Cut generation for facial disjunctive problems with two-term disjunctions

Tamás Kis

MTA SZTAKI
tamas.kis@sztaki.hu

Outline of the talk

- Disjunctive programming with two-term disjunctions
- The cut generating linear program
- Special case: mixed 0-1 linear programming
- Generalisation to two term disjunctions
- Evaluation
- Final remarks

Disjunctive programming with two-term disjunctions

- Disjunctive programs:

$$DP: \quad \begin{aligned} \min_x \; & c^T x \\ \text{s.t.} \; & Ax \geq b, \\ & x \geq 0, \\ \bigvee_{i \in Q_k} & d_i x \geq d_{i0}, \quad \forall\, k \in D \end{aligned}$$

- If $|Q_k| = 2$, then it is a *two-term disjunction*
- Let $P := \{x \in \mathbb{R}^n \mid Ax \geq b, x \geq 0\}$
- *DP* is *FACIAL* if every $d_i x \geq d_{i0}$ represents a *face* of P:

$$F_i = \{x \in P \mid d_i x \geq d_{i0}\} \text{ is a face of } P$$

- In this talk only FACIAL disjunctive programs with two-term disjunctions occur

Two-term disjunctions

- If DP is facial, w.l.o.g. every two-term disjunction

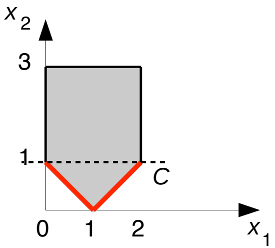
$$d_i x \geq d_{i0} \vee d_j x \geq d_{j0}$$

is equivalent to

$$-\tilde{A}_{k_1} x \geq -\tilde{b}_{k_1} \vee -\tilde{A}_{k_2} x \geq -\tilde{b}_{k_2}$$

for some $(k_1, k_2) \in \{1 \ldots, n+m\}^2$, where $(\tilde{A}, \tilde{b}) = \begin{pmatrix} A & b \\ I_n & 0 \end{pmatrix}$

Two-term disjunctions (cont.)



- Example:

$$
\begin{aligned}
\min_{x_1,x_2}\ & -x_2 \\
\text{s.t.}\ \ & x_1 + x_2 \geq 1 \\
& -x_1 + x_2 \geq -1 \\
& -x_2 \geq -3 \\
& -x_1 \geq -2 \\
& x_1, x_2 \geq 0 \\
& -x_1 - x_2 \geq -1 \lor x_1 - x_2 \geq 1.
\end{aligned}
$$

Disjunctive cuts

- To derive cuts, we add slack variables $s = (s_1 \ldots, s_m)$ to the linear program:

$$\begin{aligned} Ax - I_m s &= b \\ x &\geq 0 \end{aligned}$$

- Let J index the non-basic variables in a basis of the linear program (J may index structural and slack variables as well, and we denote them together by s_J). Suppose s_{k_1} and s_{k_2} are basic, i.e., $k_1, k_2 \notin J$, and the simplex tableau has rows

$$s_{k_1} + \sum_{j \in J} \bar{a}_{k_1,j} s_j = \bar{a}_{k_1,0} \text{ and } s_{k_2} + \sum_{j \in J} \bar{a}_{k_2,j} s_j = \bar{a}_{k_2,0}.$$

- The disjunctive cut with respect to J is

$$\sum_{j \in J} \pi_j s_j \geq \pi_0, \text{ where } \begin{cases} \pi_j = \max\{\bar{a}_{k_1,j}\bar{a}_{k_2,0}, \bar{a}_{k_2,j}\bar{a}_{k_1,0}\}, & j \in J \\ \pi_0 = \bar{a}_{k_1,0}\bar{a}_{k_2,0} \end{cases}$$

The cut generating linear program

- Given $\hat{x} \in P$ such that $\tilde{A}_{k_1}\hat{x} > \tilde{b}_{k_1}$ and $\tilde{A}_{k_2}\hat{x} > \tilde{b}_{k_2}$, find an inequality $\alpha x \geq \beta$ which is valid for the convex hull of feasible solutions of DP, and cuts off $\hat{x}$.
- Applying Balas' results, α, β can be found by solving the following LP:

$$
\begin{aligned}
(CGLP)_{k_1,k_2} \qquad & \min_{(\alpha,\beta,u,v,u_0,v_0)} \quad \alpha\hat{x} - \beta \\
& \text{s.t.} \quad \alpha - u\tilde{A} + u_0\tilde{A}_{k_1} = 0, \\
& \qquad \alpha - v\tilde{A} + v_0\tilde{A}_{k_2} = 0, \\
& \qquad \beta - u\tilde{b} + u_0\tilde{b}_{k_1} = 0, \\
& \qquad \beta - v\tilde{b} + v_0\tilde{b}_{k_2} = 0, \\
& \qquad u\mathbb{1} + v\mathbb{1} + u_0 + v_0 = 1, \\
& \qquad u, v, u_0, v_0 \geq 0.
\end{aligned}
$$

- $\alpha x \geq \beta$ is called lift-and-project cut.

The mixed 0-1 programming special case

- The mixed integer program is

$$
\begin{aligned}
MIP \quad \min_{x,s} \; & c^T x \\
& Ax - I_m s = b \\
& x_i \in \{0,1\}, \quad i = 1, \ldots, p \\
& x, s \geq 0
\end{aligned}
$$

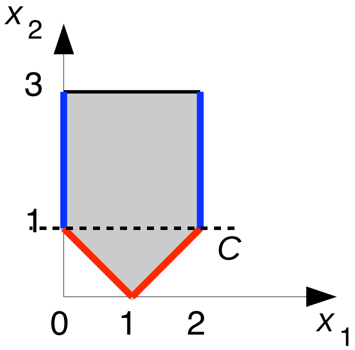
- Disjunctions: $x_i \leq 0 \vee x_i \geq 1$ for $i = 1, \ldots, p$ (binary variables)
- They correspond to the negation of constraints $x_i \geq 0$ and $x_i \leq 1$ of the LP relaxation of MIP
- Let LP be the linear relaxation of MIP

Theorem (Balas & Perregaard)

Let $\alpha, \beta, u_{M_1}, v_{M_2}, u_0, v_0$ constitute the basic variables in a feasible basis of $(CGLP)_{k_1,k_2}$. Then $M_1 \cap M_2 = \emptyset$, $|M_1 \cup M_2| = n$ and $J = M_1 \cup M_2$ is the set of non-basic variables in a basis of LP.

Counterexample for general two-term disjunctions

- In the general case, the previous Theorem does not hold:



- The red and the blue faces constitute a feasible basis of $(CGLP)$, but it is impossible that the slack variables corresponding to the blue faces are simultaneously non-basic in the linear program.

The converse Theorem in the general case

Theorem

Let J be the index-set of non-basic variables in a basis of the linear program. Then $(CGLP)$ has a feasible basis with basic variables $\alpha, \beta, u_{M_1}, v_{M_2}, u_0, v_0$ such that $M_1 \cup M_2 = J$. Moreover, the disjunctive cut $\pi_J s_J \geq \pi_0$ and the lift-and-project cut $\alpha x \geq \beta$ are equivalent.

This results is more general than the following one:

Theorem (Balas & Perregaard)

In the mixed 0-1 programming case, let J be a set of n row indexes such that the square matrix $\tilde{A}_J$ is non-singular. Then J corresponds to the n non-basic variables in some basis of the linear program. Moreover, $(CGLP)$ admits a feasible basis with basic variables $\alpha, \beta, u_{M_1}, v_{M_2}, u_0, v_0$ such that $M_1 \cup M_2 = J$. Moreover, the disjunctive cut $\pi_J s_J \geq \pi_0$ and the lift-and-project cut $\alpha x \geq \beta$ are equivalent.

Computations in the 0-1 case

- Main idea of Balas and Perregaard: mimic the steps of the simplex method for $(CGLP)$ in the original tableau
- Observation: for $i \notin J$ and $j \in J$,

$$\bar{a}_{ij} = -(\tilde{A}_i \tilde{A}_J^{-1})$$
$$\bar{a}_{i0} = \tilde{A}_i \tilde{A}_J^{-1} \tilde{b}_J - \tilde{b}_i$$

- If $0 < \hat{x}_k < 1$, the corresponding row of the simplex tableau is

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

- Balas and Perregaard have shown that the reduced costs r_{u_i} and r_{v_i} for $i \notin J$ can be expressed with the $\bar{a}_{ij}$, $\bar{a}_{i0}$, $\bar{a}_{kj}$ and the $\hat{s}_j$. If $i \notin J$, and $j \in J$, the value of the objective function in the basis $(J \setminus \{j\}) \cup \{i\}$ can be expressed similarly.
- Using the above, $(CGLP)$ can be "solved" in the original tableau with the revised simplex method.

Computations in the general case

- We can generalise the formulae for r_{u_i} and r_{v_i} for general two-term disjunctions:

$$r_{u_i} = \sum_{j \in M_1} \bar{a}_{ij}\hat{s}_j - \sigma(1+\xi_i) + \hat{s}_i - \frac{\bar{a}_{i0}\omega}{\theta}$$

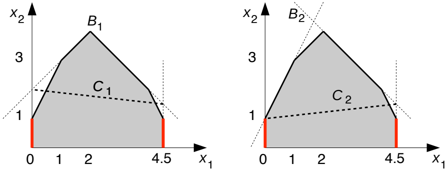
$$r_{v_i} = \sum_{j \in M_1} -\bar{a}_{ij}\hat{s}_j - \sigma(1-\xi_i) + \frac{\bar{a}_{i0}\omega}{\theta}$$

  where ξ_i, σ, θ, and ω are expressions with the $\bar{a}_{ij}$, $\bar{a}_{k_1,j}, \bar{a}_{k_2,j}$ and $\hat{s}_j$.
- We can also generalise the formula for computing the objective function after an exchange of variables in J.

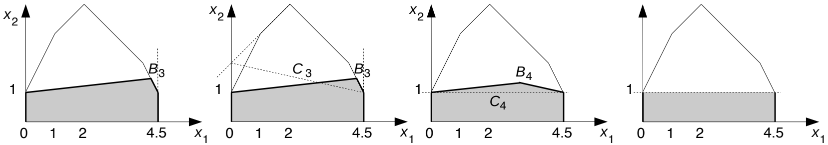
The limitations of the approach

- Consider the problem

$$\max x_2 \ \text{ s.t. } (x_1, x_2) \in P, x_1 \leq 0 \vee x_1 \geq 4.5.$$



(a) (b)



(1) (2) (3) (4)

Computational evaluation

- The LPEC instances

$$
LPCC: \quad \begin{aligned} \min_{(x,y)} \; & c^T x + d^T y \\ \text{s.t.} \; & Ax + By \geq f, \\ & 0 \leq y \perp q + Nx + My \geq 0, \end{aligned}
$$

  where $A \in \mathbb{R}^{k\times n}$, $B \in \mathbb{R}^{k\times m}$, $N \in \mathbb{R}^{m\times n}$, $M \in \mathbb{R}^{m\times m}$, $f \in \mathbb{R}^k$, $c \in \mathbb{R}^n$, and $d \in \mathbb{R}^m$.

- Problem data from
  Hu, J., Mitchell, J.E., Pang, J-S., Bennett, K.P., Kunapuli, G., On the Global Solution of Linear Programs with Linear Complementarity Constraints, SIAM J. Optimization, 19 (2008) 445-471.

LPEC results

Table: General LPCCs with $B \neq 0$, $n = m = 50$, $k = 55$.

#	*lb*	*opt*	Hu et al.		L&P			
			LPs	IPs	nodes	cuts	rounds	time
1	28.7739	29.0501	21	2	8	29	7	0.43
2	36.1885	37.5509	229	9	14	51	8	0.83
3	33.8630	37.0022	4842	696	39	105	10	1.87
4	33.7618	34.2228	102	7	23	43	8	0.55
5	21.4187	22.2835	209	24	16	47	12	0.75
6	29.8919	30.0829	108	13	17	49	10	0.74
7	37.6712	38.0405	92	7	9	27	8	0.38
8	20.8210	22.3969	187	21	19	44	8	0.46
9	39.0227	40.3380	321	14	16	71	8	1.12
10	40.0135	41.3957	190	19	27	66	12	1.20

At most 50 pivots for generating a single cut.

Comparison of cut generation methods

- On these instances, the best is not to generate any cuts: there are about 20% more nodes, but no time is spent on cut generation. Without cuts, the computation time is within 0.2 seconds on all instances.
- L&P cut generation is more than two times faster than CGLP cut generation.

Final remards

- Issues already solved: L&P cut generation, lifting of cuts (not included in this talk)
- Remaining open problems:
  - Strengthening of disjunctive cuts
  - Remark: In the mixed 0-1 programming case, integer modularisation gives cuts at least as strong as Gomory's mixed integer cuts, but without modularisation, the cuts are weak.
- Look at special cases with integer variables
  - The coefficients of the integer variables of a L&P cut $\pi x \geq \pi_0$ can be strengthened by integer modularisation