Outer-Product-Free Sets for Polynomial Optimization and Oracle-Based Cuts

Daniel Bienstock[1], Chen Chen[1], and Gonzalo Muñoz[1]

[1]Industrial Engineering & Operations Research, Columbia University, New York, NY, USA

April 26, 2017

Abstract

Cutting planes are derived from specific problem structures, such as a single linear constraint from an integer program. This paper introduces cuts that involve minimal structural assumptions, enabling the generation of strong polyhedral relaxations for a broad class of problems. We consider valid inequalities for the set $S \cap P$, where S is a closed set, and P is a polyhedron. Given an oracle that provides the distance from a point to S we construct a pure cutting plane algorithm; if the initial relaxation is a polytope, the algorithm is shown to converge. Cuts are generated from convex forbidden zones, or S-free sets derived from the oracle. We also consider the special case of polynomial optimization. Polynomial optimization may be represented using a symmetric matrix of variables, and in this lifted representation we can let S be the set of matrices that are real, symmetric outer products. Accordingly we develop a theory of *outer-product-free* sets. All maximal outer-product-free sets of full dimension are shown to be convex cones and we identify two families of such sets. These families can be used to generate intersection cuts that can separate any infeasible extreme point of a linear programming relaxation in polynomial time. Moreover, in the special case of polynomial optimization we derive strengthened oracle-based intersection cuts that can also ensure separation in polynomial time.

1 Introduction

Consider a generic mathematical program of the following form

$$\min c^T x$$
$$\text{subject to } x \in S \cap P.$$

Here $P := \{x \in \mathbb{R}^n | Ax \leq b\}$ is a polyhedral set, $c \in \mathbb{R}^n$, and $S \subset \mathbb{R}^n$ is a closed set. This paper concerns the construction of linear programming (LP) relaxations of the form $\min c^T x | x \in P_0$ with some polyhedron P_0 defining the relaxed feasible region, $P_0 \supseteq S \cap P$. A natural choice of relaxation is to set $P_0 := P$; however, this may be a poor approximation of the original problem. This paper focuses generating effective polyhedral relaxations via cutting-plane algorithms; this cutting plane approach is crucial to branch-and-cut methods (e.g. [7, 20, 65, 69, 82]) for global optimization. At each iteration we have a vector $\bar{x} \notin S \cap P$ which is an extreme point optimal solution to the current relaxation; a halfspace H is generated such that $H \supseteq S \cap P$ and $\bar{x} \notin H$. We obtain a sequence $P_0, P_1, ..., P_k$ of polyhedra containing the feasible region; thus the associated relaxations can all be solved efficiently with linear programming (as long as P_0 and the cuts are described with rational data), providing iteratively tighter dual bounds for the original problem.

There are many ways to generate cuts such as: disjunctions [10], lift-and-project [59], algebraic arguments (e.g. [6, 41, 43, 62]), combinatorics (see [85]), and convex outer-approximation (e.g. [48]). We adopt the geometric perspective, in which cuts are derived from convex forbidden zones, or S-free sets. The convexity requirement on S-free sets is essential in the generation of intersection cuts [9], although nonconvex forbidden zones with special structure can also be used to generate cuts (e.g. cross cuts [55]).

The S-free approach has traditionally been explored in the context of mixed-integer programming; we shall consider a different setting involving minimal structural assumptions on S. Suppose there is an oracle that provides the distance from a point to S. This distance can be approximated to arbitrary accuracy in the case of mixed-integer linear programming (using rounding operations), polynomial optimization (using eigenvalues, see Section 3), and a cardinality constraint (see Section 6.2). Theorem 3.3 establishes that, given the initial relaxation P_0 is a polytope, the oracle (or an arbitrarily close approximation thereof) enables a finite-time cutting plane algorithm that constructs a polyhedron arbitrarily close to $\text{conv}(S \cap P)$. Hence an explicit functional characterization of S is not necessary to produce a strong relaxation.

We also focus on a more specific problem, polynomial optimization (aka polynomial programming):

$$\begin{array}{ll} & \min\ p_0(x) \\ (\mathbf{PO}) & \text{s.t. } p_i(x) \leq 0 \quad i = 1, ..., m, \end{array}$$

where each p_k is a polynomial function with respect to the decision vector $x \in \mathbb{R}^n$. Polynomial optimization generalizes important classes of problems such as quadratic programming, and has numerous applications in engineering; moreover, the quadratic representation of binary variables can also be useful for generating strong relaxations of discrete problems (e.g. MAXCUT [40]).

We work with a representation of **PO** that uses a symmetric matrix of decision variables, and let S be the set of symmetric matrices that can be represented as a real, symmetric outer product — accordingly we study *outer-product-free* sets. Two families of full-dimensional maximal outer-product-free sets are identified in Theorems 4.7 and 4.10, which characterize all such sets in the space of 2×2 symmetric real matrices. Furthermore, we derive an oracle-based outer-product-free set in Section 4.2. We derive cuts (see Section 5) from these outer-product-free sets using the intersection cut, which enables separation of any infeasible extreme point of a (lifted) polyhedral relaxation of **PO** in polynomial time. Computational experiments in Section 6 demonstrate the potential of these intersection cuts.

In global optimization, cuts typically rely on particular substructures (see e.g. [47]) and target single terms or functions to derive cuts [15,56,60,64,73,79–81]. In contrast, we develop cuts that can account for all variables of the problem simultaneously. To the best of our knowledge there are two papers (applicable to polynomial optimization) that are similar in this regard. The disjunctive cuts of Saxena, Bonami, and Lee [74,75] apply to mixed-integer programming with nonconvex quadratic constraints (MIQCP) with all variables bounded; bounded polynomial optimization problems can be transformed to bounded MIQCP. Their disjunctive cuts are derived from a mixed-integer linear programming (MILP) approximation of the problem. Ghaddar, Vera, and Anjos [39] propose a lift-and-project method, generating cuts to strengthen a given moment relaxation by separating over a higher-moment relaxation. They show that the method can be interpreted as a generalization of the lift-and-project of Balas, Ceria, and Cornuéjols [11] for mixed-integer linear programming. Polynomial-time separation for their procedure is not guaranteed in general, but some guarantees regarding separation and convergence can be made in the special case of nonnegative variables and in the case of binary variables.

The remainder of the paper is organized as follows. Section 2 describes S-free sets, the intersection cut, and a cut strengthening procedure. Section 3 develops the oracle-based cut. Section 4 studies outer-product-free sets. Section 5 describes cut generation using outer-product-free sets. Section 6 provides numerical examples and details computational experiments. Section 7 concludes.

1.1 Notation

Denote the interior of a set int(.) and its boundary bd(.). The convex hull of a set is denoted conv(.), and its closure is clconv(.); likewise, the conic hull of a set is cone(.), and its closure clcone(.). The set of extreme points of a convex set is ext(.). For a point x and nonempty set S in $\mathbb{R}^n$, we define $d(x, S) := \inf_{s \in S}\{\|x - s\|_2\}$; note that for S closed we can replace the infimum with minimum. Denote the ball with center x and radius r to be $\mathcal{B}(x, r)$. For a square matrix X, $X_{[i,j]}$ denotes the 2×2 principal submatrix induced by indices $i \neq j$. $\langle \cdot, \cdot \rangle$ denotes the matrix inner product. A positive semidefinite matrix may be referred to as a PSD matrix for short, and likewise NSD refers to negative semidefinite.

2 S-free Sets and the Intersection Cut

Definition 2.1. A set $C \subset \mathbb{R}^n$ is *S-free* if $\text{int}(C) \cap S = \emptyset$ and C is convex.

For any S-free set C we have $S \cap P \subseteq \text{clconv}(S \setminus \text{int}(C))$, and so any valid inequalities for $\text{clconv}(S \setminus \text{int}(C))$ are valid for $S \cap P$. Larger S-free sets can be useful for generating deeper cuts [26].

Definition 2.2. An S-free set C is *maximal* if $V \not\supset C$ for all S-free V.

Under certain conditions (see [17, 26, 28, 50]), maximal S-free sets are sufficient to generate all nontrivial cuts for a problem. When $S = \mathbb{Z}^n$, C is called a lattice-free set. Maximal lattice-free sets are well-studied in integer programming theory [4, 5, 16, 23, 32, 43, 50, 58], and the notion of S-free sets was introduced as an extension [33].

So far we have left aside discussion on how precisely one can derive cuts from an S-free set C. Averkov [8] provides theoretical consideration on the matter; for instance, characterizing when $\text{conv}(P \setminus \text{int}(C))$ is a polyhedron. In specific instances, $\text{conv}(P \setminus \text{int}(C))$ can be fully described; for example, Bienstock and Michalka [22] provide a characterization of the convex hull when S is given by the epigraph of a convex function excluding a polyhedral or ellipsoidal region (also see [66], [19], [50]). A standard procedure for generating cuts (but not necessarily the entire hull) is to find a simplicial cone P' containing P and apply Balas' intersection cut [9] for $P' \setminus \text{int}(C)$. We shall adopt this approach, which has been studied in great detail (see [26–28]). The first use of simplicial cones to generate cuts can be attributed to Tuy [83] for minimization of a concave function over a polyhedron; such cuts are named Tuy cuts, concavity cuts, or convexity cuts. The distinction is that Tuy cuts are objective cuts whereas intersection cuts are feasibility cuts. Balas and Margot [13] propose strengthened intersection cuts by using a tighter relaxation of P. Porembski [70, 71] proposes a method for strengthening the Tuy cut by using different conic relaxations.

2.1 The Intersection Cut

Let $P' \supseteq P$ be a simplicial conic relaxation of P: a displaced polyhedral cone with apex $\bar{x}$ and defined by the intersection of n linearly independent halfspaces. Any n linearly independent constraints describing P can be used to define a simplicial conic relaxation; consequently any extreme point of P can be used as the apex of P'. A simplicial cone may be written as follows:

$$P' = \{x|x = \bar{x} + \sum_{j=1}^{n} \lambda_j r^{(j)}, \lambda \geq 0\}. \tag{1}$$

Each extreme ray is of the form $\bar{x} + \lambda_j r^{(j)}$, where each $r^{(j)} \in \mathbb{R}^n$ is a direction and $\lambda \in \mathbb{R}^n_+$ is a vector of scaling factors.

We shall assume $\bar{x} \notin S$, so that $\bar{x}$ is to be separated from S. So let C be an S-free set with $\bar{x}$ in its interior. Since $\bar{x} \in \text{int}(C)$, there must exist $\lambda > 0$ such that $\bar{x} + \lambda_j r^{(j)} \in \text{int}(C)\ \forall j$. Also, each extreme ray is either entirely contained in C, i.e. $\bar{x} + \lambda_j r^{(j)} \in \text{int}(C)\, \forall \lambda_j \geq 0$, or else there is an intersection point with the boundary bd(.): $\exists \lambda_j^* : \bar{x} + \lambda_j^* r^{(j)} \in \text{bd}(C)$. The intersection cut is the halfspace whose boundary contains each intersection point (given by λ_j^*) and that is parallel to all extreme rays contained in C.

Balas [9, Theorem 2] provides a closed-form expression for the cut $\pi^T x \leq \pi_0$. Consider P' in inequality form, $P' = \{x|Ax \leq b\}$, where $\bar{x} = A^{-1}b$.

Then the coefficients can be expressed as

$$\pi_0 = \sum_{\forall i} (1/\lambda_i^*) b_i - 1,\ \pi_j = \sum_{\forall i} (1/\lambda_i^*) a_{ij}. \tag{2}$$

Note that the rays are assumed to be aligned with the inequality form, in the sense that for each j, $r^{(j)}$ is the jth column of the inverse of A. Furthermore, $1/\lambda_j^*$ is treated as zero if the step length is infinite [9, p. 34].

Having obtained the hyperplane coefficients π, π_0 we must determine the correct direction of the cut, so let $\beta_0 := \text{sgn}(\pi^T \bar{x} - \pi_0)$, where sgn(.) denotes the sign function. The intersection cut is:

$$\beta_0(\pi^T x - \pi_0) \leq 0. \tag{3}$$

Let $V := \{x|\beta_0(\pi^T x - \pi_0) \leq 0\}$ be the halfspace defined by the intersection cut. For completeness we include a proof of validity of V (a fact established in the original paper by Balas [9, Theorem 1]), and furthermore establish a condition in which the cut gives the convex hull of $P' \setminus \text{int}(C)$.

Proposition 2.3. $V \supseteq P' \setminus \text{int}(C)$. *Furthermore, if all step lengths are finite, then* $V \cap P' = \text{conv}(P' \setminus \text{int}(C))$.

Proof. Let $\bar{V}$ be the complement of V, i.e. an open halfspace. It suffices to establish that $Q := \bar{V} \cap P' \subseteq \text{int}(C)$. Let K be the index set of extreme rays of P' which have finite step lengths λ^* for the intersection cut, and let $\bar{K}$ be the (complementary) index set of extreme rays corresponding to infinite step lengths. By construction of the cut

$$Q = \{x | x = \bar{x} + \sum_{i \in K} \lambda_i r^{(i)} + \sum_{j \in \bar{K}} \lambda_j r^{(j)}, \lambda \geq 0, \lambda_i < \lambda_i^* \forall i \in K\}.$$

Denote $v := \bar{x} + \sum_{i \in K} \lambda_i r^{(i)}$ and $w := \sum_{j \in \bar{K}} \lambda_j r^{(j)}$ so that $x = v + w$. Now v is in the polytope $P'' := \{x | x = \bar{x} + \sum_{i \in K} \lambda_i r^{(i)}, 0 \leq \lambda_i \leq \lambda_i^*\}$. The extreme points of P'' are $\bar{x}$ and some intersection points forming the intersection cut. v cannot be described solely as the convex combination of intersection points, as this would imply $v \in V$. Hence either $v = \bar{x}$ or v is the strict convex combination of $\bar{x} \in \text{int}(C)$ and some of the intersection points — in either case $v \in \text{int}(C)$.

Now observe that w is in the recession cone of C since by construction each extreme ray indexed by $\bar{K}$ is contained in C. Hence $x = v + w$ is also in the interior of C, which establishes $V \supseteq P' \setminus \text{int}(C)$.

Now suppose all step lengths are finite. In this case we may write

$$V \cap P' = \{x | x = \bar{x} + \sum_{i=1}^{n} \lambda_i r^{(i)}, \lambda \geq \lambda^*\}.$$

Alternatively,

$$V \cap P' = \{x | x = \sum_{i=1}^{n} \lambda_i (\bar{x} + \sum_{j=1}^{n} \lambda_j r^{(i)}) / \sum_{j=1}^{n} \lambda_j, \lambda \geq \lambda^*\}.$$

Hence every point in $V \cap P'$ is the convex combination of points $m_i := \bar{x} + \sum_{j=1}^{n} \lambda_j r^{(i)},\ i = 1, ..., n$. Since $\sum_{j=1}^{n} \lambda_j r^{(i)} \geq \lambda_i^* \forall i$ and the ray $r^{(i)}$ emanates from an interior point of C passing through the boundary at step length λ_i^*, then $m_i \in P' \setminus \text{int}(C)\ \forall i$. Hence $V \cap P' \subseteq \text{conv}(P' \setminus \text{int}(C))$.

□

2.2 Strengthening the Intersection Cut

If some of the extreme rays of P' are contained in C, then the intersection cut is not in general sufficient to capture the convex hull of $P' \setminus \text{int}(C)$. Negative step lengths λ_j^* can be used to derive stronger cuts. This was noted for the case of polyhedral S-free sets by Dey and Wolsey [33], and later for general S-free sets by Basu, Cornuéjols, and Zambelli [18] (see also [16]). We shall provide a general procedure for determining such step lengths.

A simple example of the issue is shown in Figure 1. Here $S \subset \mathbb{R}^2$ is given by the halfspace in $x_1 + x_2 \geq 1$. We let C be the (unique) maximal S-free set, $x_1 + x_2 \leq 1$. Now define the simplicial cone P' with inequalities $x_1 \geq 0$ and $x_2 \leq 0$. The apex of P' is at the origin, and the extreme ray directions are $r^{(1)} = (1, 0), r^{(2)} = (0, -1)$. The intersection points with C are at $(1, 0)$ along $r^{(1)}$ and infinity along $r^{(2)}$. The intersection cut is given by $x_1 \geq 1$. The best cut possible from C, however is $x_1 + x_2 \geq 1$, i.e. the set S itself. To obtain this cut from P' one can use the original intersection point $(1, 0)$ together with $(0, 1)$, which can be obtained by taking a step of length -1 from the origin along $r^{(2)}$.

Changing, say the kth step length λ_k, to some new value λ'_k, and holding all others fixed rotates the intersection cut about the axis defined by the $(n-1)$ fixed intersection points. This rotated cut can equivalently be generated by an intersection cut with C and a simplicial cone P'' that shares its apex and all extreme ray directions with P' except for the kth direction — call it $r_*^{(k)}$ — to which we assign infinite step length. Suppose the mth ray has finite intersection with C (the problem is infeasible if no such m exists). Then we have the parallel condition $r_*^{(k)} = \bar{x} + \lambda_m r^{(m)} - (\bar{x} + \lambda'_k r^{(k)})$, i.e. the direction $r_*^{(k)}$ is equal to the direction from the new intersection point (attained at negative step length) to one of the other intersection points. The rotated cut is valid if P'' is valid, i.e. $P'' \cap S = P' \cap S$. This can be guaranteed by ensuring $r_*^{(k)}$ is contained in the recession cone of C since by convexity of C any point in $P' \setminus P''$ would be in the interior of C hence outside S. Maximal rotation is given by $r_*^{(k)}$ parallel to an extreme ray of C.

Thus the strengthening procedure is to iterate over all extreme rays of P' that have infinite step length, i.e. with direction vectors strictly contained in the recession cone rec(C). For each such ray $r^{(k)}$, we update the step length λ_k:

$$\lambda'_k := \max\{y | \lambda_m r^{(m)} - y r^{(k)} \in \text{rec}(C)\}. \tag{4}$$

In Section 5 we show that this procedure can be performed efficiently for our proposed families of S-free sets.

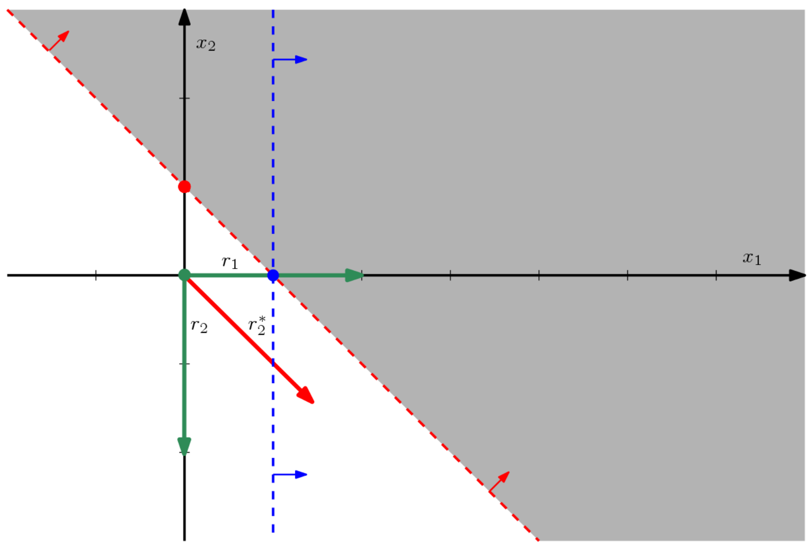

Figure 1: Example of cut tightening. In green, simplicial cone P' with rays r_1, r_2; origin marked with green dot. In grey, feasible set S, and the S-free set C is the complement of S. Blue dashes indicate the standard intersection cut; a blue dot marks the intersection point with bd(C). The strengthened intersection cut is shown with red dashes; a red dot marks the intersection point obtained with a negative step length. The tightened extreme ray $r_2 \to r_2^*$ is shown in red.

2.3 Implementation of the Intersection Cut

An intersection cut for P requires the following:

1. A simplicial cone $P' \supseteq P$ with apex at $\bar{x}$.

2. An S-free set C containing $\bar{x}$ in its interior.
3. For each extreme ray of P', either the intersection with the boundary of C, or else proof that the ray is contained entirely in C.

Step 1 is satisfied using any n defining halfspaces of P with $\bar{x}$ on their boundaries, i.e. setting P' to be a simplicial conic relaxation of P. This is possible provided P has at least one extreme point. Different choices of P' derived from P can affect the depth of the resulting cut. A natural approach is to select a cone corresponding to an optimal basic solution found from a LP relaxation, such as suggested by Gomory's initial development of the cutting plane [42]. Since there may be many optimal solutions, the results are affected by the choice of LP solver. The dual simplex method is typically employed due to fast updating after applying a cut. Choices in the simplex pivoting rule can lead to substantially different results (e.g. the lexicographic rule [12,86]). Other basic (possibly infeasible) solutions can also be used to generate intersection cuts; studies of optimizing over cut closures indicate the potential benefits of doing so (e.g. [14,35]).

Step 2 shall be the focus of Section 4. Note there always exists an S-free set that can be used to separate $\bar{x}$. Since $\bar{x} \in \text{ext}(P)$, then $\bar{x} \notin \text{clconv}(S \cap P) \implies \bar{x} \notin S$. Since S is a closed set, then there exists a ball centered around $\bar{x}$ that is S-free; for example, a lattice-free ball or cylinder is used in the original development of the intersection cut [9]. The S-free ball is a key notion used in Section 3.1.

Step 3 shall be the focus of Section 5. Note that a point precisely on the boundary is not necessary for a valid cut — points in the interior of C suffice. A simple way in practice to ensure numerical validity of the cut, for instance, is to take a sufficient step backwards from the boundary of C when generating intersection points. Computing valid points to generate a cut is computationally straightforward provided one can practically (e.g. in polynomial-time) determine membership in C.

Many cuts in mixed-integer linear programming can be interpreted as intersection cuts, as intersection cuts produce all nontrivial facets of the corner polyhedron [27]. For instance, split cuts are an important class of intersection cuts [2] due to their simplicity (derived from simple maximal lattice-free sets) and practical effectiveness. Several papers [3,29,66,67] have worked to extend the intersection cut via split cuts to mixed-integer conic optimization.

An important overarching consideration for cuts is cut pool management: using a subset of generated cuts when solving the relaxation in a given iteration (e.g. [1,36,61]). Judicial cut selection can improve convergence rate and numerical stability. Balas and Cornuéjols [13] suggest managing a pool of intersection points derived from the intersection cut procedure with similar aims.

3 Oracle Cuts

Suppose we have an oracle that provides for any given point $\bar{x} \notin S$ the nonzero Euclidean distance $d(\bar{x}, S)$ between $\bar{x}$ and the nearest point in S.

Remark. The (closed) ball $\mathcal{B}(\bar{x}, d(\bar{x}, S))$ is S-free.

Suppose P is a polytope. We can show that this S-free ball can be used to construct a pure cutting plane algorithm that will converge in the limit to the convex hull of $S \cap P$. Furthermore, an arbitrarily precise approximation of $\mathcal{B}(\bar{x}, d(\bar{x}, S))$ suffices to obtain an arbitrarily precise approximation of $\text{conv}(S \cap P)$ using the aforementioned algorithm. This is not as strong as convergence in finite time, which can be established for simpler problems (e.g. [42, 71]); such a guarantee is not possible here since $\text{conv}(S \cap P)$ may be nonlinear. Finite convergence, however, is not strictly necessary for effective practical implementation in branch-and-cut (e.g. split cuts [31]).

3.1 Separation

For any infeasible extreme point $\bar{x}$ of P the intersection cut may be applied to $\mathcal{B}(\bar{x}, d(\bar{x}, S))$ to ensure separation (recall Proposition 2.3). The ball will not in general be a maximal S-free set, but together with the intersection cut it provides a simple, efficient (modulo the oracle call), and broadly applicable tool. Now suppose instead of generating the intersection cut for $P' \setminus \text{int}(\mathcal{B}(\bar{x}, d(\bar{x}, S)))$ we seek a (stronger) cut $\alpha^T(x - \bar{x}) \geq \delta$ that separates $\bar{x}$ from $P \setminus \text{int}(\mathcal{B}(\bar{x}, d(\bar{x}, S)))$. The cut coefficients can be determined via the following master cut generation problem,

$$\begin{aligned} & \max\ \delta \\ (\mathbf{MC})\quad & \text{s.t. } \alpha^T(x - \bar{x}) \geq \delta\ \forall x \in \text{conv}(P \setminus \text{int}(\mathcal{B}(x, d(x, S)))), \qquad (5a) \\ & \|\alpha\|_1^2 \leq 1. \qquad (5b) \end{aligned}$$

The objective is to maximize the linear cut violation for the point $\bar{x}$. The cut normalization constraint (5b) is replaceable, for instance, with the 2-norm. Norm selection has been subject to extensive testing and discussion in mixed-integer programming (e.g. [37]), but we leave alternative formulations of **MC** out of scope of this initial proposal. In contrast to the intersection cut we subtract a ball over the entire polyhedron rather than a simplicial conic relaxation, which increases the computational burden of cut generation. Averkov [8] provides sufficient conditions under which $\text{conv}(P \setminus \text{int}(C))$ is polyhedral for convex C, and this includes a closed ball. Figure 2 demonstrates that one must separate over more than one nontrivial facet in general, and indeed the problem is NP-Hard [38]. The increased computational expense, however, provides us with strong cuts that provide favourable convergence properties.

One way to solve **MC** is to decompose in the fashion of Benders [21] by relying on a separation algorithm to handle constraint (5a). Given a proposed candidate cut $\hat{\alpha}$, $\hat{\delta}$, we wish to find a point $\hat{x} \in \text{conv}(P \setminus \text{int}(\mathcal{B}(\hat{x}, d(\hat{x}, S))))$ for which $\hat{\alpha}^T(\hat{x} - \bar{x}) < \hat{\delta}$, or else certify that the candidate cut is valid for constraint (5a). This task may be formulated as the subproblem

$$\begin{aligned} & z^*_{\text{SC}} := \max_x\ \|x - \bar{x}\|_1 \\ (\mathbf{SC})\quad & \text{s.t. } \hat{\alpha}^T(x - \bar{x}) \leq \hat{\delta}, \qquad (6a) \\ & x \in P. \qquad (6b) \end{aligned}$$

If $z^*_{\text{SC}} \leq d(\hat{x}, S)$, then the cut $\hat{\alpha}, \hat{\delta}$ is valid. Otherwise, the corresponding optimal solution x^* (or a small perturbation if constraint (6b) is binding) can be used to add the cut $\alpha^T(\hat{x} - \bar{x}) \geq \delta$ to the master problem. Problem (6) can be represented as an integer program; alternatively, a 2-norm may be used in the objective function, and if P is a rational polytope then an optimal solution can be obtained in finite time using simplicial branch-and-bound [57]. The ellipsoid algorithm [44, 49] can be used to solve the master problem **MC** by solving a polynomial number of instances of **SC**.

Furthermore, to accommodate algorithms that can solve the master problem only to some fixed precision, there must be an adjustment for additive error $\lambda > 0$ where a smaller ball of radius $d(\bar{x}, S) - \lambda$ is used instead. This adjustment can also be used to accommodate imprecision in estimating the distance $d(\bar{x}, S)$, which may be irrational. As such, the proposed procedure can in practical implementation only separate points sufficiently far from S. This may be a necessary tradeoff using standard numerical methods due to the possibility of, say, the feasible set being a single irrational solution.

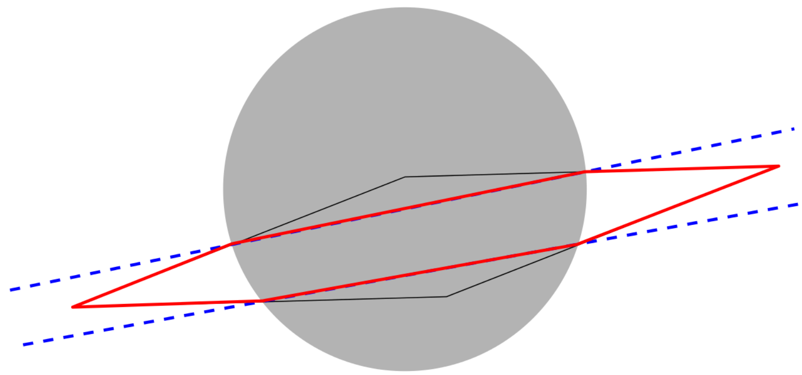

Figure 2: A parallelogram P minus a ball B. The convex hull of $P \setminus B$ shown in thick red lines; its nontrivial facets are described by the cuts with boundaries indicated by dotted blue lines.

3.2 Convergence of Cut Closures

Throughout this subsection we assume that P is a polytope. We follow closely the proof strategy of Theorem 3.6 in Averkov [8], which establishes convergence with respect to certain cuts given some structural assumptions regarding S. Our result applies to closed sets S equipped with an oracle, which is a different domain of application than that of Averkov. We also allow for a cutting plane procedure (in particular, procedure **MC**) with fixed numerical precision, where separation is only guaranteed over a ball with radius exceeding some minimum threshold $\lambda \geq 0$. More precisely, we assume that there is an oracle that, given $x \in \mathbb{R}^n$, returns an estimate $\tilde{d}(x, S)$ with $d(x, S) \leq \tilde{d}(x, S) \leq d(x, S) + \lambda$. This yields an underestimate for $d(x, S)$: $d(x, S) - \lambda \leq \tilde{d}(x, S) - \lambda \leq d(x, S)$. We will term the quantity $\tilde{d}(x, S)$ a λ-overestimate for $\tilde{d}(x, S)$. Other notions of approximations for $d(x, S)$ are similarly handled.

The Hausdorff distance between two sets X, Y, denoted $d_H(X, Y) := \max\{\sup_{x \in X} d(x, Y), \sup_{y \in Y} d(y, X)\}$, provides a natural way to describe convergence. An alternative definition of d_H is available using the notion of *ϵ-fattening*. The ϵ-fattening of a set X is $X_\epsilon := \cup_{x \in X} \mathcal{B}(x, \epsilon)$, and so $d_H(X, Y) = \inf\{\epsilon \geq 0 | X \supseteq Y_\epsilon, Y \supseteq X_\epsilon\}$.

Now relabel $P_0(\lambda) := P$, and define the rank k closure (see [25]) recursively as

$$P_{k+1}(\lambda) := \cap_{x \in \text{ext}(P_k(\lambda))} \text{conv}(P_k(\lambda) \setminus \text{int}(\mathcal{B}(x, \min\{d(x, S) - \lambda, 0\})))$$

Furthermore define the compact convex set $P_\infty(\lambda) := \cap_{k=0}^{\infty} P_k(\lambda)$, which is the infinite rank cut closure.

Two lemmas are used, with proofs that can be found in Schneider [76]. The first lemma gives us Hausdorff convergence in the sequence of cut closures [76, Lemma 1.8.2 & p. 69 Note 4].

Lemma 3.1. *Let $(C_k)_{k\in\mathbb{N}}$ be a sequence of nonempty compact sets in $\mathbb{R}^n$, and denote $C_\infty := \cap_{i=0}^{\infty} C_k$. If $C_k \supseteq C_{k+1} \forall k$ then $C_\infty = \lim_{k\to\infty} C_k$ and $\lim_{k\to\infty} d_H(C_k, C_\infty) = 0$.*

The second lemma [76, Lemma 1.4.6] ensures the existence of a ball cut that can separate an extreme point of a convex relaxation.

Lemma 3.2. *Let $C \subset \mathbb{R}^n$ be a closed, convex set and let $x \in C$. Then x is an extreme point of C iff for every open neighbourhood U around x there exists a hyperplane H defining the boundary of two (separate) halfspaces H^-, H^+ such that $x \in \text{int}(H^-), C \setminus U \in \text{int}(H^+)$.*

Theorem 3.3. *$P_\infty(\lambda) \subseteq \text{conv}(P \cap S_\lambda)$.*

Proof. By construction $P_\infty(\lambda) \subseteq P_0(\lambda) = P$, so it suffices to show that $\text{ext}(P_\infty(\lambda)) \in S_\lambda$. We shall do so by way of contradiction. Suppose there exists $\bar{x} \in \text{ext}(P_\infty(\lambda))$ such that $\bar{x} \notin S_\lambda$; observe that $\bar{x} \notin S_\lambda$ implies $d(\bar{x}, S) - \lambda > 0$. Then let U be an open ball of radius $r := (d(\bar{x}, S) - \lambda)/3$ centered at $\bar{x}$. By Lemma 3.2 there exist two opposite-facing halfspaces H^+, H^- such that $\bar{x} \in \text{int}(H^-)$ and $P_\infty(\lambda) \setminus U \in \text{int}(H^+)$. Since U is open and $P_\infty(\lambda) \cap H^-$ is in the interior of U, there exists sufficiently small $\epsilon > 0$ such that $(P_\infty(\lambda))_\epsilon \cap H^-$ is also contained in U. Now Lemma 3.1 gives us some rank k_0 for which we have the sandwich $(P_\infty(\lambda))_\epsilon \supseteq P_{k_0}(\lambda) \supseteq P_\infty(\lambda)$. Furthermore, since H^- separates an extreme point of $P_\infty(\lambda)$, it also separates an extreme point of the superset $P_{k_0}(\lambda)$. Thus there exists some extreme point $x_{k_0} \in \text{ext}(P_{k_0}(\lambda))$ in $P_{k_0}(\lambda) \cap H^- \subset U$, and so $d(x_{k_0}, \bar{x}) < r$. Thus we have

$$\begin{aligned} d(x_{k_0}, S) - \lambda &\geq d(\bar{x}, S) - d(x_{k_0}, \bar{x}) - \lambda, \\ &> d(\bar{x}, S) - \lambda - r, \\ &= 2r. \end{aligned}$$

Since U has diameter $2r$, then $U \subset \text{int}(\mathcal{B}(x_{k_0}, d(x_{k_0}, S) - \lambda))$. As H^+ is valid for $P_{k_0}(\lambda) \setminus U$, then H^+ is also valid for the nested set $P_{k_0}(\lambda) \setminus \text{int}(\mathcal{B}(x_{k_0}, d(x_{k_0}, S) - \lambda))$. However, $\bar{x} \in H^-$, which implies $\bar{x} \notin P_{k_0+1}(\lambda) \supseteq P_\infty(\lambda)$, giving us a contradiction. □

A consequence of Theorem 3.3 is that the cutting plane procedure proposed in Section 3.1 can get arbitrarily close in Hausdorff distance to $\text{conv}(P \cap S)$ in finite time given sufficient numerical precision. Formally,

Theorem 3.4. *Assume that $P \cap S$ is contained in a ball of radius $R \geq 1$ around the origin. Assume there is an oracle that given $x \in \mathbb{R}^n$ returns a rational underestimate for $d(x, S)$ that is lower bounded by $d(x, S) - \lambda$ and of size (number of bits) upper bounded by some fixed integer p (proportional to $\log R + \log \Lambda$). Then, given rational $\epsilon > 0$ there is a finite algorithm that either proves that $P \cap S$ is empty, or computes a rational polyhedral relaxation $\tilde{P}$ for $\text{conv}(P, S)$ such that $d_H(\tilde{P}, \text{conv}(P, S)) \leq \epsilon + \lambda$. The algorithm works over the rationals, and all the rationals produced by the algorithm have size bounded by a function of p, $n \log R$ and the size of ϵ.*

Proof. (Sketch.) Let P_0 be the standard hypercube with vertices whose coordinates are all $\pm R$. The P_0 contains $P \cap S$. For the generic step, so long as an extreme point v of $P_k(\lambda)$ satisfies $d(v, S) > \lambda$ then it can be separated. Here, in problems **MC** and **SC** we use the 1-norm. Theorem 3.3 shows that for any $\epsilon > 0$ there is a finite value $\bar{k}$ such that $d_H(P_{\bar{k}} \cap S_\lambda) \leq \epsilon + \lambda$. If v is an extreme point of $P_{\bar{k}}$ then the underestimate for $d(v, S)$ provided by the algorithm will be at most ϵ – this condition provides the termination criterion. Since we are using the 1-norm balls problem **SC** is a linear program, from which the result follows. □

For each extreme point $\bar{x}$ of P_0 the cutting plane procedure can be applied to $\bar{x}$ with the ball $d(\bar{x}, S-\lambda)$, where λ is a parameter accounting for numerical tolerance in the solution to **MC**. This application of cuts yields $P_1(\lambda)$, and by recursive application any $P_k(\lambda)$ is attainable. Now suppose $P \cap S$ is nonempty and we seek a relaxation R that satisfies $d_H(R, \text{conv}(P \cap S)) \le \epsilon$, where $\epsilon > 0$ is a precision parameter. Observe for any λ' such that $\epsilon > \lambda'$ we have $d_H(\text{conv}(P \cap S_{\lambda'}), \text{conv}(P \cap S)) < \epsilon$. Furthermore Theorem 3.3 gives us some r such that $d_H(P_r(\lambda'), \text{conv}(P \cap S_{\lambda'})) < \epsilon - \lambda'$. Hence, $P_r(\lambda')$, which is attainable by cutting plane algorithm (with precision satisfying tolerance λ'), satisfies our requirement for R. By similar argument, if $P \cap S$ is empty, then with sufficiently high numerical precision we can find a relaxation that is contained in an arbitrarily small neighbourhood. We note that if the 1-norm cuts are used instead of (5b) then, additionally, all computations can be assumed to take place over rationals.

4 Outer-Product-Free Sets

Our approach amounts to an augmentation of the moment/sum-of-squares approach to polynomial optimization (see [53, 54]). Let d be the maximum degree of any monomial among the polynomials p_i. Let $m_r = [1, x_1, ..., x_n, x_1x_2, ..., x_n^2, ...x_n^r]$ be a vector of all monomials up to degree r. Any polynomial may be written in the form $p_i(x) = m_r^T A_i m_r$ (provided r is sufficiently large), where A_i is a symmetric matrix derived from coefficients of p_i. For instance,

$$x_1^2 + 2x_1x_2 + 2x_2 = \begin{bmatrix} 1 \\ x_1 \\ x_2 \end{bmatrix}^T \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 1 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ x_1 \\ x_2 \end{bmatrix}.$$

We can apply this transformation to **PO**:

$$\begin{aligned} & \min\ \langle A_0, X \rangle \\ (\mathbf{LPO})\ & \text{s.t.}\ \langle A_i, X \rangle \le b_i,\ i = 1, ..., m, && \text{(7a)} \\ & X = m_r m_r^T. && \text{(7b)} \end{aligned}$$

Denote $n_r := \binom{n+r}{r} + 1$, i.e the length of m_r. Here $A_i \in \mathbb{S}^{n_r \times n_r}$ are symmetric real matrices of data, and $X \in \mathbb{S}^{n_r \times n_r}$ is a symmetric real matrix of decision variables. The problem has linear objective function, linear constraints (7a), and nonlinear constraints (7b). One can replace the moment matrix condition $X = m_r m_r^T$ with the equivalent conditions of $X \succeq 0$, $\text{rank}(X) = 1$, and consistency constraints enforcing entries representing the same monomial terms to be equal. An example of a consistency constraint is that $X_{12}, X_{21} := x_1$, and so we impose the linear equality constraint $X_{12} = X_{21}$. Dropping the nonconvex rank one constraint yields the standard semidefinite relaxation called Shor's relaxation [78]. The relaxation is said to be exact when there is an optimal solution where $\text{rank}(X) = 1$ since the solution can be factorized to obtain an optimal solution vector for **PO**. In special cases (e.g. [53, 54, 59]) one can establish that there exists sufficiently large r to ensure an exact relaxation. However, there is a combinatorial explosion in the size of **LPO** (hence the size of any associated relaxation) with respect to r.

From **LPO** we have a natural definition of S: the set of symmetric matrices that are outer products: $\{xx^T : x \in \mathbb{R}^n\}$. Furthermore, P naturally corresponds to the linear constraints (7a) together with the consistency constraints and $X_{11} = 1$. Accordingly, we shall study sets that are *outer-product-free*, where no matrix representable as a symmetric outer product is in the interior. For symmetric $X \in \mathbb{S}^{n\times n}$ we can consider the vectorized matrix $\text{vec}(X) \in \mathbb{R}^{n(n+1)/2}$, where entries from the upper triangular part of the matrix map to the vector in some order. Notions of the interior, convexity, and so forth are with respect to this vector space. For simplicity we shall drop the explicit vectorization where obvious. Matrix theory will be used in the derivation of reciprocal step lengths β to determine intersection points.

Suppose we have an extreme point of a polyhedron P with spectral decomposition $\bar{X} := \sum_{i=1}^{n} \lambda_i d_i d_i^T$ and ordering $\lambda_1 \geq ... \geq \lambda_n$. We seek to separate $\bar{X}$ when it is not representable as a symmetric outer product. Recall from Section 2.3 that maximal S-free sets help generate stronger intersection cuts. We shall explore in the following subsection the characteristics of maximal outer-product-free sets. In Section 4.2 we shall improve upon the generic proposal in Section 3 by deriving an outer-product-free extension of the S-free ball.

4.1 Maximal Outer-Product-Free Sets

We establish in Theorem 4.3 that full-dimensional maximal outer-product-free sets are cones. Then, such cones will be classified in terms of the coefficient matrices corresponding to their supports. This classification identifies two families, described in Theorems 4.7 and 4.10, with which we can perform polynomial-time separation. These two families are shown in Theorem 4.15 to describe all full-dimensional maximal outer-product-free sets in the space of 2×2 symmetric matrices. In higher dimensions we provide Propositions 4.17 and 4.18 as indications of other types of maximal outer-product-free sets.

Lemma 4.1. *Let $C \subseteq \mathbb{R}^n$ be a convex set. Every ray emanating from the origin and contained in cone(C) passes through a point in C that is not the origin.*

Proof. Consider a ray r emanating from the origin and in cone(C). Any point x along r that is not the origin has conic representation $\sum_{i=1}^{k} \lambda_i c_i$ where $k \geq 1, \lambda > 0$, and $c_i \in C \; \forall i$. Consequently, r passes through the point $x/(\sum_{i=1}^{k} \lambda_i)$, which is a convex combination of points in and therefore itself an element of C. □

Lemma 4.2. *Let $C \subseteq \mathbb{R}^n$ be a full-dimensional convex set. Every ray emanating from the origin and contained in cone(C) either intersects with the interior of C or is contained in the boundary of cone(C).*

Proof. Suppose not. Then there exists a ray r emanating from the origin with direction $d \in \mathbb{R}^n$ that is contained in the interior of cone(C) but does not intersect with int(C). Furthermore since r is in the interior of cone(C), there exists $\epsilon > 0$ such that $(d + \epsilon p)t \in \text{cone}(C)$ for all $p \in \mathbb{R}^n, t \geq 0$. However, Lemma 4.1 implies r is tangent to the boundary of C, and so there exists some $\hat{p}$ such that $(d + \epsilon\hat{p})t$ is separated by a hyperplane supporting C for all $t > 0$. Hence the ray emanating from the origin in the direction $(d + \epsilon\hat{p})$ can only intersect with C at the origin; by Lemma 4.1 this implies the ray is not contained in cone(C). □

Theorem 4.3. *Let $C \subset \mathbb{S}^{n\times n}$ be a full-dimensional outer-product-free set. Then clcone(C) is outer-product-free.*

Proof. Suppose clcone(C) is not outer-product-free; since it is closed and convex, then by definition of outer-product-free sets there must exist $d \in \mathbb{R}^n$ such that dd^T is in its interior. If d is the zeros vector, then $\text{int}(C)$ also contains the origin, which contradicts the condition that C be outer-product-free. Otherwise the ray r emanating from the origin with nonzero direction dd^T is entirely contained in $\text{clcone}(C)$ and passes through its interior. By convexity the interior of $\text{cone}(C)$ is the same as the interior of its closure, so r is also passes through the interior of cone(C). By Lemma 4.2, r intersects with the interior of C. But every point along r is an outer-product, which again implies that C is not outer-product-free. □

Corollary 4.4. *Every full-dimensional maximal outer-product-free set is a convex cone.*

Proof. Follows directly from Theorem 4.3. □

Definition 4.5. A supporting halfspace of a closed, convex set S contains S and has as boundary a supporting hyperplane of S. Thus for some $a, b \in \mathbb{R}^n \times \mathbb{R}$, $a^T x \leq b \; \forall x \in S$ and there exists $\hat{x} \in S$ such that $a^T \hat{x} = b$.

Lemma 4.6. *Let C be a full-dimensional maximal outer-product-free set. Any supporting halfspace of C is of the form $\langle A, X \rangle \geq 0$ for $A \in \mathbb{S}^{n \times n}$.*

Proof. From Corollary 4.4 we have that C is a convex cone. A generic way of writing a halfspace is $\langle A, X \rangle \geq b$. If $b \neq 0$, then there exists some nonzero X^* at the boundary of the cone such that $\langle A, X^* \rangle = b$. Since $\langle \alpha A, X^* \rangle \geq b$ for all $\alpha \geq 0$, we conclude $b = 0$, a contradiction. □

From Lemma 4.6 we may thus characterize a maximal outer-product-free set as $C = \{X \in \mathbb{S}^{n \times n} | \cap \langle A_i, X \rangle \geq 0 \; \forall i \in I$, where I is an index set that is not necessarily finite. It will be useful to classify each supporting halfspace in terms of its coefficient matrix: A is either positive semidefinite, negative semidefinite, or indefinite.

Theorem 4.7. *The halfspace $\langle A, X \rangle \geq 0$ is maximal outer-product-free iff A is negative semidefinite.*

Proof. Suppose A has a strictly positive eigenvalue, with corresponding eigenvector d. Then $\langle A, dd^T \rangle > 0$, and so the halfspace is not outer-product-free.

If A is negative semidefinite we have $\langle A, dd^T \rangle = d^T A d \leq 0 \; \forall d \in \mathbb{R}^n$, so the halfspace is outer-product-free. For maximality, suppose the halfspace is strictly contained in another outer-product-free set $\bar{C}$. Then there must exist some $\bar{X} \in \text{int}(\bar{C})$ such that $\langle A, \bar{X} \rangle < 0$. However, $\langle A, (-\bar{X}) \rangle > 0$, and so the zeros matrix is interior to $\bar{C}$ since it lies between $\bar{X}$ and $-\bar{X}$. Thus $\bar{C}$ cannot be outer-product-free. □

Corollary 4.8. *Let C be a full-dimensional maximal outer-product-free set with supporting halfspaces $\{\langle A_i, X \rangle \geq 0 : i \in I\}$. Here, the set I is possibly infinite. If there exists $i \in I$ such that A_i is negative semidefinite, then C is exactly the halfspace $\langle A_i, X \rangle \geq 0$.*

Proof. Suppose C is contained in the halfspace $\langle A_i, X \rangle \geq 0$ with A_i NSD. By Theorem 4.7 the halfspace is outer-product-free, and so C is maximal only if it is the supporting halfspace itself. □

Definition 4.9. A 2×2 PSD cone is of the form $\mathcal{C}^{i,j} := \{X \in \mathbb{S}^{n \times n} | X_{[i,j]} \succeq 0\}$, for some pair $1 \leq i \neq j \leq n$.

Remark. Let $\mathcal{C} = \mathcal{C}^{i,j}$ for some pair i, j. Then $\mathcal{C}$ is a closed, convex set. Furthermore, all its supporting halfspaces are of the form $\langle (cc^T), X \rangle \geq 0$ where $c_k = 0 \, \forall \, k \neq i, j$. Consequently, the boundary of $\mathcal{C}$ is the set of $X \in \mathbb{S}^{n \times n}$ such that there exists some such c such that $c^T X c = 0$, i.e. $X_{[i,j]}$ has rank one and is PSD.

Theorem 4.10. *Every 2×2 PSD cone is maximal outer-product-free.*

Proof. We shall first establish the outer-product-free property, and then prove maximality. Every real, symmetric outer product has rank one and is PSD, and clearly all its 2×2 principal submatrices are PSD and have rank one. Thus every symmetric outer product is on the boundary of every 2×2 PSD cone; consequently no symmetric outer product is in the interior of a 2×2 PSD cone. Since every 2×2 PSD cone is closed and convex, it is therefore outer-product-free.

Now by way of contradiction suppose for some pair (i, j) $\mathcal{C}^{i,j}$ is not maximal outer-product-free. Then it is strictly contained in an outer-product-free set $\bar{C}$. Let $\bar{X} \in \bar{C} \setminus \mathcal{C}_{ij}$; then $\bar{X}_{[i,j]}$ is not positive semidefinite. Thus we may write a spectral decomposition $\bar{X}_{[i,j]} := \lambda_1 d_1 d_1^T + \lambda_2 d_2 d_2^T$ where $\lambda_2 < 0$. We shall now construct a rank-1 PSD matrix Z in the interior of the $\mathcal{C}^{i,j}$, and hence in the interior of $\bar{C}$. This will prove a contradiction given the choice of $\bar{C}$. To that effect, choose some $\epsilon > 0$ and set $Y_{[i,j]} := (|\lambda_1| + \epsilon)d_1 d_1^T + |\lambda_2| d_2 d_2^T$. Furthermore for all other entries of Y, set $Y_{k,\ell} = -\bar{X}_{k,\ell}$. By construction $Y_{[i,j]}$ is positive definite and therefore Y is in the interior of $\bar{C}$. Now let $Z := (\bar{X} + Y)/2$ be the midpoint between $\bar{X}$ and Y. Then we have $Z = zz^T$, where the (i, j) entries of z are equal to the (i, j) entries of $\sqrt{\frac{|\lambda_1|+\lambda_1+\epsilon}{2}}$ and all other entries of z are zero. Thus $Z = zz^T$ the convex combination of $\bar{X} \in \bar{C}$ and $Y \in \text{int}(\bar{C})$, as desired. □

We can also demonstrate existence of other maximal outer-product-free sets that have supporting halfspaces with PSD coefficient matrices. First let us characterize all maximal outer-product-free sets containing the cone of positive semidefinite matrices, $\mathbb{S}_+^{n\times n}$. Our approach involves Weyl's inequality [84]. Let A, B be Hermitian matrices and define $C := A + B$. Let the eigenvalues of A, B, C be μ, ν, ρ respectively, with orderings $\mu_1 \leq ... \leq \mu_n, \nu_1 \leq ... \leq \nu_n, \rho_1 \leq ... \leq \rho_n$.

Theorem 4.11 (Weyl's Inequality)**.** *For $i = 1, ..., n$ and $j + k - n \leq i \leq r + s - 1$, we have*

$$\nu_j + \rho_k \leq \mu_i \leq \nu_r + \rho_s.$$

Lemma 4.12. *Let $\bar{X} \in \mathbb{S}^{n\times n}$ be a symmetric matrix with k nonnegative eigenvalues.* $\text{conv}(\bar{X} \cup \mathbb{S}_+^{n\times n})$ *is outer-product-free iff $k \geq 2$.*

Proof. We shall write a spectral decomposition $\bar{X} = \sum_{i=1}^n \lambda_i d_i d_i^T$ with ordering $\lambda_1 \geq ... \geq \lambda_n$. The outer-product-free condition is equivalent to the condition that there does not exist $c \in \mathbb{R}^n$ and radius $\epsilon > 0$ such that $\mathcal{B}(cc^T, \epsilon) \subset \text{conv}(\bar{X} \cup \mathbb{S}_+^{n\times n})$. The ball condition can be restated as follows: for each $Q \in \mathbb{S}^{n\times n}$ with bounded Frobenius norm $\|Q\|_F \leq 1$ there exists $\alpha \in [0, 1], R \in \mathbb{S}_+^{n\times n}$ so that $cc^T + \epsilon Q = \alpha\bar{X} + (1 - \alpha)R$, or

$$cc^T + \epsilon Q - \alpha\bar{X} \succeq 0.$$

We will show the ball condition holds for $k \leq 1$, and show that no such construction is possible for $k \geq 2$.

Suppose $k \leq 1$. We shall demonstrate that $\beta d_1 d_1^T$ is in the interior of $\text{conv}(\bar{X} \cup \mathbb{S}_+^{n\times n})$, for $\beta > |\lambda_1|$. By construction, we have that $\beta d_1 d_1^T - \alpha\bar{X}$ is a strictly positive definite matrix. From Theorem 4.11 we have for any symmetric A, B that the minimum eigenvalue of the sum is bounded as follows: $\lambda_{\min}(A+B) \geq \lambda_{\min}(A)+\lambda_{\min}(B)$. Setting ϵ to the minimum eigenvalue of $\beta d_1 d_1^T - \alpha\bar{X}$ gives the desired result:

$$
\begin{aligned}
&\lambda_{\min}(\beta d_1 d_1^T + \epsilon Q - \alpha \bar{X}),\\
\geq &\lambda_{\min}(\beta d_1 d_1^T - \alpha \bar{X}) + \lambda_{\min}(\epsilon Q),\\
\geq &\lambda_{\min}(\beta d_1 d_1^T - \alpha \bar{X}) - \epsilon,\\
\geq &0.
\end{aligned}
$$

Note that we have relied on the fact that the Frobenius norm of Q is bounded by 1, and so each singular value (thus the magnitude of any eigenvalue) is at most 1.

Now suppose $k \geq 2$. Consider some $c \in \mathbb{R}^n, \epsilon > 0$. Let $C^\perp$ be the orthogonal complement of c, and let $G = \text{span}\{d_1, d_2\}$. Then, using a standard dimension argument from linear algebra (e.g. [68, p. 48]), we can show the intersection of these sets is nonempty:

$$
\begin{aligned}
\dim(C^\perp \cap G) = &\dim(C^\perp) + \dim(G) - \dim(C^\perp + G),\\
\geq &(n-1) + 2 - n,\\
\geq &1.
\end{aligned}
$$

Now consider some nonzero $v \in C^\perp \cap G$. Setting $Q = -vv^T/\|v\|_2^2$ We have

$$
\begin{aligned}
&v^T(cc^T + \epsilon Q - \alpha \bar{X})v,\\
= &\ v^T(-\epsilon vv^T/\|v\|_2^2 - \alpha \bar{X})v,\\
\leq &\ v^T(-\epsilon vv^T)v/\|v\|_2^2 < 0.
\end{aligned}
$$

Thus in this case no such ball $\mathcal{B}(cc^T, \epsilon)$ can be in $\text{conv}(\bar{X} \cup \mathbb{S}_+^{n \times n})$, and so the convex hull is outer-product-free. □

As a consequence of Lemma 4.12 we can characterize all outer-product-free sets in $\mathbb{S}^{2\times 2}$.

Corollary 4.13. *$\mathbb{S}_+^{n\times n}$ is maximal outer-product-free iff $n = 2$.*

Proof. Follows immediately from Lemma 4.12. □

Lemma 4.14. *In $\mathbb{S}^{2\times 2}$ the cone of positive semidefinite matrices is the unique maximal outer-product-free set containing at least one positive semidefinite matrix in its interior.*

Proof. From Corollary 4.13 we have that the cone of PSD matrices is maximal for $n = 2$. Hence, if there exists a maximal outer-product-free set containing a PSD matrix in its interior, it consequently contains in its interior a boundary point of $\mathbb{S}_+^{2\times 2}$ — otherwise, it is a subset of the PSD cone. However, every boundary point of the PSD cone is a symmetric, real outer product for $n = 2$. □

Theorem 4.15. *In $\mathbb{S}^{2\times 2}$ every full-dimensional maximal outer-product-free set is either the cone of positive semidefinite matrices or a halfspace of the form $\langle A, X\rangle \geq 0$, where A is a symmetric negative semidefinite matrix.*

Proof. From Lemma 4.14, we have that every maximal outer-product-free set is either the cone of positive semidefinite matrices or it does not contain a PSD matrix in its interior. Now suppose $C \in \mathbb{S}^{n\times n}$ is a maximal outer-product-free set that is not the cone of positive semidefinite matrices. C is thus a closed, convex set with interior that does not intersect with $\mathbb{S}_+^{2\times 2}$. Then by separating hyperplane theorem there exists a supporting hyperplane of $\mathbb{S}_+^{2\times 2}$, which by Lemma 4.6 and Lemma 4.14 is of the form $\langle A, X\rangle = 0$, such that C is contained in the halfspace $\langle A, X\rangle \geq 0$. But if A has a positive eigenvalue then the halfspace includes at least one PSD matrix; thus so to maintain separation A is necessarily negative semidefinite. Furthermore, for any negative semidefinite A the halfspace $\langle A, X\rangle \geq 0$ is outer-product-free by Theorem 4.7 so C must be the halfspace itself in order to be maximal outer-product-free. □

We can show that for $n \geq 3$ there exist other types of maximal outer-product-free sets that are composed entirely of supporting halfspaces with PSD coefficient matrices, as well as maximal outer-product-free sets at least one supporting halfspace that has indefinite coefficient matrix. This can be done by constructing an appropriate outer-product-free set and applying the following result from Conforti et al. [26] (the proof of which assumes the axiom of choice):

Theorem 4.16 (CCDLM Theorem)**.** *Every S-free set is contained in a maximal S-free set.*

Their definition of S-free set includes a further requirement that S does not contain the origin, however for our purposes this is a nonrestrictive assumption as affine shifts can be applied as needed.

Proposition 4.17. *For $n \geq 3$ there exists a maximal outer-product-free set $C \subset \mathbb{S}^{n\times n}$ with full dimension such that the supporting halfspaces of C all have PSD coefficient matrices. Furthermore, C is not a 2×2 PSD cone.*

Proof. Define in $\mathbb{S}^{n\times n}$ the matrix $\bar{X} := I - \mathbf{1}\mathbf{1}^T$, where $\mathbf{1}$ is the vector of all ones. Observe $\bar{X}\mathbf{1} = (1-n)\mathbf{1}$, and so $(1-n)$ is an eigenvalue. Furthermore, solving for an eigenvector d with $(I - \mathbf{1}\mathbf{1}^T)d = d$ we have $\mathbf{1}\mathbf{1}^T d = 0$, and so the corresponding eigenvalue 1 has multiplicity $n-1$. Then for $n \geq 3$, we have by Lemma 4.12 that $Y := \operatorname{conv}(\bar{X} \cup \mathbb{S}_+^{n\times n})$ is outer-product-free. By Theorem 4.16 Y is contained in some maximal outer-product-free set C, and since C contains the PSD cone it cannot have a supporting halfspace with indefinite or negative semidefinite coefficient matrix by Lemma 4.6. Furthermore, all the 2×2 principal submatrices of $\bar{X}$ are indefinite, and so C is not a 2×2 PSD cone. □

Proposition 4.18. *For $n \geq 3$ there exists a maximal outer-product-free set $C \subset \mathbb{S}^{n\times n}$ with full dimension such at least one supporting halfspace of C is of the form $\langle A, X\rangle \geq 0$, where A is an indefinite symmetric coefficient matrix.*

Proof. We shall construct an outer-product-free set $\bar{C}$ with full dimension containing both a positive definite and negative definite matrix. Let Y be the diagonal matrix $Y := \operatorname{diag}(-1, ..., -1, -2)$, and let ℓ be the line defined by $I + \alpha Y, \alpha \in \mathbb{R}$. The eigenvalues at a point along the line parameterized by α are the diagonal entries $(1-\alpha, ..., 1-\alpha, 1-2\alpha)$. By Theorem 4.19 the Euclidean distance to the nearest real, symmetric outer product is

$$\begin{array}{ll} \sqrt{n-1}(1-\alpha), & \alpha \leq 0, \\ \sqrt{(n-2)(1-\alpha)^2 + (1-2\alpha)^2}, & 1 > \alpha > 0, \\ \sqrt{(n-1)(1-\alpha)^2 + (1-2\alpha)^2}, & \alpha \geq 1. \end{array}$$

As each segment is a convex function with respect to α, the minimum distance to an outer product along line ℓ can be calculated by setting derivatives to zero or taking extreme values of the interval, yielding the following minima:

$$\begin{array}{ll} \sqrt{n-1}, & \alpha \leq 0, \\ \sqrt{(n-2)/(n+2)}, & 1 > \alpha > 0, \\ 1, & \alpha \geq 1. \end{array}$$

The minimum radius along ℓ is therefore $\sqrt{(n-2)/(n+2)}$, and so the full-dimensional cylinder $\bar{C}$ with ℓ as its axis and radius $\sqrt{(n-2)/(n+2)}$ is outer-product-free. By Theorem 4.16, $\bar{C}$ is contained in some maximal outer-product free set C. Lemma 4.6 tells us that the supporting halfspaces are all of the form $\langle A, X\rangle \leq 0$, and C contains the identity matrix, so no such halfspace can have a negative semidefinite coefficient matrix. Furthermore, if all supporting halfspaces were to have positive semidefinite coefficient matrices, then C would contain the PSD cone. However, C contains negative definite matrices, for instance $\text{diag}(-1, ..., -1, -3)$, and so by Lemma 4.12 it cannot contain the PSD cone and be maximal outer-product-free. Hence at least one supporting halfspace of C must have indefinite (symmetric) coefficient matrix.

□

4.2 Oracle-based Outer-Product-Free Sets

Recall from Section 3 that our oracle cut requires the distance to S, in this case the distance to the nearest symmetric outer product, or equivalently the nearest positive semidefinite matrix with rank at most one. This distance can be obtained as a special case of the following positive semidefinite matrix approximation problem, given integer $r > 0$:

$$\begin{array}{rrl} & \min\limits_{Y} & \|\bar{X} - Y\| \\ (\textbf{PMA}) & \text{s.t.} & \text{rank}(Y) \leq r, \\ & & Y \succeq 0. \end{array}$$

Here $\|.\|$ is a unitarily invariant matrix norm such as the Frobenius norm, $\|.\|_F$, which is the 2-norm of the vector of singular values. Dax [30] proves the following:

Theorem 4.19 (Dax's Theorem). *Let k be the number of nonnegative eigenvalues of $\bar{X}$. For $1 \leq r \leq n-1$, an optimal solution to* **PMA** *is given by $Y = \sum_{i=1}^{\min\{k,r\}} \lambda_i d_i d_i^T$.*

This can be considered an extension of an earlier result by Higham [45], wihch provided the solution for $r = n$. When $\bar{X}$ is not negative semidefinite, the solution from Dax's theorem coincides with Eckart-Young-Mirsky [34, 63] solution to **PMA** without the positive semidefinite constraint. The optimal positive semidefinite approximant allows us to construct an outer-product-free ball:

$$\mathcal{B}_{\text{oracle}}(\bar{X}) := \begin{cases} \mathcal{B}(\bar{X}, \|\bar{X}\|_F), & \text{if } \bar{X} \text{ is NSD}, \\ \mathcal{B}(\bar{X}, \|\sum_{i=2}^{n} \lambda_i d_i d_i^T\|_F), & \text{otherwise}. \end{cases}$$

Corollary 4.20. *$\mathcal{B}_{\text{viol}}(\bar{X})$ is outer-product-free.*

Proof. Setting $r = 1$ in Dax's Theorem, we see that the nearest symmetric outer product is either $\lambda_1 d_1 d_1^T$ if $\lambda_1 > 0$, or else the zeros matrix. □

In the generic construction the oracle ball is centered around $\bar{X}$ since no further structure is assumed upon S when using an oracle. However, for **LPO** we can in certain cases use a simple geometric construction to obtain a larger ball containing the original one, as follows. Consider a ball $\mathcal{B}(X, r)$. Let $s > 0$ and suppose Q is in the boundary of the ball. We construct the "shifted" ball $\mathcal{B}(Q + (s/r)(X - Q), s)$. This ball has radius s and its center is located on the ray through X emanating from Q.

Remark. If $s > r$ then the shifted ball contains the original ball. Algebraically we may write that for any $s > r$ we have $\mathcal{B}((s/r)X + (1-s/r)Q, s) \supset \mathcal{B}(X, r)\ \forall Q \in \mathbb{S}^{n\times n}$, or $\mathcal{B}(Q + (s/r)(X - Q), s) \supset \mathcal{B}(X, r)\ \forall Q$ in the boundary of $\mathcal{B}(X, r)$.

Hence we can design a shifted oracle ball by choosing a point on the boundary of $\mathcal{B}_{\text{viol}}$ and proceeding accordingly. Let us use the nearest symmetric outer product as the boundary point in our construction:

$$\mathcal{B}_{\text{shift}}(\bar{X}, s) := \begin{cases} \mathcal{B}(s\bar{X}/\|\bar{X}\|_F, s), & \text{if } \bar{X} \text{ is NSD}, \\ \mathcal{B}\left(\lambda_1 d_1 d_1^T + \frac{s}{\|\bar{X}-\lambda_1 d_1 d_1^T\|_F}(\bar{X} - \lambda_1 d_1 d_1^T),\ s\right), & \text{otherwise}. \end{cases}$$

Proposition 4.21. *Suppose $\bar{X}$ is not a symmetric outer product. If $\lambda_2 \le 0$ then $\mathcal{B}_{\text{shift}}(\bar{X}, \|\bar{X}\|_F + \epsilon)$ is outer-product-free and strictly contains $\mathcal{B}_{\text{oracle}}(\bar{X})$ for $\epsilon > 0$. If $0 < \lambda_2 < \lambda_1$, then for $\left\|\sum_{i=2}^n \lambda_i d_i d_i^T\right\|_F < s \le \frac{\lambda_1}{\lambda_2}\left\|\sum_{i=2}^n \lambda_i d_i d_i^T\right\|_F$, $\mathcal{B}_{\text{shift}}(\bar{X}, s)$ is outer-product-free and strictly contains $\mathcal{B}_{\text{oracle}}(\bar{X})$.*

Proof. Strict containment is assured by construction, so it suffices to show that $\mathcal{B}_{\text{shift}}$ is outer-product-free.

First suppose $\bar{X}$ is negative semidefinite. Then by definition

$$\begin{aligned}\mathcal{B}_{\text{shift}}(\bar{X}, \|\bar{X}\|_F + \epsilon) &= \mathcal{B}((\|\bar{X}\|_F + \epsilon)\bar{X}/\|\bar{X}\|_F, \|\bar{X}\|_F + \epsilon), \\ &= \mathcal{B}\left((1 + \frac{\epsilon}{\|\bar{X}\|_F})\bar{X}, \|\bar{X}\|_F + \epsilon\right).\end{aligned}$$

The matrix $(1 + \epsilon/\|\bar{X}\|_F)\bar{X}$ is negative semidefinite due to our negative semidefinite assumption on $\bar{X}$, so from Dax's theorem we know the nearest outer-product is the all zeros matrix. Hence for $\mathcal{B}_{\text{shift}}$ to be outer-product-free, its radius can be no more than the Frobenius norm of its center, $(1 + \epsilon/\|\bar{X}\|_F)\bar{X}$, which by observation is indeed the case.

Now suppose $\bar{X}$ has at least one positive eigenvalue. Then by definition for $s > 0$

$$\begin{aligned}\mathcal{B}_{\text{shift}}(\bar{X}, s) &= \mathcal{B}\left(\lambda_1 d_1 d_1^T + \frac{s}{\|\bar{X} - \lambda_1 d_1 d_1^T\|_F}(\bar{X} - \lambda_1 d_1 d_1^T), s\right), \\ &= \mathcal{B}\left(\lambda_1 d_1 d_1^T + \frac{s}{\|\sum_{i=2}^n \lambda_i d_i d_i^T\|_F}\sum_{i=2}^n \lambda_i d_i d_i^T, s\right).\end{aligned}$$

If $\lambda_2 \le 0$ then the nearest outer product to the center of the shifted ball, by Dax's theorem, is $\lambda_1 d_1 d_1^T$. Thus, the maximum radius of an outer-product-free ball centered at

$$\lambda_1 d_1 d_1^T + \frac{s}{\|\sum_{i=2}^n \lambda_i d_i d_i^T \|_F)} \sum_{i=2}^n \lambda_i d_i d_i^T \ = \ \lambda_1 d_1 d_1^T + \frac{s}{\|\bar{X}\|_F} \sum_{i=2}^n \lambda_i d_i d_i^T$$

is thus $\|(s/\| \sum_{i=2}^n \lambda_i d_i d_i^T \|_F) \sum_{i=2}^n \lambda_i d_i d_i^T \|_F = s$, and so $\mathcal{B}_{\text{shift}}(\bar{X}, \|\bar{X}\|_F + \epsilon)$ is outer-product-free for all $\epsilon > 0$.

If $0 < \lambda_2 < \lambda_1$, then again by Dax's theorem the nearest outer product to the center of the shifted ball is $\lambda_1 d_1 d_1^T$ iff

$$\lambda_1 \ \ge \ \frac{s}{\|\sum_{i=2}^n \lambda_i d_i d_i^T \|_F} \lambda_2, \quad \text{i.e, iff} \quad s \ \le \ \frac{\lambda_1}{\lambda_2} \left\| \sum_{i=2}^n \lambda_i d_i d_i^T \right\|_F .$$

This gives us a maximum radius of

$$\left\| \frac{s}{\|\sum_{i=2}^n \lambda_i d_i d_i^T \|_F} \sum_{i=2}^n \lambda_i d_i d_i^T \right\|_F = s.$$

□

Theorem 4.3 gives us a further extension: provided s is chosen as prescribed by Proposition 4.21, the closure of the conic hull, $\text{clcone}(\mathcal{B}_{\text{shift}}(\bar{X}, s))$, is also outer-product-free. In the proof of Theorem 4.3 we showed that if $\bar{X}$ is NSD, then for $\epsilon > 0$ the outer-product-free set $\mathcal{B}_{\text{shift}}(\bar{X}, \|\bar{X}\|_F + \epsilon)$ contains the origin (the all zeros matrix) in its boundary. In this case the closure of the conic hull is a halfspace tangent to the ball at the origin. Otherwise if $\bar{X}$ is not NSD then we must consider two cases: either λ_2 is nonpositive or it is positive. If λ_2 is nonpositive, then we can set ϵ to any large number, and so we have a shifted ball with arbitrarily large radius that is tangent to $\lambda_1 d_1 d_1^T$. Thus in the limit as ϵ approaches infinity we obtain an outer-product-free halfspace that is tangent to $\lambda_1 d_1 d_1^T$ with normal parallel to the vector from $\bar{X}$ to $\lambda_1 d_1 d_1^T$. If λ_2 is positive, then for $\left\|\sum_{i=2}^n \lambda_i d_i d_i^T\right\|_F < s \le \frac{\lambda_1}{\lambda_2} \left\|\sum_{i=2}^n \lambda_i d_i d_i^T\right\|_F$, the outer-product-free ball $\mathcal{B}_{\text{shift}}(\bar{X}, s)$ does not contain the origin. In this case the conic hull is equal to its closure [46, Prop 1.4.7].

5 Intersection Cuts for Polynomial Optimization

In this section we discuss the implementation of Step 3 of the intersection cut as described in Section 2.3. In particular, given a simplicial cone P' with apex $\bar{X}$, we discuss how to select appropriate outer-product-free sets among those given in Section 4 and how to generate the step lengths λ.

5.1 Oracle Cuts

As shown in Section 4.2, the oracle ball $\mathcal{B}_{\text{oracle}}(\bar{X})$ by construction can generate a separating intersection cut for any $\bar{X}$ that is not a symmetric, real outer-product. Calculation of the radius and center of either ball can be done using the spectral decomposition of $\bar{X}$, as discussed in Section 4.2. For $\mathcal{B}_{\text{oracle}}$, the step lengths λ are all equal to the radius of the ball. We can strengthen the cut further by using a larger outer-product-free set, namely the conic extension $\text{clcone}(\mathcal{B}_{\text{shift}}(\bar{X}, s))$ (see Proposition 4.21 and Theorem 4.3).

If $\bar{X}$ is negative semidefinite, then by Proposition 4.21 for any $\epsilon > 0$ the shifted ball $\mathcal{B}((1 + \frac{\epsilon}{\|\bar{X}\|_F})\bar{X}, \|\bar{X}\|_F + \epsilon)$ is outer-product-free. The closed conic hull of this ball (irrespective of ϵ) is a halfspace tangent at the origin to the ball. A normal vector of this halfspace is thus $\bar{X}/\|X\|_F$, and so the equation of the halfspace is $\langle \bar{X}/\|X\|_F, X\rangle \geq 0$. The best possible cut from this maximal (recall Theorem 4.7) outer-product-free halfspace is a halfspace in the opposite direction,

$$\langle \bar{X}/\|X\|_F, X\rangle \leq 0.$$

Otherwise, if $\bar{X}$ is not NSD, then we must determine the sign of λ_2. If λ_2 is nonpositive, then we may use the halfspace that contains $\lambda_1 d_1 d_1^T$ on its boundary and that is perpendicular to the vector from $\bar{X}$ to $\lambda_1 d_1 d_1^T$, i.e. $\langle \bar{X} - \lambda_1 d_1 d_1^T, X - \lambda_1 d_1 d_1^T\rangle \geq 0$. Again, the best possible cut is a halfspace in the opposite direction:

$$\langle \bar{X} - \lambda_1 d_1 d_1^T, X - \lambda_1 d_1 d_1^T\rangle \leq 0$$

If $\bar{X}$ is not NSD and λ_2 is positive, then we may use the maximum shift prescribed by Proposition 4.21: $s = \frac{\lambda_1}{\lambda_2}\left\|\sum_{i=2}^n \lambda_i d_i d_i^T\right\|_F$. This gives us a shifted ball with centre

$$X_C := \lambda_1 d_1 d_1^T \;+\; \frac{\lambda_1}{\lambda_2}(\bar{X} - \lambda_1 d_1 d_1^T)$$

and radius

$$q := \frac{\lambda_1}{\lambda_2}\left\|\bar{X} - \lambda_1 d_1 d_1^T\right\|_F.$$

The ball does not touch the origin (see proof of Proposition 4.21), so $\text{cone}(\mathcal{B}(X_C, q))$ is outer-product-free and contains $\bar{X}$. Given the kth extreme ray of P', emanating from $\bar{X}$ along the direction $D^{(k)}$, we wish to determine the intersection point $Z_0 := \bar{X} + \lambda_k D^{(k)}$ with the boundary of $\text{cone}(\mathcal{B}(X_C, q))$. First we must check if the ray is contained in the cone, i.e. if the intersection is at infinity. The scalar projection of the direction vector onto the axis of the cone is $\langle D^{(k)}, X_C\rangle/\|X_C\|_F$. If the scalar is negative, then the ray passes through the cone. If the scalar is nonnegative, then the radius of the cone at the projected point is $r_1 := \langle D^{(k)}, X_C\rangle q/(\|X_C\|_F\sqrt{\|X_C\|_F^2 - q^2})$ (see Equation (15)). The distance from $D^{(k)}$ to the cone's axis is $d_1 := \|D^{(k)} - (\langle D^{(k)}, X_C\rangle/\langle X_C, X_C\rangle)X_C\|_F$. If $d_1 \geq r_1$ then the ray intersects with the boundary of the cone. Otherwise, we may apply the strengthening procedure of Section 2.2. Let m be the index of an extreme ray of P' with finite intersection. Applying Equation (4) yields

$$\begin{aligned}\lambda_k' := \max\{y | \|\lambda_m D^{(m)} - yD^{(k)} - (\langle \lambda_m D^{(m)} - yD^{(k)}, X_C\rangle/\langle X_C, X_C\rangle)X_C\|_F \\ \leq \langle \lambda_m D^{(m)} - yD^{(k)}, X_C\rangle q/(\|X_C\|_F\sqrt{\|X_C\|_F^2 - q^2})\}\end{aligned} \tag{9}$$

The maximum occurs when the inequality is set to equality. Squaring both sides of the equality and rearranging terms, we obtain a quadratic equation of the form $ay^2 + by + c = 0$, where

$$
\begin{aligned}
a &:= -q^2\langle D^{(k)}, X_C\rangle^2/(\|X_C\|_F^2 - q^2) - \|Z_2\|_F^2,\\
b &:= -2q^2\langle \lambda_m D^{(m)}, X_C\rangle\langle D^{(k)}, X_C\rangle/(\|X_C\|_F^2 - q^2) - 2\langle Z_1, Z_2\rangle,\\
c &:= q^2\langle \lambda_m D^{(m)}, X_C\rangle^2/(\|X_C\|_F^2 - q^2) - \|Z_1\|_F^2,\\
Z_1 &:= \|X_C\|_F\lambda_m D^{(m)} - \langle \lambda_m D^{(m)}, X_C\rangle X_C/\|X_C\|_F,\\
Z_2 &:= \langle D^{(k)}, X_C\rangle X_C/\|X_C\|_F - \|X_C\|_F D^{(k)}.
\end{aligned}
$$

Thus, $\lambda_k^{'}$ is equal to the greater root of this quadratic equation.

Now suppose the step length is finite. The scalar projection of Z_0 onto the axis of cone($\mathcal{B}(X_C, q)$) is given by $\langle Z_0, X_C\rangle/\|X_C\|_F$. The radius of the cone at the projected point is $r_2 := \langle Z_0, X_C\rangle q$ $/(\|X_C\|_F\sqrt{\|X_C\|_F^2 - q^2})$. The distance from Z_0 to the axis is $d_2 := \|Z_0 - (\langle Z_0, X_C\rangle/\langle X_C, X_C\rangle)X_C\|_F$. Intersection occurs at $d_2 = r_2$, and squaring both sides yields a quadratic equation of the form $a'\lambda_k^2 + b'\lambda_k + c' = 0$, where

$$
\begin{aligned}
a' &:= q^2\langle D^{(k)}, X_C\rangle^2/(\|X_C\|_F^2 - q^2) - \|Z_4\|_F^2,\\
b' &:= 2q^2\langle \bar{X}, X_C\rangle\langle D^{(k)}, X_C\rangle/(\|X_C\|_F^2 - q^2) - 2\langle Z_3, Z_4\rangle,\\
c' &:= q^2\langle \bar{X}, X_C\rangle^2/(\|X_C\|_F^2 - q^2) - \|Z_3\|_F^2,\\
Z_3 &:= \|X_C\|_F\bar{X} - \langle \bar{X}, X_C\rangle X_C/\|X_C\|_F,\\
Z_4 &:= \|X_C\|_F D^{(k)} - \langle D^{(k)}, X_C\rangle X_C/\|X_C\|_F.
\end{aligned}
$$

$\lambda_k^{'}$ is equal to the positive root of this quadratic equation.

5.2 2×2 PSD Cone

Some, but not all indefinite matrices lie strictly inside a 2×2 PSD cone. Furthermore, the following proposition of Chen, Atamtürk and Oren [24] ensures that every positive semidefinite matrix with rank greater than one is in a 2×2 cone:

Proposition 5.1 (CAO Proposition)**.** *For $n > 1$ a nonzero Hermitian positive semidefinite $n \times n$ matrix X has rank one iff all of its 2×2 principal minors are zero.*

Selecting appropriate 2×2 submatrices is straightforward: enumerate over all 2×2 submatrices and check for positive definiteness, i.e. $X_{ii}, X_{jj} > 0, X_{ii}X_{jj} > X_{ij}^2$. For a given (i, j)-cone, intersections can be found be found by taking the 2×2 principal submatrices $\bar{X}_{[i,j]}, D^{(k)}_{[i,j]}$ respectively corresponding to $\bar{X}$ and some extreme ray direction $D^{(k)}$.

First suppose $D^{(k)}_{[i,j]}$ is not positive semidefinite. Then we seek the step length $\lambda_k \geq 0$ such that $\bar{X}_{[i,j]} + \lambda_k D^{(k)}_{[i,j]}$ lies on the boundary of the 2×2 cone, i.e. the minimum eigenvalue is zero. Now for a symmetric matrix $\begin{bmatrix} a & b \\ b & c \end{bmatrix}$ the minimum eigenvalue can be expressed as

$$(a + c - \sqrt{a^2 - 2ac + 4b^2 + c^2})/2.$$

The eigenvalue is zero when $ac = b^2, a + c \geq 0$. Applying this to $\bar{X}_{[i,j]} + \lambda_k D^{(k)}_{[i,j]}$, we have

$$((D_{ij}^{(k)})^2 - D_{ii}^{(k)} D_{jj}^{(k)})\lambda_k^2 + (2D_{ij}^{(k)} X_{ij} - D_{ii}^{(k)} X_{jj} - D_{jj}^{(k)} X_{ii})\lambda_k + X_{ij}^2 - X_{ii} X_{jj} = 0. \quad (10)$$

The desired step length is given by the greater root of this quadratic equation with respect to λ_k (the lesser root sets the maximum eigenvalue to zero).

Now suppose $D_{[i,j]}^{(k)}$ is positive semidefinite. Then $\bar{X}_{[i,j]} + \lambda_k D_{[i,j]}^{(k)}$ is positive semidefinite for all nonnegative λ_k, giving us an intersection at infinity; thus we can apply the strengthening procedure outlined in Section 2.2. Let m be the index of an extreme ray of P' with finite intersection. Applying equation 4 we have:

$$\lambda_k' := \max\{y | \lambda_m D^{(m)} - y D^{(k)} \succeq 0\}$$

Note that λ_k' is bounded since $D^{(k)}$ is strictly positive definite. When the PSD constraint is binding we have $\lambda_{\min}(\lambda_m D^{(m)} - y D^{(k)}) = 0$. Setting the 2×2 determinant to zero yields the necessary condition

$$(\lambda_m D_{11}^{(m)} - y D_{11}^{(k)})(\lambda_m D_{22}^{(m)} - y D_{22}^{(k)}) = (\lambda_m D_{12}^{(m)} - y D_{12}^{(k)})^2$$

This is a quadratic equation in y with at least one solution guaranteed by definiteness of $D^{(k)}$.

5.3 Outer-Approximation Cuts

The maximal outer-product-free sets described by Theorem 4.7 are halfspaces, and so the best possible cuts that can be derived from these are of the form $\langle A, X \rangle \leq 0$, where A is NSD. However, observe that if A has rank $k > 1$, then the cut is of the form $\sum_{i=1}^{k} \lambda_i d_i^T X d_i \leq 0$, where each λ_i is a negative eigenvalue. Then the inequality is implied by and thus weaker than the individual inequalities of the form $\lambda_i d_i^T X d_i \leq 0$. These individual inequalities are valid as they are necessary for the positive semidefinite condition on X. Indeed, imposing all cuts of the form $c^T X c \leq 0$ is equivalent to enforcing the convex constraint $X \succeq 0$, and so the halfspaces described by Theorem 4.7 characterize the outer-approximation cuts of the SDP relaxation to **LPO**. Therefore separation is only possible if the given point is NSD or indefinite. A natural approach to separation that we shall adopt is to use all negative eigenvectors of a given extreme point of P as cut coefficient vectors. This is a well-studied procedure for semidefinite programming problems [52, 72, 75, 77].

5.4 Summary of Cuts

The proposed cuts are summarized in Table 1. Since the 2×2 cut will separate any rank two or greater PSD matrix and the outer approximation cuts will separate NSD or indefinite matrices, then the two families of cuts can be applied in tandem for global separation. $\mathcal{B}_{\text{shift}}$ provides a stronger cut than $\mathcal{B}_{\text{oracle}}$ and can be used alone or in combination with the other cuts.

6 Numerical Examples and Experiments

6.1 Example: Polynomial Optimization

We begin with a simple example in $\mathbb{S}^{2\times 2}$, a case that is fully understood and can be represented directly in three dimensions. Consider the following polynomial optimization problem:

Table 1: Proposed Cuts

Cut Name	S-free set	Separation Condition
2×2 Cut	$X_{[i,j]} \succeq 0$	$\bar{X}_{[i,j]} \succ 0$
Outer Approximation Cut	$c^T X c \leq 0$	$\bar{X}$ NSD or indefinite
Oracle Ball Cut	$\mathcal{B}_{\text{oracle}}$	$\bar{X}$ not an outer product
Strengthened Oracle Cut	$\text{clcone}(\mathcal{B}_{\text{shift}})$	$\bar{X}$ not an outer product

$$\begin{aligned} \min\ & x_1^2 + x_2^2 \\ \text{s.t.}\ & -x_1^2 - x_2^2 + x_1 x_2 \leq -2, \\ & -x_1^2 - x_2^2 - x_1 x_2 \leq -2, \\ & -x_1^2 + x_2^2 - x_1 x_2 \leq 0. \end{aligned}$$

An **LPO** representation is

$$\begin{aligned} \min\ & X_{11} + X_{22} \\ \text{s.t.}\ & -X_{11} - X_{22} + X_{12} \leq -2, && \text{(11a)} \\ & -X_{11} - X_{22} - X_{12} \leq -2, && \text{(11b)} \\ & -X_{11} + X_{22} - X_{12} \leq 0, && \text{(11c)} \\ & X = xx^T. && \text{(11d)} \end{aligned}$$

Dropping the outer product constraint (11d) results in a linear program — indeed, by construction the linear constraints describe a simplicial cone. The optimal basic solution $\bar{X}$ to this linear relaxation is at the apex of the simplicial cone,

$$\bar{X} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}.$$

Taking the negative basis inverse,

$$-\begin{bmatrix} -1 & -1 & 1 \\ -1 & -1 & -1 \\ -1 & 1 & -1 \end{bmatrix}^{-1} = \begin{bmatrix} 0.5 & 0 & 0.5 \\ 0 & 0.5 & -0.5 \\ -0.5 & 0.5 & 0 \end{bmatrix},$$

we obtain the following extreme ray directions

$$D^{(1)} = \begin{bmatrix} 0.5 & -0.5 \\ -0.5 & 0 \end{bmatrix}, D^{(2)} = \begin{bmatrix} 0 & 0.5 \\ 0.5 & 0.5 \end{bmatrix}, D^{(3)} = \begin{bmatrix} 0.5 & 0 \\ 0 & -0.5 \end{bmatrix}.$$

2×2 Cut. Equation (10) gives us the following step lengths

$$\lambda_1^* = \lambda_2^* = 2\phi \approx 3.24, \lambda_3^* = 2,$$

where $\phi := \frac{1+\sqrt{5}}{2}$ is the golden ratio. Using Equation (2) we obtain the cut

$$(0.5 + \phi^{-1})X_{11} + (\phi^{-1} - 0.5)X_{22} + 0.5X_{12} \geq 2\phi^{-1} + 1.$$

After adding this cut, the strengthened relaxation produces a rank one solution,

$$\begin{bmatrix} 2 & 0 \\ 0 & 0 \end{bmatrix}.$$

Outer Approximation Cut. As $\bar{X}$ is strictly positive definite, no outer approximation cut can separate it.

Oracle Ball Cut. Both eigenvalues of $\bar{X}$ are equal to 1, and so the radius of the ball is 1. Note that we must normalize the radius to obtain the step lengths, i.e. $\lambda_k = 1/\|D^{(k)}\|_F$. Hence the oracle ball cut is strictly dominated by the 2×2 cut: $\lambda_1 = \lambda_2 = 2/\sqrt{3} \approx 1.15, \lambda_3 = \sqrt{2} \approx 1.41$

Strengthened Oracle Cut. From Proposition 4.21 we have equal eigenvalues, and so the shifting has no effect, i.e. $X_C = \bar{X}$ and $q = 1$. $D^{(1)}$ has finite intersection with cone($\mathcal{B}(X_C, q)$) since $r_1 \approx 0.35 < d_1 \approx 0.79$. Furthermore

$$Z_3 = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix},\ Z_4 = \frac{1}{2\sqrt{2}} \begin{bmatrix} 1 & -2 \\ -2 & -1 \end{bmatrix}.$$

The quadratic equation coefficients are $a' = -1, b' = 2, c' = 4$, so $\lambda_1 = 2\phi$.

For $D^{(2)}$ we have $r_1 \approx 0.35 < d_1 \approx 0.79$, so there is a finite step length. The quadratic equation coefficients are $a' = -1, b' = 2, c' = 4$, so $\lambda_2 = 2\phi$.

For $D^{(3)}$ we have $r_1 = 0$ and $d_1 = 1/\sqrt{2}$, so there is a finite step length. The quadratic equation coefficients are $a' = -1, b' = 0, c' = 4$, so $\lambda_3 = 2$.

Thus the strengthening recovers the 2×2 cut. Note that Theorem 4.15 says that in $\mathbb{S}^{2\times 2}$ both oracle-based outer-product-free sets are contained in the 2×2 cone since they contain the strictly positive definite $\bar{X}$. Hence the 2×2 cone is the best possible outer-product-free extension of the oracle ball.

6.2 Example: Cardinality Constraint

The oracle (intersection) cut of Section 3 can be computed quickly provided $d(\bar{x}, S)$ can be determined quickly for a given set S. This is the case when S represents k-cardinality constrained vectors:

$$S := \{x \in \mathbb{R}^N | \text{card}(x) \leq k\},$$

where card(.) is the number of nonzero entries. For a given vector $\bar{x}$, the nearest point in S is a vector $\hat{x}$ equal to $\bar{x}$ at the k largest magnitude entries, and zero elsewhere.

A simple application is the statistical problem of cardinality-constrained least-absolute deviation regression (see [51]):

$$\min |Ax - b|$$

subject to

$$\text{card}(x) \leq k.$$

The cardinality constraint is used to prevent statistical overfitting. In our example let

$$A = \begin{bmatrix} 1 & 2 & 3 \\ 2 & -1 & 1 \\ 3 & 0 & -1 \end{bmatrix}, b = \begin{bmatrix} 9 \\ 8 \\ 3 \end{bmatrix},$$

and let $k = 2$. For the S-free approach we use the following formulation,

$$\begin{aligned}
\min\ & x_4 + x_5 + x_6 \\
s.t.\ & \\
& x_1 + 2x_2 + 3x_3 - 9 \le x_4, && \text{(12a)} \\
& -x_1 - 2x_2 - 3x_3 + 9 \le x_4, && \text{(12b)} \\
& 2x_1 - x_2 + x_3 - 8 \le x_5, && \text{(12c)} \\
& -2x_1 + x_2 - x_3 + 8 \le x_5, && \text{(12d)} \\
& 3x_1 - x_3 - 3 \le x_6, && \text{(12e)} \\
& -3x_1 + x_3 + 3 \le x_6, && \text{(12f)} \\
& x_4, x_5, x_6 \ge 0, && \text{(12g)} \\
& \text{card}([x_1, x_2, x_3]) \le 2. && \text{(12h)}
\end{aligned}$$

The problem is formulated in extended space with augmented variables x_4, x_5, x_6 in order to represent the objective function with linear constraints. Constraints (12a)-(12f) relate the augmented variables to the original objective function. Constraints (12g) are redundant inequalities used to form a simplicial cone after solving the linear programming relaxation. Dropping the nonconvex constraint (12h) yields a linear programming relaxation.

The linear programming relaxation has an optimal solution $x^* = [2, -1, 3, 0, 0, 0]^T$ with objective value 0. The closest vector to x^* obeying the cardinality constraint is $[2, 0, 3, 0, 0, 0]^T$ with Euclidean distance 1, giving us a step size of 1 along all directions for the intersection cut. The simplicial cone with apex x^* may be written as $Bx \le b_B$, where

$$B = \begin{bmatrix} 1 & 2 & 3 & -1 & 0 & 0 \\ 2 & -1 & 1 & 0 & -1 & 0 \\ 3 & 0 & -1 & 0 & 0 & -1 \\ 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 \end{bmatrix}, b_B = \begin{bmatrix} 9 \\ 8 \\ 3 \\ 0 \\ 0 \\ 0 \end{bmatrix}.$$

Applying Equation (2), we generate the cut

$$6x_1 + x_2 + 3x_3 - 2x_4 - 2x_5 - 2x_6 \le 19.$$

After adding the cut, the linear programming relaxation has optimal solution
$x^* \approx [1.98, -1.08, 2.95, 0.33, 0, 0]^T$ with improved objective value $\frac{1}{3}$.

6.3 Numerical Experiments

We present some numerical experiments using a pure cutting plane algorithm implementation of the cuts described in Section 5. The experiments are setup to test the stand-alone performance of our cuts. We neither claim nor intend this to be a complete practical solver for polynomial optimization; rather we consider it as one part of a full branch-and-cut procedure.

Our cutting plane algorithm solves an LP relaxation and obtains an (extreme point) optimal solution x^*, adds cuts to the relaxation separating x^*, and repeats until one of the following stopping conditions is met:

- A time limit of 600 seconds is reached
- The objective value does not improve for 10 iterations
- The maximum violation of a cut is not more than 10^{-8}. Here, if $\pi^T x \leq \pi_0$ is the cut, i.e, a valid inequality such that $\pi^T x^* > \pi_0$, we define the violation as $(\pi^T x^* - \pi_0)/\|\pi\|_1$
- The linear program becomes numerically unstable

The implementation also includes some simple cut management features, namely, adding a maximum of 5 cuts per iteration, and removing near-parallel cuts.

Computations are run on a 20-core server with an Intel Xeon E5-2687W v33.10GHz CPU and 264 GB of RAM. The simple cutting plane algorithm does not lend itself to easy parallelism; we confirmed with auxiliary experiments that similar performance can be obtained with a laptop. The code is written in Python 2.7.13 using the Anaconda 4.3 distribution. The linear programming solver is Gurobi 7.0.1 with default settings. Our code and full experimental data is available at `https://github.com/g-munoz/poly_cuts`.

Test instances are taken from two sources. First, we include all 26 problem instances from Floudas et al. [**?**] (available via GLOBALLib [**?**]) that have quadratic objective and constraints. Our cuts can accommodate arbitrary polynomial terms, however for implementation purposes reading QCQP problems is more convenient. Second, we include all 99 instances of BoxQP developed by several authors [**?**, **?**]. These problems have simple box constraints $x \in [0, 1]^n$ and a nonconvex nonhomogeneous quadratic objective function. We choose the initial LP relaxation to be the standard RLT relaxation of QCQP: setting $r = 2$ in **LPO** and including McCormick estimators for bilinear terms (see e.g. [**?**]). Problem sizes vary from 21×21 to 126×126 symmetric matrices of decision variables for BoxQP instances and from 6×6 to 63×63 for GLOBALLib instances. To obtain variable bounds for some of the GLOBALLib instances we apply a simple bound tightening procedure: minimize/maximize a given variable subject to the RLT relaxation.

Results

Results averaged over all included instances are shown in Table 2 for GLOBALLib and Table 3 for BoxQP. The columns are described as follows. Cut family indicates one of 5 cut configurations: Oracle Ball cuts (OB), Strengthened Oracle cuts (SO), Outer Approximation cuts (OA), 2x2 cuts together with OA (2x2 + OA), and finally 2x2, OA and SO cuts together (SO + 2x2 + OA). Let OPT denote the optimal value of an instance (with objective minimized), RLT the optimal value of the standard RLT relaxation, and GLB the objective value obtained after applying the cutting plane procedure. Then the optimality gaps (per instance) are

$$\text{Initial Gap} = \frac{OPT - RLT}{|OPT| + \epsilon},$$

$$\text{End Gap} = \frac{OPT - GLB}{|OPT| + \epsilon},$$

$$\text{Gap Closed} = \frac{GLB - RLT}{OPT - RLT}.$$

$\epsilon := 1$ is a parameter to avoid division by zero. Note that the Initial Gap values are the same irrespective of the cut configuration, so only one entry in the corresponding column is needed on Table 2 and Table 3. Iters is the number of cutting plane algorithm iterations per problem instance. Time is the total time in seconds spent by the algorithm. LPTime is the percentage of total time spent solving linear programs.

Large initial gaps are observed due to our choice of simple initial relaxation. The initial relaxation can be improved by problem-specific procedures, for example, or a higher order lifting (greater value of r) could be used. Several of the Floudas et al. instances are pooling problems, and for these instances adding the RRLT inequalities of Liberti and Pantelides [**?**] can result in strong (sometimes exact) relaxations. However, the scope of our experiments is limited to testing the stand-alone effects of our cuts so for consistency we have chosen the simple RLT relaxation. Since the initial gap and end gap averages are skewed by a few instances with high gap, we also provide a distribution of outcomes in Table 5. Closed gap is a more appropriate measure for comparing averages, as it is normalized to between 0% and 100%, and in Table 4 we also provide a distribution of these gaps.

As expected, the strengthened oracle cuts produce better results than the basic oracle ball cuts, as SO cuts by design dominate the equivalent cut from OB. Although both families ensure separation of infeasible extreme points, they exhibit early tailing off behaviour, and thus result in modest closed gaps. Outer approximation can only guarantee separation of non-PSD points, and hence OA can be considered an approximation of the standard RLT+SDP relaxation for QCQP. The $2 \times 2+$ OA configuration provides substantially better performance compared to OB, SO, and OA, on GLOBALLib instances, indicating the benefits of separating over the 2×2 family of cuts. OA produces the best results in terms of average end gap and closed gap on BoxQP instances; in part this is due to the known strong performance of the SDP+RLT relaxation on these instances [**?**]. On smaller instances we observe modest improvement in closed gap by adding 2×2 cuts, and worse closed gaps on larger instances. This may be related to the observation that on smaller BoxQP instances, OA provides a near-PSD solution, while on larger instances tailing off occurs before a near-PSD solution can be obtained. Results from SO+2×2+OA across all instances indicate that incorporating all cuts simultaneously can reduce the number of cuts needed; hence, despite the modest performance of the SO configuration, the strengthened oracle cuts may still warrant inclusion in a full cutting plane procedure.

The small LPTime values indicate a relatively high percent of time spent in producing cuts, managing relaxations, and overhead. The computational bottleneck in the separation procedure is a single eigenvalue computation on the $n \times n$ decision matrix solution per iteration for outer approximation and oracle cuts, and on the order of $\binom{n}{2}$ arithmetic operations for the 2×2 cuts. Overhead is a substantial contributor; our simple Python implementation stands in contrast to the optimized commercial implementation of the LP solver.

Per-instance data for 2×2+OA can be found in Appendix B; for other configurations we refer the reader to the online supplement mentioned at the beginning of this section. The appendix also provides data showing that 2×2+OA offers comparable bounds at substantially faster computation times compared to the V2 configuration of Saxena, Bonami, and Lee [74]. However, we emphasize that cuts are complementary and not competitive; as indicated by the SO+2×2+OA, it can be beneficial to include multiple families in a cutting plane procedure.

Cut Family	Initial Gap	End Gap	Closed Gap	# Cuts	Iters	Time (s)	LPTime (%)
OB	1387.92%	1387.85%	1.00%	16.48	17.20	2.59	2.06%
SO		1387.83%	8.77%	18.56	19.52	4.14	2.29%
OA		1001.81%	8.61%	353.40	83.76	33.25	7.51%
2x2 + OA		1003.33%	32.61%	284.98	118.08	30.40	15.03%
SO+2x2+OA		1069.59%	31.91%	174.79	107.16	29.55	12.56%

Table 2: Averages for GLOBALLib instances

Cut Family	Initial Gap	End Gap	Closed Gap	# Cuts	Iters	Time (s)	LPTime (%)
OB	103.59%	103.56%	0.04%	12.84	13.62	127.15	0.40%
SO		103.33%	0.34%	14.34	15.45	132.07	0.49%
OA		30.88%	75.55%	676.90	137.52	459.28	31.80%
2x2 + OA		32.84%	74.52%	349.21	140.40	473.18	28.76%
SO+2x2+OA		33.43%	74.03%	227.39	136.93	475.38	26.59%

Table 3: Averages for BoxQP instances

	GLOBALLib					**BoxQP**				
Closed Gap	OB	SO	OA	2x2 + OA	SO+2x2 +OA	OB	SO	OA	2x2 + OA	SO+2x2 +OA
>98%	0	1	0	3	3	0	0	37	37	37
90-98 %	0	1	1	2	1	0	0	16	15	15
75-90 %	0	0	0	0	0	0	0	14	11	10
50-75 %	0	0	1	3	4	0	0	13	15	15
25-50 %	0	0	1	4	4	0	0	5	7	8
<25 %	25	23	22	13	13	99	99	14	14	14

Table 4: Distribution of Closed Gaps for GLOBALLib and BoxQP

	GLOBALLib						**BoxQP**					
End Gap	Initial Gap	OB	SO	OA	2x2 + OA	SO+2x2 +OA	Initial Gap	OB	SO	OA	2x2 + OA	SO+2x2 +OA
<1 %	2	2	2	2	4	4	0	0	0	35	34	34
1 - 25 %	4	4	4	4	4	4	4	4	4	32	32	31
25-50 %	6	6	6	6	4	4	14	14	17	11	9	9
50-75 %	3	3	3	3	3	3	19	19	16	11	8	8
75-100 %	3	3	3	4	5	5	18	18	18	2	8	9
100-500 %	3	3	3	2	2	2	44	44	44	8	8	8
>500%	4	4	4	4	3	3	0	0	0	0	0	0

Table 5: Distribution of End Gaps for GLOBALLib and BoxQP

7 Conclusions

We have introduced cuts for the generic set $S \cap P$, where for the closed set S there is an oracle that provides the distance from a point to the nearest point in S. We have shown that the oracle can be used to construct a convergent cutting plane algorithm that can produce arbitrarily close approximations to $\text{conv}(S \cap P)$ in finite time. This algorithm relies on a (potentially) computationally expensive cut generation procedure, and so we have also considered a simple oracle-based intersection cut that can be easily computed. We provide applications of the intersection cut on polynomial optimization problems as well as a cardinality-constrained problem.

We have also developed an S-free approach for polynomial optimization, where S is the set of real, symmetric outer products. Our results on outer-product-free sets include a full characterization of such sets over 2×2 matrices, as well as a link between a family of maximal outer-product-free sets and outer-approximation cuts for semidefinite programming; Propositions 4.17 and 4.18 provide some open avenues that may be worth pursuing. Furthermore, the oracle-based approach can be strengthened for the special case of outer-product-free sets. We developed intersection cuts for these outer-product-free sets, including a strengthening procedure that determines negative step lengths in the case of intersections at infinity. Computational experiments on polynomial optimization problems have demonstrated the potential of our cuts as a fast way to reduce optimality gaps on a variety of problems. A full implementation could be considered for future empirical work, incorporating the cuts into a full branch-and-cut solver and developing a more sophisticated implementation, e.g. stronger initial relaxations with problem-specific valid inequalities, warm-starting the outer-approximation cuts with an initial SDP solution, etc.

References

[1] T. Achterberg, *SCIP-a framework to integrate constraint and mixed integer programming*, Konrad-Zuse-Zentrum für Informationstechnik Berlin, 2004.

[2] K. Andersen, G. Cornuéjols, and Y. Li, *Split closure and intersection cuts*, Mathematical programming, 102 (2005), pp. 457–493.

[3] K. Andersen and A. N. Jensen, *Intersection cuts for mixed integer conic quadratic sets*, in Integer Programming And Combinatorial Optimization, Springer, 2013, pp. 37–48.

[4] K. Andersen, Q. Louveaux, and R. Weismantel, *An analysis of mixed integer linear sets based on lattice point free convex sets*, Mathematics of Operations Research, 35 (2010), pp. 233–256.

[5] K. Andersen, Q. Louveaux, R. Weismantel, and L. A. Wolsey, *Inequalities from two rows of a simplex tableau*, in Integer Programming and Combinatorial Optimization, Springer, 2007, pp. 1–15.

[6] K. M. Anstreicher, *Semidefinite programming versus the reformulation linearization technique for nonconvex quadratically constrained quadratic programming*, Journal of Global Optimization, 43 (2009), pp. 471–484.

[7] A. Atamtürk and V. Narayanan, *Conic mixed-integer rounding cuts*, Mathematical programming, 122 (2010), pp. 1–20.

[8] C. Audet, P. Hansen, B. Jaumard, and G. Savard, *A branch and cut algorithm for nonconvex quadratically constrained quadratic programming*, Mathematical Programming, 87 (2000), pp. 131–152.

[9] G. Averkov, *On finite generation and infinite convergence of generalized closures from the theory of cutting planes*, arXiv preprint arXiv:1106.1526, (2011).

[10] E. Balas, *Intersection cuts—a new type of cutting planes for integer programming*, Operations Research, 19 (1971), pp. 19–39.

[11] E. Balas, *Disjunctive programming: Properties of the convex hull of feasible points*, Discrete Applied Mathematics, 89 (1998), pp. 3–44.

[12] E. Balas, S. Ceria, and G. Cornuéjols, *A lift-and-project cutting plane algorithm for mixed 0–1 programs*, Mathematical programming, 58 (1993), pp. 295–324.

[13] E. Balas, M. Fischetti, and A. Zanette, *A hard integer program made easy by lexicography*, Mathematical programming, 135 (2012), pp. 509–514.

[14] E. Balas and F. Margot, *Generalized intersection cuts and a new cut generating paradigm*, Mathematical Programming, 137 (2013), pp. 19–35.

[15] E. Balas and A. Saxena, *Optimizing over the split closure*, Mathematical Programming, 113 (2008), pp. 219–240.

[16] X. Bao, N. V. Sahinidis, and M. Tawarmalani, *Multiterm polyhedral relaxations for nonconvex, quadratically constrained quadratic programs*, Optimization Methods & Software, 24 (2009), pp. 485–504.

[17] A. Basu, M. Conforti, G. Cornuéjols, and G. Zambelli, *Maximal lattice-free convex sets in linear subspaces*, Mathematics of Operations Research, 35 (2010), pp. 704–720.

[18] ——, *Minimal inequalities for an infinite relaxation of integer programs*, SIAM Journal on Discrete Mathematics, 24 (2010), pp. 158–168.

[19] A. Basu, G. Cornuéjols, and G. Zambelli, *Convex sets and minimal sublinear functions*, Journal of Convex Analysis, 18 (2011), pp. 427–432.

[20] P. Belotti, J. C. Gez, I. Plik, T. K. Ralphs, and T. Terlaky, *On families of quadratic surfaces having fixed intersections with two hyperplanes*, Discrete Applied Mathematics, 161 (2013), pp. 2778 – 2793.

[21] P. Belotti, C. Kirches, S. Leyffer, J. Linderoth, J. Luedtke, and A. Mahajan, *Mixed-integer nonlinear optimization*, Acta Numerica, 22 (2013), pp. 1–131.

[22] J. F. Benders, *Partitioning procedures for solving mixed-variables programming problems*, Numerische mathematik, 4 (1962), pp. 238–252.

[23] D. Bienstock and A. Michalka, *Cutting-planes for optimization of convex functions over nonconvex sets*, SIAM Journal on Optimization, 24 (2014), pp. 643–677.

[24] V. Borozan and G. Cornuéjols, *Minimal valid inequalities for integer constraints*, Mathematics of Operations Research, 34 (2009), pp. 538–546.

[25] S. Burer, *Optimizing a polyhedral-semidefinite relaxation of completely positive programs*, Mathematical Programming Computation, 2 (2010), pp. 1–19.

[26] C. Chen, A. Atamtürk, and S. S. Oren, *A spatial branch-and-cut method for nonconvex QCQP with bounded complex variables*, Mathematical Programming, PP (2016), pp. 1–29.

[27] V. Chvátal, *Edmonds polytopes and a hierarchy of combinatorial problems*, Discrete mathematics, 4 (1973), pp. 305–337.

[28] M. Conforti, G. Cornuéjols, A. Daniilidis, C. Lemaréchal, and J. Malick, *Cut-generating functions and s-free sets*, Mathematics of Operations Research, 40 (2014), pp. 276–391.

[29] M. Conforti, G. Cornuéjols, and G. Zambelli, *Equivalence between intersection cuts and the corner polyhedron*, Operations Research Letters, 38 (2010), pp. 153–155.

[30] G. Cornuéjols, L. Wolsey, and S. Yildiz, *Sufficiency of cut-generating functions*, Mathematical Programming, (2013), pp. 1–9.

[31] D. Dadush, S. S. Dey, and J. P. Vielma, *The split closure of a strictly convex body*, Operations Research Letters, 39 (2011), pp. 121–126.

[32] A. Dax, *Low-rank positive approximants of symmetric matrices*, Advances in Linear Algebra & Matrix Theory, 4 (2014), pp. 172–185.

[33] A. Del Pia and R. Weismantel, *On convergence in mixed integer programming*, Mathematical programming, 135 (2012), pp. 397–412.

[34] S. S. Dey and L. A. Wolsey, *Lifting integer variables in minimal inequalities corresponding to lattice-free triangles*, in Integer Programming and Combinatorial Optimization, Springer, 2008, pp. 463–475.

[35] ——, *Constrained infinite group relaxations of MIPs*, SIAM Journal on Optimization, 20 (2010), pp. 2890–2912.

[36] C. Eckart and G. Young, *The approximation of one matrix by another of lower rank*, Psychometrika, 1 (1936), pp. 211–218.

[37] M. Fischetti and A. Lodi, *Optimizing over the first chvátal closure*, Mathematical Programming, 110 (2007), pp. 3–20.

[38] M. Fischetti and D. Salvagnin, *A relax-and-cut framework for gomory mixed-integer cuts*, Mathematical Programming Computation, 3 (2011), pp. 79–102.

[39] M. Fischetti, D. Salvagnin, and A. Zanette, *A note on the selection of benders cuts*, Mathematical Programming, 124 (2010), pp. 175–182.

[40] C. A. Floudas, P. M. Pardalos, C. Adjiman, W. R. Esposito, Z. H. Gümüs, S. T. Harding, J. L. Klepeis, C. A. Meyer, and C. A. Schweiger, *Handbook of test problems in local and global optimization*, vol. 33, Springer Science & Business Media, 2013.

[41] R. M. Freund and J. B. Orlin, *On the complexity of four polyhedral set containment problems*, Mathematical programming, 33 (1985), pp. 139–145.

[42] B. Ghaddar, J. C. Vera, and M. F. Anjos, *A dynamic inequality generation scheme for polynomial programming*, Mathematical Programming, (2011), pp. 1–37.

[43] M. X. Goemans and D. P. Williamson, *Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming*, Journal of the ACM (JACM), 42 (1995), pp. 1115–1145.

[44] R. E. Gomory, *Outline of an algorithm for integer solutions to linear programs*, Bulletin of the American Mathematical Society, 64 (1958), pp. 275–278.

[45] ——, *An algorithm for integer solutions to linear programs*, Recent advances in mathematical programming, 64 (1963), pp. 260–302.

[46] R. E. Gomory and E. L. Johnson, *Some continuous functions related to corner polyhedra*, Mathematical Programming, 3 (1972), pp. 23–85.

[47] M. Grötschel, L. Lovász, and A. Schrijver, *The ellipsoid method and its consequences in combinatorial optimization*, Combinatorica, 1 (1981), pp. 169–197.

[48] N. J. Higham, *Computing a nearest symmetric positive semidefinite matrix*, Linear algebra and its applications, 103 (1988), pp. 103–118.

[49] J.-B. Hiriart-Urruty and C. Lemaréchal, *Fundamentals of convex analysis*, Springer Science & Business Media, 2012.

[50] R. Horst and H. Tuy, *Global optimization: Deterministic approaches*, Springer Science & Business Media, 2013.

[51] J. E. Kelley Jr, *The cutting-plane method for solving convex programs*, Journal of the Society for Industrial & Applied Mathematics, 8 (1960), pp. 703–712.

[52] L. G. Khachiyan, *Polynomial algorithms in linear programming*, USSR Computational Mathematics and Mathematical Physics, 20 (1980), pp. 53–72.

[53] F. Kılınç-Karzan, *On minimal valid inequalities for mixed integer conic programs*, Mathematics of Operations Research, 41 (2015), pp. 477–510.

[54] H. Konno and R. Yamamoto, *Choosing the best set of variables in regression analysis using integer programming*, Journal of Global Optimization, 44 (2009), pp. 273–282.

[55] K. Krishnan and J. E. Mitchell, *A unifying framework for several cutting plane methods for semidefinite programming*, Optimization methods and software, 21 (2006), pp. 57–74.

[56] J. B. Lasserre, *Global optimization with polynomials and the problem of moments*, SIAM Journal on Optimization, 11 (2001), pp. 796–817.

[57] M. Laurent, *Sums of squares, moment matrices and optimization over polynomials*, in Emerging applications of algebraic geometry, Springer, 2009, pp. 157–270.

[58] Y. Li and J.-P. P. Richard, *Cook, Kannan and Schrijver's example revisited*, Discrete Optimization, 5 (2008), pp. 724–734.

[59] L. Liberti and C. C. Pantelides, *An exact reformulation algorithm for large nonconvex nlps involving bilinear terms*, Journal of Global Optimization, 36 (2006), pp. 161–189.

[60] M. Locatelli and F. Schoen, *On convex envelopes for bivariate functions over polytopes*, Mathematical Programming, (2013), pp. 1–27.

[61] M. Locatelli and N. V. Thoai, *Finite exact branch-and-bound algorithms for concave minimization over polytopes*, Journal of Global Optimization, 18 (2000), pp. 107–128.

[62] L. Lovász, *Geometry of numbers and integer programming*, Mathematical Programming: Recent Developments and Applications, (1989), pp. 177–210.

[63] L. Lovász and A. Schrijver, *Cones of matrices and set-functions and 0-1 optimization*, SIAM Journal on Optimization, 1 (1991), pp. 166–190.

[64] J. Luedtke, M. Namazifar, and J. Linderoth, *Some results on the strength of relaxations of multilinear functions*, Mathematical programming, 136 (2012), pp. 325–351.

[65] J. Lysgaard, A. N. Letchford, and R. W. Eglese, *A new branch-and-cut algorithm for the capacitated vehicle routing problem*, Mathematical Programming, 100 (2004), pp. 423–445.

[66] H. Marchand and L. A. Wolsey, *Aggregation and mixed integer rounding to solve mips*, Operations research, 49 (2001), pp. 363–371.

[67] A. Meeraus, *GLOBALLib*. `http:/http://www.gamsworld.org/global/globallib.htm`.

[68] L. Mirsky, *Symmetric gauge functions and unitarily invariant norms*, The quarterly journal of mathematics, 11 (1960), pp. 50–59.

[69] R. Misener and C. A. Floudas, *Global optimization of mixed-integer quadratically-constrained quadratic programs (miqcqp) through piecewise-linear and edge-concave relaxations*, Mathematical Programming, 136 (2012), pp. 155–182.

[70] R. Misener, J. B. Smadbeck, and C. A. Floudas, *Dynamically generated cutting planes for mixed-integer quadratically constrained quadratic programs and their incorporation into glomiqo 2*, Optimization Methods and Software, 30 (2015), pp. 215–249.

[71] S. Modaresi, M. R. Kılınç, and J. P. Vielma, *Intersection cuts for nonlinear integer programming: Convexification techniques for structured sets*, Mathematical Programming, (2015), pp. 1–37.

[72] ———, *Split cuts and extended formulations for mixed integer conic quadratic programming*, Operations Research Letters, 43 (2015), pp. 10–15.

[73] J. M. Ortega, *Matrix Theory: A Second Course*, Plenum Press, New York, 1987.

[74] M. Padberg and G. Rinaldi, *A branch-and-cut algorithm for the resolution of large-scale symmetric traveling salesman problems*, SIAM review, 33 (1991), pp. 60–100.

[75] M. Porembski, *How to extend the concept of convexity cuts to derive deeper cutting planes*, Journal of Global Optimization, 15 (1999), pp. 371–404.

[76] ——, *Cone adaptation strategies for a finite and exact cutting plane algorithm for concave minimization*, Journal of Global Optimization, 24 (2002), pp. 89–107.

[77] A. Qualizza, P. Belotti, and F. Margot, *Linear programming relaxations of quadratically constrained quadratic programs*, Mixed Integer Nonlinear Programming, (2012), pp. 407–426.

[78] A. D. Rikun, *A convex envelope formula for multilinear functions*, Journal of Global Optimization, 10 (1997), pp. 425–437.

[79] A. Saxena, P. Bonami, and J. Lee, *Convex relaxations of non-convex mixed integer quadratically constrained programs: extended formulations*, Mathematical programming, 124 (2010), pp. 383–411.

[80] ——, *Convex relaxations of non-convex mixed integer quadratically constrained programs: projected formulations*, Mathematical programming, 130 (2011), pp. 359–413.

[81] R. Schneider, *Convex bodies: the Brunn–Minkowski theory*, no. 151, Cambridge University Press, second ed., 2014.

[82] H. D. Sherali and B. M. P. Fraticelli, *Enhancing RLT relaxations via a new class of semidefinite cuts*, Journal of Global Optimization, 22 (2002), pp. 233–261.

[83] N. Z. Shor, *Quadratic optimization problems*, Soviet Journal of Circuits and Systems Sciences, 25 (1987), p. 6.

[84] F. Tardella, *Existence and sum decomposition of vertex polyhedral convex envelopes*, Optimization Letters, 2 (2008), pp. 363–375.

[85] M. Tawarmalani, J.-P. P. Richard, and C. Xiong, *Explicit convex and concave envelopes through polyhedral subdivisions*, Mathematical Programming, (2013), pp. 1–47.

[86] M. Tawarmalani and N. V. Sahinidis, *Convex extensions and envelopes of lower semi-continuous functions*, Mathematical Programming, 93 (2002), pp. 247–263.

[87] ——, *A polyhedral branch-and-cut approach to global optimization*, Mathematical Programming, 103 (2005), pp. 225–249.

[88] H. Tuy, *Concave programming under linear constraints*, Soviet Mathematics, 5 (1964), pp. 1437–1440.

[89] D. Vandenbussche and G. Nemhauser, *A branch-and-cut algorithm for nonconvex quadratic programs with box constraints*, Mathematical Programming, 102 (2005), pp. 559–575.

[90] H. Weyl, *Inequalities between the two kinds of eigenvalues of a linear transformation*, Proceedings of the National Academy of Sciences of the United States of America, 35 (1949), p. 408.

[91] L. A. Wolsey and G. L. Nemhauser, *Integer and combinatorial optimization*, John Wiley & Sons, 2014.

[92] A. Zanette, M. Fischetti, and E. Balas, *Lexicography and degeneracy: can a pure cutting plane algorithm work?*, Mathematical programming, 130 (2011), pp. 153–176.

Appendix

A Radius of the Conic Hull of a Ball

Suppose we have a ball of radius r and with centre that is distance $m > r$ from the origin. We wish to determine the radius of the conic hull of the ball a specific point along its axis. Consider a 2-dimensional cross-section of the conic hull of the ball containing the axis; this is shown in Figure 3 in rectangular (x, y) coordinates. A line passing through the origin and tangent to the boundary of the ball in the nonnegative orthant may be written in the form $y = ax$ for some $a > 0$; let $(\bar{r}, \bar{m})$ be the point of intersection between line and ball. At $(\bar{r}, \bar{m})$ we have

$$(a\bar{r} - m)^2 + \bar{r}^2 = r^2 \iff (1 + a^2)\bar{r}^2 - 2am\bar{r} + m^2 - r^2 = 0. \tag{13}$$

Now Equation (13) should only have one unique solution with respect to $\bar{r}$ since the line is tangent to the ball; thus the discriminant must be zero,

$$4a^2m^2 - 4(1 + a^2)(m^2 - r^2) = 0 \implies a = \frac{\sqrt{m^2 - r^2}}{r} \tag{14}$$

Solving Equation (13) for $\bar{r}$ with Equation (14),

$$\begin{aligned} \bar{r} &= \frac{2am}{2(1 + a^2)}, \\ &= \frac{r}{m}\sqrt{m^2 - r^2}, \\ \bar{m} &= a\bar{r}, \\ &= \frac{m^2 - r^2}{m}. \end{aligned}$$

Hence at distance d from the origin along the axis of the cone, the radius of the cone is $\frac{\bar{r}}{\bar{m}}d$, or

$$\frac{r}{\sqrt{m^2 - r^2}}d \tag{15}$$

B Supplementary Experiments

Table 6 details results for GLOBALLib instances using the 2×2+OA configuration; Table 7 shows results for BoxQP instances. OPT is the best known primal solution. RLT is the standard RLT relaxation optimal value. RLT-BT is the RLT optimal value after applying bound tightening (only applied to GLOBALLib instances). Final LB is the final lower bound obtained by the cutting plane

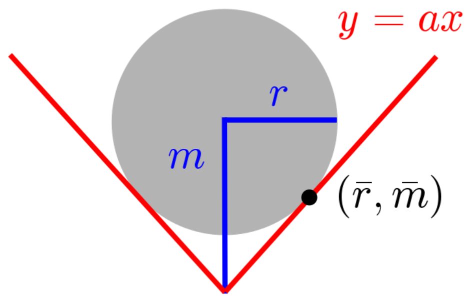

Figure 3: In grey, a ball with radius r and distance $m > r$ from the origin. In red, the boundary of its conic hull. In black, an intersection point between the boundary of the ball and its conic hull.

algorithm. OA and 2×2 are the number of outer approximation and 2×2 cuts added in total, respectively. All other columns are as described in Section 6.3.

Table 8 and Table 6 compare the 2×2+OA configuration with the reported values of algorithm V2 by Saxena, Bonami, and Lee [74, 75]. V2 Gap is the gap closed by the V2 algorithm, and V2 Time the total time spent; likewise Gap Closed and Time are the corresponding values for the 2×2+ OA configuration. For certain instances of GLOBALLib we did not obtain the same initial bound and thus excluded these from comparison. For BoxQP we start with a weak RLT relaxation (with optimal values in the wRLT column of Table 6) to match the setup of Saxena et al. Furthermore, result for V2 were reported only for the smaller instances of BoxQP. V2 uses a RLT relaxation for QCQP problems and applies two types of cuts in standard cutting plane algorithm fashion. The first is an outer-approximation of the PSD constraint of Shor's relaxation; these are convex quadratic cuts unlike the linear cuts we are adding with OA and hence provide stronger but more expensive approximations of SDP+RLT. The second is disjunctive cuts for which the separation procedure involves a mixed-integer linear program.

On GLOBALLib instances our algorithm terminates with slightly smaller gap closed on average, but does produce higher gap closed on Ex5_2_2_case1, Ex5_2_6 and Ex9_2_8. Times are substantially shorter, although the hardware is not specified for V2 results. In Table 6 we run the 2×2+OA configuration with both 600 and 3600 second time limits. The 3600 second time limit produces marginal improvements in gap closed. On smaller cases (spar020 and spar030) V2 closes more gap, but our algorithm closes more gap on the larger spar040 cases with higher density coefficient matrices (spar040-040,...,spar040-100). Due to the strong performance of RLT+SDP on these BoxQP instances, much of the difference could be attributed to differences in the outer-approximation methods used for the PSD constraint. We observe that, by and large, both cutting plane algorithms struggle and succeed to close the optimality gap on the same instances.

Instance	OPT	RLT	RLT-BT	Final LB	Initial Gap	End Gap	Gap Closed	OA	2x2	Iters	Time	LPTime
Ex2_1_1	-17.00	-18.90	-18.90	-17.89	10.6%	4.9%	53.2%	8	228	67	0.41	2.4%
Ex2_1_5	-268.01	-269.45	-269.08	-268.01	0.4%	0.0%	99.6%	2	12	4	0.14	0.0%
Ex2_1_6	-39.00	-44.40	-40.94	-39.04	5.1%	0.1%	97.8%	5	132	39	0.64	1.6%
Ex2_1_7	-4150.41	-13698.36	-5820.01	-5085.63	40.2%	22.5%	44.0%	90	127	52	7	1.0%
Ex2_1_8	15639.00	-82460.00	14439.00	15626.96	7.7%	0.1%	99.0%	222	73	59	13.82	7.2%
Ex2_1_9	-0.38	-2.20	-2.20	-1.66	131.9%	92.5%	29.9%	1400	891	474	36.9	69.8%
Ex3_1_1	7049.25	2533.20	2766.73	2788.77	60.7%	60.4%	0.5%	149	141	58	1.11	37.8%
Ex3_1_2	-30665.54	-30802.76	-30709.10	-30665.55	0.1%	0.0%	100.0%	4	8	5	0.03	0.0%
Ex3_1_3	-310.00	-310.00	-310.00	-	-	-	-	-	-	-	-	-
Ex3_1_4	-4.00	-6.00	-6.00	-5.41	40.0%	28.2%	29.5%	39	228	69	0.26	7.7%
Ex5_2_2_case1	-400.00	-599.90	-599.90	-598.97	49.9%	49.6%	0.5%	417	353	154	4.65	47.7%
Ex5_2_2_case2	-600.00	-1200.00	-1200.00	-1200.00	99.8%	99.8%	0.0%	27	28	11	0.3	3.3%
Ex5_2_2_case3	-750.00	-875.00	-874.80	-873.60	16.6%	16.5%	1.0%	188	182	74	1.67	37.7%
Ex5_2_4	-450.00	-2933.33	-2933.33	-2224.97	550.6%	393.6%	28.5%	402	744	256	3.66	15.6%
Ex5_2_5	-3500.00	-9700.00	-9700.00	-9700.00	177.1%	177.1%	0.0%	38	17	11	5.64	2.8%
Ex5_3_2	1.86	1.00	1.00	1.00	30.1%	30.1%	0.0%	38	17	11	1.97	1.5%
Ex5_3_3	3.23	1.63	1.63	1.68	37.9%	36.7%	3.1%	52	3	11	57.97	2.0%
Ex5_4_2	7512.23	2598.25	3010.67	3022.14	59.9%	59.8%	0.3%	94	81	35	0.57	22.8%
Ex8_4_1	0.62	-4.75	-3.84	-0.84	275.1%	90.0%	67.3%	5357	28	1077	600.69	63.8%
Ex9_1_4	-37.00	-63.00	-62.96	-62.85	68.3%	68.0%	0.4%	389	421	162	3.84	21.1%
Ex9_2_1	17.00	-16.00	1.00	1.00	88.9%	88.9%	0.0%	21	34	11	0.25	0.0%
Ex9_2_2	100.00	-50.00	66.67	89.79	33.0%	10.1%	69.4%	297	348	129	2.62	11.8%
Ex9_2_3	0.00	-30.00	-30.00	-30.00	3000.0%	3000.0%	0.0%	35	19	11	0.88	1.1%
Ex9_2_4	0.50	-597.00	-296.50	-296.50	19800.0%	19800.0%	0.0%	25	30	11	0.22	0.0%
Ex9_2_6	-1.00	-406.00	-201.50	-18.31	10025.2%	865.5%	91.4%	623	127	150	14.37	16.8%
Ex9_2_7	17.00	-16.00	1.00	1.00	88.9%	88.9%	0.0%	21	34	11	0.3	0.0%
Ex9_2_8	1.50	0.50	1.50	-	-	-	-	-	-	-	-	-

Table 6: Detailed results for 2x2 + OA on GLOBALLib instances

Instance	OPT	RLT	Initial Gap	End Gap	Gap Closed	OA	2x2	Iters	Time	LPTime %
spar020-100-1	-706.50	-1066.00	50.81%	0.01%	99.97%	121	63	37	5.64	6.91%
spar020-100-2	-856.50	-1289.00	50.44%	0.10%	99.80%	360	146	102	16.87	14.82%
spar020-100-3	-772.00	-1168.50	51.29%	0.00%	100.00%	30	5	7	1.23	5.69%
spar030-060-1	-706.00	-1454.75	105.91%	1.13%	98.94%	3450	15	693	600.35	58.32%
spar030-060-2	-1377.17	-1699.50	23.39%	0.00%	99.99%	93	77	34	13.41	9.55%
spar030-060-3	-1293.50	-2047.00	58.21%	0.37%	99.36%	2074	470	521	557.87	68.14%
spar030-070-1	-654.00	-1569.00	139.69%	2.87%	97.94%	3402	18	684	601.30	60.23%
spar030-070-2	-1313.00	-1940.25	47.74%	0.00%	99.99%	160	73	47	19.61	11.32%
spar030-070-3	-1657.40	-2302.75	38.91%	0.01%	99.97%	459	203	135	57.93	20.40%
spar030-080-1	-952.73	-2107.50	121.08%	1.31%	98.92%	3274	66	668	601.38	60.68%
spar030-080-2	-1597.00	-2178.25	36.37%	0.00%	100.00%	65	20	17	7.18	6.55%
spar030-080-3	-1809.78	-2403.50	32.79%	0.00%	99.99%	71	26	20	8.42	6.18%
spar030-090-1	-1296.50	-2423.50	86.86%	0.01%	99.99%	355	137	102	42.96	20.25%
spar030-090-2	-1466.84	-2667.00	81.76%	0.01%	99.99%	335	90	89	39.40	20.76%
spar030-090-3	-1494.00	-2538.25	69.85%	0.00%	99.99%	118	74	40	16.25	11.45%
spar030-100-1	-1227.13	-2602.00	111.95%	0.01%	99.99%	756	60	165	79.32	27.84%
spar030-100-2	-1260.50	-2729.25	116.43%	0.02%	99.99%	1808	222	410	257.54	44.54%
spar030-100-3	-1511.05	-2751.75	82.05%	0.15%	99.82%	793	354	234	121.12	33.78%
spar040-030-1	-839.50	-1088.00	29.57%	0.00%	99.99%	945	77	205	270.48	45.74%
spar040-030-2	-1429.00	-1635.00	14.41%	0.00%	99.99%	693	110	165	188.59	37.53%
spar040-030-3	-1086.00	-1303.25	19.99%	0.00%	99.99%	986	114	220	305.45	46.79%
spar040-040-1	-837.00	-1606.25	91.80%	6.16%	93.29%	1669	26	339	600.73	60.30%
spar040-040-2	-1428.00	-1920.75	34.48%	0.00%	99.99%	418	112	111	115.38	32.72%
spar040-040-3	-1173.50	-2039.75	73.75%	2.44%	96.69%	1591	24	323	602.73	62.16%
spar040-050-1	-1154.50	-2146.25	85.83%	2.07%	97.59%	1659	16	335	602.55	60.50%
spar040-050-2	-1430.98	-2357.25	64.68%	0.63%	99.03%	1775	10	357	602.23	57.78%
spar040-050-3	-1653.63	-2616.00	58.16%	0.41%	99.30%	1658	7	333	600.21	61.12%
spar040-060-1	-1322.67	-2872.00	117.05%	4.77%	95.92%	1667	18	337	602.26	61.10%
spar040-060-2	-2004.23	-2917.50	45.54%	0.00%	100.00%	984	48	207	268.05	45.09%
spar040-060-3	-2454.50	-3434.00	39.89%	0.00%	100.00%	165	64	47	42.73	13.95%
spar040-070-1	-1605.00	-3144.00	95.83%	0.00%	100.00%	1353	79	288	426.26	51.88%
spar040-070-2	-1867.50	-3369.25	80.37%	0.00%	100.00%	853	60	187	232.35	41.41%
spar040-070-3	-2436.50	-3760.25	54.31%	0.00%	99.99%	1368	159	313	448.92	50.06%
spar040-080-1	-1838.50	-3846.50	109.16%	0.06%	99.94%	1649	11	332	602.62	61.57%

spar040-080-2	-1952.50	-3833.00	96.26%	0.00%	100.00%	985	85	215	277.94	44.40%
spar040-080-3	-2545.50	-4361.50	71.31%	0.03%	99.96%	1776	179	391	601.69	54.47%
spar040-090-1	-2135.50	-4376.75	104.90%	0.07%	99.93%	1737	8	349	602.56	59.09%
spar040-090-2	-2113.00	-4357.75	106.18%	0.15%	99.86%	1789	1	358	601.22	58.01%
spar040-090-3	-2535.00	-4516.75	78.14%	0.00%	100.00%	441	117	117	118.24	27.11%
spar040-100-1	-2476.38	-5009.75	102.26%	0.00%	100.00%	808	92	183	228.66	42.57%
spar040-100-2	-2102.50	-4902.75	133.12%	0.64%	99.52%	1687	3	338	600.02	60.07%
spar040-100-3	-1866.07	-5075.75	171.91%	4.35%	97.47%	1617	13	326	600.11	61.83%
spar050-030-1	-1324.50	-1858.25	40.27%	2.31%	94.27%	925	15	188	603.07	52.48%
spar050-030-2	-1668.00	-2334.00	39.90%	4.45%	88.84%	936	14	190	601.47	52.07%
spar050-030-3	-1453.61	-2107.25	44.94%	7.22%	83.94%	911	14	185	600.67	53.98%
spar050-040-1	-1411.00	-2632.00	86.47%	7.56%	91.26%	906	19	185	602.20	51.35%
spar050-040-2	-1745.76	-2923.25	67.41%	5.23%	92.24%	919	11	186	602.88	53.03%
spar050-040-3	-2094.50	-3273.50	56.26%	1.09%	98.07%	890	5	179	603.93	50.85%
spar050-050-1	-1198.41	-3536.00	194.89%	50.26%	74.21%	920	65	197	601.70	49.24%
spar050-050-2	-1776.00	-3500.50	97.05%	8.48%	91.26%	968	22	198	602.63	50.32%
spar050-050-3	-2106.10	-4119.75	95.56%	6.83%	92.86%	888	17	181	603.60	54.05%
spar060-020-1	-1212.00	-1757.25	44.95%	44.95%	0.00%	54	1	11	43.39	2.72%
spar060-020-2	-1925.50	-2238.25	16.23%	16.23%	0.00%	52	3	11	42.36	3.05%
spar060-020-3	-1483.00	-2098.75	41.49%	16.44%	60.38%	551	24	115	600.17	45.57%
spar070-025-1	-2538.91	-3832.75	50.94%	17.53%	65.58%	354	36	78	603.14	31.70%
spar070-025-2	-1888.00	-3248.00	72.00%	34.35%	52.29%	355	35	78	600.63	30.92%
spar070-025-3	-2812.28	-4167.25	48.16%	13.91%	71.12%	346	39	77	609.45	32.52%
spar070-050-1	-3252.50	-7210.75	121.66%	18.93%	84.44%	350	35	77	609.62	30.01%
spar070-050-2	-3296.00	-6620.00	100.82%	14.28%	85.84%	355	25	76	607.55	30.33%
spar070-050-3	-4306.50	-7522.00	74.65%	3.60%	95.17%	364	16	76	605.96	29.81%
spar070-075-1	-4655.50	-11647.75	150.16%	12.75%	91.51%	357	13	74	603.94	30.52%
spar070-075-2	-3865.15	-10884.75	181.57%	19.27%	89.39%	350	15	73	600.17	29.43%
spar070-075-3	-4329.40	-11262.25	160.10%	15.57%	90.27%	343	37	76	601.31	27.61%
spar080-025-1	-3157.00	-4840.75	53.32%	25.75%	51.71%	224	21	49	603.39	20.23%
spar080-025-2	-2312.34	-4378.50	89.32%	54.23%	39.28%	240	15	51	601.91	20.41%
spar080-025-3	-3090.88	-5130.25	65.96%	31.26%	52.60%	237	18	51	610.58	22.29%
spar080-050-1	-3448.10	-9783.25	183.68%	48.85%	73.41%	220	30	50	609.10	17.83%
spar080-050-2	-4449.20	-9270.00	108.33%	15.77%	85.45%	229	16	49	607.95	20.13%
spar080-050-3	-4886.00	-10029.75	105.25%	15.37%	85.39%	229	26	51	608.64	19.83%
spar080-075-1	-5896.00	-15250.75	158.64%	14.65%	90.77%	220	20	48	610.97	18.21%

spar080-075-2	-5341.00	-14246.50	166.71%	17.20%	89.68%	222	18	48	605.51	17.49%
spar080-075-3	-5980.50	-14961.50	150.15%	16.46%	89.04%	217	28	49	604.13	17.64%
spar090-025-1	-3372.50	-6171.50	82.97%	55.94%	32.58%	147	28	35	611.75	13.68%
spar090-025-2	-3500.29	-6015.00	71.82%	50.93%	29.09%	145	30	35	608.43	12.53%
spar090-025-3	-4299.00	-6698.25	55.80%	32.76%	41.29%	144	31	35	616.69	15.60%
spar090-050-1	-5152.00	-12584.00	144.23%	42.44%	70.57%	133	37	34	605.23	13.35%
spar090-050-2	-5386.50	-11920.50	121.28%	33.83%	72.11%	131	44	35	613.62	12.83%
spar090-050-3	-6151.00	-12514.00	103.43%	23.61%	77.17%	137	33	34	619.27	15.37%
spar090-075-1	-6267.45	-19054.25	203.99%	52.30%	74.36%	113	47	32	610.76	9.80%
spar090-075-2	-5647.50	-18245.50	223.03%	56.16%	74.82%	116	44	32	615.87	10.41%
spar090-075-3	-6450.00	-18929.50	193.45%	40.61%	79.01%	130	35	33	608.45	12.21%
spar100-025-1	-4027.50	-7660.75	90.19%	76.91%	14.72%	100	20	24	608.35	7.84%
spar100-025-2	-3892.56	-7338.50	88.50%	76.29%	13.79%	108	12	24	610.50	8.82%
spar100-025-3	-4453.50	-7942.25	78.32%	64.89%	17.15%	109	11	24	604.63	9.01%
spar100-050-1	-5490.00	-15415.75	180.76%	98.98%	45.25%	81	34	23	610.73	6.89%
spar100-050-2	-5866.00	-14920.50	154.33%	79.56%	48.45%	89	26	23	606.93	8.30%
spar100-050-3	-6485.00	-15564.25	139.98%	71.68%	48.79%	87	33	24	620.59	8.42%
spar100-075-1	-7384.20	-23387.50	216.69%	78.70%	63.68%	79	26	21	622.82	6.96%
spar100-075-2	-6755.50	-22440.00	232.14%	93.17%	59.86%	71	24	19	607.02	6.05%
spar100-075-3	-7554.00	-23243.50	207.67%	88.07%	57.59%	70	20	18	607.93	5.51%
spar125-025-1	-5572.00	-12251.00	119.85%	119.76%	0.07%	43	2	9	632.22	0.98%
spar125-025-2	-6156.06	-12732.00	106.80%	106.80%	0.00%	44	1	9	633.90	1.08%
spar125-025-3	-6815.50	-12650.75	85.60%	85.61%	0.00%	45	0	9	640.70	1.07%
spar125-050-1	-9308.38	-24993.00	168.48%	156.65%	7.03%	31	4	7	612.13	2.84%
spar125-050-2	-8395.00	-24810.50	195.52%	190.86%	2.38%	22	8	6	632.91	1.65%
spar125-050-3	-8343.91	-24424.50	192.70%	178.70%	7.27%	31	9	8	648.46	2.89%
spar125-075-1	-12330.00	-38202.00	209.81%	190.70%	9.11%	14	6	4	609.09	1.01%
spar125-075-2	-10382.47	-37466.75	260.84%	244.29%	6.34%	15	0	3	616.68	1.02%
spar125-075-3	-9635.50	-36202.25	275.69%	272.41%	1.19%	7	3	2	607.65	0.46%

Table 7: Detailed results for 2x2 + OA cuts on BoxQP instances

Instance	V2 Gap	V2 Time	Gap Closed	Time
Ex2_1_1	72.62%	704.40	53.21%	0.41
Ex2_1_5	99.98%	0.17	99.68%	0.13
Ex2_1_6	99.95%	3397.65	93.87%	0.95
Ex2_1_8	84.70%	3632.28	73.23%	19.13
Ex2_1_9	98.79%	1587.94	29.87%	36.9
Ex3_1_1	15.94%	3600.27	0.34%	0.55
Ex3_1_2	99.99%	0.08	99.98%	0.04
Ex3_1_4	86.31%	21.26	29.49%	0.26
Ex5_2_2_case1	0.00%	0.02	2.05%	0.47
Ex5_2_2_case2	0.00%	0.05	0.00%	0.26
Ex5_2_2_case3	0.36%	0.36	0.00%	0.16
Ex5_2_4	79.31%	68.93	29.04%	5.69
Ex5_3_2	7.27%	245.82	0.00%	2.33
Ex5_4_2	27.57%	3614.38	0.24%	0.59
Ex9_1_4	0.00%	0.60	0.00%	0.34
Ex9_2_1	60.04%	2372.64	54.17%	28.37
Ex9_2_2	88.29%	3606.36	77.90%	30.84
Ex9_2_6	87.93%	2619.02	90.45%	0.12
Ex9_2_8	-	-	83.27%	0.12

Table 8: Comparison with V2 on GLOBALLib instances

	OPT	wRLT	V2 Gap	V2 Time	Gap Closed	Time	Gap Closed	Time
spar020-100-1	-706.5	-1137	95.40%	3638.2	78.32%	600.45	78.34%	3600.97
spar020-100-2	-856.5	-1328.5	93.08%	3636.665	78.61%	600.68	78.61%	1083.09
spar020-100-3	-772	-1224	97.47%	3632.56	83.68%	600.52	83.76%	3600.58
spar030-060-1	-706	-1472.5	60.00%	3823.051	58.38%	600.97	58.51%	3603.02
spar030-060-2	-1377.17	-1741	91.16%	3715.979	88.55%	601.22	88.76%	3601.78
spar030-060-3	-1293.5	-2073.5	77.41%	3696.495	77.14%	600.98	77.22%	3602.11
spar030-070-1	-654	-1647	57.39%	3786.025	53.10%	600.64	53.29%	3610.13
spar030-070-2	-1313	-1989.5	86.60%	3708.212	81.52%	601.14	81.62%	3600.68
spar030-070-3	-1657.4	-2367.5	88.66%	3744.044	86.29%	601.01	86.38%	3600.67
spar030-080-1	-952.729	-2189	69.67%	3600.777	56.96%	600.4	57.04%	3602.02
spar030-080-2	-1597	-2316	86.25%	3627.132	73.92%	601.11	73.97%	3600.21
spar030-080-3	-1809.78	-2504.5	91.42%	3666.392	85.14%	601.34	85.28%	3601.93
spar030-090-1	-1296.5	-2521	81.15%	3676.815	70.02%	600.85	70.09%	3600.91
spar030-090-2	-1466.84	-2755	82.66%	3646.756	71.56%	601.03	71.62%	3600.09
spar030-090-3	-1494	-2619.5	86.37%	3701.849	75.33%	600.57	75.56%	3602.01
spar030-100-1	-1227.13	-2683.5	81.10%	3692.504	69.68%	601.17	69.78%	3602.26
spar030-100-2	-1260.5	-2870.5	72.87%	3697.329	63.60%	600.05	63.66%	3601.47
spar030-100-3	-1511.05	-2831.5	84.10%	3606.496	75.06%	600.42	75.13%	3602.89
spar040-030-1	-839.5	-1162	31.05%	3719.223	50.80%	600.67	57.21%	3605.46
spar040-030-2	-1429	-1695	27.74%	3937.898	17.58%	15.57	17.58%	15.49
spar040-030-3	-1086	-1322	28.00%	3798.683	7.73%	15.88	7.73%	16.97
spar040-040-1	-837	-1641	33.31%	3817.844	39.77%	600.53	42.76%	3602.06
spar040-040-2	-1428	-1967.5	35.19%	3968.111	56.73%	600.08	61.35%	3600.24
spar040-040-3	-1173.5	-2089	26.71%	3972.902	43.30%	600.42	47.36%	3603.8
spar040-050-1	-1154.5	-2204	36.72%	3819.72	46.05%	600.54	49.94%	3602.77
spar040-050-2	-1430.98	-2403.5	40.87%	3610.64	50.39%	601.65	53.90%	3602.41
spar040-050-3	-1653.63	-2715	33.95%	3639.977	46.20%	600.35	48.93%	3604.85
spar040-060-1	-1322.67	-2934	47.75%	3760.964	51.73%	601.64	54.33%	3604.92
spar040-060-2	-2004.23	-3011	55.79%	3707.992	68.04%	601.1	71.04%	3600.39
spar040-060-3	-2454.5	-3532	72.63%	3764.079	74.36%	601.57	76.45%	3600.44
spar040-070-1	-1605	-3194.5	64.03%	3642.681	67.37%	601.47	69.39%	3602
spar040-070-2	-1867.5	-3446.5	57.91%	3756.377	62.79%	601.51	64.77%	3605.11
spar040-070-3	-2436.5	-3833.5	62.94%	3693.666	68.40%	600.45	70.59%	3604.16
spar040-080-1	-1838.5	-3969	58.37%	3808.258	59.04%	601.59	61.09%	3600.31

spar040-080-2	-1952.5	-3902.5	66.96%	4062.433	63.85%	600.69	65.25%	3602.2
spar040-080-3	-2545.5	-4440	72.31%	4057.149	73.35%	600.42	74.94%	3601.66
spar040-090-1	-2135.5	-4490	66.64%	3781.044	66.03%	600.92	67.48%	3603.5
spar040-090-2	-2113	-4474	66.46%	3931.349	65.74%	600.33	67.12%	3603.76
spar040-090-3	-2535	-4641	73.49%	4003.706	73.80%	601.32	75.18%	3600.81
spar040-100-1	-2476.38	-5118	76.24%	3853.573	74.81%	601.45	76.31%	3605.11
spar040-100-2	-2102.5	-5043	63.89%	3658.261	64.80%	601.7	66.14%	3603.14
spar040-100-3	-1866.07	-5196.5	59.92%	3842.685	61.46%	602.01	63.58%	3600.51

Table 9: Comparison with V2 on BoxQP instances