idea before and a finite set $\mathscr{S}$ one may further relax this problem to obtain the second order cone program

$$
\begin{aligned}
&\max_{X \in \mathbb{S}^n} \langle A, X \rangle \\
s.t: &\sum_{j=1}^{n} w_j X_{ij} - C X_{ii} \leq 0 \, \forall i \in [n], \\
&u^\top X u \geq \left(u^\top diag(X) \right)^2 \; \forall u \in \mathscr{S}.
\end{aligned} \tag{QKSSOC}
$$

4.4.1 Extended trust region

In the previous problems it is not obvious how to linearly combine the matrices A_i, $i \in [r]$, that determine the quadratic forms to form the identity matrix, and hence we cannot apply Theorem 3 directly to arbitrary QCQPs. This motivates us to consider a variation where the identity matrix is explicitly one of the constraint matrices. This is the case of the *generalized trust region* problem [45]. That type of QCQPs consists in minimizing a quadratic function over the intersection of the unit ball and some half-spaces:

$$
\begin{aligned}
&\min_{x \in \mathbb{R}^n} x^\top Q x + 2 b^\top x, \\
&\text{s.t: } x^\top x \leq 1 \\
&\quad\; Dx \leq d
\end{aligned} \tag{TR}
$$

with $Q \in \mathbb{S}^n$, $b \in \mathbb{R}^n$, $D \in k \times n$ for some $k \in \mathbb{N}$ and $d \in \mathbb{R}^k$. Notice that the constraint $x^\top x \leq 1$ can be written as $x^\top I_n \, x \leq 1$. In Sect. 5 we test our methodology on a slightly more general version of this problem, where we keep some quadratic constraints. Abusing the language, we still refer to this family of problems as extended trust region problems. The SDP relaxation of these programs is as follows:

$$\begin{aligned} \min_{x\in\mathbb{R}^n, X\in\mathbb{R}^n} \quad & \langle Q, X\rangle + 2b^\top x, \\ \text{s.t:} \quad & \langle I_n, X\rangle \le 1, \\ & Dx \le d, \\ & \begin{bmatrix} X & x \\ x^\top & 1 \end{bmatrix} \succeq 0. \end{aligned} \tag{SDPTR}$$

5 Experimental results

In this section we present experimental results exhibiting the quality of our linear relaxations for the semidefinite relaxation of max cut, Lovász's theta number and on the SDP relaxations of families of QCQPs described in Sect. 4. For each of these problems, we will compare the optimal value of the linear relaxations to the optimal value of the SDP which they respectively relax by means of the quotient of the objective values. We contrast these quotients to the alternative of using Algorithm 1, starting with $\mathscr{S} = e_1, \dots, e_n$ and iteratively generating cuts using the SDP separation oracle. Whenever we fix an semidefinite program with some label SDP, we denote by $Iter_k(SDP)$ the linear program obtained at the $k-th$ iteration of Algorithm 1. For instance, $Iter_0(SDP)$ is simply dropping the semidefinite constraint of the SDP instance. We define z_n as the optimal value of $Iter_n(SDP)$. For each family of experiments, where we consider a certain SDP, we will denote by $z_{\mathscr{S}}$ the objective value of the corresponding linear relaxation obtained by following the ideas of Sect. 1. z_{sdp} will denote the objective value of the SDP instance. Although we consider different SDPs, there will not be danger of confusion as we caption of the figures and tables indicate which SDP we are addressing.

All of the code used is available at https://github.com/dderoux/Instance_specific_relaxations. To solve the resulting optimization programs we have used Mosek [6].[6]

5.1 Max cut

Denote by z_{sdp}, z_n and $z_{\mathscr{S}}$ the objective values of programs (GW), $Iter_n(GW)$ and ($SP_{\mathscr{S}}$) with $\mathscr{S}$ chosen as in Subsect 2.2. In this particular case, since the(GW) semidefinite program does not have linear constraints beyond the ones of the diagonal, the identity I is the only constraint matrix and it commutes with the objective matrix W. This means that $\mathscr{S}$ is simply a eigenbasis for the matrix W. As we proved in Theorem 5, ($SP_{\mathscr{S}}$) satisfies the eigenvalue bound for max cut. For each n ranging from 20 up to 200 in steps of 10, we generate 5 random graphs and plot the maximum, minimum and median of the quotients $\frac{z_{\mathscr{S}}}{z_{sdp}}$. We present our results for Erdős-Rényi random graphs in Figs. 1, 2 and 3. In Figs. 4, 5 and 6 we present those for $d-$regular random graphs.

[6] The experiments were performed on a 32 GB RAM ThinkPad Lenovo T490s machine running windows 10 with a Intel(R) Core(TM) i7-8665U CPU @ 1.90GHz 2.11 GHz.

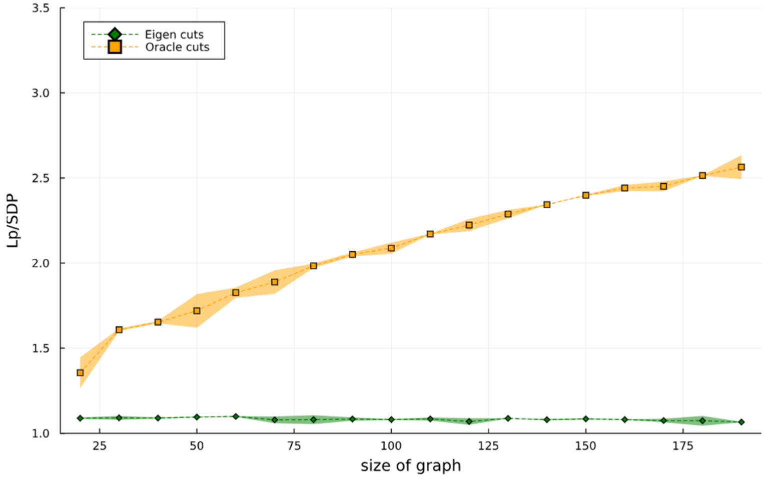

Fig. 1 Ratio of $\frac{z_{\mathscr{S}}}{z_{sdp}}$ (Eigen cuts) and $\frac{z_n}{z_{sdp}}$ (Oracle cuts) for instances of max cut where the graph has been sampled according to the Erdős-Rényi random model, with $p = 0.5$, as n grows

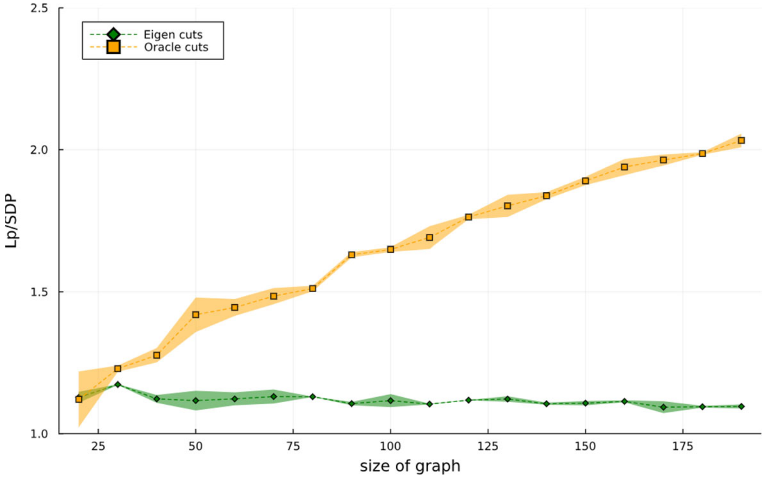

Fig. 2 Ratio of $\frac{z_{\mathscr{S}}}{z_{sdp}}$ (Eigen cuts) and $\frac{z_n}{z_{sdp}}$ (Oracle cuts) for instances of max cut where the graph has been sampled according to the Erdős-Rényi random model, with $p = 0.9$, as n grows

5.1.1 Comparison with the eigenvalue bound

In Tables 1 and 2, we compare the performance of $z_{\mathscr{S}}$ and the eigenvalue bound $\chi(G) := -n\lambda_n(G)$ on the graphs $\mathscr{G}(n, k, l)$ which we introduced Sect. 3, for different values of n, k and l. Since all of our experiments are random, we present averaged values over 5 instances, as well as the standard deviations of our results. Notice that the eigenvalue bound fails to give a small upper bound on the max cut value for this family of graphs. For the case $n = 400$, the bound fails completely, by giving a worse bound that the trivial upper bound of m for max cut. However, the linear program succeeds,

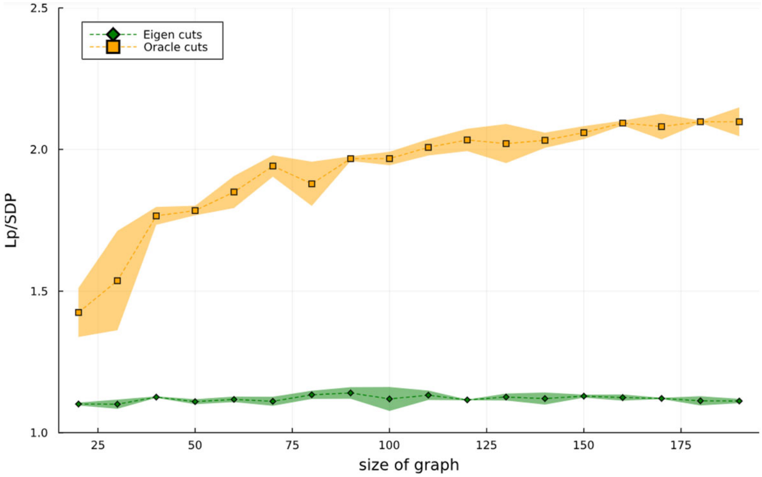

Fig. 3 Ratio of $\frac{z_{\mathscr{S}}}{z_{sdp}}$ (Eigen cuts) and $\frac{z_n}{z_{sdp}}$ (Oracle cuts) for instances of max cut where the graph has been sampled according to the Erdős-Rényi random model, with $p = \frac{3\log(n)}{n}$, as n grows

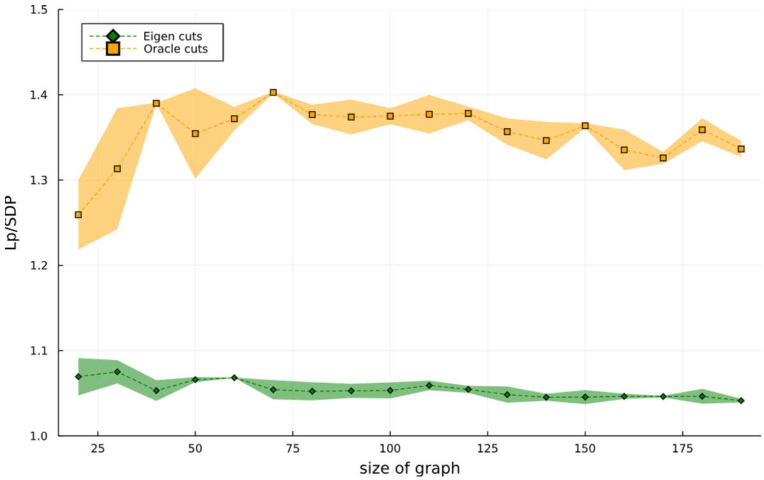

Fig. 4 Ratio of $\frac{z_{\mathscr{S}}}{z_{sdp}}$ (Eigen cuts) and $\frac{z_n}{z_{sdp}}$ (Oracle cuts) for instances of max cut where the graph is a random d-regular graph, with $d = 5$, as n grows

in all of our experiments, to have a quotient of at most 1.04 within the optimal value of the Goemans and Williamson relaxation.

5.2 Lovász theta number

Denote by z_{sdp}, z_n and $z_{\mathscr{S}}$ the objective values of programs Tn, $Iter_n(Tn)$ and LTn with $\mathscr{S}$ chosen as in Subsect. 2.2, respectively. For each n ranging from 20 to 200 in steps of 10, we generate 5 random graphs and plot the maximum, minimum and median

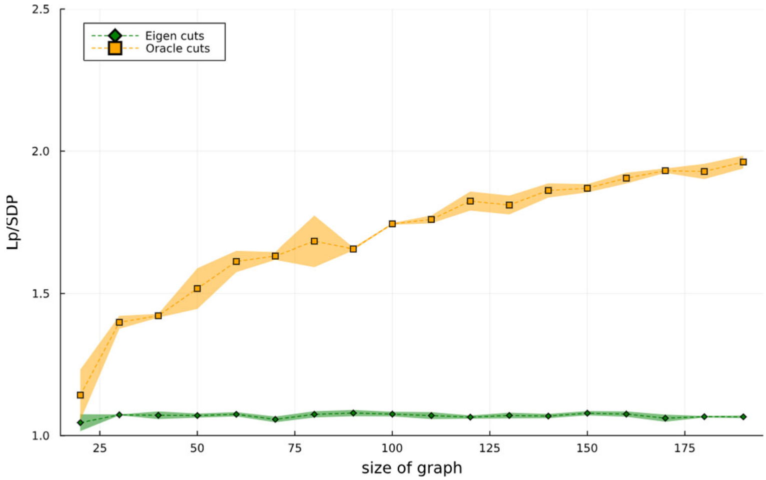

Fig. 5 Ratio of $\frac{z_{\mathcal{S}}}{z_{sdp}}$ (Eigen cuts) and $\frac{z_n}{z_{sdp}}$ (Oracle cuts) for instances of max cut where the graph is a random d-regular graph, with $d = \sqrt{n}$, as n grows

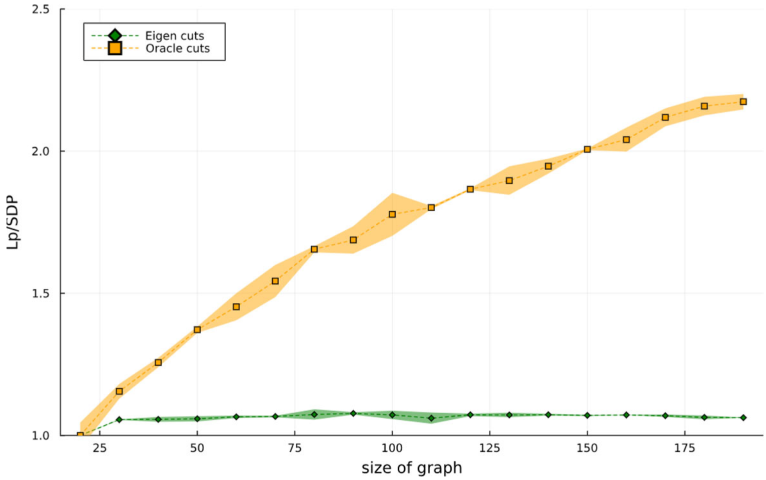

Fig. 6 Ratio of $\frac{z_{\mathcal{S}}}{z_{sdp}}$ (Eigen cuts) and $\frac{z_n}{z_{sdp}}$ (Oracle cuts) for instances of max cut where the graph is a random d-regular graph, with $d = \frac{n}{10}$, as n grows

of the quotients $\frac{z_{\mathcal{S}}}{z_{sdp}}$ and $\frac{z_n}{z_{sdp}}$ for these five instances. In the following subsections, we present these plots for Erdős-Rényi and random d-regular graphs.

5.2.1 Erdős-Rényi random graphs

In Figs. 7, 8 and 9 we plot the mentioned quotients for Erdős-Rényi random graph while we vary p, the probability of connecting two edges.

Table 1 Ratio of $\chi(G)$ to z_{sdp} and ratio of $z_{SP_{\mathscr{S}}}$ to z_{sdp} for $k = 4$ and $l = 5$.

n	$\chi(G)/z_{sdp}$: average(sd)	$z_{\mathscr{S}}/z_{sdp}$: average(sd)
64	1.241 (0.008)	1.020 (0.002)
100	1.417 (0.007)	1.017 (0.002)
196	1.760 (0.003)	1.012 (0.001)
400	2.289 (0.003)	1.010 (0.001)

Table 2 Ratio of $\chi(G)$ to z_{sdp} and ratio of $z_{\mathscr{S}}$ to z_{sdp} for $k = 6$ and $l = 10$

n	$\chi(G)/z_{sdp}$: average(sd)	$z_{\mathscr{S}}/z_{sdp}$: average(sd)
64	1.137 (0.008)	1.029 (0.002)
100	1.278 (0.007)	1.024 (0.001)
196	1.546 (0.005)	1.020 (0.001)
400	1.962 (0.002)	1.013 (0.001)

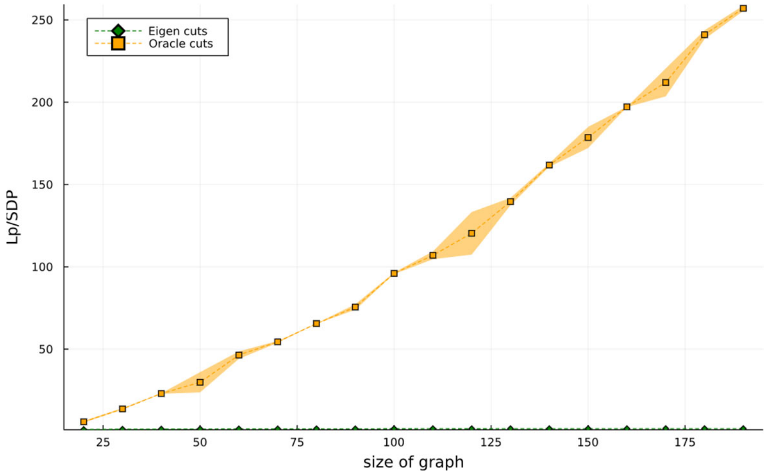

Fig. 7 Quotients for the Lovász theta number $\frac{z_{\mathscr{S}}}{z_{sdp}}$ (Eigen cuts) and $\frac{z_n}{z_{sdp}}$ (Oracle cuts) as n grows for Erdős-Rényi random graphs with $p = 0.5$

5.2.2 d-Regular random graphs

In Figs. 10, 11 and 12 we plot the mentioned quotients for $d-$regular random graph while we vary d.

5.3 Quadratically constrained quadratic problems

In this subsection we test the proposed methodology on the different QCQPs introduced in Sect. 4.

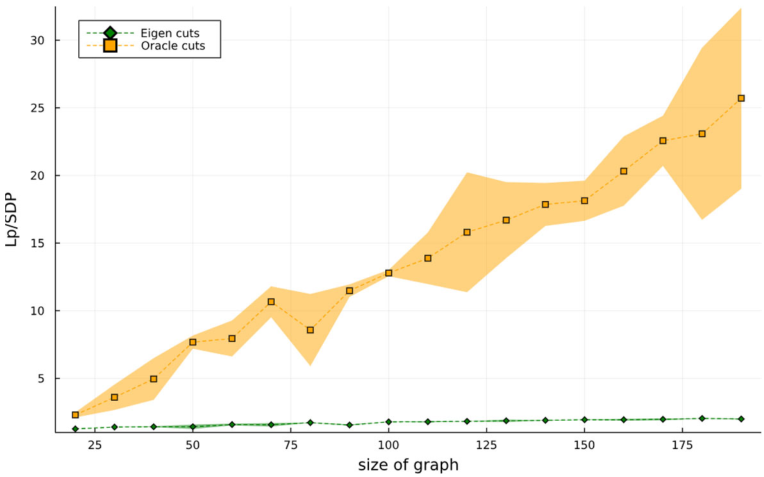

Fig. 8 Quotients for the Lovász theta number $\frac{z_{\mathscr{L}}}{z_{sdp}}$ (Eigen cuts) and $\frac{z_n}{z_{sdp}}$ (Oracle cuts) as n grows for Erdős-Rényi random graphs with $p = 0.9$

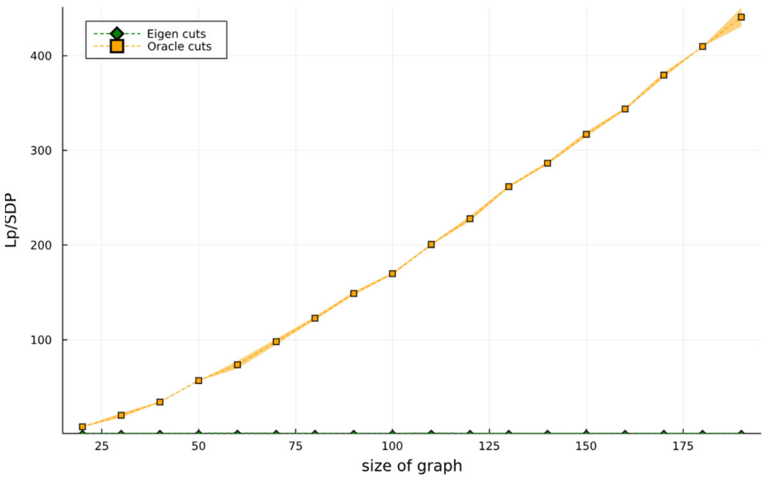

Fig. 9 Quotients for the Lovász theta number $\frac{z_{\mathscr{L}}}{z_{sdp}}$ (Eigen cuts) and $\frac{z_n}{z_{sdp}}$ (Oracle cuts) as n grows for Erdős-Rényi random graphs with $p = \frac{3\log(n)}{n}$

5.3.1 Random QCQPs

We generate random QCQPs following the review [8], where the authors compare various SDP relaxations of QCQPs in terms of percentage distance to the objective and solution time. For these instances, the x variables are bounded in an unit box $[0, 1]^n$ and the number of variables is varied from 20 up to 100 in steps of 10. The vectors c, d in $\mathbb{R}^{r+1}$ and $\mathbb{R}^q$ respectively and the matrices $D \in \mathbb{R}^{q\times n}$ and $A_i \in \mathbb{S}^n$, $i \in \{0, \ldots, r\}$ have entries drawn uniformly and independently at random from an

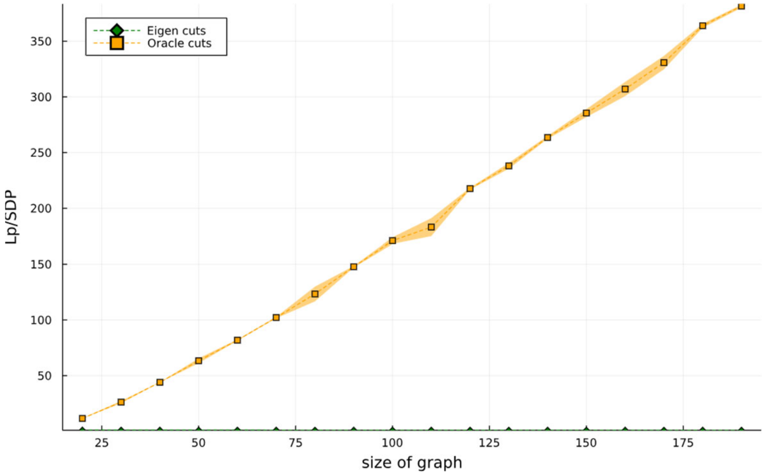

Fig. 10 Quotients for the Lovász theta number $\frac{z_{\mathscr{S}}}{z_{sdp}}$ and $\frac{z_n}{z_{sdp}}$ as n grows for random d-regular graphs with $d = 5$

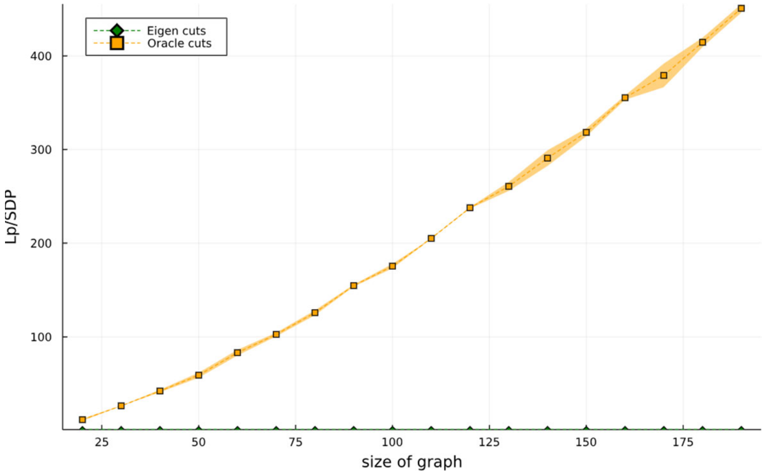

Fig. 11 Quotients for the Lovász theta number $\frac{z_{\mathscr{S}}}{z_{sdp}}$ and $\frac{z_n}{z_{sdp}}$ as n grows for random d-regular graphs with $d = \sqrt{n}$

uniform distribution supported in $[-1, 1]$. The vector $b \in \mathbb{R}^{r+1}$ has entries sampled uniformly at random from an uniform distribution supported in $[0, 100]$. We set the density Δ of A_0 to 0.5, which corresponds to the percentage of nonzero elements of the matrix, on average.

Since QCPQs are highly sensitive to the number of quadratic constraints, we test different combinations of number of quadratic and linear constraints, according to the following combinations:

- QCQPs with $r = 1, q = \frac{n}{10}$.
- QCQPs with $r = 1, q = \frac{n}{5}$.

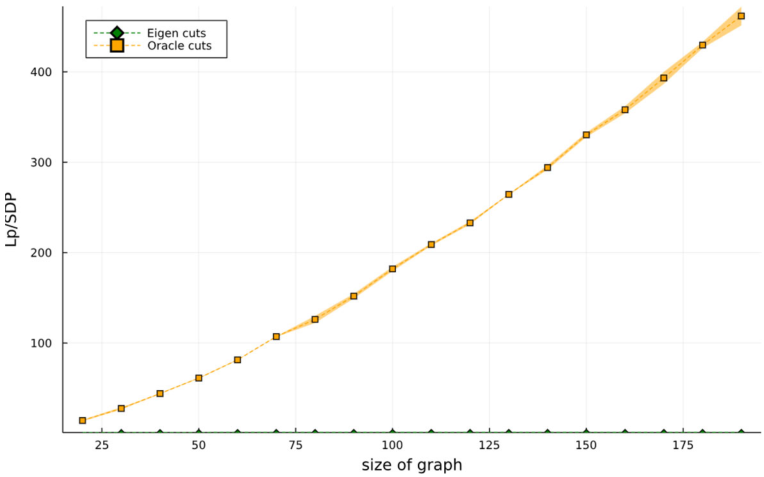

Fig. 12 Quotients for the Lovász theta number $\frac{z_{\mathscr{S}}}{z_{sdp}}$ and $\frac{z_n}{z_{sdp}}$ as n grows for random d-regular graphs with $d = \frac{n}{10}$

- QCQPs with $r = \frac{n}{2}, q = \frac{n}{10}$
- QCQPs with $r = n, q = \frac{n}{10}$.

For a given combination of these parameters and a value of n we generate 5 random instances and solve the following optimization programs for each:

- Problem $QSDP$. We denote the objective value of this semidefinite program by z_{sdp}.
- The linear relaxation $L_{\mathscr{S}}$ of $QSDP$ where we let $\mathscr{S}$ the elements of a eigenvector basis of the matrix A_0. We denote the objective value of this problem by $z_{\mathscr{S}}$.
- The LP $Iter_n(QSDP)$. We denote by z_n the objective value of this program.
- The LP $Iter_0(QSDP)$. We denote by z_0 the objective value of this program.
- The second order cone program obtained by dropping the constraint $\hat{X} \succeq 0$ from $QSDP$ adding the constraints (12) with $\mathscr{S}$ the elements of a eigenvector basis of the matrix A_0. We denote the objective value of this problem by z_{soc}.

We average the values of ratios $\frac{z_{\mathscr{S}}}{z_{sdp}}$, $\frac{z_n}{z_{sdp}}$, $\frac{z_0}{z_{sdp}}$ and $\frac{z_{soc}}{z_{sdp}}$ over the five instances, and plot the results in Figs. 13, 14, 15 and 16. In Appendix C[7] we include figures for varying values of the density Δ.

We observe that due to randomness, it will not be possible to linearly combine the matrices $A_1, \ldots, A_m, A_{m+1}$ so that they commute with the objective matrix A_0. Therefore, program CG will typically return the 0 matrix, and $\mathscr{S}$ will simply be a basis of eigenvectors of A_0.

For these instances, the quality of all the relaxations is encouraging, with the trivial relaxation obtained by dropping the semidefinite constraint getting a ratio of at most 4 in all of our experiments. The second order cone relaxation is typically the better as soon as n exceeds 50. Whenever the density increases, we notice that the ratios $\frac{z_{\mathscr{S}}}{z_{sdp}}$

[7] Appendix C is included in the online companion of this paper.

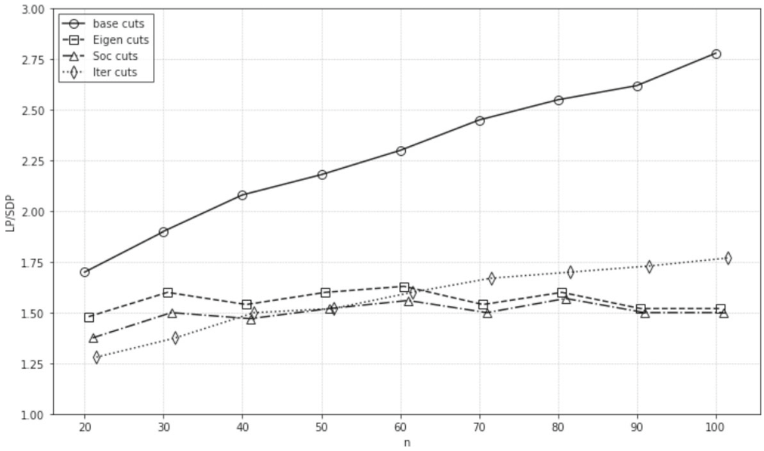

Fig. 13 Quality of the ratios $\frac{z_{\mathscr{S}}}{z_{sdp}}$ (eigen cuts), $\frac{z_n}{z_{sdp}}$ (oracle cuts), $\frac{z_0}{z_{sdp}}$ (base cuts), and $\frac{z_{soc}}{z_{sdp}}$ for random QCQP instances with $r = 1$, and $q = \frac{n}{5}$

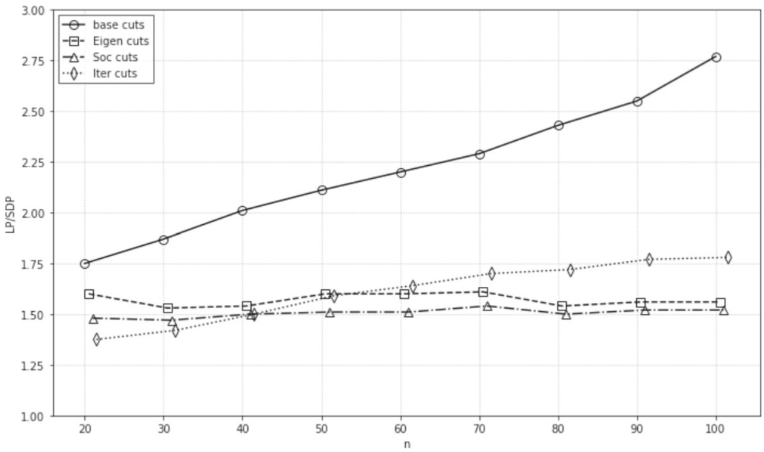

Fig. 14 Quality of the ratios $\frac{z_{\mathscr{S}}}{z_{sdp}}$ (eigen cuts), $\frac{z_n}{z_{sdp}}$ (oracle cuts), $\frac{z_0}{z_{sdp}}$ (base cuts), and $\frac{z_{soc}}{z_{sdp}}$ for random QCQP instances with $r = 1$, and $q = \frac{n}{10}$

and $\frac{z_{soc}}{z_{sdp}}$ get closer and closer, hinting at that the second order cone relaxation is not much stronger than the linear relaxation. Although the LP $Iter_n$ achieves a better ratio for small n, this is no longer true for larger values of n. In addition, notice that for a value of n this LP requires solving n LPs and n eigenvector decompositions.

5.3.2 Extended trust region problems

We now consider instances of the extended trust region problem with extra quadratic constraints, as presented in Subsect. 4.3. These instances are the same as in the previous subsection, but with the added quadratic constraint $x^\top I_n x \leq 1$. We present our results

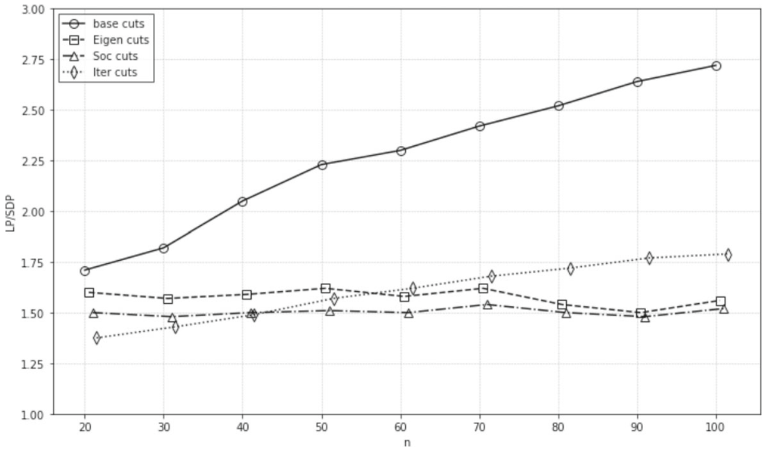

Fig. 15 Quality of the ratios $\frac{z_{\mathscr{S}}}{z_{sdp}}$ (eigen cuts), $\frac{z_n}{z_{sdp}}$ (oracle cuts), $\frac{z_0}{z_{sdp}}$ (base cuts), and $\frac{z_{soc}}{z_{sdp}}$ for random QCQP instances with $r = \frac{n}{2}$, and $q = \frac{n}{10}$

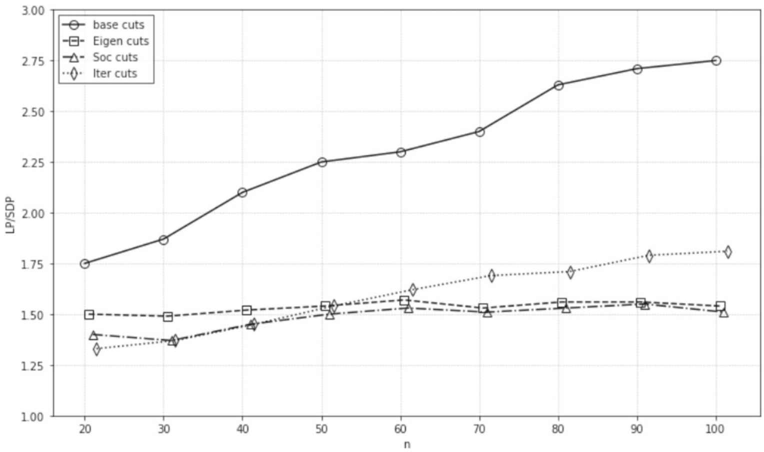

Fig. 16 Quality of the ratios $\frac{z_{\mathscr{S}}}{z_{sdp}}$ (eigen cuts), $\frac{z_n}{z_{sdp}}$ (oracle cuts), $\frac{z_0}{z_{sdp}}$ (base cuts), and $\frac{z_{soc}}{z_{sdp}}$ for random QCQP instances with $r = n$, and $q = \frac{n}{10}$

in Figs. 17, 18, 19 and 20, where the density of the matrix A_0 is set to $\Delta = 0.5$. In Appendix C we include figures for other values of Δ.

The results for these experiments are similar across the different densities. In all of our experiments, the second order cone relaxation and the linear relaxation $L_{\mathscr{S}}$ of the extended trust region problem are very strong with the ratio to the SDP relaxation being very close to 1. Moreover, this ratio does not get worse as n increases, quite in sharp contrast to the base relaxation $Iter_0$ of objective value z_0 and the LP $Iter_n$, which gets a ratio worse than 50 whenever n exceeds 100. For these instances, program $L_{\mathscr{S}}$ specialized to the extended trust region problem and program QKSSOC certify the dual bounds provided by Theorem 3, which we believe is the reason of the effectiveness of these relaxations.

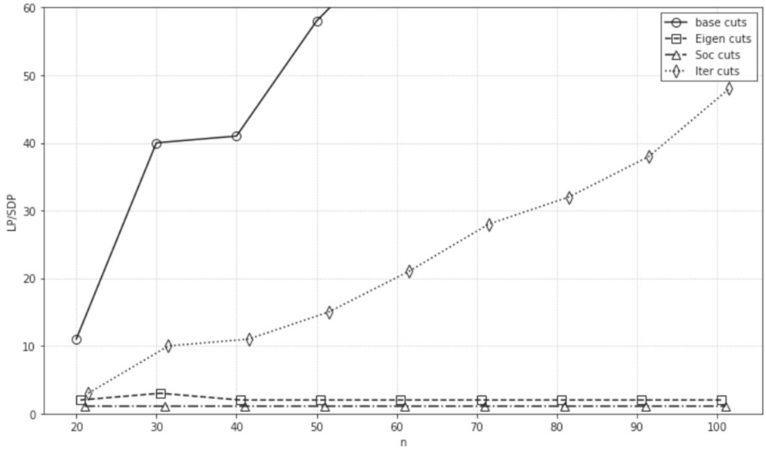

Fig. 17 Quality of the ratios $\frac{z_{\mathscr{S}}}{z_{sdp}}$ (eigen cuts), $\frac{z_n}{z_{sdp}}$ (oracle cuts), $\frac{z_0}{z_{sdp}}$ (base cuts), and $\frac{z_{soc}}{z_{sdp}}$ for instances of the extended trust region problem with $r = 1$, and $q = \frac{n}{5}$

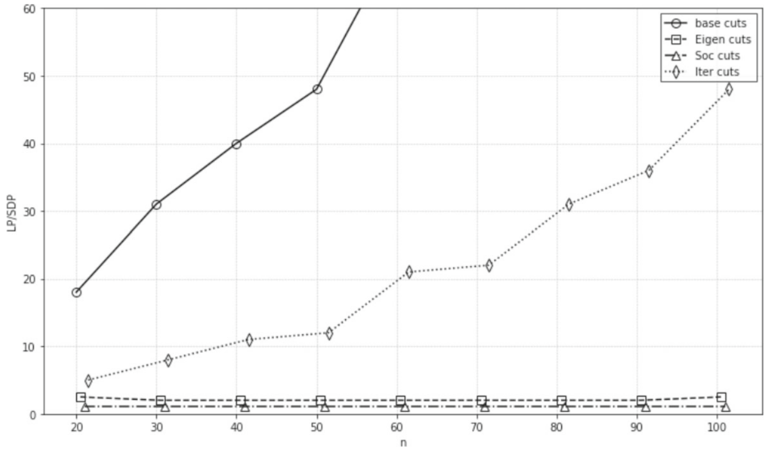

Fig. 18 Quality of the ratios $\frac{z_{\mathscr{S}}}{z_{sdp}}$ (eigen cuts), $\frac{z_n}{z_{sdp}}$ (oracle cuts), $\frac{z_0}{z_{sdp}}$ (base cuts), and $\frac{z_{soc}}{z_{sdp}}$ for instances of the extended trust region problem with $r = 1$, and $q = \frac{n}{10}$

5.3.3 Quadratic knapsack problem

We now consider instances of the quadratic knapsack problem as presented in Subsect. 4.3. In this family of problems, the linear term b_0 in the objective is 0, and therefore we can consider the different strategies mentioned in Sect. 4.3. Hence, for each instance we solve 6 programs, as follows:

– Problem QKPSDP. We denote the objective value of this semidefinite program by z_{sdp}.

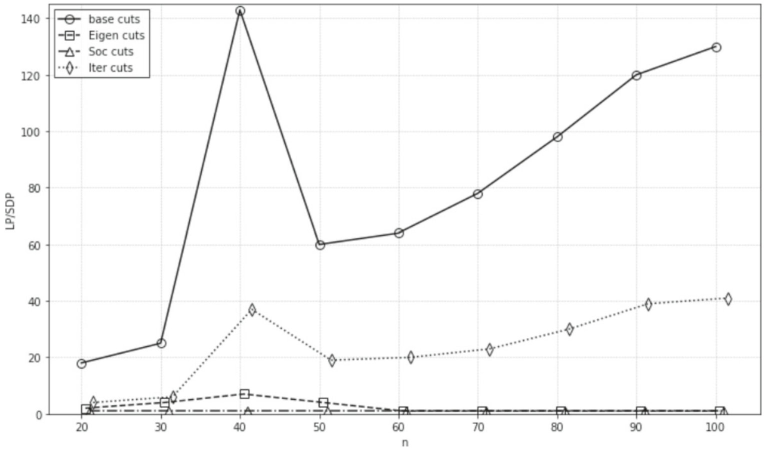

Fig. 19 Quality of the ratios $\frac{z_{\mathscr{S}}}{z_{sdp}}$ (eigen cuts), $\frac{z_n}{z_{sdp}}$ (oracle cuts), $\frac{z_0}{z_{sdp}}$ (base cuts), and $\frac{z_{soc}}{z_{sdp}}$ for instances of the extended trust region problem with $r = \frac{n}{2}$, and $q = \frac{n}{10}$

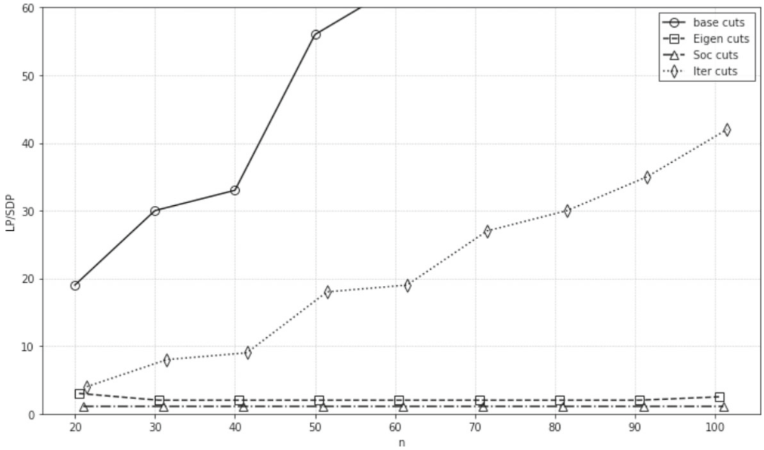

Fig. 20 Quality of the ratios $\frac{z_{\mathscr{S}}}{z_{sdp}}$ (eigen cuts), $\frac{z_n}{z_{sdp}}$ (oracle cuts), $\frac{z_0}{z_{sdp}}$ (base cuts), and $\frac{z_{soc}}{z_{sdp}}$ for instances of the extended trust region problem with $r = n$, and $q = \frac{n}{10}$

- The linear relaxation $L_{\mathscr{S}}$ of QKPSDP where we let $\mathscr{S}$ the elements of a eigenvector basis of the matrix A_0. We denote the objective value of this problem by $z_{\mathscr{S}}$.
- The LP obtained by ignoring that the b_i are 0. We denote the objective value of this problem by $z'_{\mathscr{S}}$.
- The LP $Iter_n(QSDP)$. We denote by z_n the objective value of this program.
- The LP $Iter_0(QSDP)$. We denote by z_0 the objective value of this program.
- The second order cone relaxation of QKPSDP given by program QKSSOC. We denote by z_{SOC} the objective value of this program.

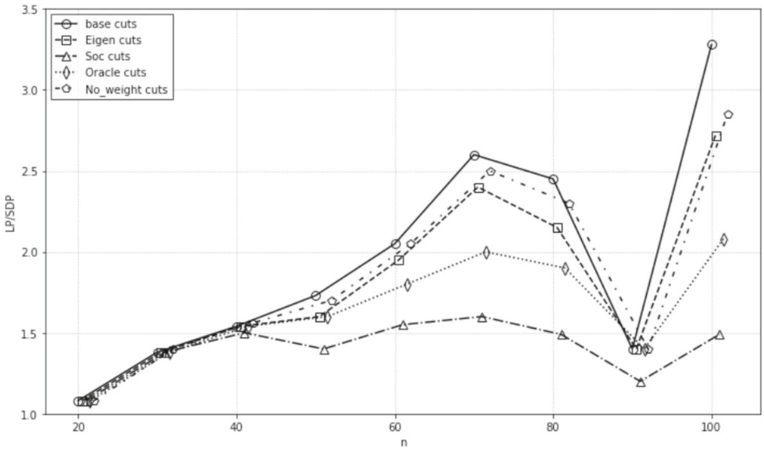

Fig. 21 Quality of the ratios $\frac{z_{\mathscr{S}}}{z_{sdp}}$ (eigen cuts), $\frac{z'_{\mathscr{S}}}{z_{sdp}}$ (no_weight cuts), $\frac{z_n}{z_{sdp}}$ (oracle cuts), $\frac{z_0}{z_{sdp}}$ (base cuts), and $\frac{z_{soc}}{z_{sdp}}$ for instances of the quadratic knapsack problem with density $\Delta = 0.05$

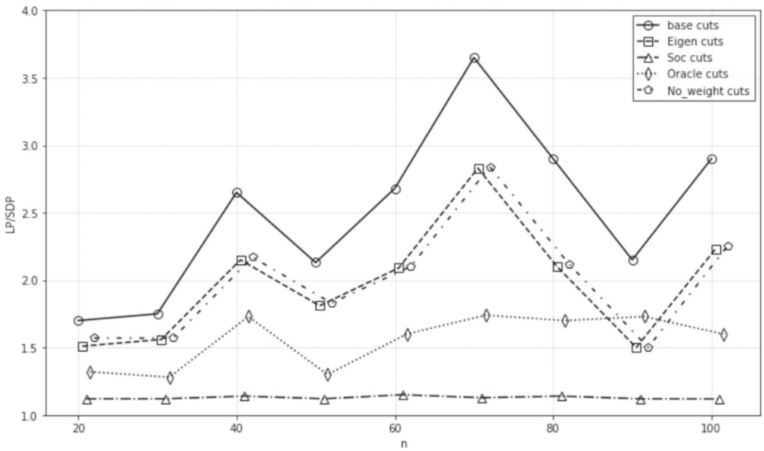

Fig. 22 Quality of the ratios $\frac{z_{\mathscr{S}}}{z_{sdp}}$ (eigen cuts), $\frac{z'_{\mathscr{S}}}{z_{sdp}}$ (no_weight cuts), $\frac{z_n}{z_{sdp}}$ (oracle cuts), $\frac{z_0}{z_{sdp}}$ (base cuts), and $\frac{z_{soc}}{z_{sdp}}$ for instances of the quadratic knapsack problem with density $\Delta = 0.25$

The instances were generated following [59], who specify instances that have become the standard to computationally test this optimization problem. Namely, we first set a *density* value $\Delta \in [0, 1]$, which corresponds to the percentage of nonzero elements of the matrix A_0. Each weight w_j, $j \in [n]$ is uniformly randomly distributed in $[1, 50]$. The ij entry of A_0 equals the ji entry and is nonzero with probability Δ, in which case it is uniformly distributed in $[1, 100]$, $i, j \in [n]$. The capacity C of the knapsack is taken uniformly at random from the interval $[50, \sum_{j=1}^{n} w_j]$. We present our results in Figs 21, 22, 23, 24, 25 and 26.

For this family of problems, all relaxations are within reasonable bounds of the SDP objective value. It is nonetheless appealing that the second order cone relaxation

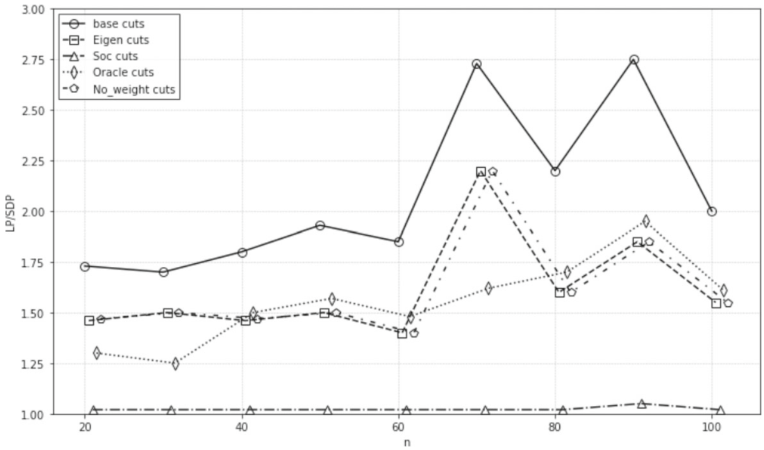

Fig. 23 Quality of the ratios $\frac{z_{\mathscr{S}}}{z_{sdp}}$ (eigen cuts), $\frac{z'_{\mathscr{S}}}{z_{sdp}}$ (no_weight cuts), $\frac{z_n}{z_{sdp}}$ (oracle cuts), $\frac{z_0}{z_{sdp}}$ (base cuts), and $\frac{z_{soc}}{z_{sdp}}$ for instances of the quadratic knapsack problem with density $\Delta = 0.5$

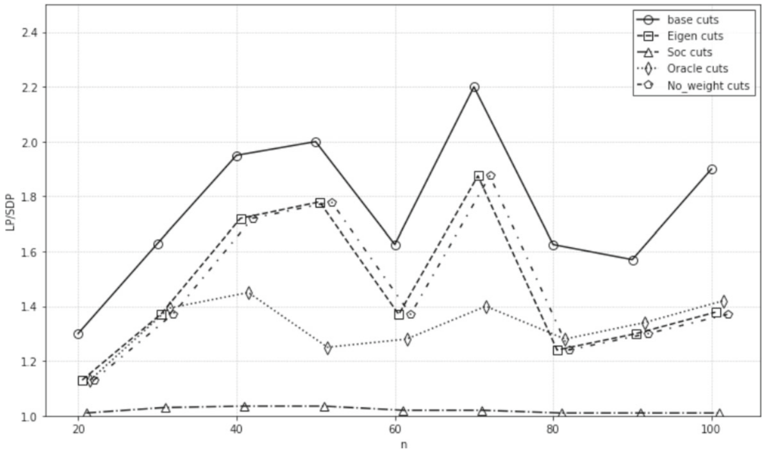

Fig. 24 Quality of the ratios $\frac{z_{\mathscr{S}}}{z_{sdp}}$ (eigen cuts), $\frac{z'_{\mathscr{S}}}{z_{sdp}}$ (no_weight cuts), $\frac{z_n}{z_{sdp}}$ (oracle cuts), $\frac{z_0}{z_{sdp}}$ (base cuts), and $\frac{z_{soc}}{z_{sdp}}$ for instances of the quadratic knapsack problem with density $\Delta = 0.75$

performs very well, with the ratio to the objective of the SDP nearly 1, regardless of the value of n. The relaxation $L_{\mathscr{S}}$ seems to perform similarly to $Iter_n$.

5.4 Computational time considerations

Algorithm 1 offers a meta-algorithm to solve semidefinite programs. Ideally, choosing appropriate starting sets $\mathscr{S}$ to initialize the algorithm will result in better solving times. It is critical then that solving program $L_{\mathscr{S}}$ or a second order cone strenghening takes significantly less time than solving the SDP. In what follows, we report solving times of the different programs proposed.

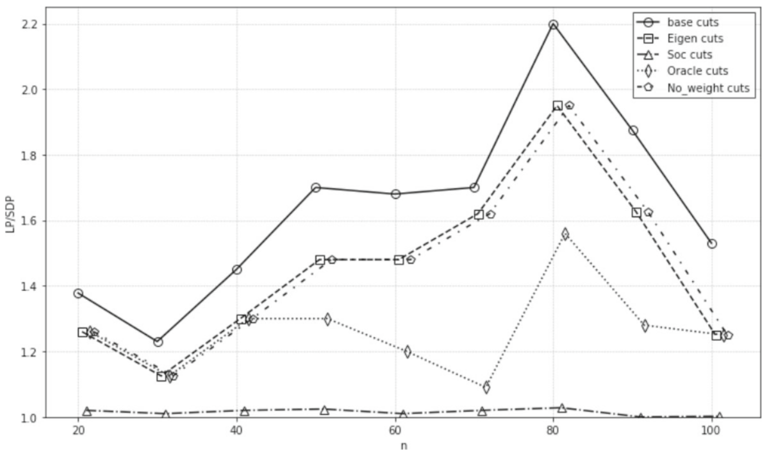

Fig. 25 Quality of the ratios $\frac{z_{\mathscr{S}}}{z_{sdp}}$ (eigen cuts), $\frac{z'_{\mathscr{S}}}{z_{sdp}}$ (no_weight cuts), $\frac{z_n}{z_{sdp}}$ (oracle cuts), $\frac{z_0}{z_{sdp}}$ (base cuts), and $\frac{z_{soc}}{z_{sdp}}$ for instances of the quadratic knapsack problem with density $\Delta = 0.95$

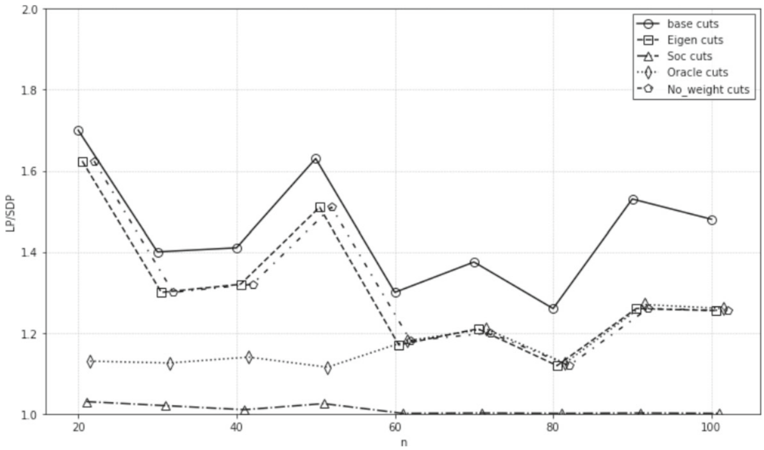

Fig. 26 Quality of the ratios $\frac{z_{\mathscr{S}}}{z_{sdp}}$ (eigen cuts), $\frac{z'_{\mathscr{S}}}{z_{sdp}}$ (no_weight cuts), $\frac{z_n}{z_{sdp}}$ (oracle cuts), $\frac{z_0}{z_{sdp}}$ (base cuts), and $\frac{z_{soc}}{z_{sdp}}$ for instances of the quadratic knapsack problem with density $\Delta = 1$

For the max cut and the Lovász theta number we consider Erdős-Rényi random graphs on 270 and 200 vertices respectively. The probability of adding an edge between two vertices is set to $p = 0.75$. We repeat the experiments for 3 instances and report the average solving time and worst ratio of the LP to the SDP objective value among the three instances.

- **Max cut:** The worst ratio found was 1.08. The average solving time of the SDP was 0.47 seconds. The average solving time of the LP was 9.77 seconds.
- **Theta number:** The worst ratio found was 6.7. The average solving time of the SDP was 2994 seconds. The average solving time of the LP was 39 seconds.

We proceed by reporting the solving times for the quadratic knapsack, random QCQPs and the Extended Trust Region problem. We consider problems with 270 variables. For the Trust Region and random QCQPs we set the number of quadratic constraints to 10, and the number of linear constraints to 20. For each problem, we generate 3 instances as described previously, setting the density Δ to 0.75. We report the average solving time, worst ratio of the LP to the SDP objective and worst ratio of the SOC to the SDP value among the three instances.

– **Trust region:** The average solving time of the SDP was 7476 seconds. The average solving time of the LP was 216 seconds. The average solving time of the SOC was 8.9 seconds. The worst ratio found for the LP was 2.49, and the worst ratio found for the SOC was 1.17.
– **Random QCQPs:** The average solving time of the SDP was 6510 seconds. The average solving time of the LP was 13 seconds. The average solving time of the SOC was 21 seconds. The worst ratio found for the LP was 1.49, and the worst ratio found for the SOC was 1.48.
– **Knapsack:** The average solving time of the SDP was 7422 seconds. The average solving time of the LP was 15 seconds. The average solving time of the SOC was 20 seconds. The worst ratio found for the LP was 1.844, and the worst ratio found for the SOC was 1.01.

It is noteworthy that solving the max cut SDP is faster by 4 orders of magnitude than the all of the other semidefinite programs considered in this paper. In addition, it is quite surprising that the SOC relaxations of the QCQPs have solving times comparable to that of the LPs. In particular, the solving time of the SOC is two orders of magnitude faster than the LP for the trust region problems. We point out that very strong, fast and scalable, specialized algorithms for semidefinite programs such as the max cut problem and the Lováz theta number exist, such as [32, 75, 77], and therefore alternatives such as an outer approximation algorithm as 1 might not be appealing for these problems.

6 Summary and future work

In this work, we introduced a generic technique to obtain linear and second order cone relaxations of semidefinite programs with provable guarantees based on the commutativity of the constraints and objective matrices. We believe that other algebraic properties of these matrices can be exploited to obtain further stronger relaxations. Although we believe solving semidefinite programs with linear programs is an interesting topic is its own right, we posit that our ideas can be exploited in settings where linear approximations of convex regions is an essential component of state-of-the-art algorithms, such as in copositive programming [16] and outer approximation algorithms for semidefinite integer programs [47].

On the theoretical side, the main remaining question regarding the max cut problem is if the proposed linear program $SP_{\mathscr{S}}$ provides a better-than-2 approximation algorithm. From our computational tests, we are not aware of any instance where the approximation factor is worse than 1.8.

For the Lovász theta number, the main theoretical question is if our proposed linear program satisfies the same inequalities that $\vartheta(\bar{G})$ does. Namely,

$$\alpha(G) \leq \vartheta(\bar{G}) \leq \chi(G)$$

where $\alpha(G)$ and $\chi(G)$ are the clique and chromatic numbers of G, respectively. It would be interesting as well to find out if program Tn satisfies the bound (9) for $d-$regular graphs. Finally, the second order cone relaxations for the knapsack and extended trust region problems performed well in terms of both solving time and objective value. It would be then worthwhile to explore the specialization of Algorithm 1 to these problems, and to compare its behaviour to state of the art algorithms for those problems.

A Missing proofs

Here we present a proof of Observation 1.

Proof We first describe the dual of program $L_{\mathscr{S}}$ for a generic set $\mathscr{S} = \{s_1, \ldots, s_k\}$. This program is given by:

$$\begin{aligned} \max_{y\in\mathbb{R}^n,\ \alpha\in\mathbb{R}^n_+} \quad & b^\top y \\ \text{s.t: } C - \sum_{i=1}^{r} y_i A_i &= \sum_{i=1}^{k} \alpha_i s_i s_i^\top. \end{aligned} \tag{$DL_{\mathscr{S}}$}$$

Notice that for any set $\mathscr{S} = \{s_1, \ldots, s_k\}$, program ($DL_{\mathscr{S}}$) is a restriction of (DSDP) as the matrices $C - \sum_i y_i A_i$ are restricted to belong to the convex cone generated by the PSD matrices $s_i s_i^T$, $i \in [k]$, rather than the whole set of positive semidefinite matrices. It follows that the optimal value of (DSDP) upper bounds the optimal value of ($DL_{\mathscr{S}}$) for any set $\mathscr{S}$. By hypothesis, both (SDP) and its dual are strictly feasible and therefore solvable by strong conic duality. Hence, we let $\mathscr{S}^*$ be the elements of a basis of $\mathbb{R}^n$ of orthonormal eigenvectors of an optimal solution S^* of program DSDP. The dual of $L_{\mathscr{S}^*}$ is then $\max_{y\in\mathbb{R}^n\ \alpha\in\mathbb{R}^n_+} b^\top y$ subject to $C - \sum_{i=1}^m y_i A_i = \sum_{i=1}^k \alpha_i v_i v_i^\top$. Hence, letting $y_i = y_i^*$ and $\alpha_i = \beta_i$ gives a feasible solution to $DL_{\mathscr{S}^*}$ which matches the optimal value of DSDP and hence is optimal. To conclude, observe that strong linear duality holds and therefore $L_{\mathscr{S}^*}$ is solvable and its optimal value equals that of DSDP. □

We now present a direct proof of Theorem 5.

Proof of Theorem 5 (Proof of Theorem 5)
The inequality holds if we are able to show a feasible solution of the dual program of ($SP_{\mathscr{S}}$) whose objective value equals $-n\lambda_n$. Let $\mathscr{S} = \{v_1, v_2, \ldots, v_n\} = \mathscr{E}(W)$. Consider an eigenvector v of W with corresponding eigenvalue λ, so that $Wv = \lambda v$. Observe that $\lambda - \lambda_n \geq 0$ since λ_n is the most negative eigenvalue of W. Clearly, the vector v is a eigenvector of the matrix $W - \lambda_n I_n$ with corresponding eigenvalue

$\lambda - \lambda_n$. By the spectral theorem, we have $W - \lambda_n I_n = \sum_{i=1}^n (\lambda_i - \lambda_n) v_i v_i^T$, where $v_1, \ldots, v_n$ are an orthonormal eigenbasis of W. In other words, we have:

$$\lambda_n I_n + \sum_{i=1}^n (\lambda_i - \lambda_n) v_i v_i^T = W.$$

This yields the desired feasible solution $\Lambda = -\lambda_n I_n$ which has an objective value $\frac{1}{2}m - \frac{n}{4}\lambda_n$ for $DL_{\mathscr{S}}$. □

We next derive the dual of program $SP_{\mathscr{S}}$.

Lemma 5 *The dual of program* $SP_{\mathscr{S}}$ *is given by program* $DSP_{\mathscr{S}}$.

Proof For this proof we ignore the constant $\frac{1}{2}m$ in the objective together with the multiplicative term $\frac{1}{4}$. Introduce dual variables $\lambda_i \in \mathbb{R}$ for $i \in [n]$ corresponding to the constraints $X_{ii} = 1$, $\beta_i \in \mathbb{R}^n_+$ for $i \in [k]$ corresponding to $v^\top X v \geq 0$ and $\alpha_{i,j}, \delta_{i,j} \geq 0$ for $i \neq j \in [n]$, corresponding to $X_{ij} \leq 1$ and $X_{ij} \geq -1$ respectively, for $i \neq j \in [n]$ (in fact we need only to consider the indices $i < j$ since X is symmetric but we will ignore this as it only complicates the proof). Multiplying the dual variables with the constraints accordingly gives the inequality

$$\sum_{i=1}^n \lambda_i X_{ii} + \sum_{i=1}^k \beta_i \langle X, v_i v_i^\top \rangle - \sum_{i \neq j} \alpha_{ij} X_{ij} + \sum_{i \neq j} \delta_{ij} X_{ij} \geq \sum_{i=1}^n \lambda_i - \sum_{i \neq j} \delta_{ij} - \sum_{i \neq j} \alpha_{ij}$$

Let $\Lambda_{ij} = \delta_{ij} - \alpha_{ij}$ for $i \neq j$ and $\Lambda_{ii} = \lambda_i$ for all $i \in [n]$. This gives the inequality

$$\langle \Lambda, X \rangle + \sum_{i=1}^k \beta_i \langle X, v_i v_i^\top \rangle \geq \sum_{i=1}^n \Lambda_{ii} - \sum_{i \neq j} \Lambda_{ij}.$$

If we let $\Lambda + \sum_{i=1}^k \beta_i v_i v_i^\top = W$ we get $\langle -W, X \rangle \leq \sum_{i \neq j} \Lambda_{ij} - \sum_{i=1}^n \Lambda_{ii}$ and this completes the proof. □

Acknowledgements The authors wish to thank the associate editor and three anonymous reviewers, whose insightful comments led us to extend our ideas to the broader setting of semidefinite optimization programs.

Funding Open Access funding provided by Carnegie Mellon University This material is based upon work supported by the U. S. Office of Naval Research under award number N00014-21-1-2243.

Data availibility All data used to conduct our experiments is publicly available.

Code Availability The full code was made available for review, and can be found at https://github.com/dderoux/Instance_specific_relaxations. We remark that a set of packages were used in this article, that were either open source or available for academic use. Specific references are included in the reference section.

Declarations

Conflict of interest The authors declare that they have no Conflict of interest.

References

1. Ahmadi, A.A.: Algebraic relaxations and hardness results in polynomial optimization and lyapunov analysis. arXiv preprint arXiv:1201.2892 (2012)
2. Ahmadi, A.A., Hall, G.: Sum of squares basis pursuit with linear and second order cone programming. Algebr. Geom. Methods Discrete Math. **685**, 27–53 (2017)
3. Ahmadi, A.A., Majumdar, A.: Dsos and sdsos optimization: more tractable alternatives to sum of squares and semidefinite optimization. SIAM J. Appl. Algebr. Geom. **3**(2), 193–230 (2019)
4. Alizadeh, F.: Interior point methods in semidefinite programming with applications to combinatorial optimization. SIAM J. Optim. **5**(1), 13–51 (1995)
5. Alon, N., Sudakov, B.: Bipartite subgraphs and the smallest eigenvalue. Comb. Probab. Comput. **9**(1), 1–12 (2000)
6. ApS, M.: The MOSEK optimization toolbox for MATLAB manual. Version 10.0. (2022). http://docs.mosek.com/9.0/toolbox/index.html
7. Baltean-Lugojan, R., Bonami, P., Misener, R., Tramontani, A.: Scoring positive semidefinite cutting planes for quadratic optimization via trained neural networks. preprint: http://www.optimization-online.org/DB_HTML/2018/11/6943.html (2019)
8. Bao, X., Sahinidis, N.V., Tawarmalani, M.: Semidefinite relaxations for quadratically constrained quadratic programming: A review and comparisons. Math. Program. **129**, 129–157 (2011)
9. Barahona, F., Mahjoub, A.R.: On the cut polytope. Math. Program. **36**(2), 157–173 (1986)
10. Ben-Tal, A., Den Hertog, D.: Hidden conic quadratic representation of some nonconvex quadratic optimization problems. Math. Program. **143**, 1–29 (2014)
11. Ben-Tal, A., Nemirovski, A.: Lectures on modern convex optimization: analysis, algorithms, and engineering applications. SIAM (2001)
12. Bertsimas, D., Cory-Wright, R.: On polyhedral and second-order cone decompositions of semidefinite optimization problems. Oper. Res. Lett. **48**(1), 78–85 (2020)
13. Bhardwaj, A., Rostalski, P., Sanyal, R.: Deciding polyhedrality of spectrahedra. SIAM J. Optim. **25**(3), 1873–1884 (2015)
14. Braun, G., Fiorini, S., Pokutta, S., Steurer, D.: Approximation limits of linear programs (beyond hierarchies). Math. Oper. Res. **40**(3), 756–772 (2015)
15. Brouwer, A.E., Cohen, Neumaier, A.: Distance-Regular Graphs. Springer-Verlag (1989)
16. Bundfuss, S., Dür, M.: An adaptive linear approximation algorithm for copositive programs. SIAM J. Optim. **20**(1), 30–53 (2009)
17. Chan, S.O., Lee, J.R., Raghavendra, P., Steurer, D.: Approximate constraint satisfaction requires large lp relaxations. J. ACM (JACM) **63**(4), 1–22 (2016)
18. Charikar, M., Makarychev, K., Makarychev, Y.: Integrality gaps for sherali-adams relaxations. In: Proceedings of the forty-first annual ACM Symposium on Theory of Computing, pp. 283–292 (2009)
19. Chung, F., Radcliffe, M.: On the spectra of general random graphs. The Electronic Journal of Combinatorics pp. P215–P215 (2011)
20. Coey, C., Lubin, M., Vielma, J.P.: Outer approximation with conic certificates for mixed-integer convex problems. Math. Program. Comput. **12**(2), 249–293 (2020)
21. Conrad, K.: Simultaneous commutativity of operators. University of Connecticut (1992)
22. d'Aspremont, A., Ghaoui, L., Jordan, M., Lanckriet, G.: A direct formulation for sparse pca using semidefinite programming. Advances in neural information processing systems **17** (2004)
23. Delorme, C., Poljak, S.: Laplacian eigenvalues and the maximum cut problem. Math. Program. **62**(1–3), 557–574 (1993)

24. Dey, S.S., Kazachkov, A.M., Lodi, A., Munoz, G.: Cutting plane generation through sparse principal component analysis. URL http://www.optimization-online.org/DB_HTML/2021/02/8259.html (2021)
25. Deza, M.M., Laurent, M.: Geometry of cuts and metrics. Springer, Cham (2009)
26. Fawzi, H.: On polyhedral approximations of the positive semidefinite cone. Math. Oper. Res. **46**(4), 1479–1489 (2021)
27. Feige, U., Ofek, E.: Spectral techniques applied to sparse random graphs. Random Struct. Algorithms **27**(2), 251–275 (2005)
28. Friedman, J., Kahn, J., Szemeredi, E.: On the second eigenvalue of random regular graphs. In: Proceedings of the twenty-first annual ACM Symposium on Theory of Computing, pp. 587–598 (1989)
29. Frieze, A., Karoński, M.: Introduction to Random Graphs. Cambridge University Press, Cambridge (2016)
30. Gally, T., Pfetsch, M.E.: Computing restricted isometry constants via mixed-integer semidefinite programming. preprint, submitted (2016)
31. Gally, T., Pfetsch, M.E., Ulbrich, S.: A framework for solving mixed-integer semidefinite programs. Optimization Methods and Software **33**(3), 594–632 (2018)
32. Garstka, M., Cannon, M., Goulart, P.: Cosmo: a conic operator splitting method for convex conic problems. J. Optim. Theory Appl. **190**(3), 779–810 (2021)
33. Goemans, M.X., Williamson, D.P.: Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. J. ACM (JACM) **42**(6), 1115–1145 (1995)
34. Golub, G.H., Van Loan, C.F.: Matrix computations. JHU press (2013)
35. Helmberg, C., Rendl, F.: A spectral bundle method for semidefinite programming. SIAM J. Optim. **10**(3), 673–696 (2000)
36. Helmberg, C., Rendl, F., Weismantel, R.: A semidefinite programming approach to the quadratic knapsack problem. J. Comb. Optim. **4**, 197–215 (2000)
37. Henrion, D., Garulli, A.: Positive polynomials in control, vol. 312. Springer Science & Business Media, Cham (2005)
38. Hojny, C., Pfetsch, M.E.: Handling symmetries in mixed-integer semidefinite programs. In: International Conference on Integration of Constraint Programming, Artificial Intelligence, and Operations Research, pp. 69–78. Springer (2023)
39. Horn, R.A., Johnson, C.R.: Matrix analysis. Cambridge University Press, Cambridge (2012)
40. Kelley, J.E., Jr.: The cutting-plane method for solving convex programs. J. Soc. Ind. Appl. Math. **8**(4), 703–712 (1960)
41. Kothari, P.K., Meka, R., Raghavendra, P.: Approximating rectangles by juntas and weakly exponential lower bounds for lp relaxations of csps. SIAM Journal on Computing **0**(0), STOC17–305 (2021)
42. Krishnan, K., Mitchell, J.E.: A unifying framework for several cutting plane methods for semidefinite programming. Optimization Methods Softw. **21**(1), 57–74 (2006)
43. Lanckriet, G.R., Cristianini, N., Bartlett, P., Ghaoui, L.E., Jordan, M.I.: Learning the kernel matrix with semidefinite programming. J. Mach. Learn. Res. **5**, 27–72 (2004)
44. Lasserre, J.B.: Global optimization with polynomials and the problem of moments. SIAM J. Optim. **11**(3), 796–817 (2001)
45. Locatelli, M.: Exactness conditions for an sdp relaxation of the extended trust region problem. Optim. Lett. **10**(6), 1141–1151 (2016)
46. Lovász, L.: On the shannon capacity of a graph. IEEE Trans. Inf. Theory **25**(1), 1–7 (1979)
47. Lubin, M., Yamangil, E., Bent, R., Vielma, J.P.: Extended formulations in mixed-integer convex programming. In: Integer Programming and Combinatorial Optimization: 18th International Conference, IPCO 2016, Liège, Belgium, June 1-3, 2016, Proceedings 18, pp. 102–113. Springer (2016)
48. Lubotzky, A., Phillips, R., Sarnak, P.: Ramanujan graphs. Combinatorica **8**(3), 261–277 (1988)
49. Majumdar, A., Hall, G., Ahmadi, A.A.: A survey of recent scalability improvements for semidefinite programming with applications in machine learning, control, and robotics. arXiv preprint arXiv:1908.05209 (2019)
50. Majumdar, A., Hall, G., Ahmadi, A.A.: Recent scalability improvements for semidefinite programming with applications in machine learning, control, and robotics. Ann. Rev. Control Robot. Auton. Syst. **3**, 331–360 (2020)
51. Mirka, R., Williamson, D.P.: An experimental evaluation of semidefinite programming and spectral algorithms for max cut. In: 20th International Symposium on Experimental Algorithms (SEA 2022). Schloss Dagstuhl-Leibniz-Zentrum für Informatik (2022)

52. Mohar, B., Poljak, S.: Eigenvalues and the max-cut problem. Czechoslov. Math. J. **40**(2), 343–352 (1990)
53. Nemirovski, A.: Interior point polynomial time methods in convex programming. Lecture notes pp. 3215–3224 (2004)
54. Nesterov, Y., et al.: Semidefinite relaxation and nonconvex quadratic optimization. Université catholique de Louvain, Center for Operations Research and Econometrics, Tech. rep. (1997)
55. O'Donnell, R., Schramm, T.: Sherali–adams strikes back. arXiv preprint arXiv:1812.09967 (2018)
56. O'donoghue, B., Chu, E., Parikh, N., Boyd, S.: Conic optimization via operator splitting and homogeneous self-dual embedding. J. Optim. Theory Appl. **169**, 1042–1068 (2016)
57. Parrilo, P.A.: Structured semidefinite programs and semialgebraic geometry methods in robustness and optimization. California Institute of Technology (2000)
58. Parrilo, P.A.: Semidefinite programming relaxations for semialgebraic problems. Math. Program. **96**(2), 293–320 (2003)
59. Pisinger, D.: The quadratic knapsack problem-a survey. Discret. Appl. Math. **155**(5), 623–648 (2007)
60. Poljak, S., Rendl, F.: Nonpolyhedral relaxations of graph-bisection problems. SIAM J. Optim. **5**(3), 467–487 (1995)
61. Poljak, S., Tuza, Z.: The expected relative error of the polyhedral approximation of the max-cut problem. Oper. Res. Lett. **16**(4), 191–198 (1994)
62. Qualizza, A., Belotti, P., Margot, F.: Linear programming relaxations of quadratically constrained quadratic programs. In: Mixed Integer Nonlinear Programming, pp. 407–426. Springer (2012)
63. Ramana, M.V.: Polyhedra, spectrahedra, and semidefinite programming. Topics in semidefinite and interior-point methods, Fields Institute Communications **18**, 27–38 (1997)
64. Reinelt, G.: Tsplib-a traveling salesman problem library. ORSA J. Comput. **3**(4), 376–384 (1991)
65. Rossi, R.A., Ahmed, N.K.: The network data repository with interactive graph analytics and visualization. In: AAAI (2015). http://networkrepository.com
66. Sherali, H.D., Fraticelli, B.M.: Enhancing rlt relaxations via a new class of semidefinite cuts. J. Global Optim. **22**(1), 233–261 (2002)
67. Shor, N.Z.: Dual quadratic estimates in polynomial and boolean programming. Ann. Oper. Res. **25**(1), 163–168 (1990)
68. Sivaramakrishnan, K.K.: Linear programming approaches to semidefinite programming problems. Rensselaer Polytechnic Institute (2002)
69. Spielman, D.: Spectral graph theory. Combinatorial scientific computing **18** (2012)
70. Tikhomirov, K., Youssef, P.: The spectral gap of dense random regular graphs. Ann. Probab. **47**(1), 362–419 (2019)
71. Vandenberghe, L., Boyd, S.: Semidefinite programming. SIAM review **38**(1), 49–95 (1996)
72. de la Vega, W.F., Kenyon-Mathieu, C.: Linear programming relaxations of maxcut. In: Proceedings of the eighteenth annual ACM-SIAM Symposium on Discrete Algorithms, pp. 53–61 (2007)
73. Vinzant, C.: What is a spectrahedron. Notices Amer. Math. Soc **61**(5), 492–494 (2014)
74. Wang, A.L., Kılınç-Karzan, F.: On the tightness of sdp relaxations of qcqps. Math. Program. **193**(1), 33–73 (2022)
75. Wang, Y., Deng, K., Liu, H., Wen, Z.: A decomposition augmented lagrangian method for low-rank semidefinite programming. SIAM J. Optim. **33**(3), 1361–1390 (2023)
76. Wang, Y., Tanaka, A., Yoshise, A.: Polyhedral approximations of the semidefinite cone and their application. Comput. Optim. Appl. **78**(3), 893–913 (2021)
77. Wen, Z., Yin, W.: A feasible method for optimization with orthogonality constraints. Math. Program. **142**(1–2), 397–434 (2013)
78. Yang, H., Liang, L., Carlone, L., Toh, K.C.: An inexact projected gradient method with rounding and lifting by nonlinear programming for solving rank-one semidefinite relaxation of polynomial optimization. Math. Program. **201**(1–2), 409–472 (2023)
79. Yang, L., Sun, D., Toh, K.C.: Sdpnal+: a majorized semismooth newton-cg augmented lagrangian method for semidefinite programming with nonnegative constraints. Math. Program. Comput. **7**(3), 331–366 (2015)

80. Yonekura, K., Kanno, Y.: Global optimization of robust truss topology via mixed integer semidefinite programming. Optim. Eng. **11**, 355–379 (2010)
81. Zhao, X.Y., Sun, D., Toh, K.C.: A newton-cg augmented lagrangian method for semidefinite programming. SIAM J. Optim. **20**(4), 1737–1765 (2010)