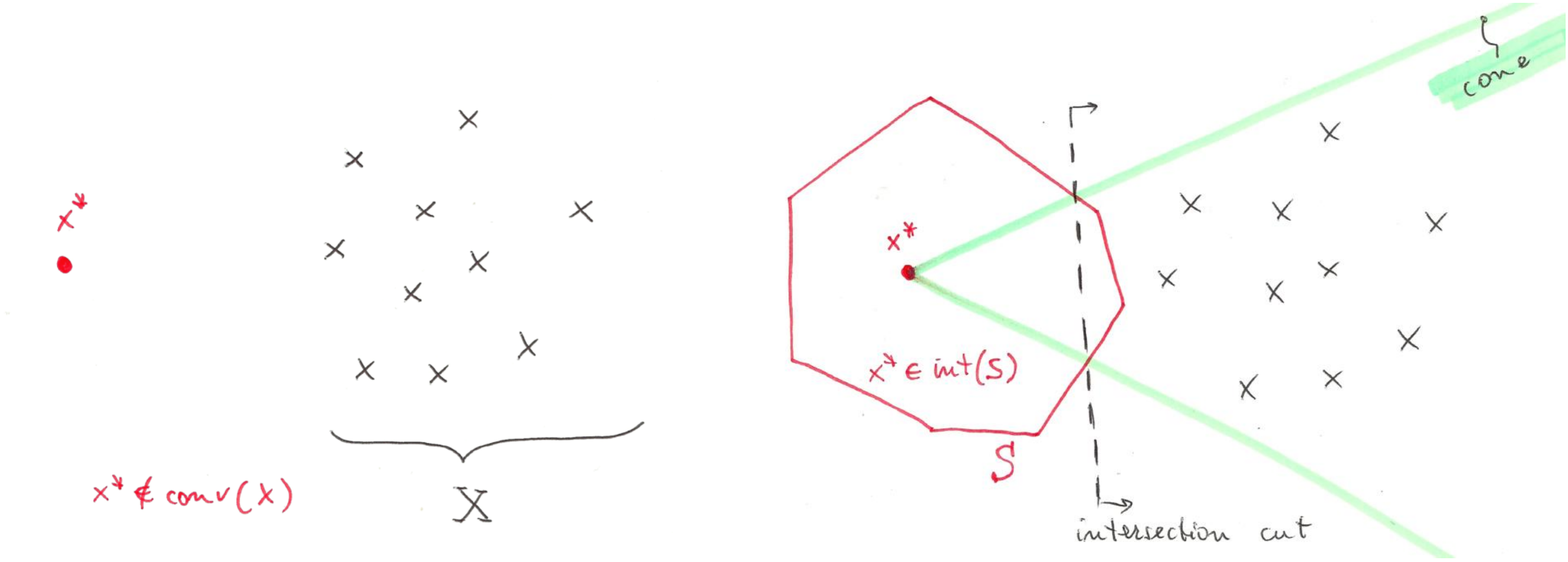
Intersection Cuts (ICs)

- **Intersection cuts** (Balas, 1971): a powerful tool to separate, by a liner cut, a point **x*** from a set **X**



- All you need is (love, but also)
  - a **cone** pointed at **x*** containing all **x ε X**
  - a **convex set S** with x* (but no **x ε X**) in its strict **interior**

Branch or Cut?

Observations:

- If x* **vertex** of an LP relaxation, a suitable cone comes for free from the **LP basis**

- Given an infeasible point x*, any **branching** (linear) disjunction violated by x* implicitly defines a **convex set S** with x* (but no feasible x) in its **interior**

$$\bigvee_{i=1}^{k} (g_i^T x \geq g_{i0}) \quad \rightarrow \quad S = \{x : g_i^T x \leq g_{0i},\ i = 1, \ldots, k\}$$

- Thus, **in principle,** one could always generate an IC instead of branching → not always advisable because of numerical issues, slow convergence, tailing off, cut saturation, etc.

IC separation issues

- IC separation can be probematic, as we need to read the cone rays from the LP tableau → **numerical accuracy** can be a big issue here!

- **Notation**: consider w.l.o.g. an LP in standard form (no var. ub's) and let
  - $\min\{\hat{c}^T\xi : \hat{A}\xi = \hat{b}, \xi \geq 0\}$ be the LP relaxation at a given node
  - $S = \{\xi : g_i^T\xi \leq g_{0i},\ i = 1, \ldots, k\}$ be a given bilinear-free set
  - $\bigvee_{i=1}^{k} (g_i^T\xi \geq g_{i0})$ be the disjunction to be satisfied by all feas. sol.s

Numerically safe ICs

A **single** valid inequality can be obtained by taking, for each variable, the worst LHS Coefficient (and RHS) in each disjunction

$$\bigvee_{i=1}^{k} (g_i^T \xi \geq g_{i0})$$

$$\bigvee_{i=1}^{k} (\overline{g}_i^T \xi \geq \overline{g}_{i0})$$

Better applied to a normalized **reduced form** of each disjunction where the coeff. of all basic var.s is zero (kind of LP reduced costs)

$$\bigvee_{i=1}^{k} (\frac{\overline{g}_i^T}{\overline{g}_{i0}} \xi \geq 1)$$

Algorithm 1: Intersection cut separation

Input : An LP vertex ξ^* along with its a associated LP basis $\hat{B}$;
the feasible-free polyhedron $S = \{\xi : g_i^T \xi \leq g_{0i},\ i = 1, \ldots, k\}$ and the associated valid disjunction $\bigvee_{i=1}^{k} (g_i^T \xi \geq g_{i0})$ whose members are violated by ξ^*;

Output: A valid intersection cut violated by ξ^*;

for $i := 1$ **to** k **do**
  $(\overline{g}_i^T, \overline{g}_{i0}) := (g_i^T, g_{i0}) - u_i^T(\hat{A}, \hat{b})$, where $u_i^T = (g_i)_{\hat{B}}^T \hat{B}^{-1}$
end
for $j := 1$ **to** n **do** $\gamma_j := \max\{\overline{g}_{ij} / \overline{g}_{i0} : i \in \{1, \ldots, k\}\}$;
return *the violated cut* $\gamma^T \xi \geq 1$

Bilevel Optimization

- The general **Bilevel Optimization Problem** (optimistic version) reads:

$$\min_{x \in \mathbb{R}^{n_1}, y \in \mathbb{R}^{n_2}} F(x,y)$$
$$G(x,y) \leq 0$$
$$y \in \arg \min_{y' \in \mathbb{R}^{n_2}} \{ f(x,y') : g(x,y') \leq 0 \}$$

  where *x* var.s only are controlled by the **leader**, while *y* var.s are computed by another player (the **follower**) solving a different problem.

- A very very hard problem even in a **convex setting with continuous var.s** only

- **Convergent** solution algorithms are problematic and typically require additional assumptions (binary/integer var.s or alike)

Reformulation

$$\min_{x \in \mathbb{R}^{n_1}, y \in \mathbb{R}^{n_2}} F(x,y)$$
$$G(x,y) \leq 0$$
$$y \in \arg \min_{y' \in \mathbb{R}^{n_2}} \{f(x,y') : g(x,y') \leq 0\}$$

- By defining the **value function**

$$\Phi(x) = \min_{y \in \mathbb{R}^{n_2}} \{f(x,y) : g(x,y) \leq 0\},$$

the problem can be restated as

$$\begin{aligned} \min F(x,y)& \\ G(x,y) &\leq 0 \\ g(x,y) &\leq 0 \\ f(x,y) &\leq \Phi(x) \\ (x,y) &\in \mathbb{R}^n. \end{aligned}$$

- Dropping the nonconvex condition $f(x,y) \leq \Phi(x)$ one gets the so-called **High Point Relaxation** (HPR)

Mixed-Integer Bilevel Linear Problems

- We will focus the **Mixed-Integer Bilevel Linear** case (MIBLP)

$$
\begin{aligned}
\min F(x,y)&\\
G(x,y) &\le 0\\
g(x,y) &\le 0\\
(x,y) &\in \mathbb{R}^n\\
f(x,y) &\le \Phi(x)\\
x_j \text{ integer},&\quad \forall j \in J_1\\
y_j \text{ integer},&\quad \forall j \in J_2,
\end{aligned}
$$

where *F, G, f* and *g* are **linear** (actually, affine) **functions**

- Note that $f(x,y) \le \Phi(x)$ is **nonconvex** even when all y var.s are continuous

MIBLP statement

- Using standard LP notation, our MIBLP reads

$$
\begin{aligned}
\min_{x,y}\ & c_x^T x + c_y^T y \\
& G_x x + G_y y \leq q \\
& Ax + By \leq b \\
& l \leq y \leq u \\
& x_j \text{ integer}, \quad \forall j \in J_x \\
& y_j \text{ integer}, \quad \forall j \in J_y \\
& d^T y \leq \Phi(x)
\end{aligned}
$$

where for a given $x = x^*$ one computes the value function by solving the following MILP:

$$\Phi(x^*) := \min_{y \in \mathbb{R}^{n_2}} \{d^T y \ : \ By \leq b - Ax^*, \quad l \leq y \leq u, \quad y_j \text{ integer } \forall j \in J_y\}.$$

Example

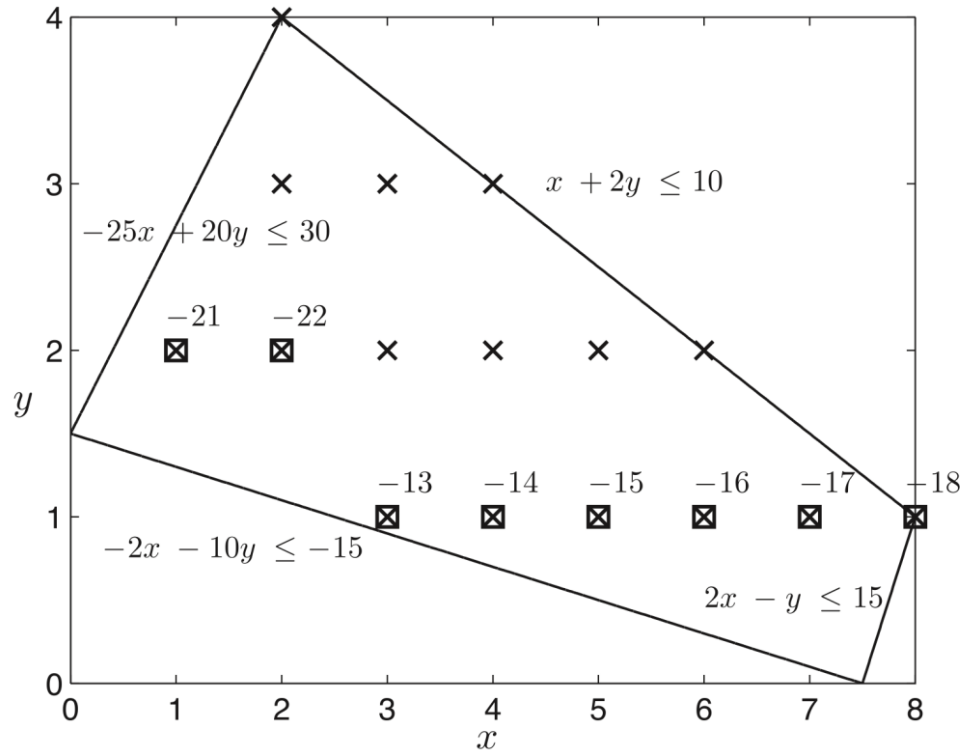
- A notorious example from

J. Moore and J. Bard. The mixed integer linear bilevel programming problem. *Operations Research*, 38(5):911–921, 1990.

$$\min_{x \in \mathbb{Z}} \; -x - 10y$$

$$y \in \arg\min_{y' \in \mathbb{Z}} \{ \; y' :$$

$$\begin{aligned} -25x + 20y' &\leq 30 \\ x + 2y' &\leq 10 \\ 2x - y' &\leq 15 \\ 2x + 10y' &\geq 15 \; \} \end{aligned}$$



where *f(x,y) = y*

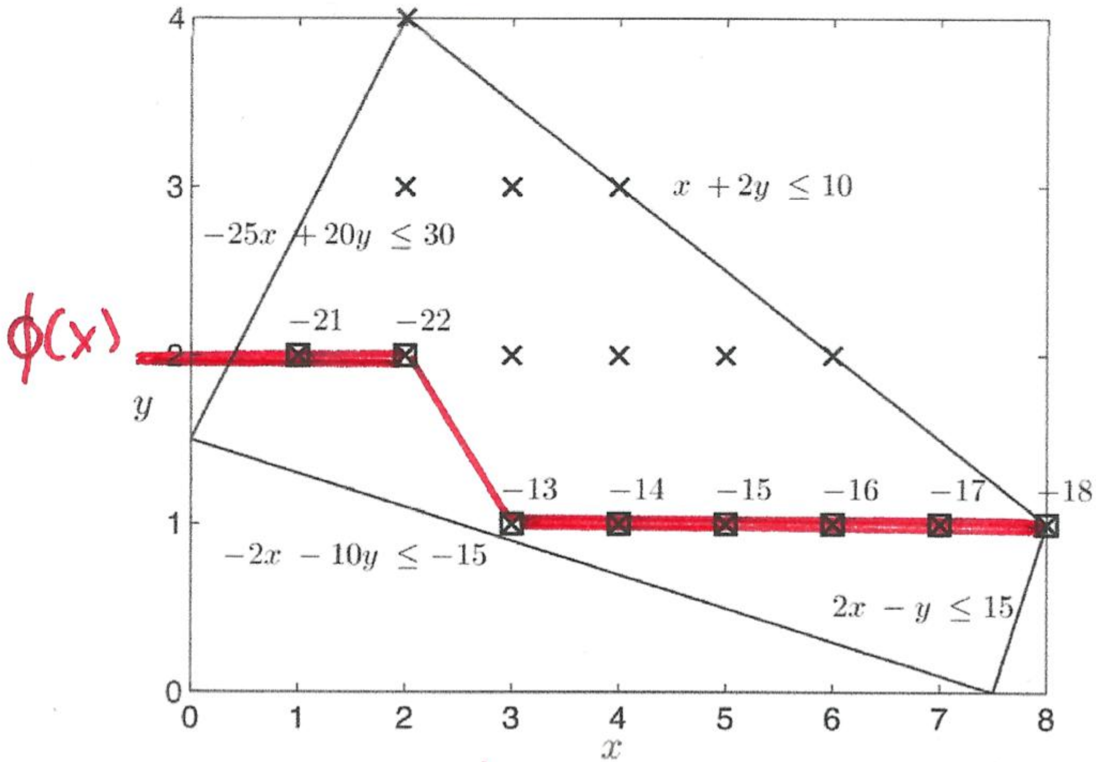
x points of HPR relax.

_____ LP relax. of HPR

Example (cont.d)

Value-function reformulation

$$\begin{aligned} \min\ & -x - 10y \\ & -25x + 20y \leq 30 \\ & x + 2y \leq 10 \\ & 2x - y \leq 15 \\ & -2x - 10y \leq -15 \\ & x, y \in \mathbb{Z} \\ & y \leq \Phi(x) \end{aligned}$$



A MILP-based B&C solver

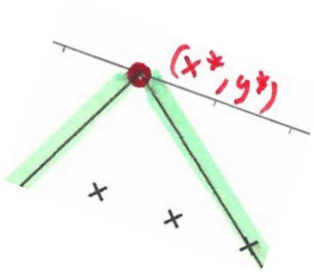
- Suppose you want to apply a **Branch-and-Cut** MILP solver to HPR
- Forget for a moment about internal heuristics (i.e., deactivate all of them), and assume the LP relaxation at each node is solved by the simplex algorithm
- **What do we need to add to the MILP solver to handle a MIBLP?**
- At each node, let ***(x*,y*)*** be the current **LP optimal vertex**:

  ***if (x*,y*)* is fractional → branch as usual**

  ***if (x*,y*)* is integer and** $f(x^*, y^*) \leq \Phi(x^*)$ **→ update the incumbent as usual**

The difficult case

- But, what can we do in third possible case, namely ***(x*,y*)* is integer but** **not bilevel-feasible**, i.e., when $f(x^*, y^*) > \Phi(x^*)$?



- **Question: how can we cut this integer *(x*,y*)* ?**
  Possible answers from the literature

  – If *(x,y)* is restricted to be **binary**, add **a no-good cut** requiring to flip at least one variable w.r.t. *(x*,y*)* or w.r.t. *x**

  – If *(x,y)* is restricted to be **integer** and all MILP coeff.s are integer, add a cut requiring a slack of 1 for the sum of all the inequalities that are tight at *(x*,y*)*

  – Are weak conditions as they do not addresses the **reason of infeasibility** by trying to enforce $f(x^*, y^*) \le \Phi(x^*)$ somehow

Use Intersection Cuts!

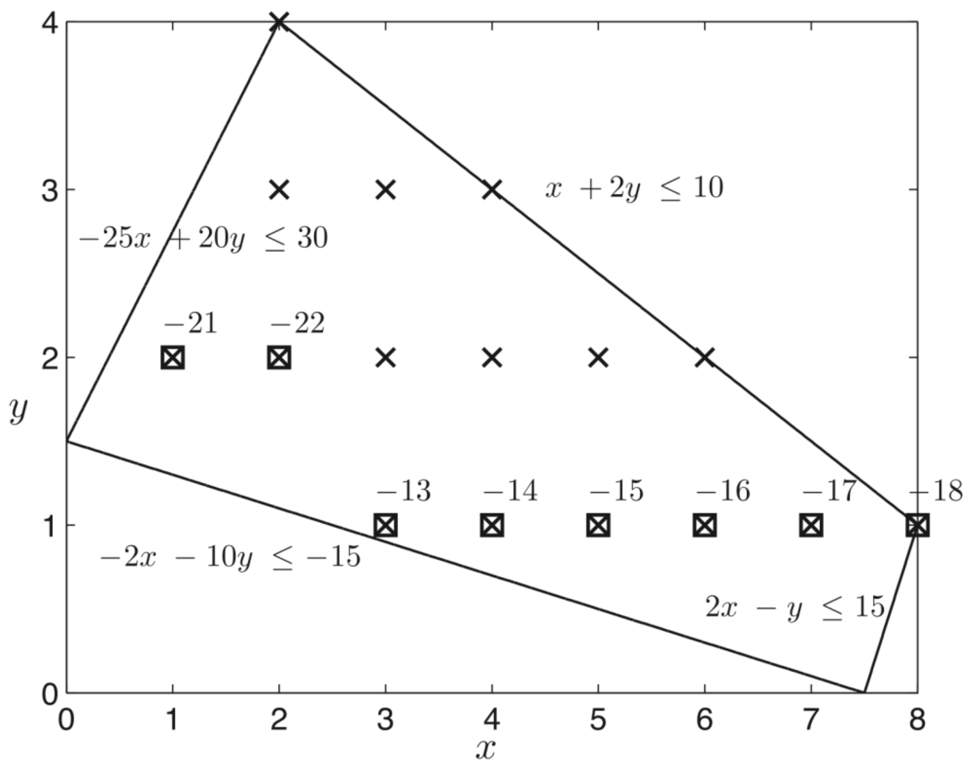
- Our idea is first illustrated on the Moore&Bard example

$$\min_{x \in \mathbb{Z}} \; -x - 10y$$
$$y \in \arg\min_{y' \in \mathbb{Z}} \{ \; y' :$$
$$\begin{aligned} -25x + 20y' &\leq 30 \\ x + 2y' &\leq 10 \\ 2x - y' &\leq 15 \\ 2x + 10y' &\geq 15 \; \} \end{aligned}$$

where *f(x,y) = y*

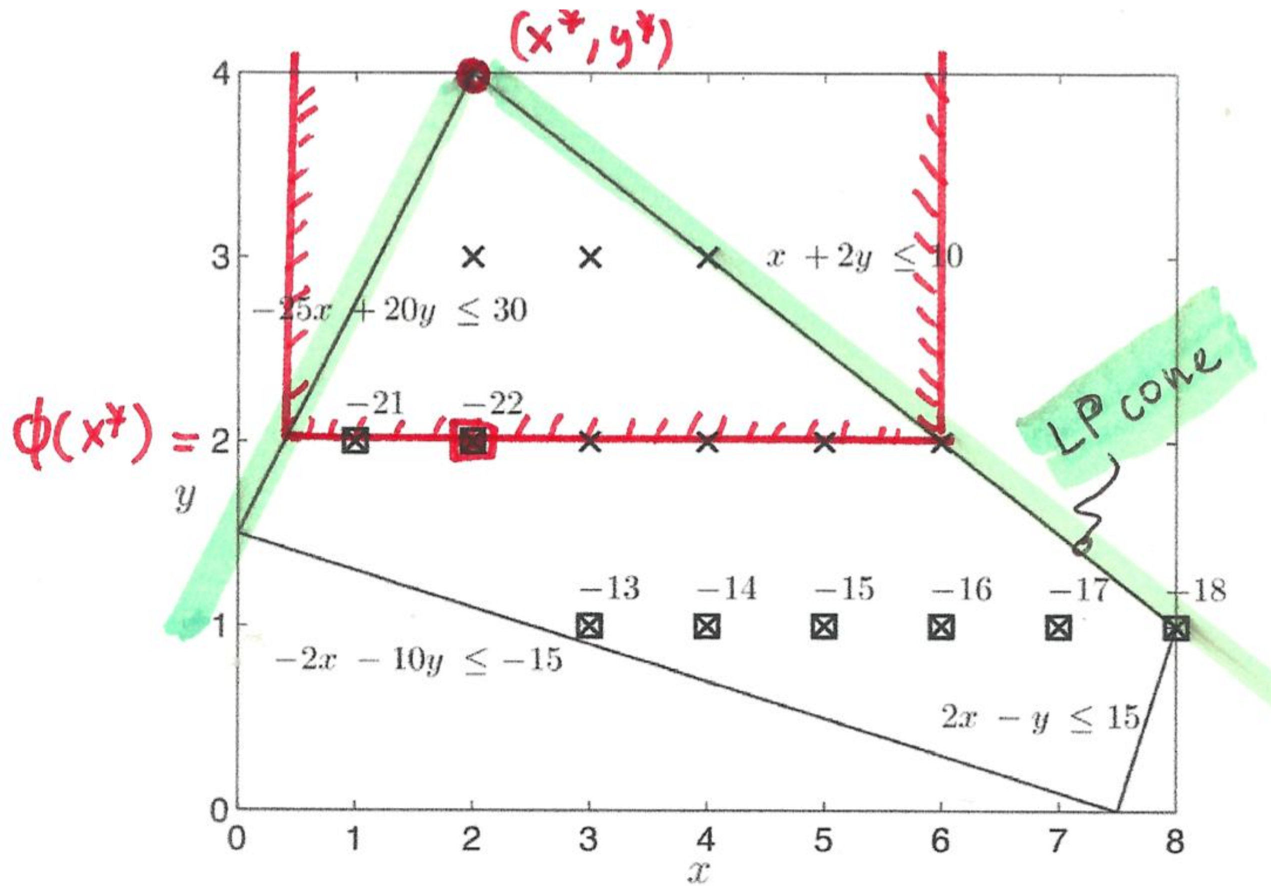
x points of HPR relax.

_____ LP relax. of HPR



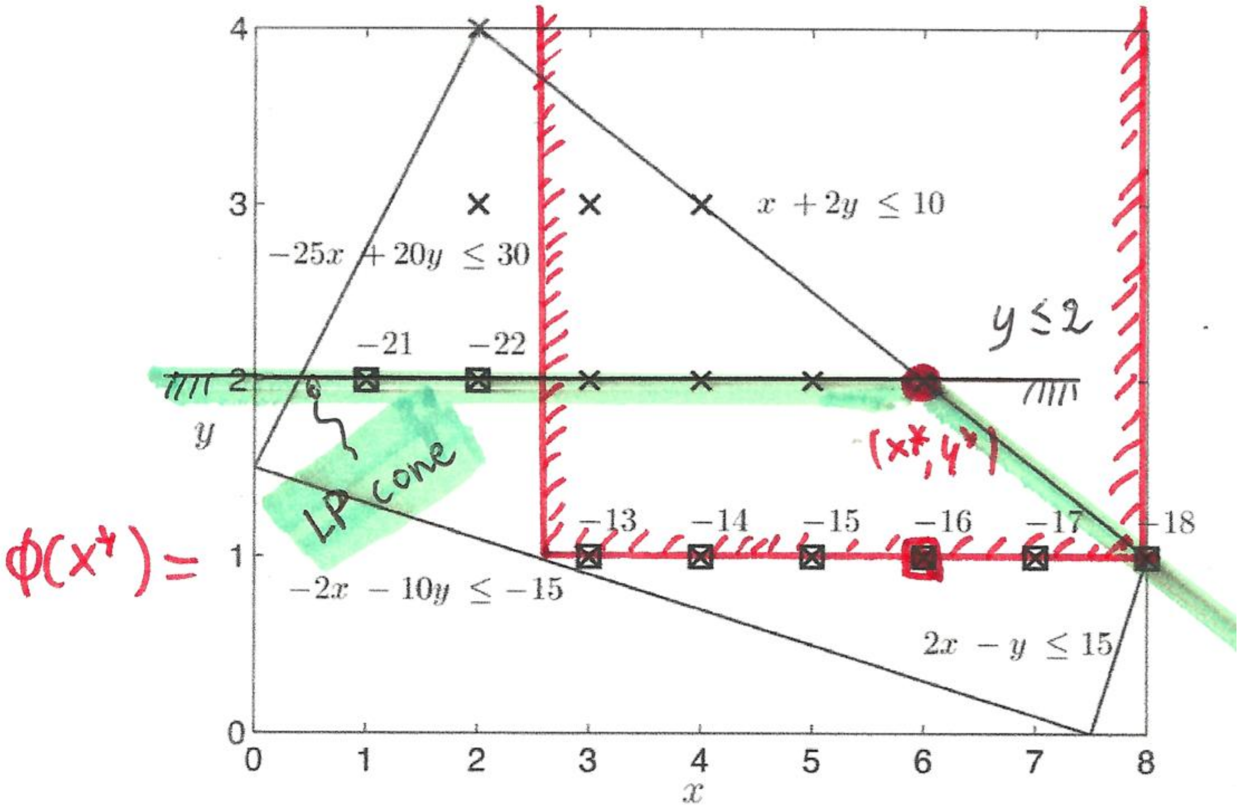
Define a suitable bilevel-free set

- Take the LP vertex *(x*,y*) = (2,4)* → *f(x*,y*) = y* = 4 > Phi(x*) = 2*



Intersection cut

- We can therefore generate the intersection cut *y ≤ 2* and repeat



A basic bilevel-free set

Lemma 1. *For any feasible solution $\hat{y}$ of the follower, the set*

$$S(\hat{y}) = \{(x, y) \in \mathbb{R}^n : f(x, y) \geq f(x, \hat{y}),\ g(x, \hat{y}) \leq 0\} \tag{10}$$

does not contain any bilevel-feasible point in its interior.

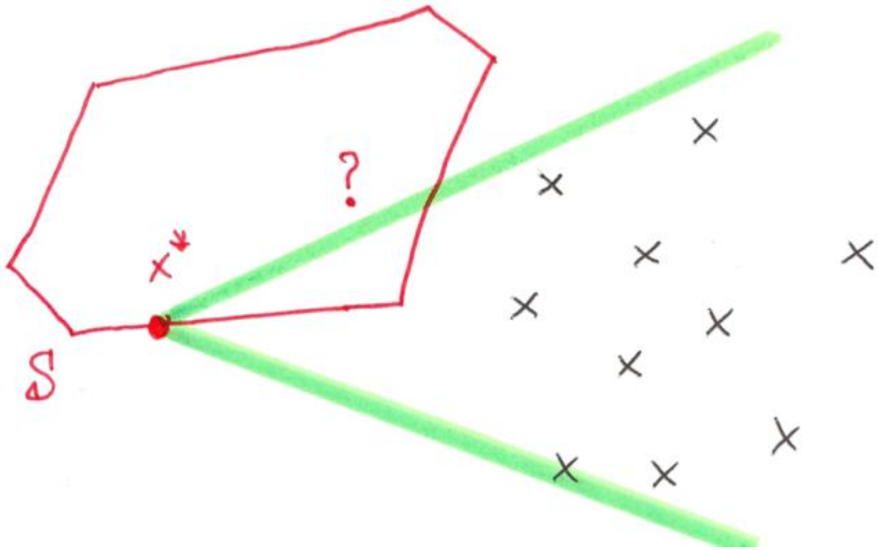
- **Note**: $S(\hat{y})$ is a convex set (actually, a **polyhedron**) when *f* and *g* are affine functions, i.e., in the MIBLP case

- **Separation algorithm**: given an optimal <u>vertex</u> *(x*,y*)* of the LP relaxation of HPR
  - Solve the follower for *x*=*x** and get an optimal sol., say $\hat{y}$
  - **if** *(x*,y*)* <u>strictly</u> inside $S(\hat{y})$ **then**

    generate a violated IC using the LP-cone pointed at *(x*,y*)* together with the bilevel-free set $S(\hat{y})$

A technical issue…

- However, the above does not lead to a proper MILP algorithm as a **bilevel-infeasible** integer vertex ***(x*,y*)*** can be on the **frontier** of the bilevel-free set *S,* so we cannot be sure to cut it by using our IC's



- We need to define the bilevel-free set in a **more clever way** if we want be sure to cut *(x*,y*)*

An enlarged bilevel-free set

- **Assuming *g(x,y)* is integer for all integer HPR solutions**, one can "move apart" the frontier of $S(\hat{y})$ so as be sure that vertex *(x*,y*)* belongs to its interior

Theorem 1. *Assume that $g(x,y)$ is integer for all HPR solutions (x,y). Then, for any feasible solution $\hat{y}$ of the follower, the extended set*

$$S^+(\hat{y}) = \{(x,y) \in \mathbb{R}^n : f(x,y) \geq f(x,\hat{y}),\ g(x,\hat{y}) \leq 1\} \tag{11}$$

does not contain any bilevel-feasible point in its interior, where 1 *denotes a vector of all one's.*

- The corresponding IC is **always violated** by *(x*,y*)* → IC separation to be implemented in a lazy constraint/usercut callback to produce a (locally valid) violated cut → **B&C solver for MIBLP**

- Note: **alternative bilevel-free sets** can be defined to produce hopefully deeper ICs

Conclusions

- Mixed-Integer Bilevel Linear Programming is a **MILP** plus additional constr.s
- **Intersection cuts** can produce valuable information at the B&B nodes
- Sound MIBLP **heuristics**, **preprocessing** etc. (not discussed here) available
- Many instances from the literature can be **solved in a satisfactory way**
- **Binary** code available (ask Markus Sinnl for a free license)

Slides http://www.dei.unipd.it/~fisch/papers/slides/

Reference papers:

M. Fischetti, I. Ljubic, M. Monaci, M. Sinnl, "Intersection cuts for bilevel optimization", in Integer Programming and Combinatorial Optimization: 18th International Conference, IPCO 2016 Proceedings, 77-88, *Mathematical Programming* 172(1), 77-103, 2018

M. Fischetti, I. Ljubic, M. Monaci, M. Sinnl, "A new general-purpose algorithm for mixed-integer bilevel linear program", *Operations Research* 63 (7), 2146-2162, 2017.

M. Fischetti, I. Ljubic, M. Monaci, M. Sinnl, "Interdiction Games and Monotonicity", *INFORMS Journal on Computing* 31(2), 390-410, 2019.