On the Separation of Split Inequalities for Non-Convex Quadratic Integer Programming

Christoph Buchheim, Emiliano Traversi
Fakultät für Mathematik, Technische Universität Dortmund

February 21, 2013

Abstract

We investigate the computational potential of split inequalities for non-convex quadratic integer programming, first introduced by Letchford [11] and further examined by Burer and Letchford [8]. These inequalities can be separated by solving convex quadratic integer minimization problems. For small instances with box-constraints, we show that the resulting dual bounds are very tight; they can close a large percentage of the gap left open by both the RLT- and the SDP-relaxations of the problem. The gap can be further decreased by separating so-called non-standard split inequalities, which we examine in the case of ternary variables.

1 Introduction

The standard formulation of an (unconstrained) Integer Quadratic Programming Problem (IQP) is the following:

$$\min\{x^\top Qx + L^\top x + c \mid x \in \mathbb{Z}^n,\ l \leq x \leq u\} \tag{1}$$

with $Q \in \mathbb{Q}^{n\times n}$, $L \in \mathbb{Q}^n$, $c \in \mathbb{Q}$, $l \in (\mathbb{Z} \cup \{-\infty\})^n$, and $u \in (\mathbb{Z} \cup \{\infty\})^n$. We assume Q to be symmetric without loss of generality. However, we do not require Q to be positive semidefinite. In other words, we do not assume convexity of the objective function

$$f(x) := x^\top Qx + L^\top x + c\ .$$

Problem (1) is thus NP-hard both by the non-convexity of the objective function and by the integrality constraints on the variables. More precisely, the problem remains NP-hard in the convex case, i.e., when $Q \succeq 0$, even if all bounds are infinite or if all variables are binary. In the first case, Problem (1) is equivalent to the *closest vector problem* [17]; in the second case, it is equivalent to *binary quadratic programming* and *max-cut* [16]. Moreover, if f is non-convex and integrality is relaxed, i.e., if the variable x_i can be chosen in the interval $[l_i, u_i]$, the resulting problem is called *BoxQP* and is again NP-hard.

One approach for solving (1) is based on the idea of getting rid of the non convexity of f and then using a convex IQP solver. Billionnet et al. [3, 4, 5] proposed an approach consisting in reformulating the objective function and obtaining an equivalent one with a convex quadratic objective function. The approach aims at finding a convex reformulation that gives the highest value of its continuous relaxation. However, convexification requires a binary expansion of each non-binary variable, resulting in a large number of additional variables and possibly leading to numerical problems.

Another natural approach to get rid of the non-convexity of f consists in linearization. This approach has been investigated intensively in the literature. Let S_k be the set of symmetric matrices of dimension k. Using the linearization function $\ell\colon \mathbb{Q}^n \to S_{n+1}$ defined by

$$\ell(x) = \begin{pmatrix} 1 \\ x \end{pmatrix} \begin{pmatrix} 1 \\ x \end{pmatrix}^\top$$

and setting

$$\tilde{Q} = \begin{pmatrix} c & \frac{1}{2}L^\top \\ \frac{1}{2}L & Q \end{pmatrix}$$

we can replace Problem (1) by the following equivalent problem:

$$\begin{array}{lrcll} \min & \langle \tilde{Q}, \ell(x) \rangle & & & \\ \text{s.t.} & x & \in & \mathbb{Z}^n & \\ & x_i & \in & [l_i, u_i] & \text{for } i = 1, \ldots, n \end{array} \tag{2}$$

where $\langle ., . \rangle$ denotes the inner product.

We can hence work in an extended space introducing a new set of variables X_{ij} with $i = 0 \ldots, n$. The new linearized formulation is obtained by substituting each product $x_i x_j$, appearing in row i and column j of $\ell(x)$, by a new variable X_{ij}. For consistency, in this new formulation the linear component x_i is substituted by the new variable X_{0i}. Finally, for a reason that will later become clearer, we also introduce a new variable X_{00}. In this way the new space contains the original n variables and $1 + \binom{n}{2}$ new variables. All variables are collected in a symmetric matrix X of dimension $n+1$, representing $\ell(x)$. The dimension of the extended space is thus

$$d(n) = \frac{(n+1)(n+2)}{2} - 1 \, .$$

The main challenge is now to ensure $X = \ell(x)$. This is equivalent to requiring

$$X_{00} = 1, \quad \text{rank}(X) = 1, \quad \text{and } X \succeq 0 \, .$$

Hence, one way to reformulate Problem (1) is as follows:

$$\begin{array}{lrcll} \min & \langle \tilde{Q}, X \rangle & & & \\ \text{s.t.} & X_{00} & = & 1 & \\ & \text{rank}(X) & = & 1 & \\ & X & \succeq & 0 & \\ & X_{i0} & \in & \mathbb{Z} & \text{for } i = 1, \ldots, n \\ & X_{i0} & \in & [l_i, u_i] & \text{for } i = 1, \ldots, n \end{array} \tag{3}$$

Working in the X-space allows more freedom and several reformulations and relaxations of Problem (3) can be defined. By eliminating the rank constraint and the integrality constraints we obtain the SDP-relaxation (SDP) of Problem (3). Buchheim and Wiegele [6] devise a branch-and-bound algorithm based on this continuous relaxation.

An alternative is to work with an ILP formulation and then use its continuous relaxation for computing bounds. Sherali and Adams [15] proposed a unifying framework for strengthening the linearization using the so-called RLT inequalities. By taking into account the bounds on the original variables $l_i \leq x_i \leq u_i$, we can add the following RLT-inequalities:

$$\begin{aligned} X_{ij} - l_i X_{j0} - l_j X_{i0} &\geq -l_i l_j \\ X_{ij} - u_i X_{j0} - u_j X_{i0} &\geq -u_i u_j \\ -X_{ij} + l_i X_{j0} + u_j X_{i0} &\geq l_i u_j \\ -X_{ij} + u_i X_{j0} + l_j X_{i0} &\geq u_i l_j \end{aligned}$$

The four inequalities above were originally introduced by McCormick [12]. Anstreicher [1] uses RLT-inequalities for strengthening SDP-relaxations for BoxQP problems. He investigates the relation between the SDP-relaxation, the linear relaxation with RLT inequalities and the SDP-relaxation with RLT inequalities. He shows that adding RLT inequalities to the SDP-relaxation improves the resulting bounds. From a practical point of view, a drawback of this approach is that SDP solvers have problems in handling the additional RLT-inequalities.

Another class of valid inequalities that can be used for strengthening the extended formulation are the so-called psd inequalities, introduced by Laurent and Poljak [10]:

$$\langle vv^\top, X \rangle \geq 0 \quad \forall v \in \mathbb{Q}^{n+1} \tag{4}$$

By definition, a matrix X is positive semidefinite if and only if it satisfies all psd inequalities, so that (4) is equivalent to $X \succeq 0$. In theory, the psd constraints can thus replace the constraint $X \succeq 0$ in an LP-based approach. Qualizza et al. [13] try to solve quadratically constrained quadratic problems using psd inequalities; these are separated heuristically on the fly. The drawback of this approach is the huge number of inequalities needed in order to model $X \succeq 0$ appropriately.

Most literature on non-convex quadratic optimization does not take integrality into account. One exception is the SDP-based approach mentioned above [6]. Moreover, Saxena et al. [14] investigate the use of disjunctive cuts for non-convex quadratic programming; these cuts are derived both from the integrality constraints of the variables and from the non-convex constraint $X = \binom{1}{x}\binom{1}{x}^\top$.

The main objective of this paper is to investigate split inequalities introduced by Letchford [11] and further examined by Burer and Letchford [8]. Split inequalities result from tightening psd inequalities, exploiting integrality of all variables. In our framework, they can be formulated as

$$\langle v(v + e_0)^\top, X \rangle \geq 0 \quad \forall v \in \mathbb{Z}^{n+1} \ , \tag{5}$$

where e_0 denotes the unit vector for dimension zero. In particular, we are interested in the potential of split inequalities for improving dual bounds compared to the RLT-relaxation and the SDP-relaxation. We perform numerical experiments to compare the RLT-relaxation and the SDP-relaxation to the relaxation given by all split inequalities.

Unfortunately, we do not know whether split inequalities can be separated in polynomial time, but we agree with the conjecture of Burer and Letchford [8] that this separation problem is NP-hard. However, from a practical point of view, this separation problem is considerably easier than Problem (1): if $X \succeq 0$, it can be reduced to a *convex* quadratic integer minimization problem; otherwise it essentially suffices to compute an eigenvector corresponding to a negative eigenvalue of X. Using an algorithm for convex quadratic integer programming proposed by Buchheim et al. [7], we can thus solve the separation problem of split inequalities fast enough to derive conclusions about the strength of split inequalities. It turns out that split inequalities can close a huge percentage of the gap even when compared to RLT-and SDP-relaxations.

In Section 2, we introduce basic definitions used throughout the paper and recall some fundamental properties of the convex hull of feasible solutions. In Section 3, we recall the definition of split inequalities and collect some theoretical results concerning the split closure. In Section 4, we explain our separation algorithm for split inequalities. In Section 5, we extend our analysis to non-standard split inequalities, focussing on the ternary case. Different from standard split inequalities, non-standard split inequalities take variable bounds into account. Finally, in Section 6, the strength of the split closure is determined experimentally.

2 Preliminaries

The aim of this paper is to compare various relaxations of Problem (1). These relaxations correspond to different sets containing the convex hull of feasible solutions in the extended space. We start by introducing some basic notation.

Definition 1 *For $n \in \mathbb{N}$, we define*

(a) $\mathscr{S}_n := \{X \in \mathbb{Q}^{(n+1)\times(n+1)} \mid X_{00} = 1,\ X_{ij} = X_{ji} \text{ for all } i, j\}$

(b) $\mathscr{S}_n^+ := \{X \in \mathscr{S}_n \mid X \succeq 0\}$

For technical reasons, we include the constraint $X_{00} = 1$ in the definitions of the sets $\mathscr{S}_n$ and $\mathscr{S}_n^+$. The set $\mathscr{S}_n^+$ is the feasible region of the SDP-relaxation obtained from (3) when relaxing integrality, box constraints, and the rank constraint. The convex hull of feasible solutions is now given as follows:

Definition 2 *For $n \in \mathbb{N}$, we define*

(a) $\mathrm{IQ}_n := \mathrm{clconv}\,\{\ell(x) \mid x \in \mathbb{Z}^n\}$

(b) $\mathrm{IQ}_n^b := \mathrm{conv}\,\{\ell(x) \mid x \in \{-b, \dots, b\}^n\}$

The set IQ_n has been introduced by Letchford [11]. By definition, Problem (1) is equivalent to minimizing the linear function $\langle \tilde{Q}, . \rangle$ over IQ_n^b, if $l_i = -b$ and $u_i = b$ for all $i = 1, \dots, n$. From the definition, we immediately derive

$$\mathrm{IQ}_n = \bigcup_{b \in \mathbb{N}} \mathrm{IQ}_n^b \,.$$

The following two results have been shown by Letchford [11]. For sake of completeness, and since we use a slightly different notation, we repeat the proofs here.

Theorem 1 *The extreme points of IQ_n are exactly the points $\ell(x)$ with $x \in \mathbb{Z}^n$.*

Proof. By definition, each vertex of IQ_n is of the form $\ell(x)$ for $x \in \mathbb{Z}^n$. Now for $\overline{x} \in \mathbb{Z}^n$, define

$$\tilde{Q} = \begin{pmatrix} \overline{x}^\top \overline{x} & -\overline{x}^\top \\ -\overline{x} & I_n \end{pmatrix} .$$

Then $\langle \tilde{Q}, \ell(x) \rangle = (x - \overline{x})^\top (x - \overline{x})$, thus $\langle \tilde{Q}, \ell(\overline{x}) \rangle = 0$ and $\langle \tilde{Q}, \ell(x) \rangle \geq 1$ for all $x \in \mathbb{Z}^n \setminus \{\overline{x}\}$. This shows that $\ell(\overline{x})$ is an extreme point of IQ_n. $\square$

Lemma 1 *The automorphism group* $\mathrm{Aut}(\mathrm{IQ}_n)$ *acts transitively on the extreme points of* IQ_n.

Proof. Choose $\overline{x} \in \mathbb{Z}^n$. It suffices to construct an automorphism φ of IQ_n that maps $\ell(0)$ to $\ell(\overline{x})$. For $X \in \mathscr{S}_n$, let $X_0 \in \mathbb{R}^{n+1}$ denote the first column of X. It is easy to verify that the affine map

$$\begin{aligned} \varphi \colon \mathscr{S}_n &\to \mathscr{S}_n \\ X &\mapsto X + 2\begin{pmatrix} 0 \\ \overline{x} \end{pmatrix} X_0^\top + \begin{pmatrix} 0 \\ \overline{x} \end{pmatrix}\begin{pmatrix} 0 \\ \overline{x} \end{pmatrix}^\top \end{aligned}$$

maps $\ell(x)$ to $\ell(x+\overline{x})$ for all $x \in \mathbb{Z}^n$ and hence induces a bijection on the set of extreme points of IQ_n with $\varphi(\ell(0)) = \ell(\overline{x})$. □

Lemma 1 shows that all extreme points of IQ_n are isomorphic from the polyhedral point of view. In particular, it suffices to examine the extreme point $\ell(0)$.

As obvious from Figure 2, the set IQ_n is not a polyhedron in the unbounded case if $n = 1$. This is due to the fact that IQ_n has an infinite number of facets. On the other hand, taking bounds into account explicitly, it is clear by definition that the set IQ_n^b is a polytope. The following theorem shows that IQ_n is not even locally polyhedral for $n \geq 2$:

Theorem 2 *For* $n \geq 2$, *every extreme point of* IQ_n *belongs to infinitely many facets of* IQ_n.

Proof. By Lemma 1, it suffices to show that $\ell(0)$ belongs to infinitely many facets of IQ_n. Moreover, we may assume $n = 2$. For $k \in \mathbb{N}$, define $v_k = (0, 1, k)^\top$. We claim that the inequality $\langle v_k(v_k + e_0)^\top, X\rangle \geq 0$ is facet-inducing for IQ_2, for all $k \in \mathbb{N}$. Define

$$X_k^0 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}, X_k^1 = \begin{pmatrix} 1 & -1 & 0 \\ -1 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}, X_k^2 = \begin{pmatrix} 1 & 0 & -\frac{1}{k} \\ 0 & 0 & 0 \\ -\frac{1}{k} & 0 & \frac{1}{k^2} \end{pmatrix},$$

$$X_k^3 = \begin{pmatrix} 1 & \frac{1}{k} & -\frac{1}{k^2} \\ \frac{1}{k} & 1 & -\frac{1}{k} \\ -\frac{1}{k^2} & -\frac{1}{k} & \frac{1}{k^2} \end{pmatrix}, X_k^4 = \begin{pmatrix} 1 & -\frac{1}{k} & \frac{1}{k^2} \\ -\frac{1}{k} & 1 & -\frac{1}{k} \\ \frac{1}{k^2} & -\frac{1}{k} & \frac{1}{k^2} \end{pmatrix}.$$

By definition, each matrix X_k^i satisfies $\langle v_k(v_k + e_0)^\top, X\rangle = 0$ and $X_k^0 = \ell(0)$. Moreover, it is easy to check that all X_k^i are affinely independent. It thus remains to show that each X_k^i belongs to IQ_2. For $i = 0, 1, 2$, this is clear. For $i = 3, 4$, we have that

$$k^2 X_k^i - (k^2 - 1)X_k^0 = \begin{pmatrix} 1 & \pm k & \mp 1 \\ \pm k & k^2 & -k \\ \mp 1 & -k & 1 \end{pmatrix} \in \mathrm{IQ}_2\,,$$

so that by convexity $X_k^i = \frac{1}{k^2}(k^2 X_k^i - (k^2 - 1)X_k^0) + \frac{k^2-1}{k^2} X_k^0 \in \mathrm{IQ}_2$. □

By Theorem 2, the set IQ_n does not only have an infinite number of facets, but even locally an infinite subset of these facets is needed to define IQ_n. This means that even in the bounded case, the intersection of IQ_n with the bound constraints is not a polyhedron. The facet-inducing inequalities used in the proof of Theorem 2 are split inequalities, discussed in the next section.

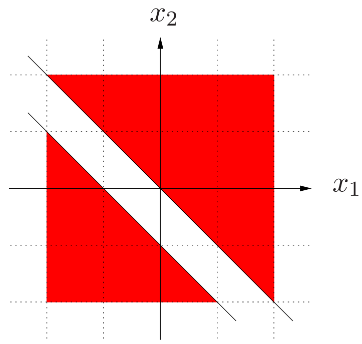


Figure 1: A standard split inequality

3 The Split Closure

A *split* corresponds to a disjunction of the form

$$(w^\top x \le s) \ \vee \ (w^\top x \ge s+1) \tag{6}$$

with $w \in \mathbb{Z}^n$ and $s \in \mathbb{Z}$, which is obviously satisfied by any $x \in \mathbb{Z}^n$. Disjunction (6) imposes to the points in the feasible region to belong either to the halfspace $w^\top x \le s$ or to the halfspace $w^\top x \ge s+1$; see Figure 1 for an illustration. In other words $w^\top x - s$ and $w^\top x - s - 1$ have to be of the same sign for each feasible x. We obtain the following valid quadratic inequality:

$$(w^\top x - s)(w^\top x - s - 1) \ge 0$$

In the extended space of X variables, the split inequalities become linear:

$$\langle v(v+e_0)^\top, X\rangle \ge 0$$

with

$$v = \begin{pmatrix} -s-1 \\ w \end{pmatrix} \in \mathbb{Z}^{n+1} .$$

We derive

Lemma 2 *For each $v \in \mathbb{Z}^{n+1}$, the* split inequality $\langle v(v+e_0)^\top, X\rangle \ge 0$ *is valid for* IQ_n.

Split inequalities for IQP have been first examined by Letchford [11]. They are different from split inequalities for linear optimization problems [2]. The main difference is that in the quadratic case we do not have to take into account the underlying polyhedron, their validity is a direct consequence of the fact that we are dealing with integer variables and that we allow quadratic constraints. Note that Lemma 2 also holds for IQ_n^b, since $\mathrm{IQ}_n^b \subset \mathrm{IQ}_n$. On the other hand, an inequality that is facet defining for IQ_n may no longer be so if box constraints are considered.

Example 1 *Figure 1 illustrates the split inequality* $(x_1+x_2 \le -1) \vee (x_1+x_2 \ge 0)$ *or, alternatively,*

$$\left\langle \begin{pmatrix} 0 \\ 1 \\ 1 \end{pmatrix} \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}^\top , X \right\rangle \ge 0 .$$

Lemma 3 *Let $v \in \mathbb{Z}^{n+1}$. Then the split inequality*

$$\langle v(v+e_0)^\top, X\rangle \geq 0$$

induces a facet of IQ_n *if and only if $v_1, \ldots, v_n$ are relatively prime.*

Lemma 3 has been shown by Letchford [11]; also cf. Theorem 5 in Burer and Letchford [8]. Notice that the facet-inducing inequalities of Theorem 2 are split inequalities, showing that every extreme point of IQ_n satisfies an infinite number of facet-inducing split inequalities with equality.

Definition 3 *For $n \geq 1$, the* split closure SC_n *is defined as*

$$\mathrm{SC}_n := \{X \in \mathscr{S}_n \mid \langle v(v+e_0)^\top, X\rangle \geq 0 \text{ for all } v \in \mathbb{Z}^{n+1}\} .$$

The objective of this paper is to investigate how closely SC_n approximates IQ_n or IQ_n^b. We first examine some basic properties of the sets $\mathscr{S}_n^+$, IQ_n, and SC_n. The following lemma is easily verified.

Lemma 4 *For $n, b \geq 1$, we have $\mathrm{IQ}_n^b \subseteq \mathrm{IQ}_n \subseteq \mathscr{S}_n^+$. All these sets are full-dimensional in $\mathscr{S}_n$.*

In particular, as $X_{00} = 1$, every valid linear inequality for each of the three sets can be written as $\langle A, X\rangle \geq 0$ for a symmetric matrix A that is unique up to scaling.

Theorem 3 *For all $n \geq 1$, we have $\mathrm{IQ}_n \subseteq \mathrm{SC}_n \subseteq \mathscr{S}_n^+$.*

Proof. The first inclusion follows immediately from Lemma 2. The second inclusion follows from Theorem 4 in [8]. □

The second inclusion $\mathrm{SC}_n \subseteq \mathscr{S}_n^+$ of Theorem 3 implies that a suitable selection of split inequalities provides a bound at least as good as the SDP-relaxation given by X_{00} and $X \succeq 0$. In Section 6.1, we report the results of a series of experiments performed in order to establish the bound improvement on random instances.

Lemma 5 *For all $n \geq 1$, we have $\mathrm{SC}_n \neq \mathscr{S}_n^+$.*

Proof. Consider the matrix $X \in \mathscr{S}_n$ with $X_{00} = 1$, $X_{01} = X_{10} = \frac{3}{8}$, $X_{11} = \frac{1}{4}$ and zero otherwise. It can be checked that $X \in \mathscr{S}_n^+$. On the other hand, the matrix X violates the split inequality $X_{11} - X_{01} \geq 0$, so that $X \notin \mathrm{SC}_n$. □

Lemma 5 shows that split inequalities yield a strictly tighter relaxation than the SDP-relaxation. On the other hand, it implies that, when considering the separation problem for split inequalities, the point to be separated could belong to the positive semidefinite cone. This will be taken into account in the separation routine proposed in Section 4.

It remains to examine the relation between IQ_n and SC_n. It is easy to verify that $\mathrm{IQ}_1 = \mathrm{SC}_1$; see Figure 2. Moreover, we have the following result:

Theorem 4 *Every extreme point of* IQ_n *is an extreme point of* SC_n.

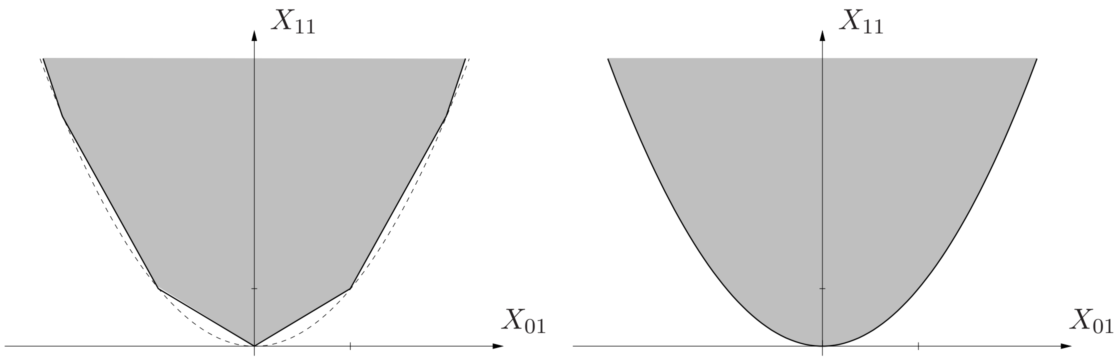


Figure 2: The sets $\mathrm{IQ}_1 = \mathrm{SC}_1$ (left) and $\mathscr{S}_1^+$ (right)

Proof. By Lemma 1 it suffices to show that $\ell(0)$ is an extreme point of SC_n. Let

$$\mathcal{V} := \{e_i, -e_i \mid i = 1, \dots, n\} \cup \{e_i + e_j \mid i, j = 1, \dots, n,\ i \neq j\} \ .$$

Each $v \in \mathcal{V}$ yields a facet-inducing inequality $\langle v(v+e_0)^\top, X\rangle \geq 0$ for SC_n. As $v_0 = 0$ for all $v \in \mathcal{V}$, we obtain $\langle v(v+e_0)^\top, \ell(0)\rangle = 0$. It is readily checked that all matrices $v(v+e_0)^\top$, $v \in \mathcal{V}$, are linearly independent. Since $|\mathcal{V}| = 2n + \binom{n}{2} = d(n)$, we obtain the result. $\square$

In [11], it was posed as an open question whether the sets IQ_2 and SC_2 coincide. In the meantime, Burer and Letchford [8] showed that this holds true, whereas $\mathrm{IQ}_6 \neq \mathrm{SC}_6$. It is not known whether IQ_n agrees with SC_n for $n \in \{3, 4, 5\}$.

In Appendix A, we present an alternative geometric proof showing $\mathrm{IQ}_2 = \mathrm{SC}_2$.

4 Separation Methods

Algorithmically speaking, Theorem 2 states that the separation problem for SC_n has to deal with an infinite number of potential cutting planes. In this section we develop such a separation algorithm. In fact, finding a violated split inequality for X^* reduces to a quadratic integer minimization problem: recall that we are looking for a $v \in \mathbb{Z}^{n+1}$ such that

$$\langle v(v+e_0)^\top, X^*\rangle < 0 \ .$$

As $\langle v(v+e_0)^\top, X^*\rangle = v^\top X^* v + v^\top X^* e_0$, the separation problem can be written as follows:

$$\begin{array}{ll} \min & v^\top X^* v + v^\top X^* e_0 \\ \text{s.t.} & v \ \in \ \mathbb{Z}^{n+1} \ . \end{array} \tag{7}$$

Notice that the separation problem (7) is of dimension $n+1$ and X^* can be an arbitrary symmetric matrix with $X^*_{00} = 1$.

As mentioned by Letchford [11], it is unclear whether the separation of split inequalities is an NP-hard problem or not. Even if the complexity of the separation problem is unknown, problem (7) becomes convex whenever $X^* \in \mathscr{S}_n^+$. From a practical point of view, convex quadratic integer minimization is much easier than its non-convex counterpart, though still NP-hard. In our implementation, we solve (7) using the algorithm introduced by Buchheim et al. [7].

In the following, assume that $X^* \notin \mathscr{S}_n^+$. Using (7) for separation would lead to an unbounded problem in this case. As mentioned in the introduction, separating a point that is not in $\mathscr{S}_n^+$ can be done by separating a violated psd inequality (4). In theory this would allow to converge to a point $X^* \in \mathscr{S}_n^+$ and then proceed as above. Even if this procedure is correct, a large number of psd inequalities is usually produced. It is preferable to generate a stronger split inequality immediately. To this end, we use an incremental rounding of eigenvectors corresponding to negative eigenvalues in order to obtain the vector v defining the split inequality. More precisely, if $v \in \mathbb{Q}^{n+1}$ is an eigenvector of X^* belonging to a negative eigenvalue, we have $\langle vv^\top, X^*\rangle < 0$. We then scale v in order to obtain an integer vector $\overline{v} \in \mathbb{Z}^{n+1}$ such that $\langle \overline{v}(\overline{v}+e_0)^\top, X^*\rangle < 0$.

Algorithm 1: Incremental Rounding Separation for Split Inequalities

---

input : a matrix $X^* \in \mathscr{S}_n \setminus \mathscr{S}_n^+$
output: a vector $\overline{v} \in \mathbb{Z}^{n+1}$ such that $\langle \overline{v}(\overline{v}+e_0)^\top, X^*\rangle < 0$
compute an eigenvector v of X^* corresponding to an eigenvalue $\lambda < 0$;
set $\varepsilon = \max\{|v_i| \mid i = 0, \dots, n\}$;
repeat
  for $i = 0, \dots, n$ **do**
    set $\overline{v}_i := \lfloor \frac{v_i}{\varepsilon} \rfloor$;
  end
  set $\varepsilon := \frac{\varepsilon}{2}$;
until $\langle \overline{v}(\overline{v}+e_0)^\top, X^*\rangle < 0$;
return $\overline{v}$;

Theorem 5 *Algorithm 1 is correct and runs in polynomial time.*

Proof. The eigenvector can be computed in polynomial time and its encoding length is polynomial. In particular, after a polynomial number k of iterations we obtain $\frac{1}{\varepsilon}v \in \mathbb{Z}^{n+1}$ and hence $\overline{v} = \frac{1}{\varepsilon}v$. Thus after $l \geq k$ iterations we have $\overline{v} = 2^{l-k}v_k$ for some $v_k \in \mathbb{Z}^{n+1}$ of polynomial size, which implies

$$\langle \overline{v}(\overline{v}+e_0)^\top, X^*\rangle = \overline{v}^\top X^* \overline{v} + \overline{v}^\top X^* e_0 = 2^{l-k}\left(2^{l-k} v_k^\top X^* v_k + v_k^\top X^* e_0\right) .$$

As $v_k^\top X^* v_k < 0$ and both $v_k^\top X^* v_k$ and $v_k^\top X^* e_0$ have polynomial encoding length, we obtain

$$\langle \overline{v}(\overline{v}+e_0)^\top, X^*\rangle < 0$$

after a polynomial number of iterations. □

Algorithm 1 produces any violated split inequality separating $X^* \notin \mathscr{S}_n^+$, with no explicit objective such as, e.g., maximizing the violation. In general, one can expect stronger cutting planes for eigenvectors corresponding to more negative eigenvalues; in our experiments in Section 6 we always use an eigenvector corresponding to the smallest eigenvalue of X^*.

By Lemma 3, each integer vector v with relatively prime entries $v_1, \dots, v_n$ yields a non-dominated split inequality. In particular, split inequalities are dense and have large coefficients in general. The incremental procedure of Algorithm 1 starts with checking the cut corresponding to a split with only one non-zero entry, more precisely the one corresponding to the biggest entry (in absolute value) of the considered eigenvector. Iteration by iteration, the number of non-zero entries increases. Similarly, also the range of possible values of the entries increases with the number of iterations; at iteration k, the splits can take values in the set $\{-2^k, -2^k - 1, \dots, 2^k - 1, 2^k\}$. This means that

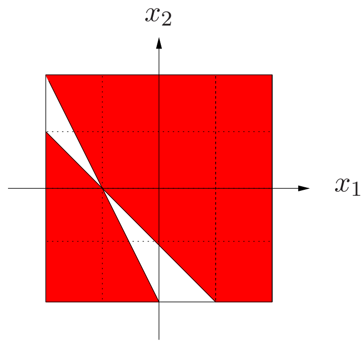


Figure 3: Example of a non-standard split inequality

the algorithm is designed to heuristically produce a cut with few non-zero entries and with small coefficients. Such cuts are preferable in order to avoid numerical problems.

An exact separation algorithm for SC_n can now be composed of both the convex quadratic minimization approach for the case $X^* \in \mathscr{S}_n^+$ and Algorithm 1 for the case $X^* \notin \mathscr{S}_n^+$. Also from the practical point of view, both procedures are crucial in order to compute exact optima over the split closure, as both cases appear regularly. However, it is possible to combine the separation of split inequalities with the SDP-relaxation, in this case only the first separation procedure is needed.

5 Bounded Case: Non Standard Split Inequalities

Up to now the presence of bounds has never been taken into account explicitly. Even for small values of n and b, split inequalities and box constraints together do not suffice to describe IQ_n^b completely. In this section we are interested in inequalities that are valid for IQ_n^b but not dominated by split inequalities. By using the software PORTA [9] it is possible to compute a complete description of the easiest non-trivial example IQ_2^1. It is composed of the following types of inequalities:

(a) split inequalities

(b) bound constraints

(c) non-standard split inequalities.

The latter can be defined (and generalized) as follows: let $v', v'' \in \mathbb{Z}^{n+1}$ and consider the inequality $\langle v'v''^\top, X\rangle \geq 0$. The validity of this inequality depends on the bounds we impose on the variables involved. As an example, consider the following inequality:

$$(x_1 + x_2 + 1)(2x_1 + x_2 + 2) \geq 0$$

which imposes that the feasible points have to either belong to both the halfspaces $x_1 + x_2 + 1 \geq 0$ and $2x_1 + x_2 + 2 \geq 0$ or to none of them. In Figure 3 the two hyperplanes are illustrated, together with the feasible region of the inequality. As is clear from the picture, the inequality is valid for IQ_n^2 (and also for IQ_n^1), but not for IQ_n^b with $b \geq 3$.

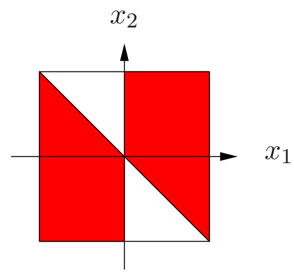


Figure 4: Example of a non-standard split inequality valid for SC_2^1

In the extended space the inequality can be written as follows:

$$\left\langle \begin{pmatrix} 2 & 2 & 1 \\ 2 & 2 & 1 \\ 2 & 2 & 1 \end{pmatrix}, X \right\rangle \geq 0$$

Inequalities of the shape $\langle v' v''^\top, X \rangle \geq 0$ have been called non-standard split inequalities and considered for the first time by Letchford [11]. Notice that only standard split inequalities are valid in the unbounded setting.

Even if the definition is intuitive, it is hard to characterize valid non-standard split inequalities for IQ_n^b, for general b. However, for $b = 1$ (the ternary case) it is easy to find the relevant non-standard split inequalities:

Lemma 6 *For any vector $v \in \mathbb{Z}^{n+1}$ with $v_0 = 0$, the non-standard split inequalities*

$$\langle v(v+e_i)^\top, X \rangle \geq 0 \ , \ \ i = 1, \ldots, n$$

are valid for IQ_n^1.

Figure 4 illustrates the non standard split inequality

$$\left\langle \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} \begin{pmatrix} 0 \\ 1 \\ 1 \end{pmatrix}^\top, X \right\rangle \geq 0$$

which is valid for IQ_2^1.

Similar to the case of standard split inequalities, the separation problem for non-standard split inequalities in the ternary case reduces to the following quadratic integer minimization problem, to be solved for each $i = 1, \ldots, n$:

$$\begin{array}{ll} \min & v^\top X^* v + v^\top X^* e_i \\ \text{s.t.} & v_0 \ = \ 0 \\ & v \ \in \ \mathbb{Z}^{n+1} \ . \end{array} \tag{8}$$

As we can see, the separation of non-standard split inequalities is analogous to the one for standard inequalities, see (7), but in one dimension less. On the other hand, we now have an additional degree of freedom represented by the choice of the index $i = 1, \ldots, n$. As in the standard case, the optimization problem (8) is convex if $X^* \in \mathscr{S}_n^+$, in this case we can solve it using the algorithm of [7] again. Similarly, if $X^* \notin \mathscr{S}_n^+$, we can apply a procedure analogous to Algorithm 1.

6 Computational Results

6.1 Quality of the Split Closure

In the following experimental evaluation, we investigate the tightness of the split closure SC_n in the ternary case, i.e., with respect to IQ_n^1. In this section, we do not consider running time issues; our objective is to determine how much of the gap can be closed by split inequalities with respect to the SDP-relaxation and the RLT-relaxation. In particular, we focus our study on the root node, i.e., we do not incorporate the bound computation into a branch-and-bound scheme.

As we are considering ternary instances, we can also consider non-standard split inequalities in our experiments. Split inequalities and non-standard split inequalities are added on the fly by using the separation techniques explained in Sections 4 and 5.

We generated a set of random ternary instances with $n = 10$. All possible levels of convexity of Q are tested: from zero negative eigenvalues (convex objective function) to ten negative eigenvalues (concave objective function). For a given number of negative eigenvalues, we randomly generated 10 different instances with 10 different seeds. The coefficients of Q, L and c are created as follows: given the number of negative eigenvalues p, we choose p negative numbers λ_i with $i = 1, \ldots, p$ uniformly at random from the interval $[-1, 0]$ and $n - p$ uniformly at random from $[0, 1]$. Next, we generate n vectors of length n each, now chosing all entries at random from $[-1, 1]$, and orthonormalize them to obtain vectors v_i. The coefficient matrix Q is then calculated as $Q = \sum_{i=1}^n \lambda_i v_i v_i^\top$. Finally, we determine L by choosing all entries uniformly at random in the interval $[-1, 1]$ and set c to zero. Therefore, the final test bed counts 110 instances.

We first compare the split closure to the LP relaxation based on the RLT-inequalities. We thus consider the following RLT-relaxation:

$$
\begin{array}{lrcll}
\min & \langle \tilde{Q}, X \rangle & & & \\
\text{s.t.} & X & \in & \mathscr{S}_n & \\
& X_{ij} + X_{0j} + X_{i0} & \geq & -1 & \text{for } i,j = 1,\ldots,n \\
& X_{ij} - X_{j0} - X_{i0} & \geq & -1 & \text{for } i,j = 1,\ldots,n \\
& -X_{ij} - X_{j0} + X_{i0} & \geq & -1 & \text{for } i,j = 1,\ldots,n \\
& -X_{ij} + X_{j0} - X_{i0} & \geq & -1 & \text{for } i,j = 1,\ldots,n \\
& X_{ii} - X_{i0} & \geq & 0 & \text{for } i = 1,\ldots,n \\
& X_{ii} + X_{i0} & \geq & 0 & \text{for } i = 1,\ldots,n \\
& X_{0i} & \in & [-1,1] & \text{for } i = 1,\ldots,n \\
& X_{ii} & \in & [0,1] & \text{for } i = 1,\ldots,n
\end{array}
$$

For a second comparison, we consider the SDP-relaxation:

$$
\begin{array}{lrcll}
\min & \langle \tilde{Q}, X \rangle & & & \\
\text{s.t.} & X & \in & \mathscr{S}_n^+ & \\
& X_{0i} & \in & [-1,1] & \text{for } i = 1,\ldots,n \\
& X_{ii} & \in & [0,1] & \text{for } i = 1,\ldots,n
\end{array}
$$

Three different separation algorithms are used:

- `STD_IR`
  Only the Incremental Rounding procedure for split inequalities described in Algorithm 1 is

applied. In particular, the separation continues only as long as the point X^* to be separated is outside $\mathscr{S}_n^+$. Hence, the final value provided depends on the sequence of points X^* to be separated. This means that the use of a different LP solver could lead to a different final solution. For this reason, the setting STD_IR only provides a heuristic separation.

- `STD_ALL`
  In addition to Algorithm 1, the convex quadratic integer minimization problem (7) is used for separating points $X^* \in \mathscr{S}_n^+$, as described in Section 4. This extension turns `STD_ALL` into an exact separation algorithm.

- `STD+NONSTD`
  This procedure consists of the exact separation algorithm for both split inequalities and non-standard split inequalities, as described in Section 5. The separation of non-standard split inequalities is performed only if no split inequalities are found. Because of Theorem 3, this means that this will only occur for points in $\mathscr{S}_n^+$. Hence the separation of non-standard split inequalities always requires solving the convex quadratic integer minimization problem (8).

Let $ROOT_{\texttt{ALGO}}$ be the root node lower bound provided by the corresponding algorithm `ALGO`. We have the following relation:

$$ROOT_{\texttt{STD\_IR}} \leq ROOT_{\texttt{STD\_ALL}} \leq ROOT_{\texttt{STD+NONSTD}}$$

In Table 1 we present the comparison of SC_n with the RLT-relaxation. Let OPT be the optimal value and $ROOT_{\texttt{RLT}}$ be the optimal value of the continuous relaxation with RLT-inequalities. For every algorithm we provide the percentage of integrality gap closed, i.e. the value

$$100 \times \frac{ROOT_{\texttt{ALGO}} - ROOT_{\texttt{RLT}}}{OPT - ROOT_{\texttt{RLT}}}$$

In order to assess the size of this gap, for every instance we report the size of the normalized integrality gap (`GAP`), i.e. the value

$$100 \times \frac{OPT - ROOT_{\texttt{RLT}}}{|OPT|} \tag{9}$$

In every row we report the average results related to ten instances with the same number of negative eigenvectors. Only instances with an open gap (i.e. with $OPT > ROOT_{\texttt{RLT}}$) are taken into account when computing the avarage of gap closed. The optimal values are computed using the SDP based algorithm devised in [6]. Table 2 reports the same type of information as Table 1 when SC_n is compared with the SDP-relaxation.

From the results in Table 1 we observe that all three separation algorithms are able to close more than 90% of the gap of the RLT-relaxation on average, even the heuristic separation `STD_IR`. These significant results are due also to the high integrality gap of the model used.

More interesting is the analysis of Table 2. First we notice that now the integrality gap is definitely more tight (4% on average). The heuristic `STD_IR` is using Algorithm 1 and it can be viewed as a way to strengthen psd inequalities (4). In practice this improvement turns out to be significant: it is able to close on average more than 50% of the gap and almost 25% in the worst scenario (instances with 9 negative eigenvalues). The separation `STD_ALL` shows our main computational results: the split closure is able to close on average around two third of the gap left open by the SDP-relaxation and almost 50% of the gap in the worst cases (instances with 9 or 5 eigenvectors), decreasing the

# neg. eigenv.	GAP	STD_IR	STD_ALL	STD+NONSTD
0	345.28	100.00	100.00	100.00
1	97.51	99.17	99.25	99.28
2	70.31	96.47	97.08	97.50
3	44.13	93.89	94.80	95.79
4	48.82	92.64	93.74	95.17
5	48.91	93.06	93.99	94.84
6	47.64	94.56	95.83	96.63
7	43.67	93.86	95.27	96.64
8	33.22	93.36	95.26	96.64
9	18.32	87.76	91.07	95.36
10	8.14	71.90	79.23	83.27
average	73.27	**92.42**	**94.14**	**95.56**

Table 1: RLT gap closed

# neg. eigenv.	GAP	STD_IR	STD_ALL	STD+NONSTD
0	4.12	100.00	100.00	100.00
1	2.71	94.92	95.71	96.46
2	4.21	73.07	79.77	81.80
3	5.75	52.16	61.12	66.42
4	5.33	48.53	56.44	67.81
5	5.71	41.55	49.70	57.11
6	3.91	41.91	64.24	72.38
7	3.59	33.73	57.50	70.55
8	2.71	29.85	60.07	69.42
9	2.27	24.67	49.25	68.02
10	1.83	30.82	55.12	66.35
average	3.83	**51.87**	**66.36**	**74.35**

Table 2: SDP gap closed

average open gap from 3.38% to 1.29%. The addition of non standard split inequalities brings the percentage of average gap closed to three quarters. This improvement is more relevant for the hardest instances to be solved, i.e., the ones with more than two negative eigenvectors. Both Tables 1 and 2 indicate that the instances with at least three negative eigenvalues are the ones with a gap most difficult to be closed.

6.2 Performance of the Separation Algorithm

In this section we investigate the behaviour in terms of time of the separation algorithm proposed in Section 4. As test bed we use four sets of instances of increasing size ($n = 20, 30, 40, 50$). For a given size n, we generated 110 instances of different convexity and seed as explained in the previous section.

As separation procedure, we chose the algorithm `STD_IR`, which has the best performance when taking running time into account. We keep track of the average gap closed (9) during the processing of the root node. In addition to the information reported, we indicate in grey the average number of inequalities added. Finally, the average gap closed by the SDP relaxation is reported as additional reference. In Figure 5 we plot the behaviour in the first 100 seconds, in Figure 6 the same information restricted to the first 10 seconds is reported.

As we can see, conducting the separation until the end is not doable in practice, as this approach takes too much time. This suggests that split inequalities should not be used as the only separation procedure in a branch-and-cut algorithm. On the other hand, less than ten cuts are enough in order to close around 40% of the gap. This shows that the separation of split inequalites can be one valuable tool within a branch-and-cut framework for non-convex integer quadratic programming.

A Alternative proof of $\mathrm{IQ}_2 = \mathrm{SC}_2$

In order to show $\mathrm{IQ}_2 = \mathrm{SC}_2$, we use the following geometric reformulation:

Theorem 6 *For $n \geq 1$, the following statements are equivalent:*

(a) $\mathrm{IQ}_n = \mathrm{SC}_n$

(b) For every lattice-point free ellipsoid $E \subset \mathbb{Q}^n$, there is a split containing $E \cap \mathbb{Z}^n$.

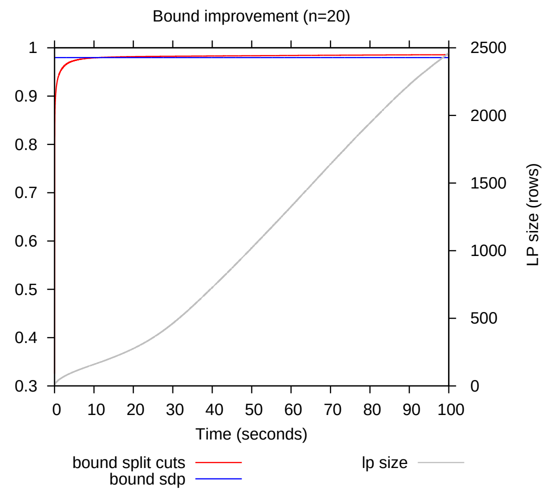
Proof. Assume that (a) holds and let E be a lattice-point free ellipsoid in $\mathbb{Q}^n$. Then there exist $A \succeq 0$ and $v \in \mathbb{Q}^n$ such that $E = E(A, v) := \{x \in \mathbb{Q}^n \mid (x - v)^\top A(x - v) \leq 1\}$. Let

$$\tilde{A} = \begin{pmatrix} v^\top Av - 1 & -(Av)^\top \\ -Av & A \end{pmatrix} .$$

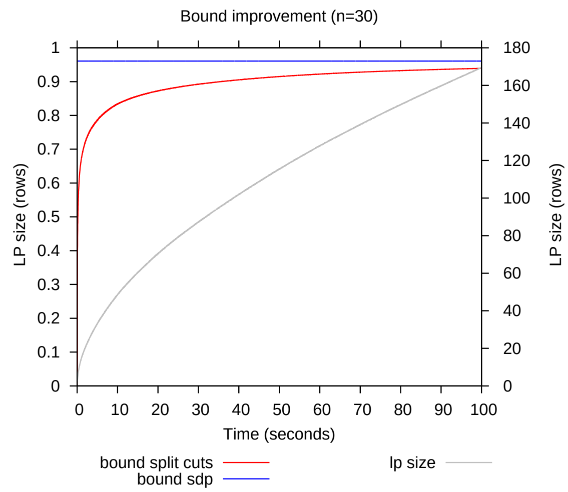
As E is lattice-point free, we derive $\langle \tilde{A}, \ell(x) \rangle = (x - v)^\top A(x - v) - 1 \geq 0$ for all $x \in \mathbb{Z}^n$ so that $\langle A, X \rangle \geq 0$ is a valid linear inequality for IQ_n. By (a), this inequality is dominated by some split inequality corresponding to a vector

$$v = \begin{pmatrix} v_0 \\ w \end{pmatrix} \in \mathbb{Z}^{n+1} .$$

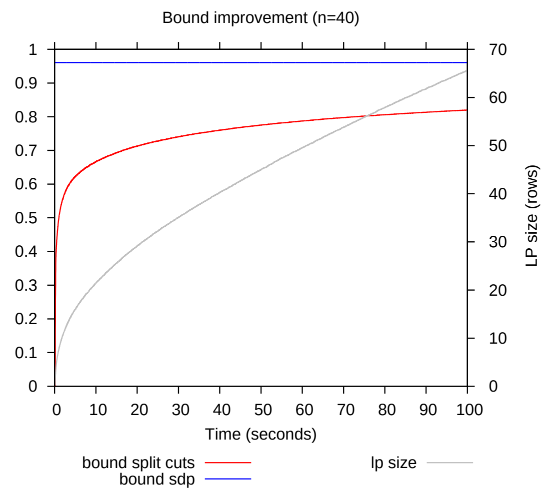
H



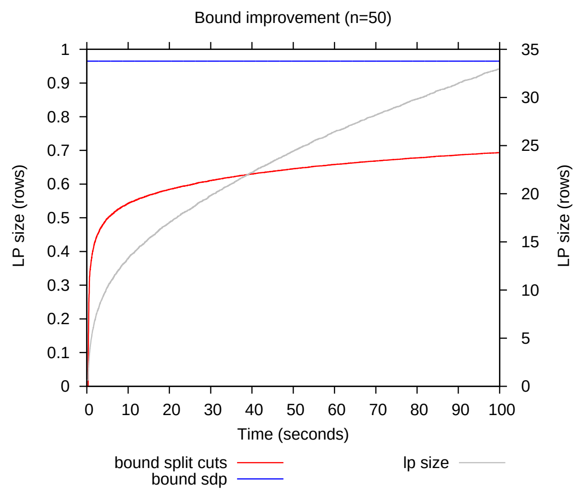
(a) $n = 20$



(b) $n = 30$



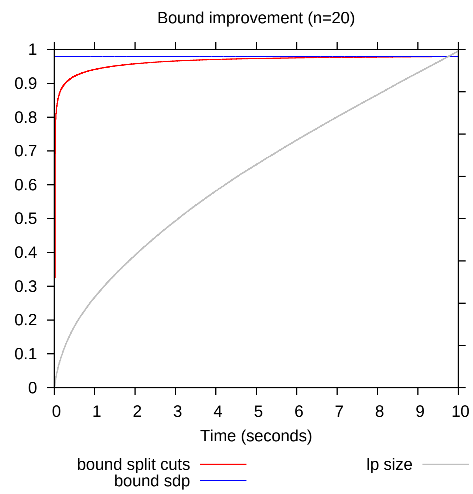
(c) $n = 40$



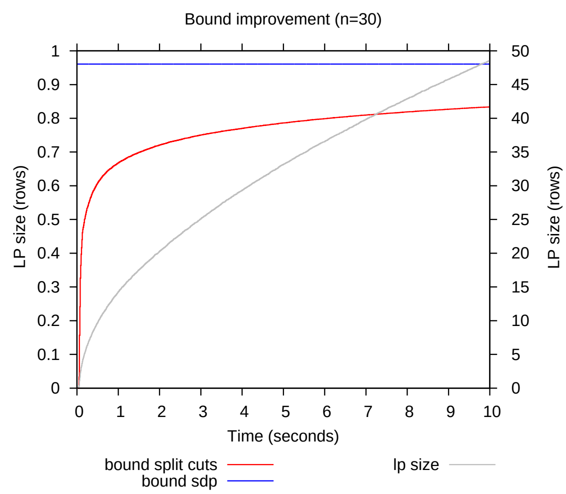
(d) $n = 50$

Figure 5: Bound improvement, first 100 seconds

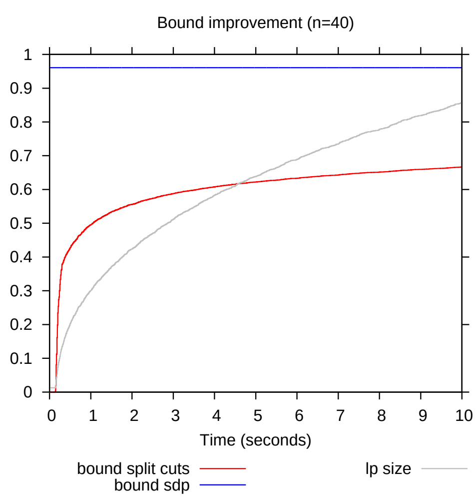
H



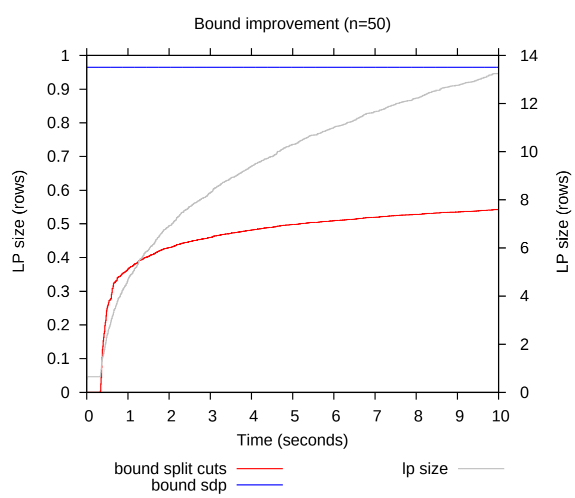
(a) $n = 20$



(b) $n = 30$



(c) $n = 40$



(d) $n = 50$

Figure 6: Bound improvement, first 10 seconds

Now $x \in E \cap \mathbb{Z}^n$ implies $\langle A, \ell(x) \rangle = 0$ and thus $\langle v(v+e_0)^\top, \ell(x) \rangle = 0$. The latter yields

$$x \in E\Big(4ww^\top, \frac{-v_0 - \frac{1}{2}}{w^\top w} w\Big) ,$$

the split corresponding to v. Now assume that (b) holds and consider any valid linear inequality $\langle \tilde{A}, X \rangle \geq 0$ for IQ_n, where

$$\tilde{A} = \begin{pmatrix} a_{00} & a^\top \\ a & A \end{pmatrix} .$$

Then $x^\top A x + 2a^\top x + a_{00} = \langle \tilde{A}, \ell(x) \geq 0$ for all $x \in \mathbb{Z}^n$, so that $A \succeq 0$ and $E := \{x \in \mathbb{Q}^n \mid \langle \tilde{A}, X \rangle \leq 0\}$ is a lattice-point free ellipsoid. By assumption, $E \cap \mathbb{Z}^n$ is contained in some split. The corresponding split inequality dominates $\langle \tilde{A}, X \rangle \geq 0$. $\square$

Lemma 7 *Let $E \subset \mathbb{Q}^n$ be a lattice-point free ellipsoid and $P := \mathrm{conv}(E \cap \mathbb{Z}^n)$. If P has any non-degenerate vertex v, then $E \cap \mathbb{Z}^n$ is contained in a split of $\mathbb{Z}^n$.*

Proof. By transition to an appropriate subspace, we may assume that E is a non-degenerate ellipsoid and that $\dim P = n$. Let x_0 be a non-degenerate vertex of P and let $x_1, \ldots, x_n$ be its neighboring vertices. As E is non-degenerate, the only integer points of P are $E \cap \mathbb{Z}^n$. In particular, we may assume by an appropriate base change that $x_0 = 0$ and $x_i = e_i$ for all $i = 1, \ldots, n$. By construction, we have $E \cap \mathbb{Z}^n \subseteq \mathbb{Z}^n_+$. We now claim that $E \cap \mathbb{Z}^n \subseteq \{0,1\}^n$. Indeed, let $x \in \mathbb{Z}^n_+$ such that $x_i \geq 2$ for some i. In particular, $e^\top x \geq 2$, where e denotes the all-ones vector. Consider $y_i \in \{0,1\}^n$ with $y_i = 1$ if and only if $x_i \geq 1$. Then $y = \sum_{i=1}^n \lambda_i e_i + \lambda x$ with

$$\lambda_i = \frac{x_i - y_i}{e^\top x - 1} \geq 0 \ \text{ and } \ \lambda = 1 - \frac{e^\top x - e^\top y}{e^\top x - 1} \geq 0 ,$$

hence y belongs to $\mathrm{conv}(\{e_i \mid x_i \geq 1\} \cup \{x\})$. This implies $x \notin E$. $\square$

Figure 7 illustrates the proof of Lemma 7. As every polytope in dimension two has only non-degenerate vertices, we obtain

Corollary 1 *The sets IQ_2 and SC_2 coincide.*

References

[1] K. Anstreicher. Semidefinite programming versus the reformulation-linearization technique for nonconvex quadratically constrained quadratic programming. *Journal of Global Optimization*, 43:471–484, 2009.

[2] E. Balas. Disjunctive programming. *Annals of Discrete Mathematics*, 5:3–51, 1979.

[3] A. Billionnet and S. Elloumi. Using a mixed integer quadratic programming solver for the unconstrained quadratic 0-1 problem. *Mathematical Programming*, 109:55–68, 2007.

[4] A. Billionnet, S. Elloumi, and A. Lambert. Extending the QCR method to general mixed-integer programs. *Mathematical Programming*.

[5] A. Billionnet, S. Elloumi, and M. Plateau. Improving the performance of standard solvers for quadratic 0-1 programs by a tight convex reformulation: The QCR method. *Discrete Appl. Math.*, 157:1185–1197, 2009.

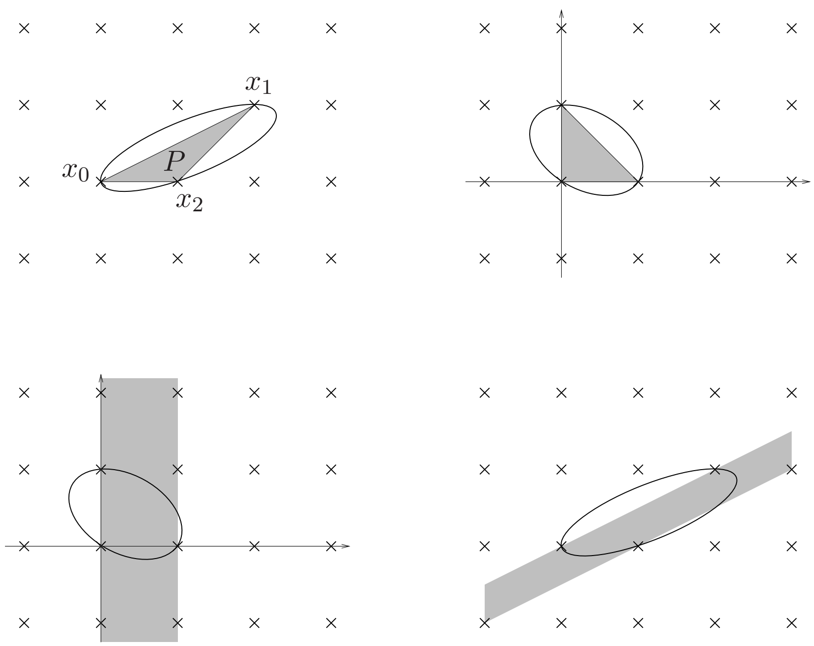


Figure 7: Finding a split containing $E \cap \mathbb{Z}^n$

[6] C. Buchheim and A. Wiegele. Semidefinite relaxations for non-convex quadratic mixed-integer programming. *Optimization Online*, 6, 2011. To appear in *Mathematical Programming A*.

[7] C. Buchheim, A. Caprara, and A. Lodi. An effective branch-and-bound algorithm for convex quadratic integer programming. In Friedrich Eisenbrand and F. Shepherd, editors, *Integer Programming and Combinatorial Optimization*, volume 6080 of *Lecture Notes in Computer Science*, pages 285–298. Springer Berlin / Heidelberg, 2010.

[8] S. Burer and A. Letchford. Unbounded convex sets for non-convex mixed-integer quadratic programming. *Mathematical Programming A*, 2012. To appear.

[9] T. Christof and A. Löbel. PORTA – POlyhedron Representation Transformation Algorithm. 2009. URL `typo.zib.de/opt-long_projects/Software/Porta`.

[10] M. Laurent and S. Poljak. On a positive semidefinite relaxation of the cut polytope. *Linear Algebra and its Applications*, 223224(0):439 – 461, 1995.

[11] A. Letchford. Integer quadratic quasi-polyhedra. In Friedrich Eisenbrand and F. Shepherd, editors, *Integer Programming and Combinatorial Optimization*, volume 6080 of *Lecture Notes in Computer Science*, pages 258–270. Springer Berlin / Heidelberg, 2010.

[12] G. McCormick. Computability of global solutions to factorable nonconvex programs: Part I Convex underestimating problems. *Mathematical Programming*, 10(1):147–175, 1976.

[13] A. Qualizza, P. Belotti, and F. Margot. Linear programming relaxations of quadratically constrained quadratic programs. 2010.

[14] A. Saxena, P. Bonami, and J. Lee. Convex relaxations of non-convex mixed integer quadratically constrained programs: extended formulations. *Mathematical Programming*, 124(1–2): 383–411, 2010.

[15] H. Sherali and W. Adams. *A Reformulation-Linearization Technique for Solving Discrete and Continuous Nonconvex Problems*. Springer, 1998.

[16] C. De Simone. The cut polytope and the boolean quadric polytope. *Discrete Mathematics*, 79: 71–75, 1989/90.

[17] P. van Emde Boas. Another NP-complete problem and the complexity of computing short vectors in a lattice. Technical report, University of Amsterdam, Department of Mathematics, 1981.