A Precise Correspondance Between Lift-and-Project Cuts, Simple Disjunctive Cuts, and Mixed Integer Gomory Cuts

Egon Balas   Michael Perregaard

September 29, 2005

Outline of the Paper

1. problem statement
2. Simple Disjuctive Cuts and Mixed Integer Gomory Cuts
3. Lift-and-Project Cuts
4. correspondance btw. Lift-and-Project Cuts and Simple Disjuctive Cuts
5. correspondance for the strengthened versions

Outline of the Paper (Cont'd)

1. bounds on the number o fundominated disjunctive cuts
2. bounds on the rank of LP polyhedron wrt. various families of cuts
3. an algorithm for solving the cut generating LP
4. computational results
5. using the algorithm for Gomory Cuts

Mixed Integer 0-1 Program

(MIP):

$$
\begin{aligned}
min \quad & cx \\
s.t \quad & \\
Ax \quad & \geq \quad b \\
x \quad & \geq \quad 0 \\
x_j \quad & \in \quad \{0, \quad 1\}, \text{ j = 1,...,p}
\end{aligned}
$$

LP Relaxation

(LP):
$min\{cx : \quad x \in P\}$,
$P := \{x \in \mathbb{R}^n_+\}$

P is sometimes denoted by $\tilde{A}x \geq \tilde{b}$, where $A := \begin{pmatrix} A \\ I \end{pmatrix}$ and $b := \begin{pmatrix} b \\ 0 \end{pmatrix}$.

- $\bar{x}$ denotes the optimum solution to the (LP)
- S is the set of surplus variables and N is the set of structural variables

Mixed Integer 0-1 Program (Cont'd)

- the simplex tableau for (LP) can be uniquely determined by the set of variables chosen to be nonbasic.
- the simplex tableau with such a choice can be writen as

$$
\begin{aligned}
x_i + \sum_{j \in N \cap J} \bar{a}_{ij} x_j + \sum_{j \in S \cap J} \bar{a}_{ij} s_j &= \bar{a}_{i0} \text{ for } i \in N \cap I \\
s_i + \sum_{j \in N \cap J} \bar{a}_{ij} x_j + \sum_{j \in S \cap J} \bar{a}_{ij} s_j &= \bar{a}_{i0} \text{ for } i \in S \cap I
\end{aligned}
$$

$\bar{a}_{ij}$ denotes the coefficient of nonbasic variable j in the row for the nonbasic variable i, and $\bar{a}_{i0}$ is the corresponding RHS

Outline

Simple Disjuctive vs. Mixed Int. Gomory

- if we identify the nonbasic variables x_j with their corresponding surplus variables s_j, row k becomes:

$$x_k + \sum_{j \in J} \bar{a}_{kj} s_j = \bar{a}_{k0}$$

- in particular, chose x_k to be s.t. $0 \leq \bar{a}_{k0} \leq 1$ and apply disjunction $x_k \leq 0 \vee x_k \geq 1$ you get $\pi s_j \geq \pi_0$ where $\pi_0 := \bar{a}_{k0}(1 - \bar{a}_{k0})$ and $\pi_j := max\{\bar{a}_{k0}(1 - \bar{a}_{k0}), -\bar{a}_{kj}\bar{a}_{k0}\}$
- the cut $\pi s_j \geq \pi_0$ depends on nonbasic set J.

Simple Disjuctive vs. Mixed Int. Gomory(Cont'd)

- if $p \geq 1$, $\pi s_j \geq \pi_0$ can be strengthened by replacing π with $\bar{\pi}$:

$$\bar{\pi} := \begin{cases} min\{f_{kj}(1-\bar{a}_{k0}), (1-f_{kj})\bar{a}_{k0}\} & j \in J \cap \{1,...p\} \\ \pi_j & j \in J - \{1,...,p\} \end{cases}$$

  with $f_{kj} := \bar{a}_{kj} - \lfloor \bar{a}_{kj} \rfloor$
- the strengthened version is the same as the Mixed Integer Gomory Cut

Outline

Lift-and-Project

- Lift and Project cuts are special disjunctive cuts of the form

$$\begin{pmatrix} Ax & \geq & b \\ x & \geq & 0 \\ -x_k & \geq & 0 \end{pmatrix} \vee \begin{pmatrix} Ax & \geq & b \\ x & \geq & 0 \\ -x_k & \geq & 1 \end{pmatrix}$$

for some $k \in \{1, ..., p\}$ such that $0 < \bar{x}_k < 1$.

Lift-and-Project [Lift]

Theorem 1([1]): Let the disjunctive constraints be

$$\bigvee_{h \in Q} (D^h x \geq d_0^h)$$

and let $A^h = \begin{pmatrix} A \\ D^h \end{pmatrix}$, $a_0^h = \begin{pmatrix} a_0 \\ d_0^h \end{pmatrix}$

Let F be the feasible set of Disjunctive Program (DP). Then

$$F = \left\{ x \in \mathbb{R}^n : \quad \bigvee_{h \in Q} (A^h x \geq a_0^h, x \geq 0) \right\}$$

Letting $F_h \{x \in \mathbb{R}^n : (A^h x \geq a_0^h, x \geq 0)\}$, we note $F = \bigcup_{h \in Q} F_h$. Let $Q^* = \{h \in Q | F_h \neq \emptyset\}$

claim: If $F \neq \emptyset$,

$$clconv\ F = \left\{ x \in \mathbb{R}^n \;\middle|\; \begin{array}{l} x = \sum_{h \in Q*} \xi^h, \\ A^h \xi^h - a_0^h \xi_0^h \geq 0, \quad h \in Q^* \\ x = \sum_{h \in Q*} \xi_0^h = 1 \end{array} \right\}$$

Lift-and-Project [Lift](Cont'd)

proof: Let S denote the RHS in the claim, so that the theorem is $F = S$. If Q is finite and $F \neq \emptyset$, then $Q^* \neq \emptyset$ and is finite. Moreover,

$$clconv\ F = clconv \left(\bigcup_{h \in Q} F_h \right)$$

(i) $F \subseteq S$:

If $x \in convF$, then x is a convex combination of at most $|Q^*|$ points, belonging to a different F_h:

$$x = \sum_{h \in Q^*} \lambda^h u^h, \quad \lambda^h \geq 0, h \in Q^*$$

where $\sum_{h \in Q^*} \lambda^h = 1$ and for each $h \in Q^*, A^h u^h \geq a_0^h, u^h \geq 0$

Lift-and-Project [Lift](Cont'd)

- We immediately note that if $x, \lambda^h, u^h, h \in Q^*$ satisfy the above constraints, then
  $x, \quad \xi_0^h = \lambda^h, \quad \xi^h = u^h\lambda^h, \quad h \in Q^*$ satisfies S.

$\Rightarrow$**(i)**

Lift-and-Project [Lift](Cont'd)

(ii) $S \subseteq clconvF$**:**

Let $\bar{x} \in S$ with associated vectors $(\bar{\xi}^h, \bar{\xi}_0^h)$, $h \in Q^*$. Let's divide the index set of nonempty F_h sets, Q^* so that

$$Q_1^* = \{h \in Q^* | \xi_0^h > 0\}, Q_2^* = \{h \in Q^* | \xi_0^h = 0\}$$

case $h \in Q_1^*$**:** $\bar{\xi}^h / \bar{\xi}_0^h$ is a solution to $A^h x \geq a_0^h, x \geq 0$ (see RHS) thus $(\bar{\xi}^h / \bar{\xi}_0^h) \in F_h$, So

$$(\bar{\xi}^h / \bar{\xi}_0^h) = \sum_{i \in U_h} \mu^{hi} u^{hi} + \sum_{k \in V_h} \nu^{hk} v^{hk}$$

for some $u^{hi} \in vertF_h$, $i \in U_h$ and $v^{hk} \in dirF_h$, $k \in V_h$ with U_h, V_h finite inex sets, $mu^{hi}, \nu^{hk} \geq 0$, and $\sum_{i \in U_h} \mu^{hi} = 1$

Lift-and-Project [Lift](Cont'd)

By setting $\mu^{hi}\bar{\xi}_0^h = \theta^{hi}$, and $\nu^{hk}\bar{\xi}_0^h = \sigma^{hk}$ we get:

$$\bar{\xi}^h = \sum_{i \in U_h} \theta^{hi} u^{hi} + \sum_{k \in V_h} \sigma^{hk} v^{hk}$$

with $\theta^{hi} \geq 0$, $i \in U_h$, $\sigma^{hk} \geq 0$, $k \in V_h$ and $\sum_{i \in U_h} \theta^{hi} = \bar{\xi}_0^h$

Lift-and-Project [Lift](Cont'd)

case $h \in Q_2^*$**:** either $\bar{\xi}^h = 0$, or $\bar{\xi}_0^h$ is a solution to $Ax \geq 0, x \geq 0$ (extreme ray) thus

$$\bar{\xi}^h = \sum_{k \in V_h} \sigma^{hk} v^{hk}$$

with $\theta^{hi} \geq 0$, $k \in V_h$ for some $v^{hk} \in dirF_h$

Lift-and-Project [Lift](Cont'd)

Thus,

$$\bar{x} = \sum_{h \in Q^*} \bar{\xi}^h$$
$$= \sum_{h \in Q_1^*} \left(\sum_{i \in U_h} \theta^{hi} u^{hi} + \sum_{k \in V_h} \sigma^{hk} v^{hk} \right) + \sum_{h \in Q_2^*} \left(\sum_{k \in V_h} \sigma^{hk} v^{hk} \right)$$

- Noting that $\sum_{h \in Q_1^*} \sum_{i \in U_h} \theta^{hi} u^{hi} = \sum_{h \in Q_1^*} \bar{\xi}_0^h = 1$, we realize that $\bar{x}$ is a convex combination of finitely many points and directions of F.

$\Rightarrow$**(ii)**

Lift-and-Project [Lift](Cont'd)

- So, $conv \quad F \subseteq S \subseteq clconv \quad F$ and since $clconv \quad F$ is the smallest closed set containing $convF$, $clconvF = S$.

Lifting in our special case, $x_j \in \{0, 1\}$

$$P_{j0} := \{x \in \mathbb{R}^n_+ : Ax \geq b, x_j = 0\}$$

$$P_{j1} := \{x \in \mathbb{R}^n_+ : Ax \geq b, x_j = 1\}$$

$$\begin{array}{cccccccc}
x & -y & & -z & & = & 0 \\
 & Ay & -by_0 & & & \geq & 0 \\
 & -y_j & 0y_0 & & & = & 0 \\
 & & & Az & -bz_0 & \geq & 0 \\
 & & & z_j & -1z_0 & = & 0 \\
 & & y_0 & & z_0 & = & 1
\end{array}$$

Lift-and-Project [Project]

- We want a cut of the form $\alpha x \geq \beta$. To get this from the disjunctive constraint set above, let A^i be A amended with the unit vector row e_j. Let $b^1 = \begin{pmatrix} b \\ 0 \end{pmatrix}$ and $b^2 = \begin{pmatrix} b \\ 1 \end{pmatrix}$.
- Then to satisfy the constraints $A^i x \geq b^i$, we should have $\alpha x \geq A^i x \geq b^i \geq \beta$. In other words, $\alpha \geq u^i A^i$ and $\beta \leq u^i b^i$.

Lift-and-Project [Project]

the resulting feasible set for (α, β) is thus:

$$\begin{aligned}
\alpha &\geq uA - u_0 e_j \\
\alpha &\geq vA + v_0 e_j \\
\beta &\leq ub \\
\beta &\leq vb + v_0 \\
u, v &\geq 0 \\
(\alpha, \beta) &\in \mathbb{R}^{n+1}
\end{aligned}$$

Lift-and-Project (Cont'd)

- A lift-and-project cut can be obtained solving the program $(CGLP)_k$

$$
\begin{array}{llllllllll}
min & \alpha\bar{x} & -\beta & & & & & & & \\
st & & & & & & & & & \\
 & \alpha & & -uA & +u_0 e_k & & & \geq & 0 \\
 & \alpha & & & & -vA & +v_0 e_k & \geq & 0 \\
 & & -\beta & +ub & & & & = & 0 \\
 & & -\beta & & & +vb & +v_0 & = & 0 \\
 & & & \sum_{i=1}^{m+p} u_i & +u_0 & +\sum_{i=1}^{m+p} v_i & +v_0 & = & 1
\end{array}
$$

$$u, u_0, v, v_0 \geq 0$$

Lift-and-Project (Cont'd)

- this program maximizes the cut off
- α and β are urs, so they can be eliminated and can be retrieved anytime given the solution vector for u, u_0, v, v_0:

$$\beta := ub = vb + v_0$$

$$\alpha := \begin{cases} max\{ua_j, va_j\} & j \neq k \\ max\{ua_k - u_0, va_j + v_0\} & j = k \end{cases}$$

Lift-and-Project (Cont'd)

- this also can be strengthened using the integrality of the x_j , $j \in \{1, ..., p\} - \{k\}$:

$$\bar{\alpha} := \begin{cases} min\{ua_j + u_0\lceil m_j \rceil, va_j - v_0\lfloor m_j \rfloor\} & j \in \{1, ..., p\} - \{k\} \\ \alpha_j, & j \in \{k\} \cup \{p+1, ..., n\} \end{cases}$$

with $m_j := \frac{va_j - ua_j}{u_0 + v_0}$.

Correspondance btw. the Unstrengthened Cuts

- introduce surplus variables to $(CGLP)_k$ so that u, v have the surplus variables included:

$$
\begin{array}{llllllll}
min & \alpha\bar{x} & -\beta & & & & & \\
st & & & & & & & \\
 & \alpha & & -uA & +u_0 e_k & & & = 0 \\
 & \alpha & & & & -vA & +v_0 e_k & = 0 \\
 & & -\beta & +ub & & & & = 0 \\
 & & -\beta & & & +vb & +v_0 & = 0 \\
 & & & \sum_{i=1}^{m+p} u_i & +u_0 & +\sum_{i=1}^{m+p} v_i & +v_0 & = 1
\end{array}
$$

$$u, u_0, v, v_0 \geq 0$$

Correspondance btw. the Unstrengthened Cuts (Cont'd)

Lemma 1: In any basic solution to the constraint set above that gives $\alpha \geq \beta$ not dominated by the constraint set of (LP), $u_0, v_0 > 0$.
proof:

Correspondance btw. the Unstrengthened Cuts (Cont'd)

Lemma 2: Let $(\bar{\alpha}, \bar{\beta}, \bar{u}, \bar{u_0}, \bar{v}, \bar{v_0})$ be a basic solution to the above constraint set, $\bar{u_0}, \bar{v_0} > 0$ $(\bar{\alpha}, \bar{\beta})$ basic.(They are URS). Let the basic components of $\bar{u}$ and $\bar{v}$ be indexed by M_1 and M_2. Then $M_1 \cap M_2 = \emptyset$, $|M_1 \cup M_2| = n$, and submatrix $\hat{A}_{nxn}$ of $\tilde{A}$ whose rows are indexed by $M_1 \cup M_2$ is nonsingular.

proof:

Correspondance btw. the Unstrengthened Cuts (Cont'd)

- define $J := M_1 \cap M_2$
- replace n inequalities indexed by J in $\tilde{A}x \geq \tilde{b}$ this amounts to setting surplus variables to 0. Since $\hat{A}_{nxn}$ is nonsingular, these equalities define a basic solution.
- The simplex tableau associated with this solution has its *nonbasic* variables indexed by J.
- in the $(CGLP)_k$ solution was the index set of basic components of (u, v).

Correspondance btw. the Unstrengthened Cuts (Cont'd)

- we have
$$\hat{A}x - s_j = \hat{b}$$
, or equivalently
$$x = \hat{A}^{-1}\hat{b} + \hat{A}^{-1}s_j$$
.
- if we let $\bar{a}_{k0} = e_k\hat{A}^{-1}\hat{b}$ and $\bar{a}_{kj} = (\hat{A}^{-1})_{kj}$, this can be written as
$$x_k = \bar{a}_{k0} - \sum_{j \in J} \bar{a}_{kj} s_j$$
- this is same as the row of (LP) associated with basic variable x_k

Correspondance btw. the Unstrengthened Cuts (Cont'd)

Lemma 3: $0 < \bar{a_{k0}} < 1$.
proof:

Theorem 4A: Let $\alpha x \geq \beta$ be the lift-and-project cut associated with a basic solution $(\alpha, \beta, u, u_0, v, v_0)$ to $(CGLP)_k$, with $u_0, v_0 > 0$ and all components of α, β basic, and the basic components of u and v be indexed by M_1 and M_2 respectively. Let $\pi s_j \geq \pi_0$ be the simple disjunctive cut from the disjunction $x_k < 0 \vee x_k > 1$ applied to $x_k = \bar{a}_{k0} - \sum_{j \in J} \bar{a}_{kj} s_j$ with $J := M_1 \cap M_2$. Then $\pi s_j \geq \pi_0$ is equivalent to $\alpha x \geq \beta$.

Correspondance btw. the Unstrengthened Cuts (Cont'd)

sketch of proof:

Remember that $x_k < 0 \vee x_k > 1$ applied to $x_k = \bar{a}_{k0} - \sum_{j \in J} \bar{a}_{kj} s_j$

was defined by

$$\pi_0 := \bar{a}_{k0}(1 - \bar{a}_{k0})$$

and

$$\pi_j := max\{\pi_j^1, \pi_j^2\}$$

where

$$\pi_j^1 := \bar{a}_{k0}(1 - \bar{a}_{k0}), \quad \pi_j^2 := -\bar{a}_{kj}\bar{a}_{k0} = (\hat{A}^{-1})_{kj}\bar{a}_{k0}$$

Bounds on the Number of Essential Cuts

- Every valid inequality for$\{x \in P : (x_k \leq 0) \vee (x_k \geq 1)\}$ is dominated by some lift-and-project cut corresponds to a basic solution of a basic solution of $(CGLP)_k$
- The number of undominated valid inequalities is bounded by

$$\binom{2(m+p+n+1)+n+1}{2n+3}$$

- By using Theorem 4A/4B, the number of bases in a simplex tableau where x_k is basic,
  that is, the number of subsets J of cardinality n is

$$\binom{m+p+n-1}{n}$$

Bounds on the Number of Essential Cuts

- Thus the elementary closure $\bigcap_{k=1}^{p} P_k$ of P with respect to the lift-and-project operation has at most

$$p \binom{m+p+n-1}{n}$$

  facets.
- Can we extend these bounds for strengthened lift-and-project cuts?
  - That is OK for strengthened cuts derived from basic solutions
  - But a strengthened cut derived from a nonbasic solution may not be dominated by any strengthened cut derived from a basic solution

The Rank of P With Respect to Diffrent Cuts

- The rank of P with respect to each of the following families is at most p
  - unstrengthened lift-and-project cuts
  - simple disjunction cuts
  - strengthened lift-and-project cuts
  - mixed integer Gomory cuts

The Rank of P With Respect to Diffrent Cuts

- Proof:
  - $P := \{x \in R^n : \tilde{A}x \geq \tilde{b}\}$
  - $P_0 := P$
  - $P_D := conv\{x \in P : x_j \in \{0, 1\}, j = 1, \ldots, p\}$
  - $P^j := conv\{P^{j-1} \cap \{x_j \in R^n : x_j \in \{0, 1\}$
  - then $P^p = P_D$

Solving $(CGLP)_k$ on the (LP) Simplex Tableau

- A basic solution to (LP) associated with set J corresponds to a set of basic solutions to $(CGLP)_k$.
- The various solutions to $(CGLP)_k$ differ among themselves by the partition of J into M_1 and M_2.
- These solutions can be obtained by degenerate pivots in $(CGLP)_k$
- A single pivot in (LP) differs J with some element with together shifting one ore more elements from M_1 to M_2 vice-versa

Solving $(CGLP)_k$ on the (LP) Simplex Tableau

- The simple disjunction cut is defined by $\pi x_J \geq \pi_0$, where $\pi_0 = \bar{a}_{k0}(1 - \bar{a}_{k0})$ and

$$\pi_j := \{ \; max\{\bar{a}_{kj}(1 - \bar{a}_{k0}), -\bar{a}_{kj}\bar{a}_{k0}\} \quad j \in J$$

- We want to pivot on $\bar{a}_{ij}, i \neq k$
  then row k becomes

$$x_k = \bar{a}_{k0} + \gamma_j \bar{a}_{i0} - \sum_{h \in J \setminus \{j\}} (\bar{a}_{kh} + \gamma_j \bar{a}_{ih}) s_h - \gamma_j x_i$$

  where

$$\gamma_j = -\frac{\bar{a}_{kj}}{\bar{a}_{ij}}.$$

- Note that we can pivot on any nonzero $\bar{a}_{ij}$ since we do not restrict ourselves to feasible bases.

Solving $(CGLP)_k$ on the (LP) Simplex Tableau

- Pivoting the variable x_i out of basis corresponds to pivoting into the basis one of the variables u_i or v_i on $(CGLP)_k$
- Such a pivot is improving on $(CGLP)_k$ only if either u_i or v_i have a negative reduced cost
- First, we choose a row i, some multiple of which is to be added to row k,
  second, we choose a column in row i, which sets the sign and size of the multiplier

Solving $(CGLP)_k$ on the (LP) Simplex Tableau

The sketch of the algorithm:

- **Step 0.** Solve (LP). Let $\bar{x}$ be an optimal solution and let k be such that $0 < \bar{x}_k < 1$
- **Step 1.** Let J index the nonbasic variables in the current basis. Compute the reduced costs $r_{u_i} < 0$ with $M_1 = \{j \in J : \bar{a}_{kj} < 0 \vee (\bar{a}_{kj} = 0 \wedge \bar{a}_{ij} > 0)\}$, and $M_2 = J \backslash M_1$ and $r_{v_i} < 0$ with $M_1 = \{j \in J : \bar{a}_{kj} < 0 \vee (\bar{a}_{kj} = 0 \wedge \bar{a}_{ij} < 0)\}$, and $M_2 = J \backslash M_1$ of u_i, v_i corresponding to each row $i \neq j$ of the simplex tableau of LP.
- **Step 2.** Let i_* be a row with $r_{u_{i_*}} < 0$ or $r_{v_{i_*}} < 0$. If no such row exists, go to step 5.

Solving $(CGLP)_k$ on the (LP) Simplex Tableau

- **Step 3.** Identify the most improving pivot column j_* in row i_* by minimizing $f^+(\gamma_j)$ over all $j \in J$ with $\gamma_j > 0$ and $f^-(\gamma_j)$ over all $j \in J$ with $\gamma_j < 0$ and choosing the more negative of these two values.
- **Step 4.** Pivot on $\bar{a}_{i_* j_*}$ and go to Step 1.
- **Step 5.** If row k has no 0 entries, stop.Otherwise perturb row k by replacing every 0 entry by ξ^t for some small ξ and $t = 1, 2, \ldots$(different for each entry).Go to step 1.

Solving $(CGLP)_k$ on the (LP) Simplex Tableau

Let $(\alpha, \beta, u, u_0, v, v_0)$ be a basic feasible solution to CGLP with $u_0, v_0 > 0$, all components of α and β basic, and the basic components of u and v indexed by M_1 and M_2, respectively. Let $\bar{s}$ be surplus variables of $\tilde{A}x \geq \tilde{b}$ corresponding to the solution $\bar{x}$. Then the reduced costs of u_i and v_i, for $i \notin J \cup \{k\}$ in this basic solution are, respectively

$$r_{u_i} = \sigma(-\sum_{j \in M_1} \bar{a}_{ij} + \sum_{j \in M_2} \bar{a}_{ij} - 1) - \sum_{j \in M_2} \bar{a}_{ij}\bar{s}_j + \bar{a}_{i0}(1 - \bar{x}_k)$$

$$r_{v_i} = \sigma(+\sum_{j \in M_1} \bar{a}_{ij} - \sum_{j \in M_2} \bar{a}_{ij} - 1) - \sum_{j \in M_1} \bar{a}_{ij}\bar{s}_j + \bar{a}_{i0}\bar{x}_k$$

where

$$\sigma = \frac{\sum_{j \in M_2} \bar{a}_{kj}\bar{s}_j - \bar{a}_{k0}(1 - \bar{x}_k)}{1 + \sum_{j \in J} |\bar{a}_{kj}|}$$

Solving $(CGLP)_k$ on the (LP) Simplex Tableau

- Write the objective function, $\alpha\bar{x} - \beta$, of $(CGLP)_k$ in terms of u_i and v_i
- Then substitute u_i and v_i in terms of $\bar{a}_{ij}$
- During this calculation, they pointed:
  $u_j = -(u_0 + v_0)\bar{a}_{kj} + (u_i - v_i)\bar{a}_{ij}$ for $j \in M_1$
  $v_j = (u_0 + v_0)\bar{a}_{kj} - (u_i - v_i)\bar{a}_{ij}$ for $j \in M_2$

The pivot column in row i of the (LP) simplex tableau that is most improving with respect to the cut from row k, is indexed by that $l^* \in J$ that minimizes $f^+(\gamma_l)$ if $\bar{a}_{kl}\bar{a}_{il} < 0$ or $f^-(\gamma_l)$ if $\bar{a}_{kl}\bar{a}_{il} > 0$, over all $l \in J$ that satisfies $\frac{-\bar{a}_{k0}}{\bar{a}_{i0}} < \gamma_l < \frac{1-\bar{a}_{k0}}{\bar{a}_{i0}}$, where $\gamma_l := -\frac{\bar{a}_{kl}}{\bar{a}_{il}}$ and for $0 \leq \gamma < \frac{1-\bar{a}_{k0}}{\bar{a}_{i0}}$

$$f^+(\gamma) :=$$

$$\frac{\sum_{j \in J}(-(\bar{a}_{k0} + \gamma\bar{a}_{i0})\bar{a}_{kj} + max\{\bar{a}_{kj}, -\gamma\bar{a}_{ij})\bar{x}_j - (1 - \bar{a}_{k0} - \gamma\bar{a}_{i0}\})\bar{a}_{k0}}{1 + |\gamma| + \sum_{j \in J}|\bar{a}_{kj} + \gamma\bar{a}_{ij}|}$$

and for $\frac{-\bar{a}_{k0}}{\bar{a}_{i0}} < \gamma_l \leq 0$

$$f^-(\gamma) :=$$

$$\frac{\sum_{j \in J}(-(\bar{a}_{k0} + \gamma\bar{a}_{i0})\bar{a}_{kj} + max\{\bar{a}_{kj} + \gamma\bar{a}_{ij}, 0\})\bar{x}_j - (1 - \bar{a}_{k0})(\bar{a}_{k0} + \gamma\bar{a}_{i0}}{1 + |\gamma| + \sum_{j \in J}|\bar{a}_{kj} + \gamma\bar{a}_{ij}|}$$

- At termination, the simple disjuntive cut from row k is an optimal lift-and-project cut; the mixed-integer Gomory cut from row k is an optimal strengthened lift-and-project cut.
- When the algorithm comes to a point where Step 2 finds no row with negative reduced costs, we can not conclude the solution is optimal if there is entries of 0's in row k
- In this case, partition (M_1, M_2) of set J is not unique, so different partition of (M_1, M_2) may lead to a basis change where J‘ differs from J in one element.
- Perturbation in Step 5 eliminates the 0 entries in row k, thus (M_1, M_2) will be unique for set J.

Using Lift-and-Project to Strengthen Mixed Integer Gomory Cuts

- Steiner triple problem with 15 variables and 35 constraints
- LP with five fractional variables is 35.
- Generating mixed integer Gomory cut for each fractional variables yields a solution of value 39
- Using improved cuts in place of original ones we get a solution of value 41.41
- Iterating this procedure for 10 times yields a value of 42.73 for mixed integer Gomory cuts and a value of 44.85 for strengthened cuts.
- IP optimum is 45.
- Intermediate cuts resulting from the procedure are dominated by the final improved ones for the first iteration.

Concluding Remarks

- There are numerous attempts to improve mixed integer Gomory cuts but none of these attempts has succeeded in defining a procedure that is guaranteed to find an improved cuts.
- The lift-and-project approach has done that
- Does the gain in the quality of the cuts justify the computation effort for improving them?