New Branch-and-Cut Algorithms for Mixed-Integer Bilevel Linear Programs

I. Ljubić

ESSEC Business School of Paris, France

Séminaire Parisien d'Optimisation 2018, June 11, Paris

Bilevel Optimization

General bilevel optimization problem

$$\min_{x\in\mathbb{R}^{n_1}, y\in\mathbb{R}^{n_2}} F(x,y) \tag{1}$$

$$G(x,y) \leq 0 \tag{2}$$

$$y \in \arg\min_{y'\in\mathbb{R}^{n_2}} \{ f(x,y') : g(x,y') \leq 0 \} \tag{3}$$

- Stackelberg game: two-person sequential game
- Leader takes follower's optimal reaction into account
- $N_x = \{1,\ldots,n_1\}$, $N_y = \{1,\ldots,n_2\}$
- $n = n_1 + n_2$: total number of decision variables

Bilevel Optimization

General bilevel optimization problem

$$\min_{x \in \mathbb{R}^{n_1}, y \in \mathbb{R}^{n_2}} F(x, y) \quad (1)$$

$$G(x, y) \leq 0 \quad (2)$$

$$y \in \arg\min_{y' \in \mathbb{R}^{n_2}} \{ f(x, y') : g(x, y') \leq 0 \} \quad (3)$$

Leader

- Stackelberg game: two-person sequential game
- Leader takes follower's optimal reaction into account
- $N_x = \{1, \ldots, n_1\}$, $N_y = \{1, \ldots, n_2\}$
- $n = n_1 + n_2$: total number of decision variables

Bilevel Optimization

General bilevel optimization problem

$$\min_{x \in \mathbb{R}^{n_1}, y \in \mathbb{R}^{n_2}} F(x, y) \quad (1)$$

$$G(x, y) \leq 0 \quad (2)$$

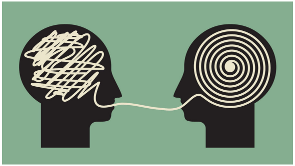
$$y \in \arg \min_{y' \in \mathbb{R}^{n_2}} \{ f(x, y') : g(x, y') \leq 0 \} \quad (3)$$

Leader

Follower

- Stackelberg game: two-person sequential game
- Leader takes follower's optimal reaction into account
- $N_x = \{1, \dots, n_1\}$, $N_y = \{1, \dots, n_2\}$
- $n = n_1 + n_2$: total number of decision variables

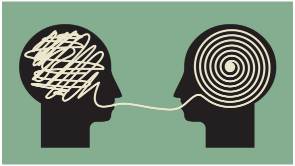
Optimistic vs Pessimistic Solution



The Stackelberg game under:

- **Perfect information:** both agents have perfect knowledge of each others strategy
- **Rationality:** agents act optimally, according to their respective goals
- **What if there are multiple optimal solutions for the follower?**
  - **Optimistic Solution:** among the follower's solution, the one leading to the best outcome for the leader is assumed
  - **Pessimistic Solution:** among the follower's solution, the one leading to the worst outcome for the leader is assumed

Optimistic vs Pessimistic Solution




The Stackelberg game under:

- **Perfect information:** both agents have perfect knowledge of each others strategy
- **Rationality:** agents act optimally, according to their respective goals
- **What if there are multiple optimal solutions for the follower?**
  - **Optimistic Solution:** among the follower's solution, the one leading to the best outcome for the leader is assumed
  - **Pessimistic Solution:** among the follower's solution, the one leading to the worst outcome for the leader is assumed

Our Focus: Mixed-Integer Bilevel Linear Programs (MIBLP)

$$
\begin{aligned}
\text{(MIBLP)} \quad & \min c_x^T x + c_y^T y && (4)\\
& G_x x + G_y y \le 0 && (5)\\
& y \in \arg\min\{d^T y : Ax + By \le 0, && (6)\\
& \qquad y_j \text{ integer}, \forall j \in J_y\} && (7)\\
& x_j \text{ integer}, \forall j \in J_x && (8)\\
& (x, y) \in \mathbb{R}^n && (9)
\end{aligned}
$$

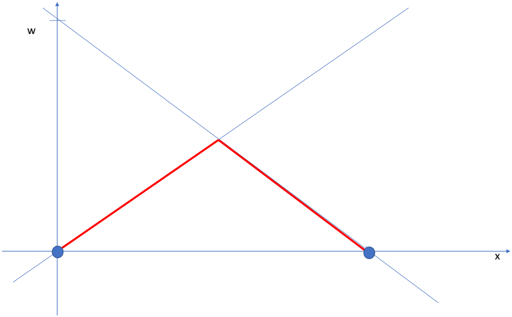
where c_x, c_y, G_x, G_y, A, B are given rational matrices/vectors of appropriate size.

Complexity

Bilevel Linear Programs

Bilevel LPs are strongly NP-hard (Audet et al. [1997], Hansen et al. [1992]).

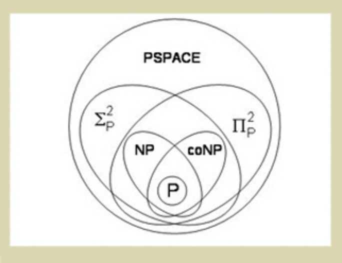
$$\begin{aligned} \min c^T x \\ Ax = b \\ x \in \{0,1\} \end{aligned} \quad \Leftrightarrow \quad \begin{aligned} \min c^T x \\ Ax = b \\ v = 0 \\ v \in \arg\max\{w : w \leq x, w \leq 1 - x, w \geq 0\} \end{aligned}$$



Complexity

Bilevel Mixed-Integer Linear Programs

MIBLP is Σ_2^P-hard (Lodi et al. [2014]): there is no way of formulating MIBLP as a MILP of polynomial size unless the polynomial hierarchy collapses.



Overview

Part I

- Branch-and-cut approach for general Mixed-Integer Bilevel Programs
- Based on intersection cuts

Part II

- Special subfamily: Interdiction-like problems (with monotonicity property)
- Specialized branch-and-cut algorithm based on interdiction cuts
- Examples: Knapsack-Interdiction and Clique-Interdiction

Based on the papers:

Part I

- M. Fischetti, I. Ljubić, M. Monaci, M. Sinnl: On the Use of Intersection Cuts for Bilevel Optimization, Mathematical Programming, to appear, 2018
- M. Fischetti, I. Ljubić, M. Monaci, M. Sinnl: A new general-purpose algorithm for mixed-integer bilevel linear programs, Operations Research 65(6): 1615-1637, 2017

Part II

- M. Fischetti, I. Ljubić, M. Monaci, M. Sinnl: Interdiction Games and Monotonicity, with Application to Knapsack Problems, INFORMS Journal on Computing, to appear, 2018
- F. Furini, I. Ljubić. P. San Segundo, S. Martin: The Maximum Clique Interdiction Game, submitted, 2018

STEP 1: VALUE FUNCTION REFORMULATION

Our Focus: Mixed-Integer Bilevel Linear Programs (MIBLP)

Value Function Reformulation:

$$\text{(MIBLP)} \quad \min c_x^T x + c_y^T y \tag{10}$$
$$G_x x + G_y y \leq 0 \tag{11}$$
$$Ax + By \leq 0 \tag{12}$$
$$(x, y) \in \mathbb{R}^n \tag{13}$$
$$d^T y \leq \Phi(x) \tag{14}$$
$$x_j \text{ integer}, \quad \forall j \in J_x \tag{15}$$
$$y_j \text{ integer}, \quad \forall j \in J_y \tag{16}$$

where $\Phi(x)$ is non-convex, non-continuous:

$$\Phi(x) = \min\{d^T y : Ax + By \leq 0, \quad y_j \text{ integer}, \forall j \in J_y\}$$

- dropping $d^T y \leq \Phi(x) \rightarrow$ **High Point Relaxation** (HPR) which is a MILP $\rightarrow$ we can use MILP solvers with all their tricks
- let $\overline{\text{HPR}}$ be LP-relaxation of HPR

Our Focus: Mixed-Integer Bilevel Linear Programs (MIBLP)

Value Function Reformulation:

I am a Mixed-Integer Linear Program (MILP) 😄

$$\text{(HPR)} \quad \min c_x^T x + c_y^T y \tag{10}$$
$$G_x x + G_y y \leq 0 \tag{11}$$
$$Ax + By \leq 0 \tag{12}$$
$$(x, y) \in \mathbb{R}^n \tag{13}$$
$$\tag{14}$$
$$x_j \text{ integer}, \quad \forall j \in J_x \tag{15}$$
$$y_j \text{ integer}, \quad \forall j \in J_y \tag{16}$$

where $\Phi(x)$ is non-convex, non-continuous:

$$\Phi(x) = \min\{d^T y : Ax + By \leq 0, \quad y_j \text{ integer}, \forall j \in J_y\}$$

- dropping $d^T y \leq \Phi(x) \rightarrow$ **High Point Relaxation** (HPR) which is a MILP $\rightarrow$ we can use MILP solvers with all their tricks
- let $\overline{\text{HPR}}$ be LP-relaxation of HPR

Our Focus: Mixed-Integer Bilevel Linear Programs (MIBLP)

Value Function Reformulation:

$$(\overline{\text{HPR}}) \quad \min c_x^T x + c_y^T y \tag{10}$$
$$G_x x + G_y y \leq 0 \tag{11}$$
$$Ax + By \leq 0 \tag{12}$$
$$(x, y) \in \mathbb{R}^n \tag{13}$$
$$\tag{14}$$
$$\tag{15}$$
$$\tag{16}$$

where $\Phi(x)$ is non-convex, non-continuous:

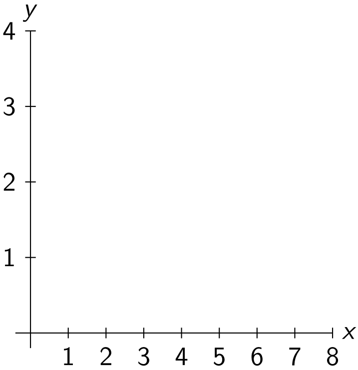
$$\Phi(x) = \min\{d^T y : Ax + By \leq 0, \quad y_j \text{ integer}, \forall j \in J_y\}$$

- dropping $d^T y \leq \Phi(x) \rightarrow$ **High Point Relaxation** (HPR) which is a MILP $\rightarrow$ we can use MILP solvers with all their tricks
- let $\overline{\text{HPR}}$ be LP-relaxation of HPR

Example

- notorious example from Moore and Bard [1990]
- HPR
- value-function reformulation

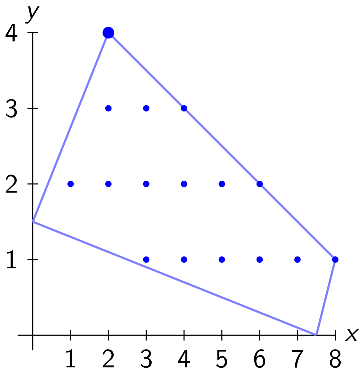
$$
\begin{aligned}
\min_{x \in \mathbb{Z}} \; & -x - 10y \\
y \in \arg\min_{y' \in \mathbb{Z}} \{ y' : & \\
-25x + 20y' & \leq 30 \\
x + 2y' & \leq 10 \\
2x - y' & \leq 15 \\
2x + 10y' & \geq 15 \}
\end{aligned}
$$



Example

- notorious example from Moore and Bard [1990]
- HPR
- value-function reformulation

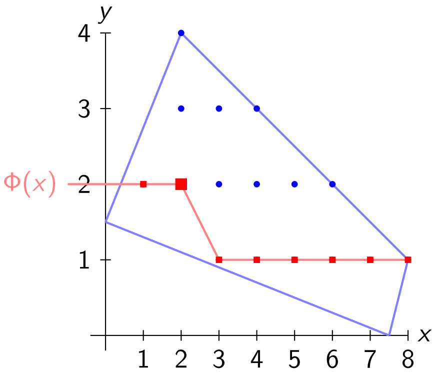
$$\min_{x,y \in \mathbb{Z}} -x - 10y$$
$$-25x + 20y \geq 30$$
$$x + 2y \leq 10$$
$$2x - y \leq 15$$
$$2x + 10y \geq 15$$



Example

- notorious example from Moore and Bard [1990]
- HPR
- value-function reformulation

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



General Idea

General Procedure

- Start with the HPR- (or $\overline{\text{HPR}}$-)relaxation
- Get rid of bilevel infeasible solutions on the fly
- Apply branch-and-bound or branch-and-cut algorithm

There are some unexpected difficulties along the way...

- Optimal solution can be unattainable
- HPR can be unbounded

(Un)expected Difficulties: **Unattainable Solutions**

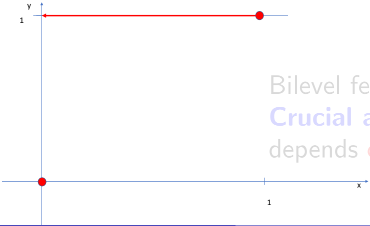
Example from Köppe et al. [2010]

Continuous variables in the leader, integer variables in the follower $\Rightarrow$ optimal solution may be **unattainable**

$$\inf_{x,y} \quad x - y$$
$$0 \leq x \leq 1$$
$$y \in \arg\min_{y'} \{y' : y' \geq x, 0 \leq y' \leq 1, y' \in \mathbb{Z}\}.$$

Equivalent to

$$\inf_{x} \{x - \lceil x \rceil : 0 \leq x \leq 1\}$$



Bilevel feasible set is neither convex nor closed.
Crucial assumption for us: follower subproblem depends only on integer leader variables $J_F \subseteq J_x$.

(Un)expected Difficulties: **Unattainable Solutions**

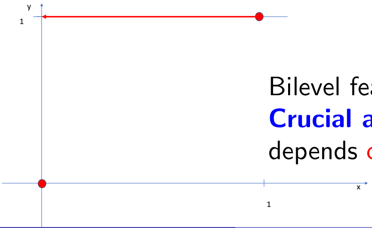
Example from Köppe et al. [2010]

Continuous variables in the leader, integer variables in the follower $\Rightarrow$ optimal solution may be **unattainable**

$$\inf_{x,y} \quad x - y$$
$$0 \leq x \leq 1$$
$$y \in \arg\min_{y'} \{y' : y' \geq x, 0 \leq y' \leq 1, y' \in \mathbb{Z}\}.$$

Equivalent to

$$\inf_{x} \{x - \lceil x \rceil : 0 \leq x \leq 1\}$$



Bilevel feasible set is neither convex nor closed.
Crucial assumption for us: follower subproblem depends only on integer leader variables $J_F \subseteq J_x$.

(Un)expected Difficulties: **Unbounded HPR-Relaxation**

Example from Xu and Wang [2014]

Unboundness of HPR-relaxation does not allow to draw conclusions on the optimal solution of MIBLP

- **unbounded**
- **infeasible**
- **admit an optimal solution**

$$\begin{aligned} \max_{x,y} \quad & x + y \\ & 0 \le x \le 2 \\ & x \in \mathbb{Z} \\ & y \in \arg\max_{y'} \{ d \cdot y' : y' \ge x, y' \in \mathbb{Z} \}. \end{aligned}$$

$$\begin{aligned} \max_{x,y} \quad & x + y \\ & 0 \le x \le 2 \\ & y \ge x \\ & x, y \in \mathbb{Z} \end{aligned}$$

$d = 1 \quad \Rightarrow \Phi(x) = \infty$ (MIBLP infeasible)

$d = 0 \quad \Rightarrow \Phi(x)$ feasible for all $y \in \mathbb{Z}$ (MIBLP unbounded)

$d = -1 \quad \Rightarrow x^* = 2, y^* = 2$ (optimal MIBLP solution)

(Un)expected Difficulties: **Unbounded HPR-Relaxation**

Example from Xu and Wang [2014]

Unboundness of HPR-relaxation does not allow to draw conclusions on the optimal solution of MIBLP

- **unbounded**
- **infeasible**
- **admit an optimal solution**

$$\max_{x,y} \quad x + y$$
$$0 \le x \le 2$$
$$x \in \mathbb{Z}$$
$$y \in \arg\max_{y'}\{d \cdot y' : y' \ge x, y' \in \mathbb{Z}\}.$$

$$\max_{x,y} \quad x + y$$
$$0 \le x \le 2$$
$$y \ge x$$
$$x, y \in \mathbb{Z}$$

$d = 1 \quad \Rightarrow \Phi(x) = \infty$ (MIBLP infeasible)

$d = 0 \quad \Rightarrow \Phi(x)$ feasible for all $y \in \mathbb{Z}$ (MIBLP unbounded)

$d = -1 \quad \Rightarrow x^* = 2, y^* = 2$ (optimal MIBLP solution)

STEP 2: BRANCH-AND-CUT ALGORITHM

Assumption

All the integer-constrained variables x and y have finite lower and upper bounds both in HPR and in the follower MILP.

Assumption

Continuous leader variables x_j (if any) do not appear in the follower problem.

If for all HPR solutions, the follower MILP is unbounded $\Rightarrow$ MIBLP is infeasible. Preprocessing (solving a single LP) allows to check this. Hence:

Assumption

For an arbitrary HPR solution, the follower MILP is well defined.

Our Goal: Design MILP-based solver for MIBLP

For the rest of presentation: Assume HPR value is bounded.

Our Goal

solve MIBLP by using a standard **simplex-based branch-and-cut** algorithm;
enforce $d^T y \leq \Phi(x)$ on the fly, by adding cutting planes

- given **optimal vertex** (x^*, y^*) of $\overline{\text{HPR}}$
  - (x^*, y^*) infeasible for HPR (i.e., fractional) $\rightarrow$ branch as usual
  - (x^*, y^*) feasible for HPR and $f(x^*, y^*) \leq \Phi(x^*)$ $\rightarrow$ update the incumbent as usual
  - (x^*, y^*) feasible for HPR and $f(x^*, y^*) > \Phi(x^*)$, i.e., **bilevel-infeasible** $\rightarrow$ we need to do something!

Our Goal: Design MILP-based solver for MIBLP

For the rest of presentation: Assume HPR value is bounded.

Our Goal

solve MIBLP by using a standard **simplex-based branch-and-cut** algorithm; enforce $d^T y \leq \Phi(x)$ on the fly, by adding cutting planes

- given **optimal vertex** (x^*, y^*) of $\overline{\text{HPR}}$
  - (x^*, y^*) infeasible for HPR (i.e., fractional) $\rightarrow$ branch as usual
  - (x^*, y^*) feasible for HPR and $f(x^*, y^*) \leq \Phi(x^*)$ $\rightarrow$ update the incumbent as usual
  - (x^*, y^*) feasible for HPR and $f(x^*, y^*) > \Phi(x^*)$, i.e., **bilevel-infeasible** $\rightarrow$ we need to do something!

Our Goal: Design MILP-based solver for MIBLP

For the rest of presentation: Assume HPR value is bounded.

Our Goal

solve MIBLP by using a standard **simplex-based branch-and-cut** algorithm; enforce $d^T y \leq \Phi(x)$ on the fly, by adding cutting planes

- given **optimal vertex** (x^*, y^*) of $\overline{\text{HPR}}$
  - (x^*, y^*) infeasible for HPR (i.e., fractional) $\rightarrow$ branch as usual
  - (x^*, y^*) feasible for HPR and $f(x^*, y^*) \leq \Phi(x^*)$ $\rightarrow$ update the incumbent as usual
  - (x^*, y^*) feasible for HPR and $f(x^*, y^*) > \Phi(x^*)$, i.e., **bilevel-infeasible** $\rightarrow$ we need to do something!

- Moore and Bard [1990] (Branch-and-Bound)
  - branching to cut-off bilevel infeasible solutions
  - no y-variables in leader-constraints
  - either all x-variables integer or all y-variables continuous

Our Goal: Design MILP-based solver for MIBLP

For the rest of presentation: Assume HPR value is bounded.

Our Goal

solve MIBLP by using a standard **simplex-based branch-and-cut** algorithm; enforce $d^T y \leq \Phi(x)$ on the fly, by adding cutting planes

- given **optimal vertex** (x^*, y^*) of $\overline{\text{HPR}}$
  - (x^*, y^*) infeasible for HPR (i.e., fractional) $\rightarrow$ branch as usual
  - (x^*, y^*) feasible for HPR and $f(x^*, y^*) \leq \Phi(x^*)$ $\rightarrow$ update the incumbent as usual
  - (x^*, y^*) feasible for HPR and $f(x^*, y^*) > \Phi(x^*)$, i.e., **bilevel-infeasible** $\rightarrow$ we need to do something!

- DeNegre [2011], DeNegre & Ralphs (Branch-and-Cut)
  - cuts based on slack
  - needs all variables and coefficients to be integer
  - open-source solver `MibS`

Our Goal: Design MILP-based solver for MIBLP

For the rest of presentation: Assume HPR value is bounded.

Our Goal

solve MIBLP by using a standard **simplex-based branch-and-cut** algorithm;
enforce $d^T y \leq \Phi(x)$ on the fly, by adding cutting planes

- given **optimal vertex** (x^*, y^*) of $\overline{\text{HPR}}$
  - (x^*, y^*) infeasible for HPR (i.e., fractional) $\rightarrow$ branch as usual
  - (x^*, y^*) feasible for HPR and $f(x^*, y^*) \leq \Phi(x^*)$ $\rightarrow$ update the incumbent as usual
  - (x^*, y^*) feasible for HPR and $f(x^*, y^*) > \Phi(x^*)$, i.e., **bilevel-infeasible** $\rightarrow$ we need to do something!

- Xu and Wang [2014], Wang and Xu [2017] (Branch-and-Bound)
  - multiway branching to cut-off bilevel infeasible solutions
  - all x-variables integer and bounded, follower coefficients of x-variables must be integer

Our Goal: Design MILP-based solver for MIBLP

For the rest of presentation: Assume HPR value is bounded.

Our Goal

solve MIBLP by using a standard **simplex-based branch-and-cut** algorithm; enforce $d^T y \leq \Phi(x)$ on the fly, by adding cutting planes

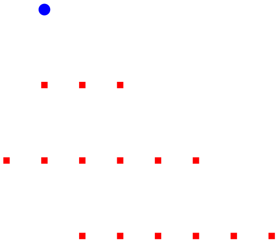
- given **optimal vertex** (x^*, y^*) of $\overline{\text{HPR}}$
  - (x^*, y^*) infeasible for HPR (i.e., fractional) $\rightarrow$ branch as usual
  - (x^*, y^*) feasible for HPR and $f(x^*, y^*) \leq \Phi(x^*)$ $\rightarrow$ update the incumbent as usual
  - (x^*, y^*) feasible for HPR and $f(x^*, y^*) > \Phi(x^*)$, i.e., **bilevel-infeasible** $\rightarrow$ we need to do something!

- **Our Approach** (Branch-and-Cut)
  - Use **Intersection Cuts** (Balas [1971]) to cut off bilevel infeasible solutions

STEP 3: INTERSECTION CUTS

Intersection Cuts (ICs)

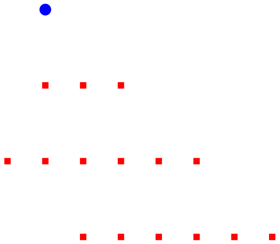
- powerful tool to separate a bilevel infeasible point (x^*, y^*) from a set of bilevel feasible points (X, Y) by a linear cut



- what we need to derive ICs
  - a cone pointed at (x^*, y^*) containing all (X, Y) (if (x^*, y^*) is a vertex of $\overline{HPR}$-**relaxation**, a possible cone comes from **LP-basis**)
  - a convex set S with (x^*, y^*) but no bilevel feasible points $((x, y) \in (X, Y))$ in its **interior**
  - important: (x^*, y^*) should not be on the frontier of S.

Intersection Cuts (ICs)

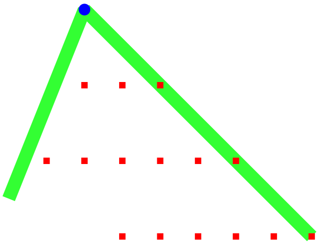
- powerful tool to separate a bilevel infeasible point (x^*, y^*) from a set of bilevel feasible points (X, Y) by a linear cut



- what we need to derive ICs
  - a cone pointed at (x^*, y^*) containing all (X, Y) (if (x^*, y^*) is a vertex of $\overline{HPR}$-**relaxation**, a possible cone comes from **LP-basis**)
  - a convex set S with (x^*, y^*) but no bilevel feasible points $((x, y) \in (X, Y))$ in its **interior**
  - important: (x^*, y^*) should not be on the frontier of S.

Intersection Cuts (ICs)

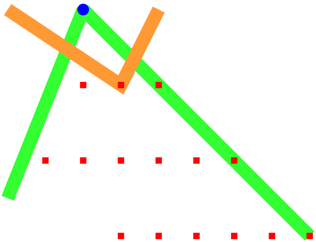
- powerful tool to separate a bilevel infeasible point (x^*, y^*) from a set of bilevel feasible points (X, Y) by a linear cut



- what we need to derive ICs
  - a cone pointed at (x^*, y^*) containing all (X, Y) (if (x^*, y^*) is a vertex of $\overline{HPR}$-**relaxation**, a possible cone comes from **LP-basis**)
  - a convex set S with (x^*, y^*) but no bilevel feasible points ($(x, y) \in (X, Y)$) in its **interior**
  - important: (x^*, y^*) should not be on the frontier of S.

Intersection Cuts (ICs)

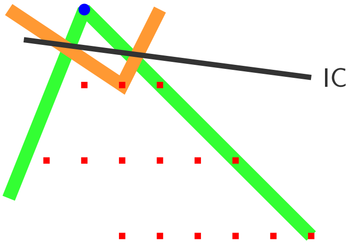
- powerful tool to separate a bilevel infeasible point (x^*, y^*) from a set of bilevel feasible points (X, Y) by a linear cut



- what we need to derive ICs
  - a cone pointed at (x^*, y^*) containing all (X, Y) (if (x^*, y^*) is a vertex of $\overline{HPR}$-**relaxation**, a possible cone comes from **LP-basis**)
  - a convex set S with (x^*, y^*) but no bilevel feasible points $((x, y) \in (X, Y))$ in its **interior**
  - important: (x^*, y^*) should not be on the frontier of S.

Intersection Cuts (ICs)

- powerful tool to separate a bilevel infeasible point (x^*, y^*) from a set of bilevel feasible points (X, Y) by a linear cut



- what we need to derive ICs
  - a cone pointed at (x^*, y^*) containing all (X, Y) (if (x^*, y^*) is a vertex of $\overline{HPR}$-**relaxation**, a possible cone comes from **LP-basis**)
  - a convex set S with (x^*, y^*) but no bilevel feasible points $((x, y) \in (X, Y))$ in its **interior**
  - important: (x^*, y^*) should not be on the frontier of S.

Intersection Cuts for Bilevel Optimization

- we need a **bilevel-free set** S

Theorem

For any feasible solution of the follower $\hat{y} \in \mathbb{R}^{n_2}$, the set

$$S(\hat{y}) = \{(x, y) \in \mathbb{R}^n : d^T y > d^T \hat{y},\ Ax + B\hat{y} \leq b\}$$

does not contain any bilevel-feasible point (not even on its frontier).

- note: $S(\hat{y})$ is a **polyhedron**
- problem: **bilevel-infeasible** (x^*, y^*) can be on the **frontier** of bilevel-free set $S \rightarrow$ IC based on $S(\hat{y})$ may not be able to cut off (x^*, y^*)

Intersection Cuts for Bilevel Optimization

- we need a **bilevel-free set** S

Theorem

For any feasible solution of the follower $\hat{y} \in \mathbb{R}^{n_2}$, the set

$$S(\hat{y}) = \{(x, y) \in \mathbb{R}^n : d^T y > d^T \hat{y},\ Ax + B\hat{y} \leq b\}$$

does not contain any bilevel-feasible point (not even on its frontier).

- note: $S(\hat{y})$ is a **polyhedron**
- problem: **bilevel-infeasible** (x^*, y^*) can be on the **frontier** of bilevel-free set $S \rightarrow$ IC based on $S(\hat{y})$ may not be able to cut off (x^*, y^*)

Intersection Cuts for Bilevel Optimization

- we need a **bilevel-free set** S

Theorem

For any feasible solution of the follower $\hat{y} \in \mathbb{R}^{n_2}$, the set

$$S(\hat{y}) = \{(x, y) \in \mathbb{R}^n : d^T y > d^T \hat{y},\ Ax + B\hat{y} \leq b\}$$

does not contain any bilevel-feasible point (not even on its frontier).

- note: $S(\hat{y})$ is a **polyhedron**
- problem: **bilevel-infeasible** (x^*, y^*) can be on the **frontier** of bilevel-free set $S \rightarrow$ IC based on $S(\hat{y})$ may not be able to cut off (x^*, y^*)

Intersection Cuts for Bilevel Optimization

Assumption

$Ax + By - b$ is integer for all HPR solutions (x, y).

Theorem

Under the previous assumption, for any feasible solution of the follower $\hat{y} \in \mathbb{R}^{n_2}$, the extended polyhedron

$$S^+(\hat{y}) = \{(x, y) \in \mathbb{R}^n : d^T y \geq d^T \hat{y},\ Ax + B\hat{y} \leq b + \mathbf{1}\}, \tag{17}$$

where $\mathbf{1} = (1, \cdots, 1)$ denote a vector of all ones of suitable size, does not contain any bilevel feasible point in its interior.

Intersection Cuts for Bilevel Optimization

Assumption

$Ax + By - b$ is integer for all HPR solutions (x, y).

Theorem

Under the previous assumption, for any feasible solution of the follower $\hat{y} \in \mathbb{R}^{n_2}$, the extended polyhedron

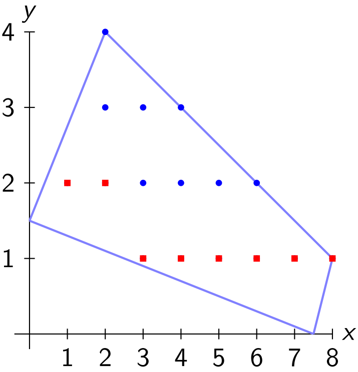
$$S^+(\hat{y}) = \{(x, y) \in \mathbb{R}^n : d^T y \geq d^T \hat{y},\ Ax + B\hat{y} \leq b + \mathbf{1}\}, \tag{17}$$

where $\mathbf{1} = (1, \cdots, 1)$ denote a vector of all ones of suitable size, does not contain any bilevel feasible point in its interior.

Intersection Cuts for Bilevel Optimization

- application sketch on the example from Moore and Bard [1990]
- solve $\overline{\text{HPR}}\rightarrow$ obtain $(x^*, y^*) = (2, 4)$ and LP-cone, take $\hat{y} = 2$
- solve $\overline{\text{HPR}}$ again $\rightarrow$ obtain $(x^*, y^*) = (6, 2)$ and LP-cone, take $\hat{y} = 1$

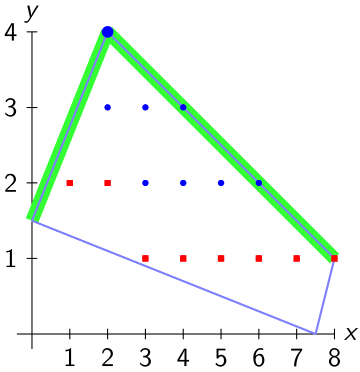
$$\begin{aligned} \min_{x \in \mathbb{Z}} & -x - 10y \\ y \in \arg\min_{y' \in \mathbb{Z}} & \{y' : \\ & -25x + 20y' \leq 30 \\ & x + 2y' \leq 10 \\ & 2x - y' \leq 15 \\ & 2x + 10y' \geq 15\} \end{aligned}$$



Intersection Cuts for Bilevel Optimization

- application sketch on the example from Moore and Bard [1990]
- solve $\overline{\text{HPR}}\rightarrow$ obtain $(x^*, y^*) = (2, 4)$ and LP-cone, take $\hat{y} = 2$
- solve $\overline{\text{HPR}}$ again $\rightarrow$ obtain $(x^*, y^*) = (6, 2)$ and LP-cone, take $\hat{y} = 1$

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



Intersection Cuts for Bilevel Optimization

- application sketch on the example from Moore and Bard [1990]
- solve $\overline{\text{HPR}} \rightarrow$ obtain $(x^*, y^*) = (2, 4)$ and LP-cone, take $\hat{y} = 2$
- solve $\overline{\text{HPR}}$ again $\rightarrow$ obtain $(x^*, y^*) = (6, 2)$ and LP-cone, take $\hat{y} = 1$

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

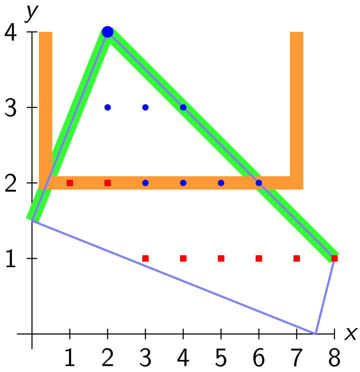
$$y \in \arg\min_{y' \in \mathbb{Z}} \{y' :$$

$$-25x + 20y' \leq 30$$

$$x + 2y' \leq 10$$

$$2x - y' \leq 15$$

$$2x + 10y' \geq 15\}$$



Intersection Cuts for Bilevel Optimization

- application sketch on the example from Moore and Bard [1990]
- solve $\overline{\text{HPR}} \rightarrow$ obtain $(x^*, y^*) = (2, 4)$ and LP-cone, take $\hat{y} = 2$
- solve $\overline{\text{HPR}}$ again $\rightarrow$ obtain $(x^*, y^*) = (6, 2)$ and LP-cone, take $\hat{y} = 1$

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

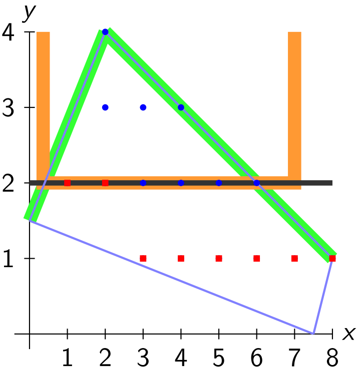
$$y \in \arg\min_{y' \in \mathbb{Z}} \{y' :$$

$$-25x + 20y' \leq 30$$

$$x + 2y' \leq 10$$

$$2x - y' \leq 15$$

$$2x + 10y' \geq 15\}$$



Intersection Cuts for Bilevel Optimization

- application sketch on the example from Moore and Bard [1990]
- solve $\overline{\text{HPR}} \rightarrow$ obtain $(x^*, y^*) = (2, 4)$ and LP-cone, take $\hat{y} = 2$
- solve $\overline{\text{HPR}}$ again $\rightarrow$ obtain $(x^*, y^*) = (6, 2)$ and LP-cone, take $\hat{y} = 1$

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

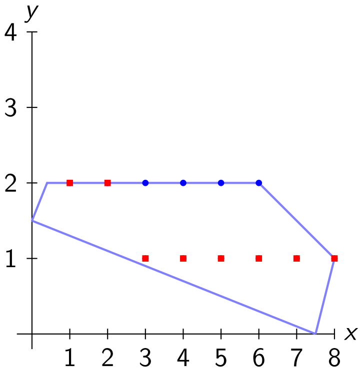
$$y \in \arg\min_{y' \in \mathbb{Z}} \{y' :$$

$$-25x + 20y' \leq 30$$

$$x + 2y' \leq 10$$

$$2x - y' \leq 15$$

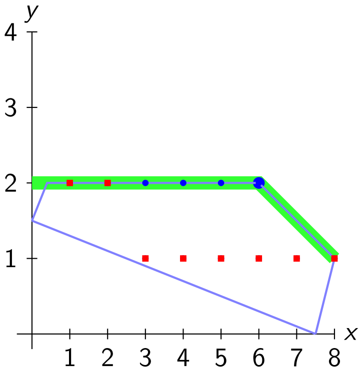
$$2x + 10y' \geq 15\}$$



Intersection Cuts for Bilevel Optimization

- application sketch on the example from Moore and Bard [1990]
- solve $\overline{\text{HPR}} \rightarrow$ obtain $(x^*, y^*) = (2, 4)$ and LP-cone, take $\hat{y} = 2$
- solve $\overline{\text{HPR}}$ again $\rightarrow$ obtain $(x^*, y^*) = (6, 2)$ and LP-cone, take $\hat{y} = 1$

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



Intersection Cuts for Bilevel Optimization

- application sketch on the example from Moore and Bard [1990]
- solve $\overline{\text{HPR}} \rightarrow$ obtain $(x^*, y^*) = (2, 4)$ and LP-cone, take $\hat{y} = 2$
- solve $\overline{\text{HPR}}$ again $\rightarrow$ obtain $(x^*, y^*) = (6, 2)$ and LP-cone, take $\hat{y} = 1$

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

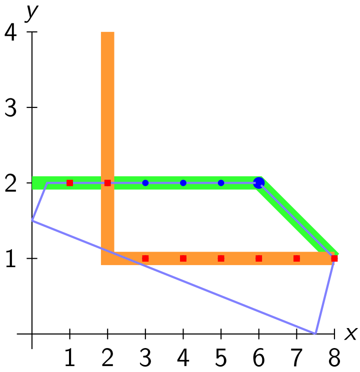
$$y \in \arg\min_{y' \in \mathbb{Z}} \{y' :$$

$$-25x + 20y' \leq 30$$

$$x + 2y' \leq 10$$

$$2x - y' \leq 15$$

$$2x + 10y' \geq 15\}$$



Intersection Cuts for Bilevel Optimization

- application sketch on the example from Moore and Bard [1990]
- solve $\overline{\text{HPR}} \rightarrow$ obtain $(x^*, y^*) = (2, 4)$ and LP-cone, take $\hat{y} = 2$
- solve $\overline{\text{HPR}}$ again $\rightarrow$ obtain $(x^*, y^*) = (6, 2)$ and LP-cone, take $\hat{y} = 1$

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

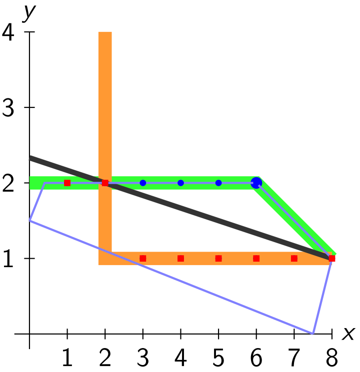
$$y \in \arg\min_{y' \in \mathbb{Z}} \{ y' :$$

$$-25x + 20y' \leq 30$$

$$x + 2y' \leq 10$$

$$2x - y' \leq 15$$

$$2x + 10y' \geq 15 \}$$



Other Bilevel-Free Sets can be defined

- **The choice of bilevel-free polyhedra is not unique.**
- The larger the bilevel-free set, the better the IC.

Theorem (Motivated by Xu [2012], Wang and Xu [2017])

Given $\Delta\hat{y} \in \mathbb{R}^n_2$ such that $d^T\Delta\hat{y} < 0$ and $\Delta\hat{y}_j$ integer $\forall j \in J_y$, the following set

$$X^+(\Delta\hat{y}) = \{(x, y) \in \mathbb{R}^n : Ax + By + B\Delta\hat{y} \leq b + \mathbf{1}\}$$

has no bilevel-feasible points in its interior.

Proof: by contradiction. Assume $(\tilde{x}, \tilde{y}) \in X^+(\Delta\hat{y})$ is bilevel-feasible. But then, $d^T\tilde{y} > d^T(\tilde{y} + \Delta\hat{y})$ and $(\tilde{x}, \tilde{y} + \Delta\hat{y})$ is feasible for the follower, hence contradiction.

SEPARATION of INTERSECTION CUTS

Separation of ICs associated to $S^+(\hat{y})$

Given $\hat{y} \in \mathbb{R}^n_2$ such that $\hat{y}_j$ integer $\forall j \in J_y$, the following set

$$S^+(\hat{y}) = \{(x, y) \in \mathbb{R}^n : d^T y \geq d^T \hat{y},\ Ax + B\hat{y} \leq b + \mathbf{1}\}$$

is bilevel-feasible free. **How to compute $\hat{y}$?**

- **SEP1**

$$\hat{y} \in \arg\min_{y \in \mathbb{R}^{n_2}} \{d^T y \ :\ By \leq b - Ax^*, \quad y_j \text{ integer } \forall j \in J_y\}.$$

  - $\hat{y}$ is the optimal solution of the follower when $x = x^*$.
  - Maximize the distance of (x^*, y^*) from the facet $d^T y \geq d^T \hat{y}$ of $S(\hat{y})$.
- **SEP2** Alternatively, try to find $\hat{y}$ such that some of the facets in $Ax + b\hat{y} \leq b$ can be removed (making thus $S(\hat{y})$ larger!)
  - A modified MIP is solved, s.t. the number of removable facets is maximized.

Separation of ICs associated to $X^+(\Delta\hat{y})$

Given $\Delta\hat{y} \in \mathbb{R}_2^n$ such that $d^T\Delta\hat{y} < 0$ and $\Delta\hat{y}_j$ integer $\forall j \in J_y$, the following set

$$X^+(\Delta\hat{y}) = \{(x,y) \in \mathbb{R}^n : Ax + By + B\Delta\hat{y} \leq b + \mathbf{1}\}$$

has no bilevel-feasible points in its interior. **How to compute $\Delta\hat{y}$?**

- **XU** (Xu [2012])

$$\begin{aligned}
\Delta\hat{y} \in \arg\min \sum_{i=1}^{\tilde{m}} t_i & \\
d^T\Delta y \leq -1 & \\
B\Delta y \leq b - Ax^* - By^* & \\
\Delta y_j \text{ integer}, & \qquad \forall j \in J_y \\
B\Delta y \leq t \text{ and } t \geq 0. &
\end{aligned}$$

  - variable t_i has value 0 in case $(\tilde{B}\Delta y)_i \leq 0$ ("removable facet");
  - "maximize the size" of the bilevel-free set associated with $\Delta\hat{y}$.

COMPUTATIONAL STUDY

Settings

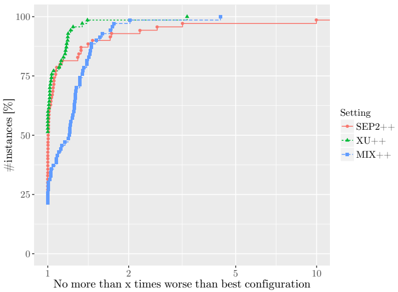
C, CPLEX 12.6.3, Intel Xeon E3-1220V2 3.1 GHz, four threads.

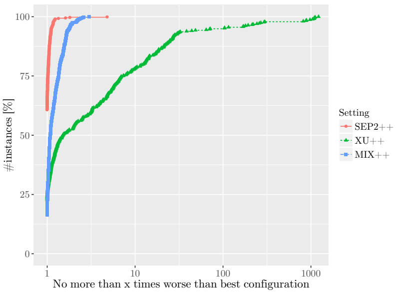
Class	Source	Type	#Inst	#OptB	#Opt
DENEGRE	DeNegre [2011],Ralphs and Adams [2016]	I	50	45	**50**
MIPLIB	Fischetti et al. [2016]	B	57	20	27
XUWANG	Xu and Wang [2014]	I,C	140	140	**140**
INTER-KP	DeNegre [2011],Ralphs and Adams [2016]	B	160	79	138
INTER-KP2	Tang et al. [2016]	B	150	53	**150**
INTER-ASSIG	DeNegre [2011],Ralphs and Adams [2016]	B	25	25	**25**
INTER-RANDOM	DeNegre [2011],Ralphs and Adams [2016]	B	80	-	**80**
INTER-CLIQUE	Tang et al. [2016]	B	80	10	**80**
INTER-FIRE	Baggio et al. [2016]	B	72	-	**72**
total			814	372	762

- #OptB = number of optimal solutions known before our work.
- #Opt = number of optimal solutions known after our work.

Effects of different ICs

- `MIX++`: combination of settings `SEP2++` and `XU++` (both ICs being separated at each separation call).
- Performance profile on the subsets of (bilevel and interdiction) instances that could be solved to optimality by all three settings within the given time-limit of one hour.





Comparison with the literature (1)

- Results for the instance set XUWANG

	MIX++											Xu and Wang [2014]
n_1	$i=1$	$i=2$	$i=3$	$i=4$	$i=5$	$i=6$	$i=7$	$i=8$	$i=9$	$i=10$	avg	avg
10	3	3	3	3	2	3	2	3	2	3	2.6	1.4
60	2	0	0	1	1	1	1	1	2	2	0.9	45.6
110	2	1	2	2	1	2	1	2	2	12	2.8	111.9
160	2	2	3	2	3	1	4	1	1	3	2.1	177.9
210	2	3	1	1	3	3	3	2	5	3	2.6	1224.5
260	3	4	3	6	3	5	6	2	7	11	5.0	1006.7
310	5	10	11	14	7	16	15	8	5	3	9.4	4379.3
360	17	28	11	13	11	15	7	19	9	14	14.4	2972.4
410	19	10	29	8	21	10	9	15	23	42	18.7	4314.2
460	22	10	22	35	21	21	32	22	23	23	23.1	6581.4
B1-110	0	0	0	0	0	1	0	1	0	9	1.3	132.3
B1-160	1	1	3	1	2	1	3	0	0	2	1.3	184.4
B2-110	16	2	2	8	1	25	15	5	1	122	19.7	4379.8
B2-160	8	38	21	91	34	4	40	3	12	123	37.4	22999.7

Comparison with the literature (2)

- Results for the instance sets `INTER-KP2` (left) and `INTER-CLIQUE` (right)

n_1	k	`MIX++` t[s]	Tang et al. [2016] t[s]	Tang et al. [2016] #unsol
20	5	5.4	721.4	0
20	10	1.7	2992.6	3
20	15	0.2	129.5	0
22	6	10.3	1281.2	6
22	11	2.3	3601.8	10
22	17	0.2	248.2	0
25	7	33.6	3601.4	10
25	13	8.0	3602.3	10
25	19	0.4	1174.6	0
28	7	97.9	3601.0	10
28	14	22.6	3602.5	10
28	21	0.5	3496.9	8
30	8	303.0	3601.0	10
30	15	31.8	3602.3	10
30	23	0.6	3604.5	10

ν	d	`MIX++` t[s]	Tang et al. [2016] t[s]	Tang et al. [2016] #unsol
8	0.7	0.1	373.0	0
8	0.9	0.2	3600.0	10
10	0.7	0.3	3600.1	10
10	0.9	0.7	3600.2	10
12	0.7	0.8	3600.3	10
12	0.9	1.9	3600.4	10
15	0.7	2.2	3600.3	10
15	0.9	12.6	3600.2	10

Conclusions (Part I)

- Branch-and-cut algorithm, a black-box solver for mixed integer bilevel programs
  - Major feature: intersection cuts, to cut away bilevel-free sets.
  - It outperforms previous methods from the literature by a large margin.
  - Byproduct: the optimal solution for more than 300 previously unsolved instances from literature is now available.

Code is publicly available:

`https://msinnl.github.io/pages/bilevel.html`

Part II

Often, the follower's subproblem has a special structure that we could exploit.

Conclusions (Part I)

- Branch-and-cut algorithm, a black-box solver for mixed integer bilevel programs
  - Major feature: intersection cuts, to cut away bilevel-free sets.
  - It outperforms previous methods from the literature by a large margin.
  - Byproduct: the optimal solution for more than 300 previously unsolved instances from literature is now available.

Code is publicly available:

`https://msinnl.github.io/pages/bilevel.html`

Part II

Often, the follower's subproblem has a special structure that we could exploit.

PART II: BRANCH-AND-CUT FOR INTERDICTION-LIKE PROBLEMS

Interdiction Games (IGs)

- special case of bilevel optimization problems
- leader and follower have **opposite objective functions**
- leader **interdicts** items of follower
  - type of interdiction: linear or **discrete**, cost increase or **destruction**
  - interdiction **budget**
- two-person, zero-sum sequential game
- studied mostly for network-based problems in the follower

Interdiction Games (IGs)

- special case of bilevel optimization problems
- leader and follower have **opposite objective functions**
- leader **interdicts** items of follower
  - type of interdiction: linear or **discrete**, cost increase or **destruction**
  - interdiction **budget**
- two-person, zero-sum sequential game
- studied mostly for network-based problems in the follower

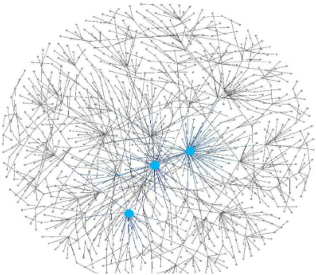
(a) Linear, cost increase

(b) Discrete, destruction

Figure: Early Applications of Interdiction, following [Livy, 218BC]

Interdiction Games (IGs): Attacker-Defender models

(a) Drug cartels



(b) Most voulnerable nodes

Figure: Modern Applications of Interdiction

- Interdiction Problems: find leader's strategy that results in the worst outcome for the follower (min-max)
- Blocker Problems: find the minimum cost strategy for the leader that guarantees a limited outcome for the follower

Interdiction Games (IGs)

We focus on:

$$\min_{x \in X} \max_{y \in \mathbb{R}^{n_2}} \; d^T y \tag{18}$$

$$Q\, y \leq q_0 \tag{19}$$

$$0 \leq y_j \leq u_j(1 - x_j), \qquad \forall j \in N \tag{20}$$

$$y_j \text{ integer}, \qquad \forall j \in J_y \tag{21}$$

- $X = \{x \in \mathbb{R}^{n_1} : Ax \leq b,\ x_j \text{ integer } \forall j \in J_x,\ x_j \text{ binary } \forall j \in N\}$ (feasible interdiction policies).
- n_1 and n_2 are the number of leader (x) and follower (y) variables, resp.
- d, Q, q_0, u, A, b are given rational matrices/vectors of appropriate size.
- u: finite upper bounds on the follower variables y_j that can be interdicted.
- The concept easily extends to blocker problems as well.

PROBLEM REFORMULATION

Problem Reformulation

For a given $x \in X$ we define the value function:

$$\Phi(x) = \max_{y \in \mathbb{R}^{n_2}} d^T y \tag{22}$$

$$Q\, y \leq Q_0 \tag{23}$$

$$0 \leq y_j \leq u_j(1 - x_j), \qquad \forall j \in N \tag{24}$$

$$y_j \text{ integer}, \qquad \forall j \in J_y \tag{25}$$

so that problem can be restated in the $\mathbb{R}^{n_1+1}$ space as

$$\min_{x \in \mathbb{R}^{n_1}, w \in \mathbb{R}} w \tag{26}$$

$$w \geq \Phi(x) \tag{27}$$

$$Ax \leq b \tag{28}$$

$$x_j \text{ integer}, \qquad \forall j \in J_x \tag{29}$$

$$x_j \in \{0, 1\}, \qquad \forall j \in N. \tag{30}$$

Try to replace the constraints (27) by linear constraints.

Benders-Like Reformulation

Find (sufficiently large) M_j's and reformulate the follower [Wood, 2010]

$$\Phi(x) = \max\{d^T y - \sum_{j \in N} M_j x_j y_j : y \in Y\}, \tag{31}$$

where

$$Y = \{y \in \mathbb{R}^{n_2} : Q\, y \le q_0, \quad 0 \le y_j \le u_j \ \forall j \in N, \quad y_j \text{ integer } \forall j \in J_y\}.$$

Let $\hat{Y}$ be extreme points of $\operatorname{conv} Y$.

Benders-Like Reformulation

$$\min_{x \in \mathbb{R}^{n_1}, w \in \mathbb{R}} w \tag{32}$$

$$w \ge d^T \hat{y} - \sum_{j \in N} M_j x_j \hat{y}_j \qquad \forall \hat{y} \in \hat{Y} \tag{33}$$

$$Ax \le b \tag{34}$$

$$x_j \text{ integer}, \qquad \forall j \in J_x \tag{35}$$

$$x_j \text{ binary}, \qquad \forall j \in N. \tag{36}$$

INTERDICTION GAMES WITH MONOTONICITY PROPERTY

Interdiction Problems with Monotonicity Property

The follower:

$$\Phi(x) = \max_{y \in \mathbb{R}^{n_2}} \; d_N^T y_N + d_R^T y_R$$

$$Q_N\, y_N + Q_R\, y_R \leq q_0$$

$$0 \leq \; y_j \leq u_j(1 - x_j), \qquad \forall j \in N$$

$$y_j \text{ integer}, \qquad \forall j \in J_y$$

- $y_N = (y_j)_{j \in N}$ variables that can be interdicted,
- $y_R = (y_j)_{j \in R}$ the remaining follower variables.
- Associated $Q = (Q_N, Q_R)$ and $d^T = (d_N^T, d_R^T)$.

Downward Monotonicity: Assume $Q_N \geq 0$

"if $\hat{y} = (\hat{y}_N, \hat{y}_R)$ is a feasible follower for a given x and $y' = (y'_N, \hat{y}_R)$ satisfies integrality constraints and $0 \leq y'_N \leq \hat{y}_N$, then y' is **also feasible** for x".

Independent Systems (y are binary and $R = \emptyset$)

$\mathcal{S} := \{S \subseteq N : Q\,\chi_S \leq q_0\} \subseteq 2^N$ forms an **independent system**.

Interdiction Problems with Monotonicity Property

The follower:

$$\Phi(x) = \max_{y \in \mathbb{R}^{n_2}} \; d_N^T y_N + d_R^T y_R$$

$$Q_N\, y_N + Q_R\, y_R \leq q_0$$

$$0 \leq y_j \leq u_j(1 - x_j), \qquad \forall j \in N$$

$$y_j \text{ integer}, \qquad \forall j \in J_y$$

- $y_N = (y_j)_{j \in N}$ variables that can be interdicted,
- $y_R = (y_j)_{j \in R}$ the remaining follower variables.
- Associated $Q = (Q_N, Q_R)$ and $d^T = (d_N^T, d_R^T)$.

Downward Monotonicity: Assume $Q_N \geq 0$

"if $\hat{y} = (\hat{y}_N, \hat{y}_R)$ is a feasible follower for a given x and $y' = (y'_N, \hat{y}_R)$ satisfies integrality constraints and $0 \leq y'_N \leq \hat{y}_N$, then y' is **also feasible** for x".

Independent Systems (y are binary and $R = \emptyset$)

$\mathcal{S} := \{S \subseteq N : Q\, \chi_S \leq q_0\} \subseteq 2^N$ forms an **independent system**.

Even with Monotonicity the Problems Remain Hard...

Complexity

- Even when the follower is a pure LP, the problem remains NP-hard (Zenklusen [2010], Dinitz and Gupta [2013]).
- In general, already knapsack interdiction is Σ_2^P-hard (Caprara et al. [2013]).

Examples

Interdicting/Blocking:

- set packing problem
- (multidimensional) knapsack problem
- prize-collecting Steiner tree
- orienteering problem
- maximum clique problem
- all kind of hereditary problems on graphs

The Choice of M_j's is Crucial

Theorem

For Interdiction Games with Monotonicity $M_j = d_j$, *i.e., we have:*

$$\min_{x \in \mathbb{R}^{n_1}, w \in \mathbb{R}} w$$

$$w \geq \sum_{j \in R} d_j \hat{y}_j + \sum_{j \in N} d_j \hat{y}_j (1 - x_j) \qquad \forall \hat{y} \in \hat{Y}$$

$$Ax \leq b$$

$$x_j \text{ integer}, \qquad \forall j \in J_x$$

$$x_j \text{ binary}, \qquad \forall j \in N.$$

- Branch-and-cut: separation of interdiction cuts is done by solving the follower's subproblem with given x^* (lazy cut separation).
- Specialized procedures/algorithms for the follower's subproblem could be exploited.

Interdiction Cuts Could be Lifted/Modified

Assumption 2

All follower variables y_N are binary and $u_j = 1$.

Theorem

Take any $\hat{y} \in \hat{Y}$. Let $a, b \in N$ with $\hat{y}_a = 1$, $\hat{y}_b = 0$, $d_a < d_b$ and $Q_a \geq Q_b$. Then the following **lifted interdiction cut** is valid:

$$w \geq \sum_{j \in R} d_j \hat{y}_j + \sum_{j \in N} d_j \hat{y}_j (1 - x_j) + (d_b - d_a)(1 - x_b).$$

Theorem

Take any $\hat{y} \in \hat{Y}$. Let $a, b \in N$ with $\hat{y}_a = 1$, $\hat{y}_b = 0$ and $Q_a \geq Q_b$. Then the following **modified interdiction cut** is valid:

$$w \geq \sum_{j \in R} d_j \hat{y}_j + \sum_{j \in N} d_j \hat{y}_j (1 - x_j) + d_b(x_a - x_b). \tag{37}$$

COMPUTATIONAL RESULTS

The Knapsack Interdiction Problem

Runtime to optimality. Our approach (B&C) vs. the cutting plane (CP) and CCLW approaches from Caprara et al. [2016].

size	instance	z^*	CP	CCLW	B&C	size	instance	z^*	CP	CCLW	B&C
35	1	279	0.34	0.79	0.12	45	1	427	1.81	2.37	0.23
	2	469	1.59	2.57	0.21		2	633	13.03	11.64	0.37
	3	448	55.61	40.39	0.66		3	548	TL	344.01	1.81
	4	370	495.50	1.48	0.87		4	611	TL	38.90	3.30
	5	467	TL	0.72	0.93		5	629	TL	3.42	2.78
	6	268	71.43	0.06	0.11		6	398	3300.76	0.07	0.17
	7	207	144.46	0.06	0.07		7	225	60.43	0.04	0.09
	8	41	0.50	0.04	0.07		8	157	60.88	0.05	0.10
	9	80	0.97	0.03	0.07		9	53	0.83	0.05	0.10
	10	31	0.12	0.03	0.08		10	110	0.40	0.05	0.11
40	1	314	0.66	1.06	0.16	50	1	502	2.86	4.55	0.21
	2	472	6.67	7.50	0.36		2	788	1529.16	1520.56	2.38
	3	637	324.61	162.80	1.02		3	631	TL	105.59	2.40
	4	388	1900.03	0.34	0.82		4	612	TL	3.64	1.27
	5	461	TL	0.22	0.58		5	764	TL	0.60	4.82
	6	399	2111.85	0.09	0.13		6	303	1046.85	0.05	0.14
	7	150	83.59	0.05	0.08		7	310	2037.01	0.09	0.11
	8	71	1.73	0.04	0.09		8	63	2.79	0.05	0.12
	9	179	137.16	0.08	0.09		9	234	564.97	0.10	0.12
	10	0	0.03	0.03	0.04		10	15	0.09	0.04	0.13

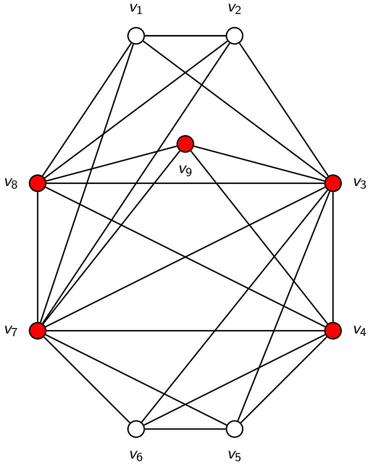
The Knapsack Interdiction Problem

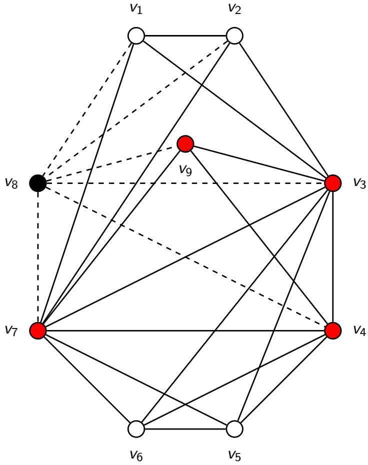
Instances from Tang et al. [2016] (TRS). Comparison with MIX++. Average results over ten instances per row. N^* #instances unsolved.

		TRS		MIX++	B&C
$\|N\|$	k	t[s]	N^*	t[s]	t[s]
20	5	721.4	0	5.4	0.1
20	10	2992.6	3	1.7	0.1
20	15	129.5	0	0.2	0.1
22	6	1281.2	6	10.3	0.1
22	11	3601.8	10	2.3	0.1
22	17	248.2	0	0.2	0.1
25	7	3601.4	10	33.6	0.2
25	13	3602.3	10	8.0	0.2
25	19	1174.6	0	0.4	0.1
28	7	3601.0	10	97.9	0.3
28	14	3602.5	10	22.6	0.3
28	21	3496.9	8	0.5	0.1
30	8	3601.0	10	303.0	0.3
30	15	3602.3	10	31.8	0.3
30	23	3604.5	10	0.6	0.1

The Clique Interdiction Problem

Example: $\omega(G) = 5$ and $k = 1$

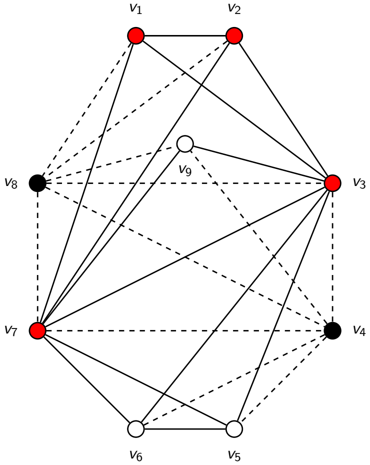


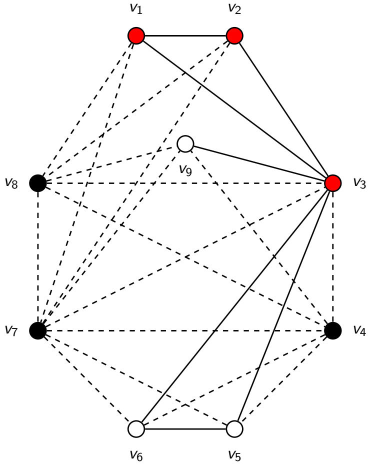


Maximum Clique $\tilde{K} = \{v_3, v_4, v_7, v_8, v_9\}$ Optimal interdiction policy $\{v_8\}$

The Clique Interdiction Problem

Example: $\omega(G) = 5$ and $k = 2, k = 3$





Optimal interdiction policy $\{v_4, v_8\}$

Optimal interdiction policy $\{v_4, v_7, v_8\}$

Branch-and-Cut for Clique Interdiction

Benders-Like Reformulation

$\mathcal{K}$: set of all cliques in G.

$$\begin{aligned} \min \quad & w \\ & w + \sum_{u \in K} x_u \geq |K| \qquad K \in \mathcal{K} \\ & \sum_{u \in V} x_u \leq k \\ & x_u \in \{0,1\} \quad u \in V. \end{aligned}$$

Ingredients:

- State-of-the-art clique solver from San Segundo et al. [2016].
- Facets, lifting.
- Combinatorial primal and dual bounds.
- Graph reductions.

Comparison with MIX++

		CLIQUE-INTER				MIX++			
$\|V\|$	#	# solved	time	exit gap	root gap	# solved	time	exit gap	root gap
50	44	**44**	**0.01**	-	0.16	28	68.58	6.44	8.50
75	44	**44**	**1.45**	-	0.41	14	120.19	9.47	10.91
100	44	**37**	**9.30**	1.00	0.98	7	164.42	12.65	13.11
125	44	**35**	**13.43**	1.33	1.20	2	135.33	13.88	14.73
150	44	**33**	**27.23**	1.91	1.43	1	397.52	16.42	16.39

Results on Real-world (sparse) networks

	$\lvert V\rvert$	$\lvert E\rvert$	ω [s]	$k = \lceil 0.005 \cdot \lvert V\rvert \rceil$		$k = \lceil 0.01 \cdot \lvert V\rvert \rceil$	
				[s]	$\lvert V_p\rvert$	[s]	$\lvert V_p\rvert$
socfb-UIllinois	30,795	1,264,421	0.5	24.4	10,456	41.6	8290
ia-email-EU	32,430	54,397	0.0	0.6	30,375	0.5	29,212
rgg_n_2_15_s0	32,768	160,240	0.0	-	-	0.2	30,848
ia-enron-large	33,696	180,811	0.0	2.2	27,791	29.5	26,651
socfb-UF	35,111	1,465,654	0.3	17.8	14,264	87.8	10,708
socfb-Texas84	36,364	1,590,651	0.3	24.6	10,706	74.3	8,704
tech-internet-as	40,164	85,123	0.0	1.4	31,783	-	-
fe-body	45,087	163,734	0.1	1.8	2,259	1.8	2259
sc-nasasrb	54,870	1,311,227	0.1	-	-	145.5	1,195
soc-themarker_u	69,413	1,644,843	2.1	T.L.	35,678	T.L.	31,101
rec-eachmovie_u	74,424	1,634,743	0.7	-	-	367.3	13669
fe-tooth	78,136	452,591	0.5	18.9	7	19.0	7
sc-pkustk11	87,804	2,565,054	1.1	70.7	2,712	57.1	2,712
soc-BlogCatalog	88,784	2,093,195	11.7	T.L.	51,607	T.L.	46,240
ia-wiki-Talk	92,117	360,767	0.2	49.2	72,678	87.4	72,678

Conclusions

Branch-and-Cuts for

- General Mixed Integer Bilevel Programs (intersection cuts)
- Interdiction-Like Bilevel Programs (interdiction cuts)
- Interdiction problems easier, and it pays off to exploit the structure
- Use interdiction cuts for blocker-type problems too

Open questions, directions for future research

- Other bilevel-free sets, tighter cuts for the generic case?
- Non-linear mixed integer bilevel problems?
- General purpose solvers for bilevel pricing problems?
- Three-level and multi-level optimization problems, DAD models?

Literature I

C. Audet, P. Hansen, B. Jaumard, and G. Savard. Links between linear bilevel and mixed 0–1 programming problems. **Journal of Optimization Theory and Applications**, 93(2):273–300, 1997.

A. Baggio, M. Carvalho, A. Lodi, and A. Tramontani. Multilevel approaches for the critical node problem. Working paper. **École Polytechnique de Montreal**, 2016.

E. Balas. Intersection cuts–a new type of cutting planes for integer programming. **Operations Research**, 19(1):19–39, 1971.

A. Caprara, M. Carvalho, A. Lodi, and G. J. Woeginger. A complexity and approximability study of the bilevel knapsack problem. In **Proceedings of the 16th International Conference on Integer Programming and Combinatorial Optimization**, IPCO'13, pages 98–109. Springer, 2013.

A. Caprara, M. Carvalho, A. Lodi, and G. J. Woeginger. Bilevel knapsack with interdiction constraints. **INFORMS Journal on Computing**, 28(2):319–333, 2016.

S. DeNegre. **Interdiction and Discrete Bilevel Linear Programming**. PhD thesis, Lehigh University, 2011.

Literature II

M. Dinitz and A. Gupta. Packing interdiction and partial covering problems. In **Proceedings of the 16th International Conference on Integer Programming and Combinatorial Optimization**, IPCO'13, pages 157–168. Springer-Verlag, 2013.

M. Fischetti, I. Ljubic, M. Monaci, and M. Sinnl. Intersection cuts for bilevel optimization. In **IPCO Proceedings**, LNCS. Springer, 2016.

P. Hansen, B. Jaumard, and G. Savard. New branch-and-bound rules for linear bilevel programming. **SIAM Journal on Scientific and Statistical Computing**, 13(5):1194–1217, 1992.

M. Köppe, M. Queyranne, and C. T. Ryan. Parametric integer programming algorithm for bilevel mixed integer programs. **Journal of Optimization Theory and Applications**, 146(1):137–150, 2010.

A. Lodi, T. K. Ralphs, and G. J. Woeginger. Bilevel programming and the separation problem. **Math. Program.**, 146(1-2):437–458, 2014.

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

T. K. Ralphs and E. Adams. Bilevel instance library, 2016. `http://coral.ise.lehigh.edu/data-sets/bilevel-instances/`.

Literature III

P. San Segundo, A. Lopez, and P. M. Pardalos. A new exact maximum clique algorithm for large and massive sparse graphs. **Computers & OR**, 66:81–94, 2016.

Y. Tang, J.-P. P. Richard, and J. C. Smith. A class of algorithms for mixed-integer bilevel min–max optimization. **Journal of Global Optimization**, 66(2): 225–262, 2016.

L. Wang and P. Xu. The watermelon algorithm for the bilevel integer linear programming problem. **SIAM Journal on Optimization**, 27(3):1403–1430, 2017.

R. K. Wood. **Bilevel Network Interdiction Models: Formulations and Solutions**. John Wiley & Sons, Inc., 2010.

P. Xu. **Three essays on bilevel optimization algorithmsand applications**. PhD thesis, Iowa State University, 2012.

P. Xu and L. Wang. An exact algorithm for the bilevel mixed integer linear programming problem under three simplifying assumptions. **Computers & Operations Research**, 41:309–318, 2014.

R. Zenklusen. Matching interdiction. **Discrete Applied Mathematics**, 158(15): 1676 – 1690, 2010.