CANADA EXCELLENCE RESEARCH CHAIR
DATA SCIENCE FOR REAL-TIME DECISION-MAKING

LATTICE REFORMULATIONS CUTS

Karen Aardal
Frederik von Heymann
Andrea Lodi
Andrea Tramontani
Laurence A. Wolsey

July 2018

DS4DM-2018-003

POLYTECHNIQUE MONTRÉAL
DÉPARTEMENT DE MATHÉMATIQUES ET GÉNIE INDUSTRIEL
Pavillon André-Aisenstadt
Succursale Centre-Ville C.P. 6079
Montréal - Québec
H3C 3A7 - Canada
Téléphone: 514-340-5121 # 3314

Mathematical Programming manuscript No.
(will be inserted by the editor)

Lattice Reformulation Cuts

Karen Aardal · Frederik von Heymann · Andrea Lodi · Andrea Tramontani · Laurence A. Wolsey

Received: date / Accepted: date

Abstract Here we consider the question whether the lattice reformulation of a linear integer program can be used to produce effective cutting planes. We consider integer programs (IP) in the form $\max\{\boldsymbol{cx} \mid \boldsymbol{Ax} = \boldsymbol{b}, \boldsymbol{x} \in \mathbb{Z}^n_+\}$, where the reformulation takes the form $\max\{\boldsymbol{cx}^0 + \boldsymbol{cQ\mu} \mid \boldsymbol{Q\mu} \geq -\boldsymbol{x}^0,\ \boldsymbol{\mu} \in \mathbb{Z}^{n-m}\}$, where $\boldsymbol{Q}$ is an $n \times (n-m)$ integer matrix. Working on an optimal LP tableau in the $\boldsymbol{\mu}$-space allows us to generate $n-m$ Gomory mixed-integer inequalities (GMIs) in addition to the m GMIs associated with the optimal tableau in the $\boldsymbol{x}$ space. These provide new cuts that can be seen as GMIs associated to $n-m$ non-elementary split directions associated with the reformulation matrix $\boldsymbol{Q}$. On the other hand it turns out that the corner polyhedra associated to an LP basis and the GMI or split closures are the same whether working in the $\boldsymbol{x}$ or $\boldsymbol{\mu}$ spaces. Computationally we show that the effectiveness of the cuts generated by this approach depends on the quality of the reformulation obtained by the reduced basis algorithm used to generate $\boldsymbol{Q}$ and that it is worthwhile to generate several rounds of such cuts. However, the effectiveness of the cuts deteriorates as the number of constraints is increased.

The research has been financed in part by The Netherlands Organisation for Scientific Research, NWO, grant number 613.000.801, which we gratefully acknowledge.

K. Aardal
Delft University of Technology, The Netherlands. E-mail: k.i.aardal@tudelft.nl

F. von Heymann
University of Cologne, Germany. E-mail: fred@fredvonheymann.eu

A. Lodi
CERC, École Polytechnique de Montréal, Canada. E-mail: andrea.lodi@polymtl.ca

A. Tramontani
CPLEX Optimization, IBM, Italy. E-mail: andrea.tramontani@it.ibm.com

L.A. Wolsey
Université catholique de Louvain, Belgium. E-mail: laurence.wolsey@uclouvain.be

Keywords Integer programming · Cutting planes · lattice reformulation · lattice basis reduction

Mathematics Subject Classification (2010) 90C10 · 11H06

1 Introduction

In a series of papers Aardal et al. [1–4] have shown that certain integer programs that cannot be solved by a standard MIP solver can be solved by using a lattice-reformulation of the problem. This raises the question studied here of whether such a lattice-reformulation can also be used to produce effective cutting planes.

Specifically we consider pure integer programs (IP) in the form

$$\max\{\boldsymbol{cx} \mid \boldsymbol{Ax} = \boldsymbol{b}, \boldsymbol{x} \in \mathbb{Z}^n_+\}\,, \tag{1}$$

where $\boldsymbol{A} \in \mathbb{Z}^{m\times n}$, $\boldsymbol{b} \in \mathbb{Z}^m$, $\boldsymbol{c} \in \mathbb{Z}^n$, and where we let $P = \{\boldsymbol{x} \in \mathbb{R}^n_+ \mid \boldsymbol{Ax} = \boldsymbol{b}\}$ and $S = P \cap \mathbb{Z}^n$.

The reformulation takes the form

$$\max\{\boldsymbol{cx}^0 + \boldsymbol{cQ\mu} \mid \boldsymbol{Q\mu} \geq -\boldsymbol{x}^0,\ \boldsymbol{\mu} \in \mathbb{Z}^{n-m}\}\,, \tag{2}$$

where $\boldsymbol{Q}$ is an $n \times (n-m)$ integer matrix and $\boldsymbol{x}^0$ is a point satisfying $\boldsymbol{Ax}^0 = \boldsymbol{b},\ \boldsymbol{x}^0 \in \mathbb{Z}^n$. Here we let $\hat{P} = \{\boldsymbol{\mu} \in \mathbb{R}^{n-m} : \boldsymbol{Q\mu} \geq -\boldsymbol{x}^0\}$ and $\hat{S} = \hat{P} \cap \mathbb{Z}^{n-m}$. The integer sets S and $\hat{S}$ are related: $\boldsymbol{x} \in S$ if and only if there exists $\boldsymbol{\mu} \in \hat{S}$ with $\boldsymbol{x} = \boldsymbol{x}^0 + \boldsymbol{Q\mu}$, or in other words $S = \text{proj}_x\{(\boldsymbol{x}, \boldsymbol{\mu}) : \boldsymbol{x} = \boldsymbol{x}^0 + \boldsymbol{Q\mu}, \boldsymbol{\mu} \in \hat{S}\}$.

The intuition behind our approach is that the polytope $\hat{P}$ has a regular shape in the sense that it does not have an obvious thin direction, if $\hat{P}$ is created after a basis reduction process of $\boldsymbol{Q}$, and that therefore split disjunctions in the coordinate directions of $\hat{P}$ are potentially interesting. This idea is supported by the computational experience with branch and bound on $\hat{P}$ rather than on P. Branching in unit directions on $\hat{P}$ has proven to be computationally more effective for certain problem types, see e.g. [2,3]. Thus our motivation is to look for Gomory Mixed-Integer (GMI) cuts [20] that are not necessarily tableau cuts for P, but are still computationally easy to generate.

A first practical observation is that if one considers the reformulated problem (2), one can generate $(n-m)$ new Chvátal-Gomory (CG) [10] or GMI cuts off an optimal linear program (LP) tableau. Here we will concentrate on GMI cuts (also viewed as split cuts [12]). These will be called (lattice) *ℓ-cuts*. This raises a series of questions both theoretical and computational. For example:

- What is the relationship between P and $\hat{P}$?
- Given a point $\boldsymbol{\mu} \in \hat{S}$, what is the corresponding point $\boldsymbol{x} \in S$?
- What do the ℓ-cuts in the $\boldsymbol{\mu}$-space give in the $\boldsymbol{x}$-space?
- Are the corner polyhedra associated to a basis in the $\boldsymbol{x}$ and $\boldsymbol{\mu}$ spaces the same?
- What, if any, is the relationship between the GMI or split closures of P and $\hat{P}$?

Computational questions that we investigate are:

- How effective are the ℓ-cuts?
- Can the ℓ-cuts associated to a basis tableau be easily generated in the $\boldsymbol{x}$-space?

We now point to some related computational work. Bixby et al. [7] observed that one round of GMI inequalities generated from an optimal basic solution closed 24% of the integrality gap on average on 41 MIPLIB instances. Cornuéjols et al. [14] suggested to multiply a row in the optimal LP tableau by an integer k, and then derive a GMI off of the resulting row. They called a cut generated in this way a k-cut. The standard tableau GMI inequality is a k-cut with $k = 1$. One motivation behind this approach is to create a large fractional right-hand side of the resulting tableau row as this intuitively could lead to a stronger inequality. Later Cornuéjols [13] suggested that one should look for deep split cuts that can be separated efficiently. This is also the viewpoint taken here.

An alternative, but very costly approach, is to generate all the inequalities from a given family, known as the closure. Balas and Saxena [6] performed a computational study of the split inequalities and concluded that the split closure closed 82% of the integrality gap, on average, on 33 mixed integer MIPLIB instances, and 71%, on average, on 24 pure integer MIPLIB instances. It is, however, NP-hard to optimize a linear function over the split closure [9], so achieving these results is computationally expensive. Of course, a vast literature has been devoted to computationally viable ways of approximating the split closure, see, e.g., Dash and Goycoolea [16] and Fischetti and Salvagnin [19].

In Section 2 we present the background material we need concerning inequalities and lattices. In Section 3 we see that most of the theoretical questions have simple and perhaps surprising answers. In particular, even though the GMI/split cuts generated may be different, the GMI/split closures are the same. We give a description of our approach for generating violated inequalities in Section 4 and present our computational results comparing different possible variants in Section 5. Finally, some conclusions are drawn in Section 6.

2 Background

2.1 Gomory mixed integer inequalities and split inequalities

We define Gomory mixed-integer inequalities (GMI) and split inequalities, k-cuts and closures. For a more general exposition we refer to [11, 27].

Consider the single row mixed-integer set

$$X = \{(\boldsymbol{x}, \boldsymbol{y}) \in \mathbb{Z}^n_+ \times \mathbb{R}^p_+ \mid \sum_{j=1}^{n} a_j x_j + \sum_{j=1}^{p} g_j y_j = b\} \tag{3}$$

and suppose that $b \notin \mathbb{Z}$. Let

$$b := \lfloor b \rfloor + f_0 \text{ with } 0 < f_0 < 1,$$
$$a_j := \lfloor a_j \rfloor + f_j \text{ with } 0 \le f_j < 1.$$

The *Gomory mixed-integer (GMI) inequality* [20] for X is

$$\sum_{\{j:f_j \le f_0\}} \frac{f_j}{f_0} x_j + \sum_{\{j:f_j > f_0\}} \frac{1-f_j}{1-f_0} x_j + \sum_{\{j:g_j>0\}} \frac{g_j}{f_0} y_j - \sum_{\{j:g_j<0\}} \frac{g_j}{1-f_0} y_j \ge 1\,. \quad (4)$$

If the row (3) is a row from a simplex tableau of a linear relaxation, the associated GMI inequality is referred to as a *tableau GMI inequality*.

Cornuéjols et al. [14] introduced *k-cuts*, which are cuts that are obtained by first multiplying (3) from an optimal tableau in which one of the $\boldsymbol{x}$-variables is basic by an integer k, and then deriving the GMI inequality. In this paper we introduce ℓ-cuts, which are tableau GMI cuts derived from an optimal tableau of the LP-relaxation of (2). In Sections 3 and 4 we explain how to generate these cuts in the space of the $\boldsymbol{x}$-variables.

Let T be a polyhedron in $\mathbb{R}^{n+p}$. Next, we consider a mixed integer set $T \cap (\mathbb{Z}^n \times \mathbb{R}^p)$. For given $(\boldsymbol{\pi}, \pi_0) \in \mathbb{Z}^{n+1}$ we define

$$\Pi_1 := T \cap \{(\boldsymbol{x}, \boldsymbol{y}) \in \mathbb{Z}^n \times \mathbb{R}^p \mid \boldsymbol{\pi x} \le \pi_0\}$$
$$\Pi_2 := T \cap \{(\boldsymbol{x}, \boldsymbol{y}) \in \mathbb{Z}^n \times \mathbb{R}^p \mid \boldsymbol{\pi x} \ge \pi_0 + 1\}\,.$$

An inequality $\boldsymbol{\alpha x} + \boldsymbol{\gamma y} \le \beta$ is called a *split inequality* [12] if there exists a $(\boldsymbol{\pi}, \pi_0) \in \mathbb{Z}^{n+1}$ such that $\boldsymbol{\alpha x} + \boldsymbol{\gamma y} \le \beta$ is valid for $\Pi_1 \cup \Pi_2$. The disjunction $\boldsymbol{\pi x} \le \pi_0 \vee \boldsymbol{\pi x} \ge \pi_0 + 1$ is called a *split disjunction*. The GMI inequality can be viewed as a split inequality for (3) with the split in which $\pi_j = \lfloor a_j \rfloor$ if $f_j \le f_0$, $\pi_j = \lceil a_j \rceil$ if $f_j > f_0$ and $\pi_0 = \lfloor b \rfloor$.

The *elementary closure*, or simply the *closure*, associated with a family F of inequalities valid for $T \cap (\mathbb{Z}^n \times \mathbb{R}^p)$ is the convex set obtained as the intersection of all inequalities in F. Not surprisingly, the GMI and split closures are equivalent [28]. The separation problem for the split closure is NP-hard [9].

Later we will encounter single row sets X both in the all integer case and in the mixed integer case in which we add GMI cuts containing slack variables that are not integer variables.

Observation 1 *If X is replaced by*

$$\bar{X} = \{(\boldsymbol{x}, \boldsymbol{y}) \in \mathbb{Z}^n_+ \times \mathbb{R}^p_+ \mid \sum_{j=1}^{n} \bar{a}_j x_j + \sum_{j=1}^{p} g_j y_j = \bar{b}\}\,,$$

where $\bar{a}_j \equiv a_j \mod 1$ for $1 \le j \le n$ and $\bar{b} \equiv b \mod 1$, the GMI (4) for X and the GMI for $\bar{X}$ are the same inequality.

2.2 Lattices and lattice reformulation

Given $l \leq n$ linearly independent vectors $\boldsymbol{b}_1, \dots \boldsymbol{b}_l \in \mathbb{R}^n$, the set $L(\boldsymbol{b}_1, \dots, \boldsymbol{b}_l) := \{\sum_{i=1}^{l} c_i \boldsymbol{b}_i,\ c_i \in \mathbb{Z}\}$ is called the *lattice* generated by $\boldsymbol{b}_1, \dots \boldsymbol{b}_l$. The vectors $\boldsymbol{b}_1, \dots \boldsymbol{b}_l$ are called a *lattice basis*, and we often represent them as a matrix $\boldsymbol{B} = (\boldsymbol{b}_1, \dots \boldsymbol{b}_l)$. Given a lattice L generated by $\boldsymbol{B}$, the basis $\boldsymbol{B}'$ is an alternative basis for L if and only if we can write $\boldsymbol{B}' = \boldsymbol{B}\boldsymbol{U}$, where $\boldsymbol{U}$ is an $l \times l$ unimodular matrix.

We now can explain the reformulation of the integer program

$$\max\{\boldsymbol{c}\boldsymbol{x} \mid \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b},\ \boldsymbol{x} \in \mathbb{Z}_+^n\}, \tag{1}$$

presented in the Section 1, due to Aardal et al. [1]. The set $\ker_{\mathbb{Z}}(\boldsymbol{A}) = \{\boldsymbol{x} \in \mathbb{Z}^n \mid \boldsymbol{A}\boldsymbol{x} = \boldsymbol{0}\}$ is a lattice, called the *kernel lattice* of $\boldsymbol{A}$.

Suppose $\boldsymbol{x}$ is a feasible solution in problem (1). If $\boldsymbol{x}^0 \in \mathbb{Z}^n$ satisfies $\boldsymbol{A}\boldsymbol{x}^0 = \boldsymbol{b}$, it follows that $\boldsymbol{A}(\boldsymbol{x} - \boldsymbol{x}^0) = \boldsymbol{0}$ and thus, if $\boldsymbol{Q}$ is a lattice basis for $\ker_{\mathbb{Z}}(\boldsymbol{A})$, this is equivalent to $(\boldsymbol{x} - \boldsymbol{x}^0) = \boldsymbol{Q}\boldsymbol{\mu}$ where $\boldsymbol{\mu} \in \mathbb{Z}^{n-m}$. Now substituting $\boldsymbol{x} = \boldsymbol{x}^0 + \boldsymbol{Q}\boldsymbol{\mu}$ and using $\boldsymbol{x} \geq \boldsymbol{0}$ gives the reformulation

$$\max\{\boldsymbol{c}(\boldsymbol{x}^0 + \boldsymbol{Q}\boldsymbol{\mu}) \mid \boldsymbol{Q}\boldsymbol{\mu} \geq -\boldsymbol{x}^0, \boldsymbol{\mu} \in \mathbb{Z}^{n-m}\}. \tag{2}$$

Let L be a lattice in a Euclidean vector space E. A subset $K \subseteq L$ is called a *pure sublattice* of L if there exists a linear subspace D of E such that $K = D \cap L$.

A matrix $\boldsymbol{A} \in \mathbb{Z}^{m \times n}$ of full row rank is in *Hermite Normal Form* if it has the form $\mathrm{HNF}(\boldsymbol{A}) = (\boldsymbol{H}, \boldsymbol{0}^{m \times (n-m)}) = \boldsymbol{A}\boldsymbol{U}$, where $\boldsymbol{H}$ is a lower triangular nonnegative $m \times m$ matrix in which the unique row maxima can be found along the diagonal, and $\boldsymbol{U}$ is an $n \times n$ unimodular matrix.

Observation 2 *A lattice L generated by the basis $\boldsymbol{B} = (\boldsymbol{b}_1, \dots, \boldsymbol{b}_l)$ is a pure sublattice of the standard lattice $\mathbb{Z}^n$ if and only if* $\mathrm{HNF}(\boldsymbol{B}^\top) = (\boldsymbol{I}, \boldsymbol{0})$.

Observation 3 *The lattice* $\ker_{\mathbb{Z}}(\boldsymbol{A})$ *is a pure sublattice of* $\mathbb{Z}^n$.

Theorem 1 (See Schrijver [29], Theorem 5.2.) *The Hermite Normal Form $(\boldsymbol{H}, \boldsymbol{0})$ of a rational matrix $\boldsymbol{A}$ of full row rank has size polynomially bounded by the size of $\boldsymbol{A}$. Moreover, there exists a unimodular matrix $\boldsymbol{U}$ with $\boldsymbol{A}\boldsymbol{U} = (\boldsymbol{H}, \boldsymbol{0})$, such that the size of $\boldsymbol{U}$ is polynomially bounded by the size of $\boldsymbol{A}$.*

Proposition 1 (See Schrijver [29], Corollary 5.3a.) *Given a rational matrix $\boldsymbol{A}$ of full row rank, a unimodular matrix $\boldsymbol{U}$ such that $\boldsymbol{A}\boldsymbol{U}$ is in Hermite Normal Form can be found in polynomial time.*

3 Relations between solutions and polyhedra in $\boldsymbol{x}$- and $\boldsymbol{\mu}$-space

Here we establish answers to the theoretical questions raised in the introduction.

3.1 Expressing $\boldsymbol{\mu} \in \hat{S}$ as a function of $\boldsymbol{x} \in S$

The lattice reformulation gives a way of expressing each feasible vector $\boldsymbol{x} \in S$ as a function of $\boldsymbol{\mu}$. A natural question is how to express a feasible vector $\boldsymbol{\mu} \in \hat{S}$ as a function of $\boldsymbol{x}$. In particular, this is our prime tool for generating general disjunctions for deriving split inequalities, as described in more detail in Section 4.

A consequence of $\ker_{\mathbb{Z}}(\boldsymbol{A})$ being a pure sublattice of $\mathbb{Z}^n$, and of Theorem 1 and Proposition 1, is that we can find, in polynomial time, a unimodular matrix $\boldsymbol{U}$ such that

$$\boldsymbol{U}^{\mathsf{T}}\boldsymbol{Q} = \begin{pmatrix} \boldsymbol{I} \\ \boldsymbol{0} \end{pmatrix}. \tag{5}$$

Let $\boldsymbol{W}$ be the matrix consisting of the first $n-m$ rows of $\boldsymbol{U}^{\mathsf{T}}$ as in (5). Since $\boldsymbol{W}$ is a submatrix of $\boldsymbol{U}^{\mathsf{T}}$ it follows that all elements of $\boldsymbol{W}$ are integral. It is also clear that

$$\boldsymbol{W}\boldsymbol{Q} = \boldsymbol{I}\,. \tag{6}$$

This was also observed by Mehrotra and Li [26]. Note that $\boldsymbol{W}$ in general is not unique: given a matrix $\boldsymbol{W}$, we can form a matrix $\boldsymbol{W}' = \boldsymbol{W} + \boldsymbol{C}$, where $\boldsymbol{C}$ is an integer $(n-m) \times n$ matrix consisting of rows obtained by taking an integer linear combination of rows of $\boldsymbol{A}$. The matrix $\boldsymbol{W}$ permits us to translate an expression in $\boldsymbol{\mu}$-variables back to an expression in $\boldsymbol{x}$-variables. Specifically we have $\boldsymbol{W}\boldsymbol{x} = \boldsymbol{W}\boldsymbol{x}^0 + \boldsymbol{W}\boldsymbol{Q}\boldsymbol{\mu}$ and thus

$$\boldsymbol{\mu} = \boldsymbol{W}\boldsymbol{x} - \boldsymbol{W}\boldsymbol{x}^0\,.$$

3.2 Relations between polyhedra in the $\boldsymbol{x}$- and $\boldsymbol{\mu}$-spaces

Here we show that not only vectors in S and $\hat{S}$ correspond one-to-one, but that there is also a one-to-one correspondence between vectors in P and $\hat{P}$.

Proposition 2 *Given $\boldsymbol{A} \in \mathbb{Z}^{m\times n}$ and $\boldsymbol{b} \in \mathbb{Z}^m$, define $P = \{\boldsymbol{x} \in \mathbb{R}^n_+ \mid \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}\}$. We can write $\boldsymbol{x} \in P \cap \mathbb{Z}^n$ as $\boldsymbol{x} = \boldsymbol{x}^0 + \boldsymbol{Q}\boldsymbol{\mu}$, where $\boldsymbol{x}^0$, $\boldsymbol{Q}$, and $\boldsymbol{\mu}$ are defined as in Section 2. Define $\hat{P} = \{\boldsymbol{\mu} \in \mathbb{R}^{n-m} \mid \boldsymbol{Q}\boldsymbol{\mu} \geq -\boldsymbol{x}^0\}$ for $\boldsymbol{Q}$ and $\boldsymbol{x}^0$ as given above. The map $f(\boldsymbol{\mu}) = \boldsymbol{Q}\boldsymbol{\mu} + \boldsymbol{x}^0$ is a bijective map from $\hat{P}$ to P.*

Proof Take $\bar{\boldsymbol{\mu}} \in \hat{P}$ and let $\bar{\boldsymbol{x}} = \boldsymbol{Q}\bar{\boldsymbol{\mu}} + \boldsymbol{x}^0$. The vector $\bar{\boldsymbol{x}}$ is nonnegative since $\boldsymbol{Q}\bar{\boldsymbol{\mu}} \geq -\boldsymbol{x}^0$. Moreover, $\boldsymbol{A}\bar{\boldsymbol{x}} = \boldsymbol{A}\boldsymbol{Q}\bar{\boldsymbol{\mu}} + \boldsymbol{A}\boldsymbol{x}^0 = \boldsymbol{A}\boldsymbol{x}^0 = \boldsymbol{b}$, where the second equality holds since $\boldsymbol{Q}$ is a basis for $\ker_{\mathbb{Z}}(\boldsymbol{A})$.

Take $\bar{\boldsymbol{x}} \in P$. Since $\boldsymbol{Q}$ spans the Euclidean vector space $\{\boldsymbol{x} \in \mathbb{R}^n \mid \boldsymbol{A}\boldsymbol{x} = \boldsymbol{0}\}$, we can write $\bar{\boldsymbol{x}}$ as $\bar{\boldsymbol{x}} = \boldsymbol{Q}\bar{\boldsymbol{\mu}} + \boldsymbol{x}^0$ for some $\bar{\boldsymbol{\mu}} \in \mathbb{R}^{n-m}$. $\bar{\boldsymbol{x}} \in P$ implies $\bar{\boldsymbol{x}} \geq \boldsymbol{0}$, and hence $\boldsymbol{Q}\bar{\boldsymbol{\mu}} + \boldsymbol{x}^0 \geq \boldsymbol{0}$, so $\bar{\boldsymbol{\mu}} \in \hat{P}$. □

Given an LP-basis, we examine the corresponding partitions of $\boldsymbol{A}, \boldsymbol{Q}$ and $\boldsymbol{W}$.

Proposition 3 *Given $\boldsymbol{A}, \boldsymbol{Q}, \boldsymbol{W}$ as described above and an LP-basis $\boldsymbol{B}$ in $\boldsymbol{x}$-space, write $\boldsymbol{A} = (\boldsymbol{B}, \boldsymbol{N})$, $\boldsymbol{Q}^\mathsf{T} = (\boldsymbol{Q}_B, \boldsymbol{Q}_N)$, $\boldsymbol{W} = (\boldsymbol{W}_B, \boldsymbol{W}_N)$ and $\boldsymbol{B}\boldsymbol{x}_B^0 + \boldsymbol{N}\boldsymbol{x}_N^0 = \boldsymbol{b}$.*
For $\boldsymbol{Q}_B$ and $\boldsymbol{Q}_N$ the following holds:

i) $\boldsymbol{Q}_B = -\boldsymbol{B}^{-1}\boldsymbol{N}\boldsymbol{Q}_N$,
ii) $\boldsymbol{Q}_N^{-1} = \boldsymbol{W}_N - \boldsymbol{W}_B\boldsymbol{B}^{-1}\boldsymbol{N}$.

Proof i) As $\boldsymbol{A}\boldsymbol{Q} = \boldsymbol{0}$, $\boldsymbol{B}\boldsymbol{Q}_B + \boldsymbol{N}\boldsymbol{Q}_N = \boldsymbol{0}$ and, as $\boldsymbol{B}^{-1}$ exists, $\boldsymbol{Q}_B = -\boldsymbol{B}^{-1}\boldsymbol{N}\boldsymbol{Q}_N$.
ii) As $\boldsymbol{W}\boldsymbol{Q} = \boldsymbol{I}$, $\boldsymbol{W}_B\boldsymbol{Q}_B + \boldsymbol{W}_N\boldsymbol{Q}_N = \boldsymbol{I}$, and using i), one has $(-\boldsymbol{W}_B\boldsymbol{B}^{-1}\boldsymbol{N} + \boldsymbol{W}_N)\boldsymbol{Q}_N = \boldsymbol{I}$. It follows as $\boldsymbol{Q}_N$ and $\boldsymbol{W}_N - \boldsymbol{W}_B\boldsymbol{B}^{-1}\boldsymbol{N}$ are both $(n-m)\times(n-m)$ matrices that $\boldsymbol{Q}_N$ is non-singular and thus $\boldsymbol{Q}_N^{-1} = \boldsymbol{W}_N - \boldsymbol{W}_B\boldsymbol{B}^{-1}\boldsymbol{N}$. □

Now we consider the representation of the basis in the $\boldsymbol{x}$- and $\boldsymbol{\mu}$-spaces. A basic solution in the $\boldsymbol{x}$-space is written as

$$\boldsymbol{x}_B + \boldsymbol{B}^{-1}\boldsymbol{N}\boldsymbol{x}_N = \boldsymbol{B}^{-1}\boldsymbol{b}, \quad \boldsymbol{x}_B, \boldsymbol{x}_N \geq \boldsymbol{0}.$$

Writing the reformulation in equality form we observe that the $\boldsymbol{x}$-variables are precisely the slack variables, i.e.,

$$\boldsymbol{x} - \boldsymbol{Q}\boldsymbol{\mu} = \boldsymbol{x}^0,$$

which we can write as

$$\begin{pmatrix} \boldsymbol{x}_B \\ \boldsymbol{x}_N \end{pmatrix} - \begin{pmatrix} \boldsymbol{Q}_B \\ \boldsymbol{Q}_N \end{pmatrix} \boldsymbol{\mu} = \begin{pmatrix} \boldsymbol{x}_B^0 \\ \boldsymbol{x}_N^0 \end{pmatrix}. \tag{7}$$

Since the $\boldsymbol{\mu}$-variables are free, they must be basic and thus the basic variables are $(\boldsymbol{x}_B, \boldsymbol{\mu})$. Multiplying the last $n-m$ rows of (7) by $-\boldsymbol{Q}_N^{-1}$ yields

$$-\boldsymbol{Q}_N^{-1}\boldsymbol{x}_N + \boldsymbol{I}\boldsymbol{\mu} = -\boldsymbol{Q}_N^{-1}\boldsymbol{x}_N^0, \text{ or equivalently } \boldsymbol{\mu} = \boldsymbol{Q}_N^{-1}\boldsymbol{x}_N - \boldsymbol{Q}_N^{-1}\boldsymbol{x}_N^0.$$

Substituting for $\boldsymbol{\mu}$ in the first m rows of (7) gives

$$\boldsymbol{x}_B - \boldsymbol{Q}_B\boldsymbol{Q}_N^{-1}\boldsymbol{x}_N = \boldsymbol{x}_B^0 - \boldsymbol{Q}_B\boldsymbol{Q}_N^{-1}\boldsymbol{x}_N^0,$$

and we obtain an expression for a basic solution:

$$\begin{pmatrix} \boldsymbol{x}_B \\ \boldsymbol{\mu} \end{pmatrix} - \begin{pmatrix} \boldsymbol{Q}_B\boldsymbol{Q}_N^{-1} \\ \boldsymbol{Q}_N^{-1} \end{pmatrix} \boldsymbol{x}_N = \begin{pmatrix} \boldsymbol{x}_B^0 - \boldsymbol{Q}_B\boldsymbol{Q}_N^{-1}\boldsymbol{x}_N^0 \\ -\boldsymbol{Q}_N^{-1}\boldsymbol{x}_N^0 \end{pmatrix}. \tag{8}$$

Now using Proposition 3, the basic solution (8) can be rewritten as

$$\begin{aligned} \begin{pmatrix} \boldsymbol{x}_B \\ \boldsymbol{\mu} \end{pmatrix} + \begin{pmatrix} \boldsymbol{B}^{-1}\boldsymbol{N} \\ -(\boldsymbol{W}_N - \boldsymbol{W}_B\boldsymbol{B}^{-1}\boldsymbol{N}) \end{pmatrix} \boldsymbol{x}_N &= \begin{pmatrix} \boldsymbol{x}_B^0 + \boldsymbol{B}^{-1}\boldsymbol{N}\boldsymbol{x}_N^0 \\ -(\boldsymbol{W}_N - \boldsymbol{W}_B\boldsymbol{B}^{-1}\boldsymbol{N})\boldsymbol{x}_N^0 \end{pmatrix} \\ &= \begin{pmatrix} \boldsymbol{B}^{-1}\boldsymbol{b} \\ -(\boldsymbol{W}_N - \boldsymbol{W}_B\boldsymbol{B}^{-1}\boldsymbol{N})\boldsymbol{x}_N^0 \end{pmatrix}. \end{aligned} \tag{9}$$

From (8) we see that, given an LP-basis, the $\boldsymbol{\mu}$-variables can be expressed solely as a function of $\boldsymbol{Q}$.

We now illustrate the different basis representations in an Example.

Example 1 Consider an instance with $m = 2,\ n = 5$,

$$(\boldsymbol{A} \mid \boldsymbol{b}) = \left(\begin{array}{ccccc|c} 0 & 5 & 3 & 1 & 7 & 9 \\ 6 & 3 & 0 & 11 & 2 & 14 \end{array}\right).$$

To obtain a reformulation, one can take

$$\boldsymbol{Q} = \begin{pmatrix} 1 & -3 & -3 \\ 3 & 3 & 0 \\ 0 & -3 & 4 \\ -1 & 1 & 2 \\ -2 & -1 & -2 \end{pmatrix}, \quad \boldsymbol{x}^0 = \begin{pmatrix} 0 \\ 1 \\ 1 \\ 1 \\ 0 \end{pmatrix}.$$

A matrix $\boldsymbol{W}$ corresponding to $\boldsymbol{Q}$ is:

$$\boldsymbol{W} = \begin{pmatrix} -2 & -1 & 0 & -4 & -1 \\ -2 & 1 & 1 & -3 & 2 \\ -3 & 0 & 1 & -5 & 1 \end{pmatrix}.$$

For the feasible basis $\boldsymbol{x}_B = (x_1, x_2)$, the corresponding $\boldsymbol{x}$-tableau is:

$$\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \frac{1}{30}\begin{pmatrix} -9 & 52 & -11 \\ 18 & 6 & 42 \end{pmatrix}\begin{pmatrix} x_3 \\ x_4 \\ x_5 \end{pmatrix} = \frac{1}{30}\begin{pmatrix} 43 \\ 54 \end{pmatrix}.$$

Now setting $\boldsymbol{\mu} = \boldsymbol{W}\boldsymbol{x} - \boldsymbol{W}\boldsymbol{x}^0$ and eliminating the basic variables $\boldsymbol{x}_B$ by substitution, the corresponding $\boldsymbol{\mu}$-tableau is:

$$\begin{pmatrix} \mu_1 \\ \mu_2 \\ \mu_3 \end{pmatrix} - \frac{1}{30}\begin{pmatrix} 0 & -10 & -10 \\ -6 & 8 & -4 \\ 3 & 6 & -3 \end{pmatrix}\begin{pmatrix} x_3 \\ x_4 \\ x_5 \end{pmatrix} = \frac{1}{30}\begin{pmatrix} 10 \\ -2 \\ -9 \end{pmatrix}.$$

From the μ_3 row, one has $f_3 = \frac{27}{30}, f_4 = \frac{24}{30}, f_5 = \frac{3}{30}$ and $f_0 = \frac{21}{30}$ giving the ℓ-cut:

$$\frac{1}{3}x_3 + \frac{2}{3}x_4 + \frac{1}{7}x_5 \geq 1.$$

□

We now turn our attention to the group problem associated with the two formulations, and the related corner polyhedra [21]. Let $\boldsymbol{A} = (\boldsymbol{B},\ \boldsymbol{N})$, where $\boldsymbol{B}$ corresponds to the basic variables in an optimal solution to the LP-relaxation of (1). The following integer optimization problem is a relaxation of (1) obtained by dropping the nonnegativity constraints on the basic variables $\boldsymbol{x}_B$.

$$\max\{\boldsymbol{c}\boldsymbol{x} \mid (\boldsymbol{B}\ \boldsymbol{N})\begin{pmatrix} \boldsymbol{x}_B \\ \boldsymbol{x}_N \end{pmatrix} = \boldsymbol{b},\ \boldsymbol{x}_N \geq \boldsymbol{0},\ \boldsymbol{x}_B, \boldsymbol{x}_N \text{ integral}\}. \tag{10}$$

Using the relation $\boldsymbol{B}\boldsymbol{x}_B + \boldsymbol{N}\boldsymbol{x}_N = \boldsymbol{b}$ and the integrality of $\boldsymbol{x}_B$ gives the equivalent formulation of (10) as:

$$\max\{\boldsymbol{c}_B\boldsymbol{B}^{-1}\boldsymbol{b} + (\boldsymbol{c}_N - \boldsymbol{c}_B\boldsymbol{B}^{-1}\boldsymbol{N})\boldsymbol{x}_N \mid \boldsymbol{B}^{-1}\boldsymbol{N}\boldsymbol{x}_N \equiv \boldsymbol{B}^{-1}\boldsymbol{b} \mod 1,\ \boldsymbol{x}_N \in \mathbb{Z}_+^{n-m}\}. \tag{11}$$

Problem (11) is referred to as the *group problem* [21].

We will now prove that the feasible sets of the group problem is the same whether we view it in the original $\boldsymbol{x}$-space or in the reformulated space.

Theorem 2 *The group problems arising in the $\boldsymbol{x}$- and $\boldsymbol{\mu}$-spaces are the same.*

Proof We consider the feasible regions corresponding to the underlying groups. Let

$$G = \{\boldsymbol{x}_N \in \mathbb{Z}_+^{n-m} \mid \boldsymbol{B}^{-1}\boldsymbol{N}\boldsymbol{x}_N \equiv \boldsymbol{B}^{-1}\boldsymbol{b} \mod 1\}$$

and

$$\hat{G} = \{\boldsymbol{x}_N \in \mathbb{Z}_+^{n-m} \mid -(\boldsymbol{W}_N - \boldsymbol{W}_B\boldsymbol{B}^{-1}\boldsymbol{N})\boldsymbol{x}_N \equiv -(\boldsymbol{W}_N - \boldsymbol{W}_B\boldsymbol{B}^{-1}\boldsymbol{N})\boldsymbol{x}_N^0 \mod 1\}.$$

As $\boldsymbol{W}_N\boldsymbol{x}_N, \boldsymbol{W}_N\boldsymbol{x}_N^0$ are integer,

$$\hat{G} = \{\boldsymbol{x}_N \in \mathbb{Z}_+^{n-m} \mid \boldsymbol{W}_B\boldsymbol{B}^{-1}\boldsymbol{N}\boldsymbol{x}_N \equiv \boldsymbol{W}_B\boldsymbol{B}^{-1}\boldsymbol{N}\boldsymbol{x}_N^0 \mod 1\}. \quad (12)$$

Now as $\boldsymbol{W}_B$ is an integral matrix, it follows that $G \subseteq \hat{G}$.

Conversely, take $\hat{G}$ in the form (see (8)):

$$\hat{G} = \{\boldsymbol{x}_N \in \mathbb{Z}_+^{n-m} \mid \boldsymbol{Q}_N^{-1}\boldsymbol{x}_N \equiv \boldsymbol{Q}_N^{-1}\boldsymbol{x}_N^0 \mod 1\},$$

Suppose $\boldsymbol{x}_N \in \hat{G}$. As $\boldsymbol{Q}_B$ is an integer matrix, $\boldsymbol{x}_N$ lies in

$$\{\boldsymbol{x}_N \in \mathbb{Z}_+^{n-m} \mid \boldsymbol{Q}_B\boldsymbol{Q}_N^{-1}\boldsymbol{x}_N \equiv \boldsymbol{Q}_B\boldsymbol{Q}_N^{-1}\boldsymbol{x}_N^0 \mod 1\},$$

which, as $\boldsymbol{Q}_B\boldsymbol{Q}_N^{-1} = -\boldsymbol{B}^{-1}\boldsymbol{N}$, is precisely G. □

As the order of the groups is given by the determinant, it follows that $|\det(\boldsymbol{B})| = |\det(\boldsymbol{Q}_N)|$ and as the corner polyhedron is the convex hull of the solutions to the group problem, it follows immediately that the corner polyhedra are the same.

Based on Observation 1, we see that the ℓ-cuts generated from the second set of equations of (8), the second set of equations of (9) or from (12) are the same.

Observation 4 *Taking $\boldsymbol{\mu} = \boldsymbol{W}(\boldsymbol{x} - \boldsymbol{x}^0)$ or $\boldsymbol{\mu}' = \boldsymbol{W}_B(\boldsymbol{x}_B - \boldsymbol{x}_B^0)$ leads to the same ℓ-cuts because $\boldsymbol{W}_N\boldsymbol{x}_N = 0 \mod 1$ and $\boldsymbol{W}_N\boldsymbol{x}_N^0 = 0 \mod 1$. Therefore a simple way to obtain the ℓ-cuts is to left multiply the $\boldsymbol{x}$-tableau by $\boldsymbol{W}_B$. It follows that $\boldsymbol{W}_B$ is an m-dimensional generalization of the k in k-cuts. In particular, if $m = 1$, taking $k = \boldsymbol{W}_B$ gives us a specific choice for k. On the other hand, if $m > 1$, the ℓ-cuts can be viewed as multi-row tableau cuts, see e.g. [15].*

Now we consider closures. Let P_S (P_{CG}) be the split (Chvátal-Gomory) closure with respect to P. Analogous notation is used for $\hat{P}$. We show that the split closures associated with P and $\hat{P}$ are equivalent.

Theorem 3 $P_{\text{S}} = \{\boldsymbol{x} \in \mathbb{R}^n \mid \boldsymbol{x} = \boldsymbol{x}^0 + \boldsymbol{Q}\boldsymbol{\mu},\ \boldsymbol{\mu} \in \hat{P}_{\text{S}}\}$.

Proof We use the definition of split cuts [12]. Let $(\boldsymbol{\pi}, \pi_0) \in \mathbb{Z}^{n+1}$ and let

$$\begin{aligned} \boldsymbol{\alpha x} - q(\boldsymbol{\pi x} - \pi_0) &\leq \alpha_0 \\ \boldsymbol{\alpha x} + r(\boldsymbol{\pi x} - \pi_0 - 1) &\leq \alpha_0 \end{aligned}$$

be valid inequalities for P with $q, r \geq 0$. Then, $\boldsymbol{\alpha x} \leq \alpha_0$ is valid for $(P \cap \{\boldsymbol{\pi x} \leq \pi_0\}) \cup (P \cap \{\boldsymbol{\pi x} \geq \pi_0 + 1\})$. The inequality $\boldsymbol{\alpha x} \leq \alpha_0$ is called a split cut.

Substitute $\boldsymbol{x}$ for $\boldsymbol{Q\mu} + \boldsymbol{x}^0$. Let

$$\begin{aligned} \hat{\boldsymbol{\pi}} &= \boldsymbol{\pi Q}\,, \\ \hat{\pi}_0 &= \pi_0 - \boldsymbol{\pi x}^0\,, \\ \hat{\boldsymbol{\alpha}} &= \boldsymbol{\alpha Q}\,, \\ \hat{\alpha_0} &= \alpha_0 - \boldsymbol{\alpha x}^0\,. \end{aligned}$$

Notice that $(\hat{\boldsymbol{\pi}}, \hat{\pi}_0) \in \mathbb{Z}^{n-m+1}$ as $\boldsymbol{Q}$ and $\boldsymbol{x}^0$ are integer. We obtain

$$\hat{\boldsymbol{\alpha}}\boldsymbol{\mu} - q(\hat{\boldsymbol{\pi}}\boldsymbol{\mu} - \hat{\pi}_0) \leq \hat{\alpha_0} \tag{13}$$

$$\hat{\boldsymbol{\alpha}}\boldsymbol{\mu} + r(\hat{\boldsymbol{\pi}}\boldsymbol{\mu} - \hat{\pi}_0 - 1) \leq \hat{\alpha_0}\,. \tag{14}$$

If inequalities (13) and (14) are valid for $\hat{P}$, then $\hat{\boldsymbol{\alpha}}\boldsymbol{\mu} \leq \hat{\alpha_0}$ is valid for $(\hat{P} \cap \{\hat{\boldsymbol{\pi}}\boldsymbol{\mu} \leq \hat{\pi}_0\}) \cup (\hat{P} \cap \{\hat{\boldsymbol{\pi}}\boldsymbol{\mu} \geq \hat{\pi}_0 + 1\})$.

Going from a split cut for $\hat{P}$ to a split cut for P is similar by using $\boldsymbol{\mu} = \boldsymbol{W}(\boldsymbol{x} - \boldsymbol{x}^0)$ and using that $\boldsymbol{W}$ and $\boldsymbol{x}^0$ are integer. □

Similarly one can show the following result:

Proposition 4 $P_{\text{CG}} = \{\boldsymbol{x} \in \mathbb{R}^n \mid \boldsymbol{x} = \boldsymbol{x}^0 + \boldsymbol{Q\mu},\ \boldsymbol{\mu} \in \hat{P}_{\text{CG}}\}$.

4 Separating Cuts from Lattice Reformulations

In Subsection 4.1 we give a high-level description of our approach. In Subsection 4.2 we describe three different reduction methods to derive the basis $\boldsymbol{Q}$ in the reformulation (2). In our computations we test how the quality of the reduction influences the effectiveness of the cuts generated. We also describe how to derive the matrix $\boldsymbol{W}$ in (6).

4.1 High-level description of our approach

As discussed in Section 3, ℓ-cuts are tableau GMI cuts derived from an optimal tableau in the space of the $\boldsymbol{\mu}$-variables. However, they can be generated by working directly in the space of the $\boldsymbol{x}$-variables. The approach for separating ℓ-cuts in the space of the $\boldsymbol{x}$-variables is as follows.

Initialization: Generate a reduced basis $\boldsymbol{Q}$ for $\ker_{\mathbb{Z}}(\boldsymbol{A})$ as in (2), and a corresponding matrix $\boldsymbol{W}$ (6) as shown in Section 4.2.

Iteration t**:** After the addition of t rounds of ℓ-cuts,

1. Solve the resulting linear program and take the rows corresponding to the $\boldsymbol{x}$-variables in the basis. The resulting set of equations is of the form:

$$\begin{aligned} \boldsymbol{x}_B + \boldsymbol{N}\boldsymbol{x}_N + \boldsymbol{S}_N \boldsymbol{s}_N = \bar{\boldsymbol{x}}_B \\ \boldsymbol{x}_B \in \mathbb{Z}_+^{|B|}, \boldsymbol{x}_N \in \mathbb{Z}_+^{|N|}, \boldsymbol{s}_N \geq \boldsymbol{0} \end{aligned} \tag{15}$$

   where $\boldsymbol{x}_N$ are the non-basic $\boldsymbol{x}$-variables, $\boldsymbol{s}_N$ are non-basic slack variables from previously added cuts and $\boldsymbol{N}$ and $\boldsymbol{S}$ are the associated matrices in this part of the optimal tableau.
2. For every row $\boldsymbol{w}_i$ of $\boldsymbol{W}_B$ such that $\boldsymbol{w}_i \bar{\boldsymbol{x}}_B \notin \mathbb{Z}$, left multiply the equations (15) by $\boldsymbol{w}_i$ and generate the GMI cut from the resulting "aggregated" row. (See (12).)
3. Add a selection of the separated cuts to the current LP.

4.2 How to generate the matrices $\boldsymbol{Q}$ and $\boldsymbol{W}$

The reformulation (2) is valid for any basis $\boldsymbol{Q}$ of the lattice $\ker_{\mathbb{Z}}(\boldsymbol{A})$. We will, however, be interested in a basis that is *reduced*. To test how the quality of the reduction plays a role in computations, we consider three different reductions.

4.2.1 LLL reductions

Given linearly independent vectors $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_l \in \mathbb{R}^n$, the corresponding Gram-Schmidt orthogonalized vectors are

$$\begin{aligned} \boldsymbol{b}_1^* &= \boldsymbol{b}_1 \,, \\ \boldsymbol{b}_j^* &= \boldsymbol{b}_j - \sum_{k=1}^{j-1} \mu_{jk} \boldsymbol{b}_k^*, \quad 2 \leq j \leq l \,, \quad \text{where} \\ \mu_{jk} &= \frac{\boldsymbol{b}_j^\top \boldsymbol{b}_k^*}{||\boldsymbol{b}_k^*||^2}, \quad 1 \leq k < j \leq l \,. \end{aligned}$$

Definition 1 (Lenstra, Lenstra, Lovász [25]) A basis $\boldsymbol{b}_1, \boldsymbol{b}_2, \ldots, \boldsymbol{b}_l$ is called *LLL-reduced* if

$$|\mu_{jk}| \leq \frac{1}{2} \quad \text{for } 1 \leq k < j \leq l \,, \tag{16}$$

$$||\boldsymbol{b}_j^* + \mu_{j,j-1} \boldsymbol{b}_{j-1}^*||^2 \geq y \cdot ||\boldsymbol{b}_{j-1}^*||^2 \quad \text{for } 1 < j \leq l \,, \tag{17}$$

for $1/4 < y < 1$.

Many quality guarantees can be given for a reduced basis. Well-known guarantees are that the first reduced basis vector is an approximation of the shortest non-zero vector in the lattice, and that all reduced basis vectors are approximations of the successive minima of the lattice. We refer to [25] for details. A reduced basis can be computed in polynomial time, and the larger the parameter y in (17), the better the quality guarantees become.

4.2.2 Korkine-Zolotarev reduction

A basis $\boldsymbol{b}_1, \ldots \boldsymbol{b}_l$ of the lattice L is reduced in the sense of Korkine and Zolotarev (KZ-reduced) [23] if it satisfies the following conditions.

1. $\boldsymbol{b}_1$ is a shortest non-zero vector of L in the Euclidean norm,
2. $|\mu_{i1}| \leq \frac{1}{2}$ for $2 \leq i \leq l$,
3. if $L^{(l-1)}$ denotes the orthogonal projection of L on the orthogonal complement $(\mathbb{R}\boldsymbol{b}_1)^{\perp}$ of $\mathbb{R}\boldsymbol{b}_1$, then the projections $\boldsymbol{b}_i - \mu_{i1}\boldsymbol{b}_1$ of $\boldsymbol{b}_2, \ldots, \boldsymbol{b}_l$ yield a KZ-reduced basis $\boldsymbol{b}_2 - \mu_{21}\boldsymbol{b}_1, \ldots, \boldsymbol{b}_l - \mu_{l1}\boldsymbol{b}_1$ of $L^{(l-1)}$.

So, the first basis vector in a KZ-reduced basis is a shortest non-zero lattice vector. Several other bounds on the quality of such a basis, along with a non-recursive definition of a KZ-reduced basis, can be found in [24]. Since a shortest lattice vector is computed, determining a KZ-reduced basis is computationally much more costly than determining an LLL-reduced basis.

In our computational study we test the following reduction methods.

LLL-low: LLL reduction with $y = 26/100$, to test a low-quality reduction.

LLL: LLL reduction with $y = 99/100$, to test a high-quality basis that is reasonably fast to compute.

KZ: Korkine-Zolotarev reduction, to test in some sense an "optimally" reduced basis.

4.2.3 Computing the matrix $\boldsymbol{W}$

As mentioned before, the matrix $\boldsymbol{W}$ is not unique. Let $\boldsymbol{e}_i$ be the ith column of the $(n-m)$-dimensional identity matrix. The matrix $\boldsymbol{W}$ can be calculated by computing the Hermite Normal Form as stated in Proposition 1. However, any method for finding a feasible solution to the $n-m$ systems of integer equations

$$\boldsymbol{Q}^{\mathsf{T}}\boldsymbol{w}_i = \boldsymbol{e}_i,\ \boldsymbol{w}_i \in \mathbb{Z}^n\ i = 1, \ldots, n-m\,, \tag{18}$$

can be used. A valid matrix $\boldsymbol{W}$ is then obtained by taking the $n-m$ vectors $\boldsymbol{w}_i$ as its rows. In our computational study we again use the lattice reformulation technique described in [1] to derive the vectors $\boldsymbol{w}_i$, as this technique yields vectors $\boldsymbol{w}_i$ in which the absolute value of the elements is relatively small. For each of the $\boldsymbol{Q}$-matrices generated according to the three reductions given above, we generated an associated matrix $\boldsymbol{W}$ and the computations for (18) are all done using LLL reduction with $y = 99/100$.

5 Computational Experiments

The goal of the computational experiments reported in this section is threefold.

- In Section 5.1 we compare the strength of ℓ-cuts generated from different reduced bases leading to different $\boldsymbol{Q}/\boldsymbol{W}$ pairs as discussed in Section 4.2.

– In Section 5.2 we compare ℓ-cuts from a single $\boldsymbol{Q}/\boldsymbol{W}$ pair (the "best" discussed in Section 5.1) against standard GMIs and k-cuts [14]. More precisely, we considered those two families of cutting planes because the former is the standard reference for cutting plane generation, while the generation of the latter has some similarities with our approach, as previously discussed.
– In Section 5.3 we compare the strength of ℓ-cuts from a single $\boldsymbol{Q}/\boldsymbol{W}$ pair obtained by iteratively separating from the tableau, i.e., by increasing the rank, with approximate closure counterparts (ℓ-cuts, lift-and-project and split closures), i.e., by optimizing over the row aggregation.

The test instances are obtained as in Cornuéjols et al. [14] except that the matrix coefficients a_{ij}, requirements b_i and variable upper bounds h_j are required to be integer. Specifically, the objective function coefficients c_j are generated uniformly at random in [1,1000] and the coefficients a_{ij} are integer-generated uniformly at random in [1,1000]. For binary instances, denoted by "B", and for instances with unbounded integer variables, denoted by "U", we compute b_i as $b_i = \lfloor 0.5 \sum_{j=1}^{n} a_{ij} \rfloor$. For instances with bounded integers, denoted by "I", the h_j are generated uniformly in [5,10] and $b_i = \lfloor 0.5 \sum_{j=1}^{n} h_j a_{ij} \rfloor$.

5.1 Comparing the effect of basis reduction algorithms

In this section, we examine the effect of the basis reduction method used to generate lattice basis matrix $\boldsymbol{Q}$ on the quality of the resulting ℓ-cuts. In addition, as a reference, we compare with GMI cuts. More precisely, we consider the three reduction methods LLL-low, LLL, and KZ mentioned in Section 4.2.

Table 1 reports on the results of the comparisons between: GMIs from the optimal LP tableau, denoted by GMI, ℓ-cuts from the reduction method LLL-low, denoted by ℓ-LLL-low, ℓ-cuts from the reduction method LLL, denoted by ℓ-LLL, ℓ-cuts from the reduction method KZ, denoted by ℓ-KZ, and a combination of GMIs and ℓ-LLL, denoted by GMI + ℓ-LLL).

The other column headings are: R for the number of rounds of cuts, followed by n and m for the number of variables and constraints, respectively, and T for the type of the instance. Then, for each approach, we report on the number of cuts generated and the percentage of the gap that is closed between the optimal LP and IP values, on average over 20 instances.

The results in Table 1 clearly show that the gap closed by ℓ-cuts, independently of the basis reduction method, is significantly larger than that closed by only using GMIs, but the number of cuts is much larger. Moreover, by using a strongly reduced lattice basis (ℓ-LLL or ℓ-KZ), we obtain a significantly larger gap reduction than with a weaker reduction (ℓ-LLL-low). The gaps closed for the ℓ-LLL and ℓ-KZ reductions are not significantly different, typically varying by less than 1%. As the LLL reduction is much cheaper to compute, we will

Table 1 Comparing how the quality of cuts depend on the basis reduction method.

				GMI		ℓ-LLL-low		ℓ-LLL		ℓ-KZ		GMI + ℓ-LLL	
R	n	m	T	cuts	%gap	cuts	%gap	cuts	%gap	cuts	%gap	cuts	%gap
1	10	1	B	1.0	18.61	5.8	44.19	9.0	50.86	9.0	51.21	9.9	51.17
	20	1	B	1.0	9.75	6.6	23.08	14.7	34.83	15.2	36.63	15.5	34.88
	50	1	B	1.0	12.41	6.3	23.62	15.1	34.40	15.1	33.55	15.4	34.46
	100	1	B	1.0	10.66	6.5	25.54	15.3	31.00	12.9	31.16	15.4	31.00
	10	1	I	1.0	15.67	6.1	45.63	8.9	55.10	8.9	56.62	9.9	55.23
	20	1	I	1.0	13.36	7.2	30.17	14.4	39.88	14.7	41.46	15.4	40.45
	50	1	I	1.0	11.59	6.6	20.48	15.5	35.22	15.3	32.62	15.6	35.22
	100	1	I	1.0	13.83	7.2	27.72	15.9	34.71	13.8	37.92	16.1	34.71
	10	1	U	1.0	37.27	5.3	70.12	7.8	76.65	7.8	74.23	8.6	76.65
	20	1	U	1.0	34.88	6.1	65.65	10.4	77.52	11.1	77.48	10.9	77.96
	50	1	U	1.0	62.36	4.5	85.17	7.5	92.13	7.5	89.71	7.6	92.13
	100	1	U	1.0	64.07	3.4	85.89	5.1	98.59	4.9	98.71	5.3	98.59
	50	2	B	2.0	5.31	10.0	9.30	26.6	12.03	31.6	12.32	27.6	12.13
	50	3	B	3.0	2.55	11.9	3.62	32.5	5.10	41.8	5.68	34.7	5.19
	50	4	B	4.0	1.17	14.8	1.66	37.0	2.06	45.4	2.25	40.4	2.14
	50	2	U	2.0	13.02	10.0	22.23	26.6	32.14	31.5	32.58	27.7	32.50
	50	3	U	3.0	5.48	11.9	9.34	32.5	12.26	41.7	13.60	34.9	12.68
	50	4	U	4.0	3.48	14.9	4.91	37.2	6.95	45.4	7.42	40.4	6.99
5	10	1	B	9.0	35.29	28.2	68.88	36.6	79.98	36.3	81.02	51.8	81.23
	20	1	B	10.4	25.32	42.2	47.77	75.3	57.27	79.6	57.93	96.1	57.04
	50	1	B	10.5	28.08	44.6	43.38	91.7	51.77	92.4	50.78	100.4	52.06
	100	1	B	10.1	28.91	45.8	44.23	95.0	50.46	87.9	50.38	99.5	50.58
	10	1	I	8.3	31.12	30.4	68.06	40.3	77.55	40.4	79.05	57.9	77.55
	20	1	I	8.0	25.27	41.4	51.85	74.0	65.32	72.6	64.08	91.9	67.32
	50	1	I	7.7	22.17	40.9	41.89	88.8	57.70	88.7	54.03	95.2	57.73
	100	1	I	7.9	25.27	46.2	48.96	93.3	57.45	87.5	57.45	97.7	59.02
	10	1	U	6.6	53.44	17.5	87.47	22.2	95.12	22.7	94.36	30.5	95.58
	20	1	U	5.7	57.01	23.2	90.73	29.5	97.82	24.1	99.04	35.0	98.65
	50	1	U	4.5	84.83	10.2	94.83	16.6	95.73	15.9	95.03	17.6	95.73
	100	1	U	4.1	81.91	8.1	97.67	7.6	100.00	6.4	100.00	8.0	100.00
	50	2	B	19.1	11.20	62.0	15.03	141.8	18.09	167.9	18.66	158.8	18.36
	50	3	B	25.5	4.66	72.1	5.73	170.4	7.59	210.9	8.30	196.4	7.61
	50	4	B	31.1	2.08	85.0	2.68	189.8	3.30	227.4	3.79	224.8	3.36
	50	2	U	15.9	19.87	59.2	33.04	139.9	40.10	162.3	40.98	156.2	41.17
	50	3	U	21.9	9.75	68.2	14.71	166.9	17.14	208.2	18.51	192.9	17.63
	50	4	U	28.2	5.45	81.7	6.87	188.7	9.11	226.9	9.66	222.0	9.18
10	10	1	B	24.6	42.46	55.7	73.29	61.2	82.80	62.1	84.12	98.0	83.78
	20	1	B	28.8	31.28	91.0	51.74	149.1	60.55	160.4	61.13	198.8	60.08
	50	1	B	30.5	32.59	100.0	45.48	192.6	54.40	194.1	53.39	214.5	54.65
	100	1	B	30.4	32.62	102.6	46.52	197.1	54.13	185.4	53.18	206.7	54.19
	10	1	I	20.6	38.43	62.2	71.23	74.8	80.21	74.7	81.23	115.9	80.16
	20	1	I	21.7	28.71	87.4	57.48	145.8	68.55	143.3	68.61	184.0	70.02
	50	1	I	19.2	26.97	88.3	46.20	184.6	61.12	184.8	58.64	201.6	60.96
	100	1	I	21.1	28.98	98.4	51.84	191.7	60.43	184.9	60.35	204.5	61.82
	10	1	U	14.2	58.53	32.9	91.43	35.8	95.98	36.4	95.91	49.9	96.27
	20	1	U	12.8	68.65	39.6	95.33	37.1	99.76	24.7	100.00	39.1	100.00
	50	1	U	9.1	89.69	13.9	95.55	19.1	96.52	21.0	95.99	20.2	96.52
	100	1	U	7.3	89.50	10.8	99.33	7.6	100.00	6.4	100.00	8.0	100.00
	50	2	B	49.2	12.32	133.8	16.04	288.7	18.92	340.3	19.55	330.7	19.04
	50	3	B	61.4	5.02	153.1	6.14	345.1	7.85	423.2	8.61	404.1	7.95
	50	4	B	71.8	2.27	179.3	2.87	383.5	3.43	455.0	3.93	463.6	3.47
	50	2	U	36.5	21.70	124.7	35.30	281.5	42.25	326.5	42.71	318.8	43.02
	50	3	U	51.0	10.41	144.0	15.45	336.5	17.93	416.7	19.23	394.2	18.24
	50	4	U	62.3	5.77	168.3	7.39	379.0	9.53	453.6	10.12	451.9	9.55

just report the ℓ-LLL results for further comparisons, although we performed the computation with both, confirming that the results are very similar.

Concerning the type of instances, we can observe that the gaps closed for unbounded integer instances are larger than those for bounded integer instances, which are in turn larger than those for binary ones. Unfortunately, as the number of rows increases from 1 to 4, the gaps closed decrease significantly, while, on the bright side, increasing the number of rounds up to 10 gives non-trivial improvements. Finally, GMIs very marginally improve on ℓ-LLL, which somehow demonstrates that the strength of ℓ-cuts shown by this experiment does not only depend on the number of cuts generated.

5.2 Comparing k-cuts and ℓ-cuts

In this section, we compare the behaviour of ℓ-cuts and k-cuts. More precisely, we separate k-cuts in the following two possible ways. For each tableau row, with basic variable, say, x_j, we

1. multiply the row by an integer value $k = 1, \dots, 10$ and we thereby generate 10 possibly different k-cuts,
2. multiply the row by an integer w_{ij}, $i = 1, \dots, n-m$, and we generate $n-m$ possibly different k-cuts.

In other words, we either use "trivial" values for k, or we use **individual** k's from the reduced basis LLL. Note that, for the latter, k-cuts and ℓ-cuts are identical for the special case of $R = 1$ and $m = 1$, see Section 4.1.

Table 2 reports on the results of the comparisons among: GMIs from the optimal LP tableau, denoted by GMI, k-cuts of type 1 above, denoted by $k-10$, k-cuts of type 2 above, denoted by k-LLL, a combination of GMIs and k-cuts, denoted by GMI + k-LLL, ℓ-cuts from LLL-reduced bases, denoted as before by ℓ-LLL, and a combination of GMIs and ℓ-LLL, denoted by GMI+ℓ-LLL. (Note that columns GMI, ℓ-LLL and GMI+ℓ-LLL are the same as in Table 1.)

The results in Table 2 clearly show that for $R > 1$ the gap closed by ℓ-LLL is significantly larger than that closed by k-LLL and with far fewer cuts. Recall that the entries for k-LLL and ℓ-LLL are necessarily identical for $R = 1$ and $m = 1$. Moreover, the gap closed by k-LLL is slightly larger than that of $k - 10$, but with more cuts in general. Finally, the improvement of GMIs + k-LLL with respect to k-LLL is much more significant than that of GMIs + ℓ-LLL with respect to ℓ-LLL.

5.3 Comparing rank and row aggregation

In this section, we compare the use of ℓ-cuts in multiple rounds, as in the previous tables, i.e., by using for separation the row aggregation provided by the simplex algorithm, with the case in which we optimize over the aggregation by solving an LP but we stay at rank 1, i.e., we only use the original constraints and the $\boldsymbol{W}$-matrix. The latter procedure, if iterated, allows to compute the *approximated strengthened* ℓ-LLL *closure*, by adapting the algorithm proposed by Bonami [8] for the *strengthened lift-and-project closure*. More precisely,

- The strengthened lift-and-project closure of a mixed integer linear program is the polyhedron obtained by intersecting all strengthened lift-and-project cuts [5, 18] obtained from its initial formulation, or equivalently all GMIs read from all tableaus corresponding to feasible and infeasible bases of the LP relaxation. An approximation of this closure is computed by iteratively generating lift-and-project cuts and strengthening them by integer lifting, see [8].

Table 2 Comparing k-cuts and ℓ-cuts

				GMI		$k-10$		k-LLL		GMI + k-LLL		ℓ-LLL		GMI + ℓ-LLL	
R	n	m	T	cuts	%gap	cuts	%gap	cuts	%gap	cuts	%gap	cuts	%gap	cuts	%gap
1	10	1	B	1.0	18.61	10.0	34.46	9.0	50.86	9.9	51.17	9.0	50.86	9.9	51.17
	20	1	B	1.0	9.75	10.0	23.74	14.7	34.83	15.5	34.88	14.7	34.83	15.5	34.88
	50	1	B	1.0	12.41	10.0	26.83	15.1	34.40	15.4	34.46	15.1	34.40	15.4	34.46
	100	1	B	1.0	10.66	10.0	29.40	15.3	31.00	15.4	31.00	15.3	31.00	15.4	31.00
	10	1	I	1.0	15.67	10.0	30.16	8.9	55.10	9.9	55.23	8.9	55.10	9.9	55.23
	20	1	I	1.0	13.36	10.0	30.92	14.4	39.88	15.4	40.45	14.4	39.88	15.4	40.45
	50	1	I	1.0	11.59	10.0	24.08	15.5	35.22	15.6	35.22	15.5	35.22	15.6	35.22
	100	1	I	1.0	13.83	10.0	31.55	15.9	34.71	16.1	34.71	15.9	34.71	16.1	34.71
	10	1	U	1.0	37.27	9.8	66.25	7.8	76.65	8.6	76.65	7.8	76.65	8.6	76.65
	20	1	U	1.0	34.88	9.7	74.18	10.4	77.52	10.9	77.96	10.4	77.52	10.9	77.96
	50	1	U	1.0	62.36	7.4	87.95	7.5	92.13	7.6	92.13	7.5	92.13	7.6	92.13
	100	1	U	1.0	64.07	5.6	98.71	5.1	98.59	5.3	98.59	5.1	98.59	5.3	98.59
	50	2	B	2.0	5.31	19.7	10.11	50.0	12.87	50.9	12.87	26.6	12.03	27.6	12.13
	50	3	B	3.0	2.55	30.0	4.22	92.5	5.54	94.5	5.61	32.5	5.10	34.7	5.19
	50	4	B	4.0	1.17	40.0	2.17	139.7	2.51	142.9	2.53	37.0	2.06	40.4	2.14
	50	2	U	2.0	13.02	20.0	24.49	42.7	33.06	43.5	33.14	26.6	32.14	27.7	32.50
	50	3	U	3.0	5.48	30.0	11.20	87.9	13.41	89.9	13.58	32.5	12.26	34.9	12.68
	50	4	U	4.0	3.48	40.0	6.32	137.2	8.50	139.9	8.50	37.2	6.95	40.4	6.99
5	10	1	B	9.0	35.29	127.6	47.98	105.7	54.14	126.1	62.70	36.6	79.98	51.8	81.23
	20	1	B	10.4	25.32	156.1	33.03	235.8	37.52	266.2	45.44	75.3	57.27	96.1	57.04
	50	1	B	10.5	28.08	157.8	38.24	283.1	41.54	300.0	42.91	91.7	51.77	100.4	52.06
	100	1	B	10.1	28.91	164.0	40.02	261.0	40.92	270.3	41.71	95.0	50.46	99.5	50.58
	10	1	I	8.3	31.12	122.9	42.23	140.5	56.14	145.2	64.05	40.3	77.55	57.9	77.55
	20	1	I	8.0	25.27	135.1	36.05	234.9	43.39	256.6	49.93	74.0	65.32	91.9	67.32
	50	1	I	7.7	22.17	132.9	32.33	236.3	42.28	243.4	44.61	88.8	57.70	95.2	57.73
	100	1	I	7.9	25.27	138.9	39.05	254.2	42.27	262.0	43.13	93.3	57.45	97.7	59.02
	10	1	U	6.6	53.44	88.9	73.83	68.2	79.29	68.2	83.30	22.2	95.12	30.5	95.58
	20	1	U	5.7	57.01	77.8	84.04	98.7	81.13	103.0	87.07	29.5	97.82	35.0	98.65
	50	1	U	4.5	84.83	30.9	91.36	43.9	94.87	45.0	94.88	16.6	95.73	17.6	95.73
	100	1	U	4.1	81.91	12.8	100.00	16.2	100.00	15.9	100.00	7.6	100.00	8.0	100.00
	50	2	B	19.1	11.20	244.3	13.47	675.0	15.27	711.1	15.98	141.8	18.09	158.8	18.36
	50	3	B	25.5	4.66	320.6	5.80	1072.4	6.43	1104.2	6.79	170.4	7.59	196.4	7.61
	50	4	B	31.1	2.08	372.8	2.79	1439.8	2.74	1472.2	2.97	189.8	3.30	224.8	3.36
	50	2	U	15.9	19.87	225.0	28.63	654.0	36.15	663.0	36.99	139.9	40.10	156.2	41.17
	50	3	U	21.9	9.75	314.9	13.29	1032.3	14.76	1031.1	15.97	166.9	17.14	192.9	17.63
	50	4	U	28.2	5.45	358.6	7.39	1432.1	8.93	1495.8	9.51	188.7	9.11	222.0	9.18
10	10	1	B	24.6	42.46	301.9	52.21	223.9	54.93	300.7	65.92	61.2	82.80	98.0	83.78
	20	1	B	28.8	31.28	359.9	37.12	510.0	38.10	645.3	47.84	149.1	60.55	198.8	60.08
	50	1	B	30.5	32.59	397.3	40.99	637.5	43.25	688.2	45.17	192.6	54.40	214.5	54.65
	100	1	B	30.4	32.62	398.8	42.82	600.7	44.02	624.3	44.82	197.1	54.13	206.7	54.19
	10	1	I	20.6	38.43	269.6	44.70	285.8	56.77	318.5	65.69	74.8	80.21	115.9	80.16
	20	1	I	21.7	28.71	308.3	37.71	499.9	44.22	551.8	52.77	145.8	68.55	184.0	70.02
	50	1	I	19.2	26.97	320.5	34.75	515.2	43.31	532.9	46.65	184.6	61.12	201.6	60.96
	100	1	I	21.1	28.98	326.8	43.72	536.0	43.22	541.4	45.74	191.7	60.43	204.5	61.82
	10	1	U	14.2	58.53	183.3	76.06	143.6	80.54	143.3	85.03	35.8	95.98	49.9	96.27
	20	1	U	12.8	68.65	158.7	87.45	212.3	82.67	212.2	89.62	37.1	99.76	39.1	100.00
	50	1	U	9.1	89.69	63.3	92.12	77.9	94.98	75.7	95.04	19.1	96.52	20.2	96.52
	100	1	U	7.3	89.50	12.8	100.00	16.2	100.00	15.9	100.00	7.6	100.00	8.0	100.00
	50	2	B	49.2	12.32	573.2	14.08	1486.7	15.69	1580.3	16.62	288.7	18.92	330.7	19.04
	50	3	B	61.4	5.02	726.6	6.05	2227.4	6.64	2318.9	6.98	345.1	7.85	404.1	7.95
	50	4	B	71.8	2.27	847.6	2.90	2895.1	2.84	3035.1	3.07	383.5	3.43	463.6	3.47
	50	2	U	36.5	21.70	486.8	29.71	1323.9	36.65	1360.1	38.22	281.5	42.25	318.8	43.02
	50	3	U	51.0	10.41	672.9	13.75	2102.6	14.95	2118.5	16.26	336.5	17.93	394.2	18.24
	50	4	U	62.3	5.77	760.1	7.64	2812.4	9.00	2960.1	9.65	379.0	9.53	451.9	9.55

– Analogously, given a reduced $\boldsymbol{W}$-matrix to generate rank-1 ℓ-cuts, the approximated strengthened ℓ-LLL closure is computed as follows. If $\boldsymbol{x}^*$ is the optimal LP solution and $\boldsymbol{w}^i x^* \notin \mathbb{Z}$, one generates an intersection cut [5] on the disjunction, $\boldsymbol{w}^i\boldsymbol{x} \leq \lfloor \boldsymbol{w}^i\boldsymbol{x}^* \rfloor$ and $\boldsymbol{w}^i\boldsymbol{x} \geq \lceil \boldsymbol{w}^i\boldsymbol{x}^* \rceil$, which is then strengthened. This is repeated for each row i of $\boldsymbol{W}$ at each iteration until no more violated cuts are found.

In terms of closures, the comparison is completed by reporting on the results for the split closure. Exploiting the result reported [17] that shows the equivalence between the split closure and the Mixed-Integer Rounding (MIR) closure, the split closure is computed by iteratively separating violated MIR cuts through the solution of a mixed-integer program as in [17].

Table 3 reports on the results on the comparisons between:

– 10 rounds of: (a) ℓ-LLL cuts, (b) a combination of GMIs and ℓ-LLL cuts, denoted by GMI+ℓ-LLL, and
– the approximated closures of: (c) strengthened lift-and-project cuts, denoted by "str. L&P", (d) strengthened ℓ-LLL cuts, denoted by "str. ℓ-LLL", (e) split cuts, denoted by "split".

In contrast to the cases of strengthened L&P and ℓ-LLL closures, the term "approximated" for the split closure refers to the fact that the computation is stopped after a time limit of 5 hours. Such a time limit affects only the multi-row instances with binary variables and this is indicated in the table by "*".

Table 3 Comparing higher rank cuts with rank-1 closures

			10 rounds				"approximated" closures						
			ℓ-LLL		GMI + ℓ-LLL		str. L&P		str. ℓ-LLL		split		
n	m	T	cuts	%gap	cuts	%gap	cuts	%gap	cuts	%gap	cuts	%gap	
10	1	B	61.2	82.80	98.0	83.78	4.1	29.87	41.6	99.73	62.3	100.00	
20	1	B	149.1	60.55	198.8	60.08	4.0	18.56	71.4	61.93	148.4	84.11	
50	1	B	192.6	54.40	214.5	54.65	4.1	22.32	48.0	50.84	174.7	88.29	
100	1	B	197.1	54.13	206.7	54.19	4.4	21.38	42.8	46.81	162.3	89.72	
10	1	I	74.8	80.21	115.9	80.16	1.3	16.89	14.3	63.87	48.3	88.97	
20	1	I	145.8	68.55	184.0	70.02	1.3	13.68	21.2	47.42	53.9	81.90	
50	1	I	184.6	61.12	201.6	60.96	1.2	12.26	21.1	42.02	61.6	82.28	
100	1	I	191.7	60.43	204.5	61.82	1.3	14.94	20.2	42.28	57.0	85.22	
10	1	U	35.8	95.98	49.9	96.27	1.0	37.27	8.8	79.31	25.9	97.58	
20	1	U	37.1	99.76	39.1	100.00	1.0	34.88	10.4	77.52	27.0	92.57	
50	1	U	19.1	96.52	20.2	96.52	1.0	62.36	7.5	92.13	38.1	99.97	
100	1	U	7.6	100.00	8.0	100.00	1.0	64.07	5.1	98.59	35.9	98.70	
50	2	B	288.7	18.92	330.7	19.04	10.1	9.95	84.4	18.96	460.8	41.24	*
50	3	B	345.1	7.85	404.1	7.95	15.8	4.23	99.7	8.09	519.0	18.95	*
50	4	B	383.5	3.43	463.6	3.47	20.7	1.99	140.5	4.05	518.6	8.49	*
50	2	U	281.5	42.25	318.8	43.02	2.0	13.02	26.7	32.28	178.6	71.87	
50	3	U	336.5	17.93	394.2	18.24	3.3	5.57	34.7	12.81	342.4	42.08	
50	4	U	379.0	9.53	451.9	9.55	4.7	3.76	40.9	7.49	372.8	22.41	

The results in Table 3 clearly show that growing the rank of the ℓ-cuts gives generally better results than optimizing over the approximate closure of the disjunctions in the $\boldsymbol{W}$-matrix although there is no domination. Nevertheless, it is confirmed that the approximated strengthened ℓ-LLL closure is way stronger than the approximated strengthened L&P closure. In other words, elementary disjunctions in the reformulated space are stronger than elementary disjunctions in the original space. With few exceptions, neither the strengthened ℓ-LLL closure nor the strengthened L&P closure provide a good approximation of the rank-1 split closure. Finally, separating both ℓ-LLL and L&P cuts together does not significantly improve over ℓ-LLL alone, although the results are not explicitly reported in the table.

6 Concluding remarks

Our ℓ-cuts are generated based on general disjunctions originating from information on the lattice structure of the underlying problem. For the test instances, which are similar to the instances used by [14] in their computational study of k-cuts, we observe that the lattice structure gives useful information to obtain cuts that improve on standard GMI/Split cuts and k-cuts. For single-row problems, a large percentage of the integrality gap is closed. For multi-row problems the results are not as good, and it remains a challenge to identify cuts that can be generated within reasonable computing time and that work well on multi-row problems.

We observe that the better the quality of the basis generating the lattice, the better the quality of the resulting ℓ-cuts. We have, however, only tried one lattice reformulation [1], and given the partial success of the approach it would be useful to investigate other reformulations, in particular a reformulation that captures multi-row problems better. Also, extending our approach to the mixed-integer case would be interesting.

References

1. Aardal, K., Hurkens, C.A.J., Lenstra, A.K.: Solving a system of linear Diophantine equations with lower and upper bounds on the variables. Mathematics of Operations Research 25(3), 427–442 (2000).
2. Aardal, K., Lenstra, A.K.: Hard equality constrained integer knapsacks. Mathematics of Operations Research 29(3), 724–738 (2004). Erratum: Mathematics of Operations Research 31(4), 846 (2006).
3. Aardal, K., Bixby, R.E., Hurkens, C.A.J., Lenstra, A.K., Smeltink, J.W.: Market split and basis reduction: Towards a solution of the Cornuéjols-Dawande instances. INFORMS Journal on Computing 12, 192–202 (2000).
4. Aardal, K., Wolsey, L.A.: Lattice based extended formulations for integer linear equality systems. Mathematical Programming 121, 337–352 (2010).
5. Balas, E.: Intersection cuts—a new type of cutting planes for integer programming. Operations Research 19, 19–39 (1971).
6. Balas, E., Saxena, A.: Optimizing over the split closure. Mathematical Programming 113(2), 219–240 (2008).
7. Bixby, R.E., Gu, Z., Rothberg, E., Wunderling, R.: Mixed integer programming: A progress report. In: Grötschel, M. (ed.), The Sharpest Cut: The Impact of Manfred Padberg and His Work, MPS/SIAM Series in Optimization, 309–326 (2004).
8. Bonami, P.: On optimizing over lift-and-project closures. Mathematical Programming Computation 4(2), 151–179 (2012).
9. Caprara, A., Letchford, A.N.: On the separation of split cuts and related inequalities. Mathematical Programming 94, 279–294 (2006).
10. Chvátal, V.: Edmonds polytopes and a hierarchy of combinatorial problems. Discrete Mathematics 4, 305–337 (1973).
11. Conforti, M., Cornuéjols, G., Zambelli, G.: Integer Programming. Springer (2014).
12. Cook, W., Kannan, R., Schrijver, A.: Chvátal closures for mixed-integer programming. Mathematical Programming 47, 155–174 (1990).
13. Cornuéjols, G.: Valid inequalities for mixed integer linear programs. Mathematical Programming 112, 3-44 (2008).
14. Cornuéjols, G., Li, Y., Vandenbussche, D.: K-cuts: A variation of Gomory Mixed Integer Cuts from the LP tableau. INFORMS Journal on Computing 15(4), 385–396 (2003).

15. Cornuéjols, G., Nannicini, G.: Practical strategies for generating rank-1 split cuts in mixed-integer linear programming. Mathematical Programming Computation 3(4), 281–318 (2011).
16. Dash, S., Goycoolea, M.: A heuristic to generate rank-1 GMI cuts. Mathematical Programming Computation 2, 231–257 (2010).
17. Dash, S., Günlük, O., Lodi, A.: MIR closures of polyhedral sets, Mathematical Programming 121, 33–60 (2010).
18. Fischetti, M., Lodi, A., Tramontani, A.: On the separation of disjunctive cuts. Mathematical Programming 128, 205–230 (2011).
19. Fischetti, M. Salvagnin, D.: A relax-and-cut framework for Gomory mixed-integer cuts. Mathematical Programming Computation 3, 79–102 (2011).
20. Gomory, R.E.: An algorithm for integer solutions to linear programs. In: Graves, R.L., Wolfe, P. (eds.), Recent Advances in Mathematical Programming, McGraw-Hill, New York, NY, 269–302 (1963).
21. Gomory, R.E.: Some polyhedra related to combinatorial problems. Linear Algebra and Its Applications 2, 451–558 (1969).
22. von Heymann, F.: On Lattice Methods in Integer Optimization. PhD Thesis, Delft University of Technology, ISBN 978-94-6191-874-1 (2013).
23. Korkine, A., Zolotareff, G.: Sur les formes quadratiques. Mathematische Annalen 6, 366–389 (1873).
24. Lagarias, J.C., Lenstra, Jr., H.W., Schnorr, C.P.: Korkin-Zolotarev bases and successive minima of a lattice and its reciprocal lattice. Combinatorica 10(4), 333–348 (1990).
25. Lenstra, A.K., Lenstra, Jr., H.W., Lovász, L.: Factoring polynomials with rational coefficients. Mathematische Annalen 261(4), 515–534 (1982).
26. Mehrotra, S., Li, Z.: Branching on hyperplane methods for mixed integer linear and convex programming using adjoint lattices. Journal of Global Optimization 49(4), 623–649 (2011).
27. Nemhauser, G.L., Wolsey, L.A.: Integer and Combinatorial Optimization. Wiley-Interscience, New York, NY (1988).
28. Nemhauser, G.L., Wolsey, L.A.: A recursive procedure to generate all cuts for 0-1 mixed integer programs. Mathematical Programming 46, 379–390 (1990).
29. Schrijver, A.: Theory of Linear and Integer Programming. John Wiley & Sons Ltd. (1986).