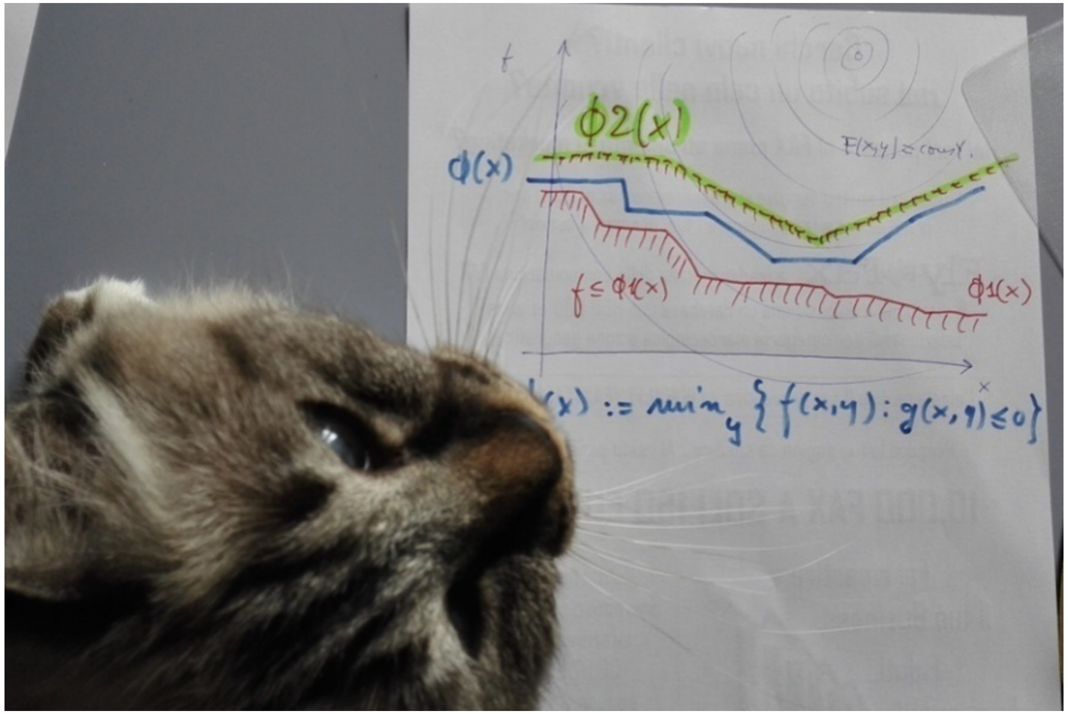
Implementing a B&C algorithm for Mixed-Integer Bilevel Linear Programming

Matteo Fischetti, University of Padova



Bilevel Optimization

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

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

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)

Example: 0-1 ILP

- A generic 0-1 ILP can be reformulated as the following **linear & continuos bilevel problem**

$$\min c^T x$$
$$Ax = b$$
$$x \in \{0,1\}^n$$

$$\min c^T x$$
$$Ax = b$$
$$x \in [0,1]^n$$
$$y = 0$$

$$y \in \arg\min_{y'} \{-\sum_{j=1}^{n} y'_j : \; y'_j \le x_j, \; y'_j \le 1 - x_j \quad \forall j = 1, \ldots, n\}$$

Note that y is fixed to 0 but it cannot be removed from the model!

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 **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

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

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

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

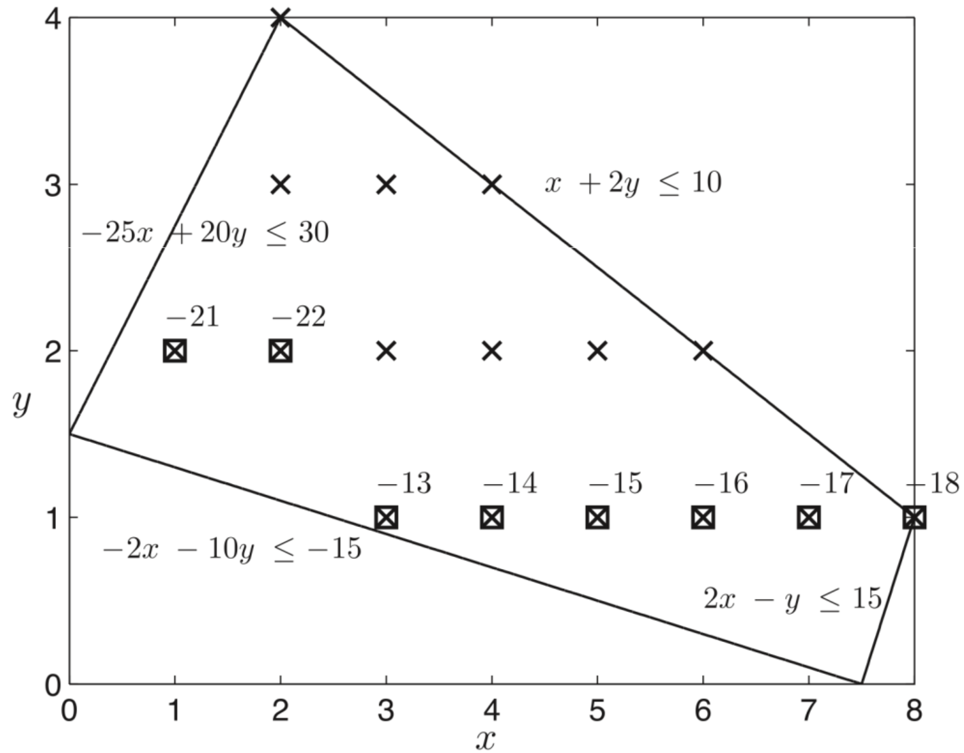
Example

- A notorious example from

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

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

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



where *f(x,y)* = *y*

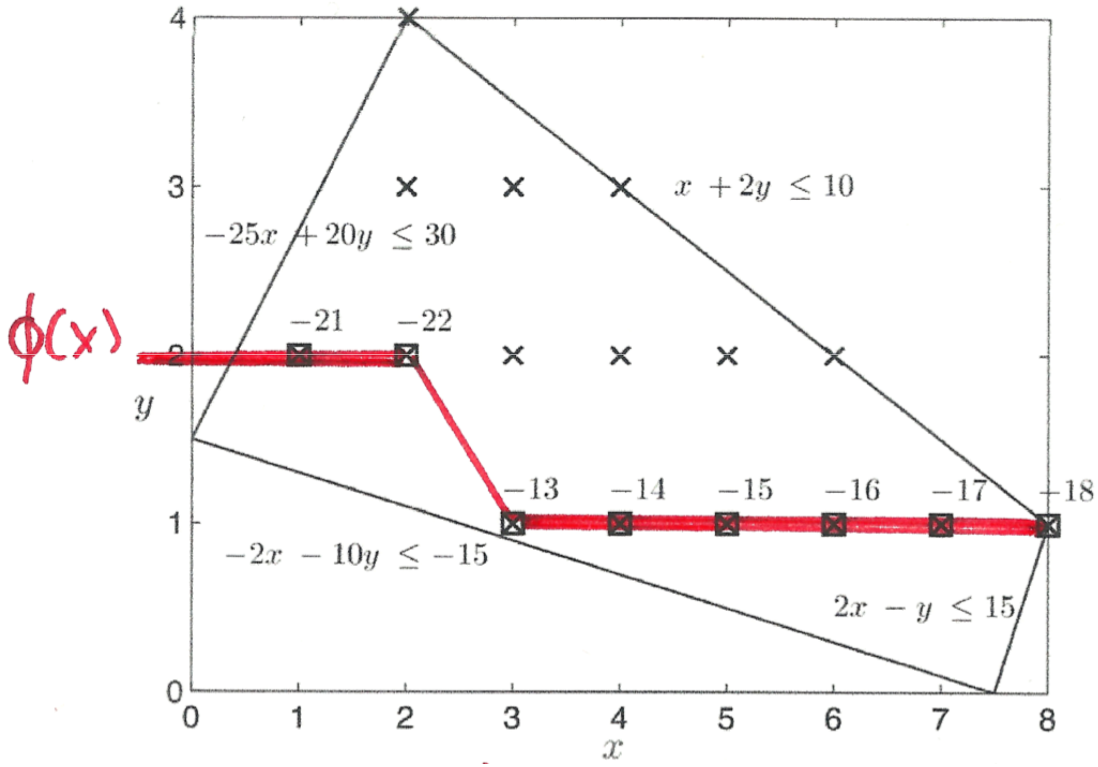
x points of HPR relax.

____ LP relax. of HPR

Example (cont.d)

Value-function reformulation

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



A convergent B&B scheme

Algorithm 2: A basic branch-and-bound scheme for MIBLP

Input : A MIBLP instance satisfying proper assumptions;
Output: An optimal MIBLP solution.

Apply a standard LP-based B&B to HPR, branching as customary on integer-constrained variables x_j and y_j that are fractional at the optimal LP solution; incumbent update is instead inhibited as it requires the bilevel-specific check described below;
for *each unfathomed B&B node where standard branching cannot be performed* **do**
  Let (x^*, y^*) be the integer HPR solution at the current node;
  Compute $\Phi(x^*)$ by solving the follower MILP for $x = x^*$;
  if $d^T y^* \leq \Phi(x^*)$ **then**
    The current solution (x^*, y^*) is bilevel feasible: update the incumbent and fathom the current node
  else
    if *not all variables* x_j *with* $j \in J_F$ *are fixed by branching* **then**
      Branch on any x_j $(j \in J_F)$ not fixed by branching yet, even if x_j^* is integer, so as to reduce its domain in both child nodes
    else
      let $(\hat{x}, \hat{y})$ be an optimal solution of the HPR at the current node amended by the additional restriction $d^T y \leq \Phi(x^*)$;
      Possibly update the incumbent with $(\hat{x}, \hat{y})$, and fathom the current node
    end
  end
end

Here J_F is the set of the leader x-variables appearing in the follower problem, all of which are assumed to be integer constrained (we also exclude HPR unboundedness)

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?**

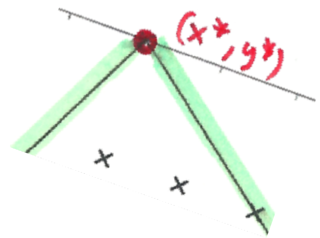
- At each node, let ***(x*,y*)*** be the current **LP optimal vertex**:

  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

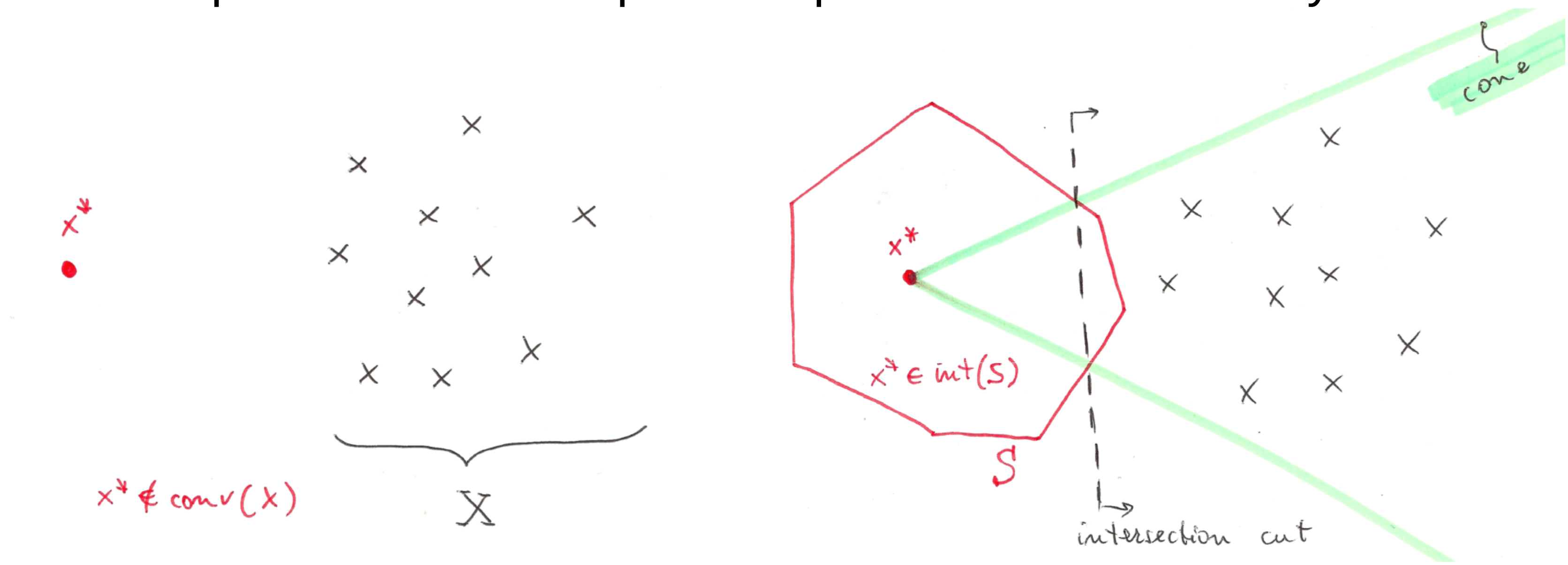
  – If *(x,y)* is restricted to be **binary**, add **a no-good cut** requiring to flip at least one variable w.r.t. *(x*,y*)* or w.r.t. *x**

  – 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*)*

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

Intersection Cuts (ICs)

- Try and use of **intersection cuts** (Balas, 1971) instead
- ICs are a powerful tool to separate a point **x*** from a set **X** by a linear cut



- All you need is
  - a **cone** pointed at **x*** containing all **x ε X**
  - a **convex set S** with x* (but no **x ε X**) in its **interior**
- If x* **vertex** of an LP relaxation, a suitable cone comes for the **LP basis**

ICs for bilevel problems

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

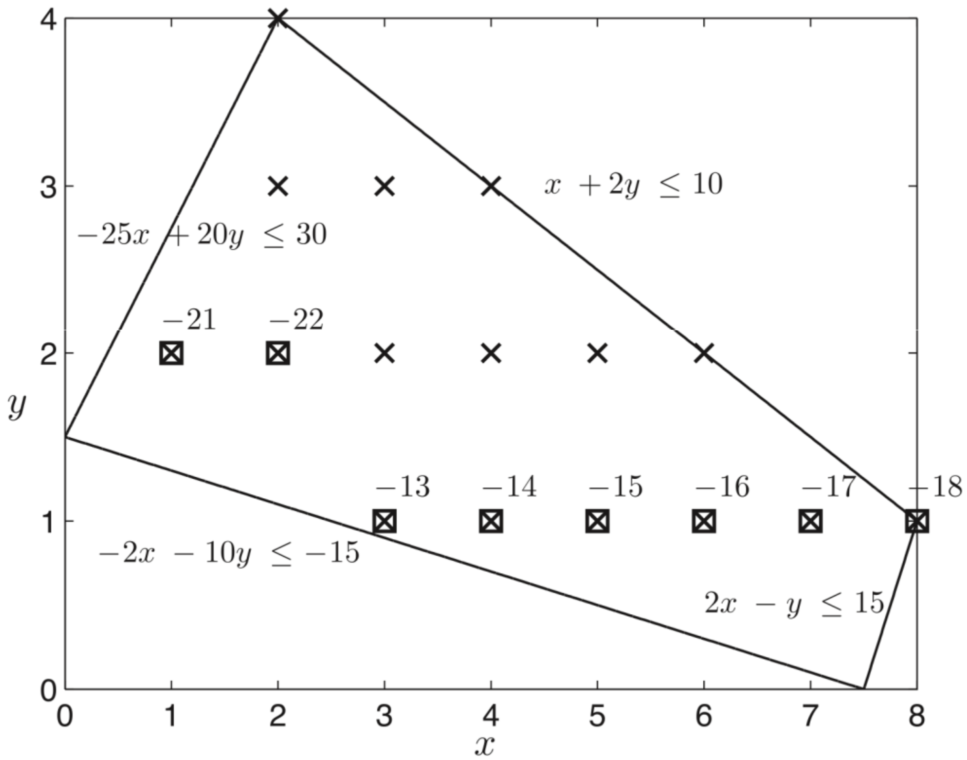
$$\min_{x \in \mathbb{Z}} \; -x - 10y$$

$$y \in \arg\min_{y' \in \mathbb{Z}} \{ \; y' : \quad -25x + 20y' \le 30, \quad x + 2y' \le 10, \quad 2x - y' \le 15, \quad 2x + 10y' \ge 15 \; \}$$

where *f(x,y) = y*

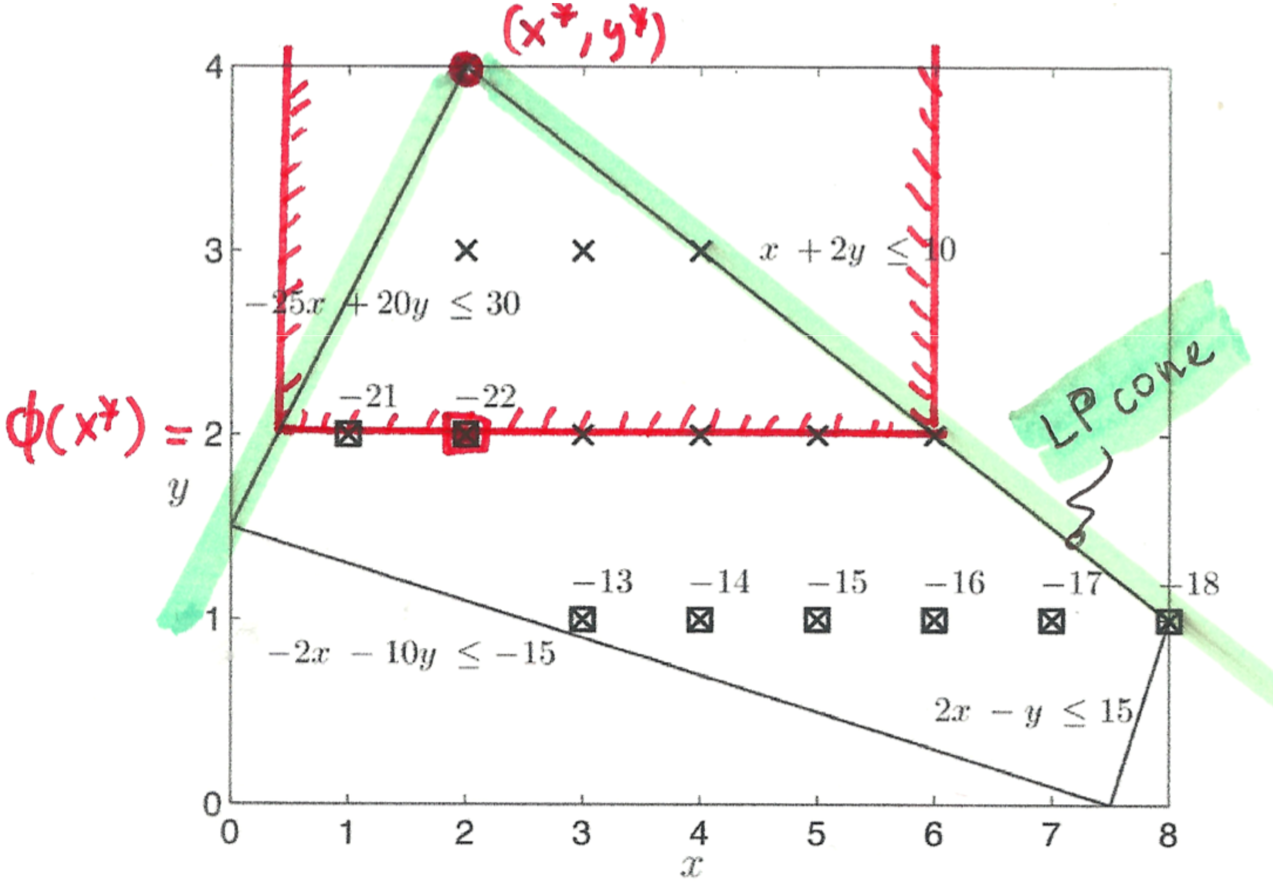
x points of HPR relax.

____ LP relax. of HPR



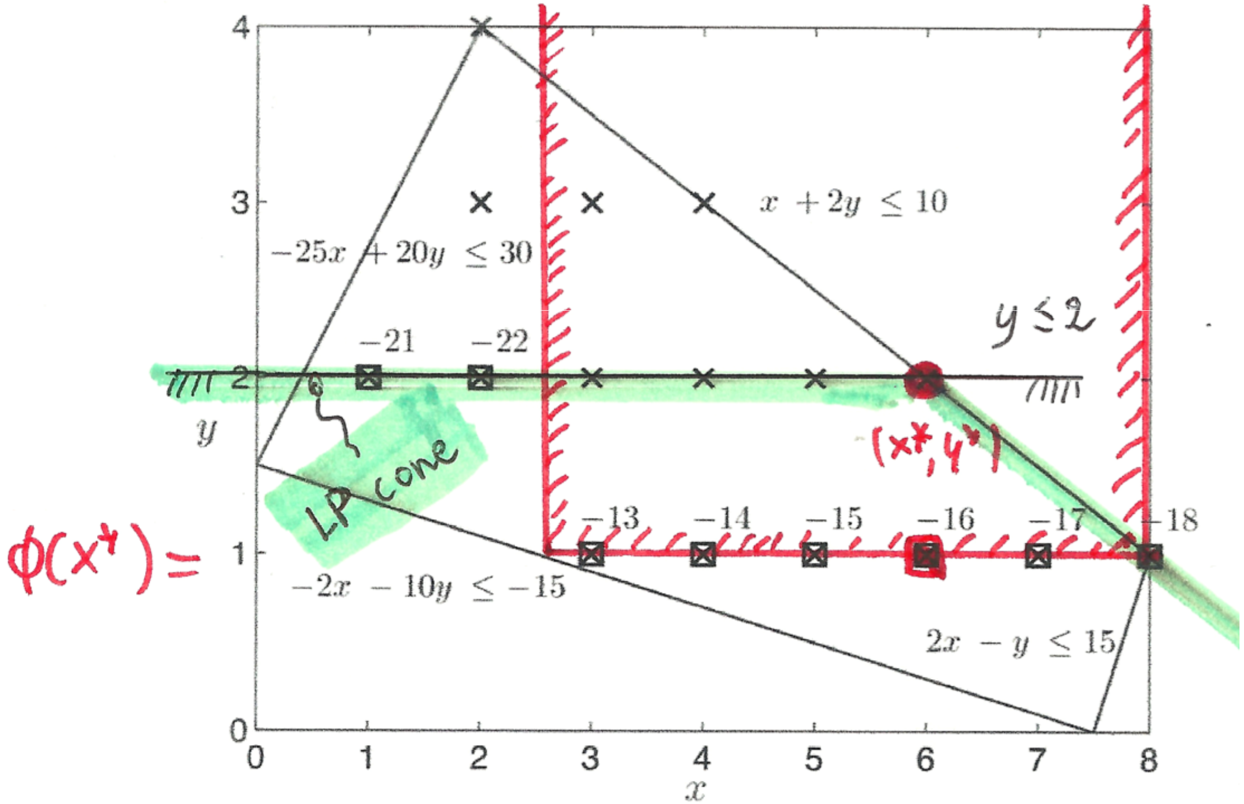
Define a suitable bilevel-free set

- 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.

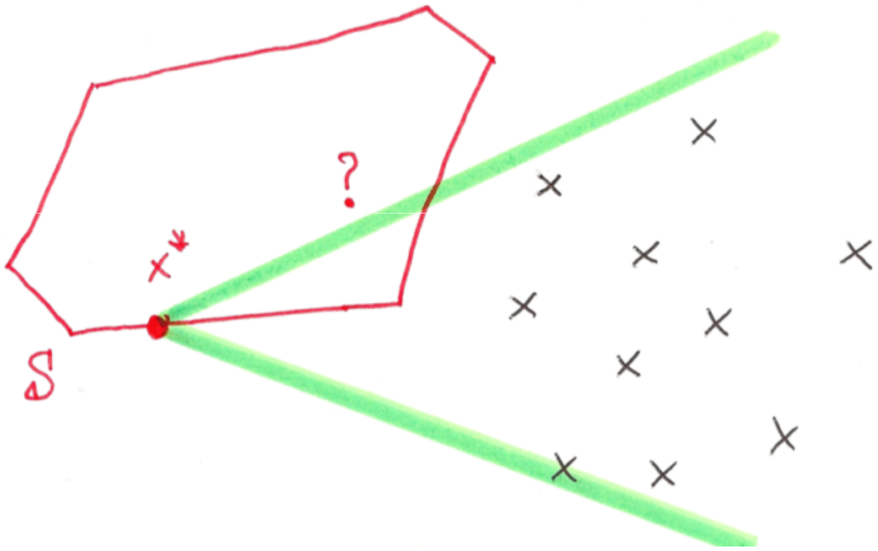
- **Note**: $S(\hat{y})$ is a convex set (actually, a **polyhedron**) when *f* and *g* are affine functions, i.e., in the MIBLP case

- **Separation algorithm**: given an optimal vertex *(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*)* strictly inside $S(\hat{y})$ **then**

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

It looks simple, but …

- 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*

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

does not contain any bilevel-feasible point in its interior, where $\mathbf{1}$ *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

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!

- For **MILP**s, ICs like Gomory cuts are **not mandatory** (so we can skip their generation in case of numerical problems), but for **MIBLP**s they are more instrumental **#SeparateOrPerish**

- **Notation change**: let $\xi = (x, y) \in \mathbb{R}^n$

  $\min\{\hat{c}^T \xi : \hat{A}\xi = \hat{b}, \xi \geq 0\}$ be the LP relaxation at a given node

  $S = \{\xi : g_i^T \xi \leq g_{0i},\ i = 1, \ldots, k\}$ be the bilevel-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

Algorithm 1: Intersection cut separation

Input : An LP vertex ξ^* along with its 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** $\gamma_j := \max\{\, g_{ij}/g_{i0} : i \in \{1, \ldots, k\}\,\}$;
if $\gamma \geq 0$ **then**
  for $j := 1$ **to** n **do**
    if ξ_j *is integer constrained* **then** $\gamma_j := \min\{\gamma_j, 1\}$;
  end
end
return *the violated cut* $\gamma^T \xi \geq 1$

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**

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, 2016 (to appear in *Mathematical Programming*)

M. Fischetti, I. Ljubic, M. Monaci, M. Sinnl, "A new general-purpose algorithm for mixed-integer bilevel linear program", to appear in *Operations Research*.

M. Fischetti, I. Ljubic, M. Monaci, M. Sinnl, "Interdiction Games and Monotonicity", Tech. Report 2016 (submitted)