Recovering Dantzig-Wolfe Bounds by Cutting Planes

Rui Chen
Cornell Tech
(Joining CUHK-Shenzhen in August)

MIP Workshop, University of Kentucky

June, 2024

Joint work with...

Figure: Andrea, me and Oktay at MIP 2022

MIPs with blocks

- We consider MI(L)Ps with the following structure:

$$
\begin{aligned}
z^* := \min\ & c^\top x \\
\text{s.t. } & x_{I(j)} \in P^j, \quad j \in \{1, \dots, q\}, && \text{(blocks)} \\
& Ax \geq b, && \text{(coupling constraints)} \\
& x \in X, && \text{(integrality)}
\end{aligned}
$$

where $I(j) \subseteq \{1, \dots, n\}$ (not necessarily disjoint), and

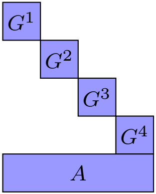
$$P^j = \{y \in \mathbb{R}^{|I(j)|} : G^j y \geq g^j\} \qquad \text{for } j = 1, \dots, q$$

and $X \subseteq \mathbb{R}^n$ represents integrality constraints on some of the variables (all data is rational, problem is feasible)

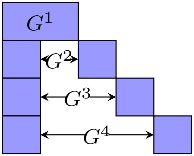
- Many important MIP problems have this structure
- Notice that P^js do not know about the integrality

MIPs with blocks: Applications

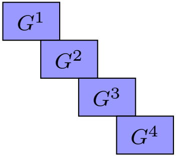
- Loosely coupled [Bodur et al. 2022]
  - Multiple knapsack assignment [Kataoka and Yamada 2014]
  - Generalized assignment [Gattal and Benrazek 2021]



- Two-stage stochastic integer programs [Ahmed 2010]



- Overlapping
  - Temporal knapsack [Bartlett et al. 2005]
  - Temporal bin packing [Dell'Amico et al. 2020]



Dantzig-Wolfe (DW) relaxation

MIP with blocks:

$$
\begin{aligned}
z^* := \min\ & c^\top x \\
\text{s.t. } & x_{I(j)} \in P^j, \quad j \in J := \{1, \ldots, q\}, && \text{(blocks)} \\
& Ax \geq b, && \text{(coupling constraints)} \\
& x \in X && \text{(integrality)}
\end{aligned}
$$

LP Relaxation:

$$
\begin{aligned}
z^{LP} := \min\ & c^\top x \\
\text{s.t. } & x_{I(j)} \in P^j, \quad j \in J \\
& Ax \geq b,
\end{aligned}
$$

DW Relaxation:

$$
\begin{aligned}
z^{DW} := \min\ & c^\top x \\
\text{s.t. } & x_{I(j)} \in \text{conv}(Q^j), \quad j \in J \\
& Ax \geq b,
\end{aligned}
$$

where $Q^j = P^j \cap X^j$ (X^j : integrality constraints on $x_{I(j)}$) and

$$z^{DW} \geq z^{LP}$$

DW Relaxation

$$
\begin{aligned}
z^{DW} = \min\ & c^\top x \\
\text{s.t. } & x_{I(j)} = \sum_{v \in V^j} \lambda_v v + \sum_{r \in R^j} \mu_r r, && j \in J, \quad (\pi^j) \\
& \sum_{v \in V^j} \lambda_v = 1, && j \in J \quad (\theta^j) \\
& Ax \geq b, && (\beta) \\
& \lambda \geq 0,\ \mu \geq 0,
\end{aligned}
$$

where V^j (R^j) is the set of extreme points (rays) of conv(Q^j)

- Solve the DW relaxation using column generation

  Pricing problem for block $j \in J$:

  $$D_j(\pi^j) := \min \left\{ (\pi^j)^\top v : v \in \text{conv}(Q^j) \right\} \; = \; \min \left\{ (\pi^j)^\top v : v \in Q^j \right\}$$

- Can be used to solve the MIP exactly if combined with branching (i.e., branch-and-price, but not available in most solvers)

DW Relaxation

$$
\begin{aligned}
z^{DW} = \min\ & c^\top x \\
\text{s.t. } & x_{I(j)} = \sum_{v \in V^j} \lambda_v v + \sum_{r \in R^j} \mu_r r, \quad j \in J, \quad (\pi^j) \\
& \sum_{v \in V^j} \lambda_v = 1, \qquad\qquad j \in J \quad (\theta^j) \\
& Ax \geq b, \qquad\qquad (\beta) \\
& \lambda \geq 0,\ \mu \geq 0,
\end{aligned}
$$

where V^j (R^j) is the set of extreme points (rays) of conv(Q^j)

- Solve the DW relaxation using column generation

  Pricing problem for block $j \in J$:

$$
D_j(\pi^j) := \min \left\{ (\pi^j)^\top v : v \in \text{conv}(Q^j) \right\} \;=\; \min \left\{ (\pi^j)^\top v : v \in Q^j \right\}
$$

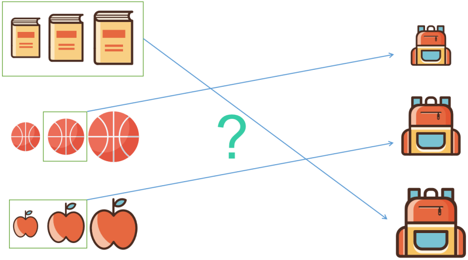
- Can be used to solve the MIP exactly if combined with branching (i.e., branch-and-price, but not available in most solvers)

Example: Multiple knapsack assignment problem (MKAP)

Given:

- N: set of items with weights
- M: set of knapsack types with capacity
- K: set of item classes
- $(S_k)_{k \in K}$: set of items that belong to item class k

- Objective: Maximize profit of packed items
- Only items from the same item class can be packed together
- Cannot exceed knapsack capacities



Example: Multiple knapsack assignment problem (MKAP)

Given:

- N: set of items with weights
- M: set of knapsack types with capacity
- K: set of item classes
- $(S_k)_{k \in K}$: set of items that belong to item class k

MKAP

$$
\begin{aligned}
\min \quad & \sum_{i \in M} \sum_{j \in N} -p_j x_{ij} & \\
\text{s.t.} \quad & \boxed{\sum_{j \in S_k} w_j x_{ij} \leq C_i y_{ik},} & i \in M,\ k \in K, \\
& \sum_{i \in M} x_{ij} \leq 1, & j \in N, \\
& \sum_{k \in K} y_{ik} \leq 1, & i \in M, \\
& x \in \{0,1\}^{M \times N},\ y \in \{0,1\}^{M \times K} &
\end{aligned}
$$

DW bound compared to LP bound

DW bound (z^{DW}) is better than LP bound (z^{LP}), and sometimes significantly so

$\|K\|$	$\|M\|$	$\|N\|$	$(z^{DW} - z^{LP})/\|z^{DW}\|$ (%)
10	10	100	**5.65**
10	10	200	**3.53**
10	10	300	**3.64**
10	20	100	0.74
10	20	200	0.02
10	20	300	0.00
10	30	100	0.88
10	30	200	0.02
10	30	300	0.00
10	40	100	**1.51**
10	40	200	0.04
10	40	300	0.00

Table: Bound gaps ($|K| = 10$)

$\|K\|$	$\|M\|$	$\|N\|$	$(z^{DW} - z^{LP})/\|z^{DW}\|$ (%)
25	10	100	**54.17**
25	10	200	**58.14**
25	10	300	**66.02**
25	20	100	**9.40**
25	20	200	**8.74**
25	20	300	**10.18**
25	30	100	**3.80**
25	30	200	**1.79**
25	30	300	**1.30**
25	40	100	**3.13**
25	40	200	0.75
25	40	300	0.25

Table: Bound gaps ($|K| = 25$)

- N: set of items
- M: set of knapsacks
- K: set of item classes

Solver cuts

Let z^{LP+} denote the bound obtained by Gurobi at the root node
(LP bound enhanced by Gurobi presolve and cuts)

$\lvert K\rvert$	$\lvert M\rvert$	$\lvert N\rvert$	$(z^{DW} - z^{LP})/\lvert z^{DW}\rvert$ (%)	$(z^{DW} - z^{LP+})/\lvert z^{DW}\rvert$ (%)
10	10	100	5.65	**4.13**
10	10	200	3.53	**2.64**
10	10	300	3.64	**2.79**
10	40	100	1.51	0.12
25	10	100	54.17	**1.76**
25	10	200	58.14	**13.64**
25	10	300	66.02	**16.63**
25	20	100	9.40	0.45
25	20	200	8.74	**6.83**
25	20	300	10.18	**8.78**
25	30	100	3.80	0.13
25	30	200	1.79	0.85
25	30	300	1.30	**1.29**
25	40	100	3.13	0.00

Table: Bound gaps

Question: How to incorporate DW information without branch-and-price?

Solver cuts

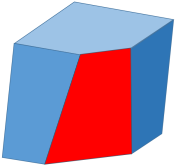
Let z^{LP+} denote the bound obtained by Gurobi at the root node (LP bound enhanced by Gurobi presolve and cuts)

$\lvert K\rvert$	$\lvert M\rvert$	$\lvert N\rvert$	$(z^{DW} - z^{LP})/\lvert z^{DW}\rvert$ (%)	$(z^{DW} - z^{LP+})/\lvert z^{DW}\rvert$ (%)
10	10	100	5.65	**4.13**
10	10	200	3.53	**2.64**
10	10	300	3.64	**2.79**
10	40	100	1.51	0.12
25	10	100	54.17	**1.76**
25	10	200	58.14	**13.64**
25	10	300	66.02	**16.63**
25	20	100	9.40	0.45
25	20	200	8.74	**6.83**
25	20	300	10.18	**8.78**
25	30	100	3.80	0.13
25	30	200	1.79	0.85
25	30	300	1.30	**1.29**
25	40	100	3.13	0.00

Table: Bound gaps

Question: How to incorporate DW information without branch-and-price?

We can add a single cutting plane $c^\top x \geq z^{DW}$ into the solver



- Straightforward, easy to implement
- Requires only one cut
- Often performs very badly in practice...

Explanation: high dual degeneracy (large/high-dim. optimal face) in LP relaxation

Simple Observation

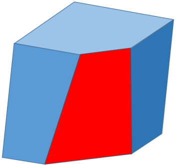
Let P be a polyhedron in $\mathbb{R}^n$. If neither $c^\top x \leq v$ nor $c^\top x \geq v$ is valid for P, then

$$\dim(P \cap \{x : c^\top x = v\}) = \dim(P) - 1.$$

- Ineffective solver cutting planes
- Ineffective branching decisions
- Serious degeneracy issues in the dual LP

Objective function cut

We can add a single cutting plane $c^\top x \geq z^{DW}$ into the solver



- Straightforward, easy to implement
- Requires only one cut
- Often performs very badly in practice...

Explanation: high dual degeneracy (large/high-dim. optimal face) in LP relaxation

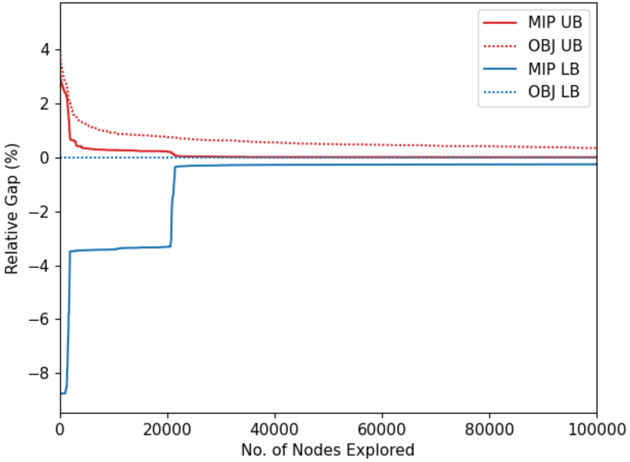
Simple Observation

Let P be a polyhedron in $\mathbb{R}^n$. If neither $c^\top x \leq v$ nor $c^\top x \geq v$ is valid for P, then

$$\dim(P \cap \{x : c^\top x = v\}) = \dim(P) - 1.$$

- Ineffective solver cutting planes
- Ineffective branching decisions
- Serious degeneracy issues in the dual LP

Original formulation vs. objective function cut



Question: Is there a better way?

Dantzig-Wolfe block cuts

Remember:

$$z^{DW} = \min\left\{c^\top x \;:\; Ax \geq b,\; x_{I(j)} \in \text{conv}(Q^j),\; j = 1, \ldots, q\right\}$$

$\Longrightarrow$ DW bound z^{DW} can be obtained by adding cuts valid for $\{Q^j\}_{j=1}^q$

Definition

We call a cut a Dantzig-Wolfe Block (DWB) cut if it is of the form

$$\pi^\top x_{I(j)} \geq D_j(\pi)$$

for some $j \in J$, where $D_j(\pi) = \min\{\pi^\top y : y \in Q^j\}$

- DWB cuts together with $Ax \geq b$ recover the DW bound z^{DW}
- Existing cutting plane approaches [Ralphs et al. 2003, Ralphs and Galati 2005, Avella et al. 2010]
- Under some conditions these cuts define high dimensional "faces" of the MIP

Question: Do we need a lot of DWB cuts to obtain z^{DW} ?

Dantzig-Wolfe block cuts

Remember:

$$z^{DW} = \min\left\{c^\top x \;:\; Ax \geq b,\; x_{I(j)} \in \text{conv}(Q^j),\; j = 1,\ldots,q\right\}$$

$\Longrightarrow$ DW bound z^{DW} can be obtained by adding cuts valid for $\{Q^j\}_{j=1}^q$

Definition

We call a cut a Dantzig-Wolfe Block (DWB) cut if it is of the form

$$\pi^\top x_{I(j)} \geq D_j(\pi)$$

for some $j \in J$, where $D_j(\pi) = \min\{\pi^\top y : y \in Q^j\}$

- DWB cuts together with $Ax \geq b$ recover the DW bound z^{DW}
- Existing cutting plane approaches [Ralphs et al. 2003, Ralphs and Galati 2005, Avella et al. 2010]
- Under some conditions these cuts define high dimensional "faces" of the MIP

Question: Do we need a lot of DWB cuts to obtain z^{DW} ?

Dantzig-Wolfe block cuts

Remember:

$$z^{DW} = \min\left\{c^\top x \ : \ Ax \geq b,\ x_{I(j)} \in \text{conv}(Q^j),\ j = 1, \ldots, q\right\}$$

$\implies$ DW bound z^{DW} can be obtained by adding cuts valid for $\{Q^j\}_{j=1}^q$

Definition

We call a cut a Dantzig-Wolfe Block (DWB) cut if it is of the form

$$\pi^\top x_{I(j)} \geq D_j(\pi)$$

for some $j \in J$, where $D_j(\pi) = \min\{\pi^\top y : y \in Q^j\}$

- DWB cuts together with $Ax \geq b$ recover the DW bound z^{DW}
- Existing cutting plane approaches [Ralphs et al. 2003, Ralphs and Galati 2005, Avella et al. 2010]
- Under some conditions these cuts define high dimensional "faces" of the MIP

Question: Do we need a lot of DWB cuts to obtain z^{DW} ?

Dantzig-Wolfe block cuts

Remember:

$$z^{DW} = \min\left\{c^\top x \ : \ Ax \geq b,\ x_{I(j)} \in \text{conv}(Q^j),\ j = 1, \ldots, q\right\}$$

$\Longrightarrow$ DW bound z^{DW} can be obtained by adding cuts valid for $\{Q^j\}_{j=1}^q$

Definition

We call a cut a Dantzig-Wolfe Block (DWB) cut if it is of the form

$$\pi^\top x_{I(j)} \geq D_j(\pi)$$

for some $j \in J$, where $D_j(\pi) = \min\{\pi^\top y : y \in Q^j\}$

- DWB cuts together with $Ax \geq b$ recover the DW bound z^{DW}
- Existing cutting plane approaches [Ralphs et al. 2003, Ralphs and Galati 2005, Avella et al. 2010]
- Under some conditions these cuts define high dimensional "faces" of the MIP

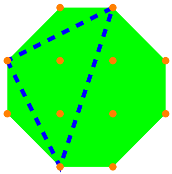
Question: Do we need a lot of DWB cuts to obtain z^{DW} ?

Column generation gives an inner approximation

Column generation

$$
\begin{aligned}
\min \ & \sum_{i \in I} c_i x_i \\
\text{s.t. } & x_{I(j)} = \sum_{v \in \hat{V}^j} \lambda_v v + \sum_{r \in \hat{R}^j} \mu_r r, \quad j \in J \quad (\pi^j) \\
& \sum_{v \in \hat{V}^j} \lambda_v = 1, \quad j \in J \quad (\theta_j) \\
& Ax \geq b, \quad (\beta) \\
& \lambda \geq 0, \ \mu \geq 0
\end{aligned}
$$

Inner approximation



- For $\tau = 1, 2, \ldots,$ we solve the following subproblem for each block $j \in J$:

$$D_j(\pi^j) := \min \left\{ (\pi^j)^\top v : v \in Q^j \right\}.$$

to generate a new point $v \in V^j$ or a ray $r \in R^j$

Notice: Such a point $v \in V^j$ gives a valid inequality for Q^j :

$$(\pi^j)^\top x_{I(j)} \geq (\pi^j)^\top v = D_j(\pi^j)$$

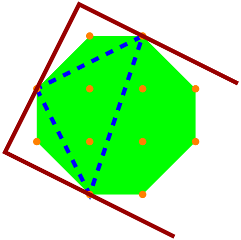
Inner and outer approximations

Build an inner approx. for each conv(Q^j)

$$
\begin{aligned}
\min \ & \sum_{i \in I} c_i x_i \\
\text{s.t. } & x_{I(j)} = \sum_{v \in \hat{V}^j} \lambda_v v + \sum_{r \in \hat{R}^j} \mu_r r, \quad j \in J \quad (\pi^j) \\
& \sum_{v \in \hat{V}^j} \lambda_v = 1, \quad j \in J \quad (\theta_j) \\
& Ax \geq b, \quad (\beta) \\
& \lambda \geq 0, \ \mu \geq 0
\end{aligned}
$$

Build an outer approx. for each conv(Q^j)

$$
\begin{aligned}
\min \ & \sum_{i \in I} c_i x_i \\
\text{s.t. } & (\pi^j(\tau))^\top x_{I(j)} \geq D_j(\pi^j(\tau)), \quad j \in J, \ \tau \in \mathcal{T} \\
& Ax \geq b
\end{aligned}
$$



Inner and outer approximations

Build an inner approx. for each conv(Q^j)

$$
\begin{aligned}
\min\ & \sum_{i\in I} c_i x_i \\
\text{s.t. } & x_{I(j)} = \sum_{v\in \hat{V}^j} \lambda_v v + \sum_{r\in \hat{R}^j} \mu_r r, \quad j \in J \quad (\pi^j) \\
& \sum_{v\in \hat{V}^j} \lambda_v = 1, \quad j \in J \quad (\theta_j) \\
& Ax \geq b, \quad (\beta) \\
& \lambda \geq 0,\ \mu \geq 0
\end{aligned}
$$

Build an outer approx. for each conv(Q^j)

$$
\begin{aligned}
\min\ & \sum_{i\in I} c_i x_i \\
\text{s.t. } & (\pi^j(\tau))^\top x_{I(j)} \geq D_j(\pi^j(\tau)), \quad j \in J,\ \tau \in \mathcal{T} \\
& Ax \geq b
\end{aligned}
$$

- For $\tau = 1, 2, \ldots,$ we solve the following subproblem for each block $j \in J$:

$$D_j(\pi^j) := \min\left\{(\pi^j)^\top v : v \in Q^j\right\}$$

- At termination we have: $z^{DW} = b^\top \bar{\beta} + \sum_{j=1}^{q} \bar{\theta}_j$, and
  1. $\bar{\theta}_j \leq D_j(\bar{\pi}^j)$ for all $j \in J$ (Nonnegative reduced costs)
  2. $c_i = A_i^\top \bar{\beta} + \sum_{j:i\in I(j)} \bar{\pi}_i^j$, for all $i = 1, \ldots, n$ (Dual feasibility of master LP)

Last-iteration DWB cuts

- Let $\bar{\pi}^j$ be the dual variables associated with the constraints

$$x_{I(j)} = \sum_{v \in \hat{V}^j} \lambda_v v + \sum_{r \in \hat{R}^j} \mu_r r, \quad j \in J$$

  at the last iteration

- We know (not only for $\bar{\pi}^j$ but for any π^j)

$$(\bar{\pi}^j)^\top x_{I(j)} \geq D_j(\bar{\pi}^j) \quad \longleftarrow \min\left\{(\pi^j)^\top v : v \in Q^j\right\}$$

  are valid for the MIP for all $i \in J$

- These cuts together with linking constraints imply the objective function cut

$$c^\top x \geq z^{DW}$$

Theorem

For $j \in J$, let $D_j(\bar{\pi}^j) = \min\{(\bar{\pi}^j)^\top y : y \in Q^j\}$ be the block subproblem in the last iteration of DW decomposition. Then

$$\begin{aligned} z^{DW} = \min\ & c^\top x \\ \text{s.t. } & (\bar{\pi}^j)^\top x_{I(j)} \geq D_j(\bar{\pi}^j), \quad j \in J, \\ & Ax \geq b \end{aligned}$$

Proof of the Theorem

The "$\geq$" direction is from validity of the DWB cuts. We only prove the "$\leq$" direction. For $j \in J$, let $D_j(\bar{\pi}^j) = \min\{(\bar{\pi}^j)^\top y : y \in Q^j\}$ be the block subproblem in the last iteration of DW decomposition. Then

$$\begin{aligned} z^{DW} = \min\ & c^\top x \\ \text{s.t. } & (\bar{\pi}^j)^\top x_{I(j)} \geq D_j(\bar{\pi}^j), \quad j \in J, \\ & Ax \geq b. \end{aligned}$$

For all x satisfying $Ax \geq b,\ (\bar{\pi}^j)^\top x_{I(j)} \geq D_j(\bar{\pi}^j),\ j \in J$, we have

$$c^\top x = \sum_{i=1}^n c_i x_i \underbrace{=}_{\text{Dual feasibility}} \sum_{i=1}^n \left[x_i A_i^\top \bar{\beta} + \sum_{j:i \in I(j)} x_i \bar{\pi}_i^j \right] = \underbrace{(\bar{\beta})}_{\geq 0}{}^\top \underbrace{Ax}_{\geq b} + \sum_{j=1}^q \underbrace{(\bar{\pi}^j)^\top x_{I(j)}}_{\geq D_j(\bar{\pi}^j)}$$

$$\geq b^\top \bar{\beta} + \sum_{j=1}^q D_j(\bar{\pi}^j) \underbrace{\geq}_{\text{Nonneg. reduced costs}} b^\top \bar{\beta} + \sum_{j=1}^q \bar{\theta}_j = z^{DW}. \quad \square$$

Comparing different approaches

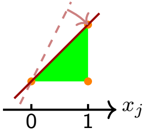
$(\|K\|, \|M\|, \|N\|)$	# solved instances			Avg B&C time (s)			Avg opt gap (%)		
	MIP	OBJ	DWB	MIP	OBJ	DWB	MIP	OBJ	DWB
(10,10,100)	28/30	18/30	**30**/30	$\geq$ 65	$\geq$298	2	0.04	0.03	0.00
(10,10,200)	11/30	10/30	**30**/30	$\geq$424	$\geq$457	6	0.27	0.09	0.00
(10,10,300)	7/30	11/30	**30**/30	$\geq$489	$\geq$454	41	0.21	0.09	0.00
(25,10,100)	**30**/30	**30**/30	**30**/30	$<$ 1	1	$<$ 1	0.00	0.00	0.00
(25,10,200)	**30**/30	**30**/30	**30**/30	2	25	$<$ 1	0.00	0.00	0.00
(25,10,300)	**30**/30	29/30	**30**/30	6	$\geq$ 40	$<$ 1	0.00	0.00	0.00
(25,20,200)	29/30	9/30	**30**/30	$\geq$ 44	$\geq$499	3	0.02	0.08	0.00
(25,20,300)	22/30	2/30	**29**/30	$\geq$224	$\geq$590	$\geq$ 48	0.24	0.17	0.00
(25,30,300)	**1**/30	0/30	0/30	$\geq$595	$\geq$600	$\geq$600	0.84	0.53	0.40

- MIP: original formulation
- OBJ: original formulation + objective cut
- DWB: original formulation + last-iteration DWB cuts

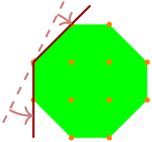
  (Note: Even if we terminate early, we still obtain DWB cuts)

- DWB cuts have good computational performance, moreover,
- It is possible to strengthen the DWB cuts to further reduce dual degeneracy

1. Disjunctive coefficient strengthening
   - Variant of [Andersen and Pochet 2010]
   - Sequentially strengthen the coefficients of the cut
   - Each step requires solving one block MIP



2. Strengthening via tilting
   - Variant of local cuts [Chvatal et al. 2013]
   - Each tilting requires solving two sequences of block MIPs
   - (Depth-d) recursive tilting: one cutting plane $\Rightarrow$ multiple (2^d) cutting planes



Strengthened DWB cuts

$(\lvert K\rvert, \lvert M\rvert, \lvert N\rvert)$	# solved instances			Avg B&C time (s)			Avg opt gap (%)		
	DWB	STR	D6T	DWB	STR	D6T	DWB	STR	D6T
(10,10,100)	**30**/30	**30**/30	**30**/30	2	$<$ 1	2	0.00	0.00	0.00
(10,10,200)	**30**/30	**30**/30	**30**/30	6	1	7	0.00	0.00	0.00
(10,10,300)	**30**/30	**30**/30	**30**/30	41	2	17	0.00	0.00	0.00
(25,10,100)	**30**/30	**30**/30	**30**/30	$<$ 1	$<$ 1	$<$ 1	0.00	0.00	0.00
(25,10,200)	**30**/30	**30**/30	**30**/30	$<$ 1	$<$ 1	3	0.00	0.00	0.00
(25,10,300)	**30**/30	**30**/30	**30**/30	$<$ 1	$<$ 1	9	0.00	0.00	0.00
(25,20,200)	**30**/30	**30**/30	**30**/30	3	2	4	0.00	0.00	0.00
(25,20,300)	29/30	**30**/30	**30**/30	$\geq$ 48	5	22	0.00	0.00	0.00
(25,30,300)	0/30	2/30	**6**/30	$\geq$600	$\geq$585	$\geq$559	0.40	0.31	0.19

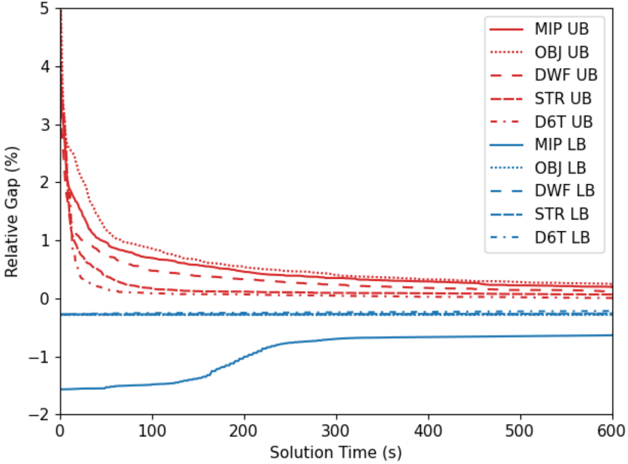
- STR: original formulation + last-iteration DWB cuts with disjunctive coefficient strengthening
- D6T: original formulation + last-iteration DWB cuts with disjunctive coefficient strengthening and tilting of depth 6

Comparing computational performance of cuts

$(\|K\|, \|M\|, \|N\|)$	# B&C nodes				
	MIP	OBJ	DWB	STR	D6T
(10,10,100)	$\geq$ 22761	$\geq$1698156	4481	151	1
(10,10,200)	$\geq$2778880	$\geq$3898948	6828	258	8
(10,10,300)	$\geq$2616971	$\geq$2781174	131139	495	5
(25,10,100)	93	1123	1	1	1
(25,10,200)	3853	20272	1	1	1
(25,10,300)	8488	$\geq$ 25864	1	1	1
(25,20,200)	$\geq$ 27987	$\geq$ 438310	2255	335	1
(25,20,300)	$\geq$ 113353	$\geq$ 410728	$\geq$ 13861	1003	1
(25,30,300)	$\geq$ 81545	$\geq$ 379999	$\geq$105115	$\geq$67556	$\geq$6312

- MIP: original formulation
- OBJ: original formulation + objective cut
- DWB: original formulation + last-iteration DWB cuts
- STR: original formulation + last-iteration DWB cuts with coeff. strengthening
- D6T: original formulation + last-iteration DWB cuts with coeff. strengthening and tilting of depth 6

Comparing computational performance of cuts



Time permitting: Dual degeneracy

Proposition

Assume $w^* \in \mathbb{R}^m_+$ is a dual basic optimal solution of an LP with n variables and m inequality constraints. Then, the optimal face of the LP has dimension at most $n - \|w^*\|_0$. Furthermore, if w^* is the unique dual optimal solution, then the optimal face of the LP has dimension exactly $n - \|w^*\|_0$.

Table: Relative Dual Degeneracy Levels ($1 - \|w^*\|_0/n$) for Different Formulations

$(\|K\|, \|M\|, \|N\|)$	$(1 - |w^*|_0/n)\times$100%					
	MIP	OBJ	DWB	STR	D3T	D6T
(10,10,100)	50.65%	99.91%	56.98%	35.05%	1.36%	1.54%
(10,10,200)	53.30%	99.95%	53.77%	38.02%	7.19%	0.28%
(10,10,300)	54.23%	99.97%	53.50%	39.77%	17.99%	0.61%
(25,10,100)	44.57%	99.92%	40.47%	30.85%	3.70%	3.70%
(25,10,200)	49.75%	99.96%	47.23%	37.99%	3.71%	1.68%
(25,10,300)	51.73%	99.97%	56.84%	48.89%	8.05%	1.17%
(25,20,200)	52.44%	99.98%	62.38%	38.30%	0.93%	1.25%
(25,20,300)	54.69%	99.98%	62.72%	42.19%	4.78%	0.62%
(25,30,300)	55.68%	99.99%	78.99%	48.56%	8.59%	3.27%

Using ML(!) to distinguish between easy and hard instances

Train:

- Run half of the MKAP instances with and without DWB cuts (10 min time limit)
- Record simple features: $z^{LP}, z^{LP+}, z^{DW}, z^{UB}$ and LB at termination for both
- Label instances if DW performs better (10% faster or better gap at termination)
- Run a shallow decision tree to obtain the simple rule:

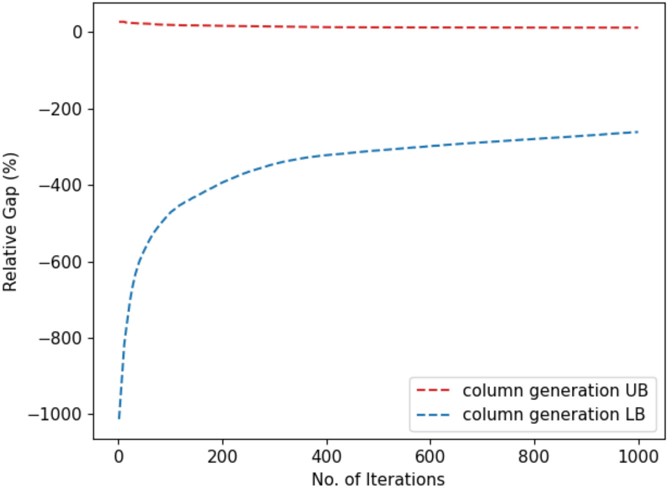
$$\text{DWB cuts are better if :} \quad (z^{DW} - z^{\text{LB}})/z^{\text{UB}} > 0.05\%$$

Test:

- On the rest, run DW on 1 thread and MIP in 3, stop when DW is done
- Allocate all processors to the more promising method

	MIP	DW-STR	HYB
Number of Instances Solved	113/155	133/155	134/155
Average Optimality Gap (%)	0.13%	0.05%	0.05%
Average Solution Time (s)	205	116	109

Time permitting: Solving the DW relaxation via column generation



Time permitting: Solving the DW relaxation in practice

- We do not try to solve the DW relaxation (or, to generate cuts)
- Instead we solve the Lagrangian relaxation of

$$
\begin{aligned}
z^{DW} = \min \quad & c^\top x \\
\text{s.t.} \quad & y^j \in \operatorname{conv}(Q^j), && j = 1, \dots, q, \\
& y^j = x_{I(j)}, && j = 1, \dots, q, && (\pi^j) \\
& Ax \geq b && && (\beta)
\end{aligned}
$$

- After dualizing the coupling constraints

$$
z^{DW} = \max_{\beta \geq 0, \pi} \; z(\pi, \beta)
$$

where

$$
\begin{aligned}
z(\pi, \beta) = \min \quad & \overbrace{c^\top x + \beta^\top (b - Ax) \; + \; \sum_{j=1}^{q} (\pi^j)^\top (y^j - x_{I(j)})}^{\text{coefficients of } x \text{ must be } 0}, \\
\text{s.t.} \quad & y^j \in Q^j, \quad j = 1, \dots, q
\end{aligned}
$$

- Az $z(\pi, \beta)$ is a piecewise linear concave function, computing z^{DW} is a (nonsmooth) convex optimization problem

Time permitting: Computing (last-iteration) cuts in practice

$$z^{DW} = \max_{\beta \geq 0, \pi} \; z(\pi, \beta) \quad \text{s.t.} \quad \sum_{j : i \in I(j)} \pi_i^j + \beta^\top A_i = c_i, \quad i = 1, \dots, n,$$

where

$$z(\pi, \beta) = \min \quad \beta^\top b \; + \; \sum_{j=1}^{q} (\pi^j)^\top y^j, \quad \text{s.t.} \quad y^j \in Q^j, \quad j = 1, \dots, q$$

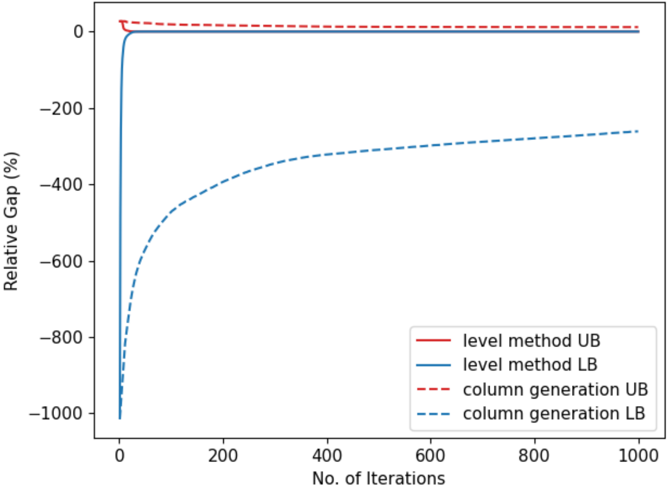
(We use the level method to update π and β for better computational performance)

We can show that at any iteration τ we have

$$\begin{aligned} z\big(\pi(\tau), \beta(\tau)\big) \leq \min \; & c^\top x \\ \text{s.t.} \; & \big(\pi^j(\tau)\big)^\top x_{I(j)} \geq D_j\big(\pi^j(\tau)\big) \quad \longleftarrow \min\left\{(\pi^j)^\top v : v \in Q^j\right\} \\ & Ax \geq b \end{aligned}$$

- If the Lagrangian dual is solved to optimality, we have cuts that recover z^{DW}
- If terminated early, we have a set of cuts recovering the current dual bound

Time permitting: Column generation vs. Level method



Thank you!

Contact:
rui.chen@cornell.edu

Reference:
Chen, R., Günlük, O. and Lodi, A., 2024. Recovering Dantzig–Wolfe Bounds by Cutting Planes. *Operations Research*.

Algorithm The Level Method for Solving Lagrangian Dual

Initialize:

$\hat{V}^j \leftarrow \emptyset$, $\hat{R}^j \leftarrow \emptyset$, $j = 1, 2, \ldots, p$

$\bar{z} \leftarrow$ an upper bound of z_D

LB $\leftarrow -\infty$, UB $\leftarrow \infty$, $t \leftarrow 0$

Main Loop: $t \leftarrow t + 1$, **solve:**

$$\text{UB} \leftarrow \max \sum_{j=1}^{q} \theta_j + b^\top \beta \tag{1a}$$

$$\text{s.t. } \theta_j \leq v^\top \pi^j, \qquad v \in \hat{V}^j,\ j = 1, \ldots, q, \tag{1b}$$

$$r^\top \pi^j \geq 0, \qquad r \in \hat{R}^j,\ j = 1, \ldots, q, \tag{1c}$$

$$\sum_{j=1}^{q} \theta_j + b^\top \beta \leq \bar{z}, \qquad \text{(some known UB on } z^{DW}\text{)} \tag{1d}$$

$$\sum_{j: i \in I(j)} \pi_i^j + \beta^\top A_i = c_i, \qquad i = 1, \ldots, n, \tag{1e}$$

$$\tag{1g}$$

$$\beta \geq 0. \tag{1f}$$

if $LB = -\infty$ **then**

$(\bar{\pi}, \bar{\beta}) \leftarrow$ optimal value of (π, β) in (1)

else

solve:

$$\min \left\| \left(\pi - \bar{\pi}, \beta - \bar{\beta}\right) \right\|_2^2$$

$$\text{s.t. } \sum_{j=1}^{q} \theta_j + b^\top \beta \geq 0.7 \cdot \text{UB} + 0.3 \cdot \text{LB} \tag{2}$$

$$\text{(1b)} - \text{(1g)}$$

$(\bar{\pi}, \bar{\beta}) \leftarrow$ optimal value of (π, β) in (2)

end if

Algorithm The Level Method for Solving Lagrangian Dual – continued

1: **for** $j = 1, 2, \ldots, q$ **do**
2:     **solve pricing problem for** $\pi^j = \bar{\pi}^j$
3:     **if** bounded **then**
4:         let v^j denote an optimal solution
5:         $\hat{V}^j \leftarrow \hat{V}^j \cup \{v^j\}$
6:     **else**
7:         let r^j denote an extreme ray of conv(Q^j) with $(\pi^j)^\top r^j < 0$
8:         $\hat{R}^j \leftarrow \hat{R}^j \cup \{r^j\}$
9:     **end if**
10: **end for**
11: LB $\leftarrow \max\left\{\text{LB}, \sum_{j=1}^{q} D_j(\bar{\pi}^j) + b^\top \bar{\beta}\right\}$
12: **if** UB-LB is small enough **then**
13:     **return** LB
14: **else**
15:     **go to** step 2
16: **end if**