Eliminating redundant columns from column generation subproblems using classical Benders' cuts

Marco Lübbecke [1] Stephen Maher [2] *Jonas Witt* [1]

[1]RWTH Aachen University

[2]Lancaster University

SCIP workshop 2018 · Aachen · 08/03/2018

Dantzig-Wolfe reformulation for IPs

$$\begin{aligned} \min \quad & c^T x \\ \text{s.t.} \quad & Ax \geq b \\ & Dx \geq d \\ & x \in \mathbb{Z}^n_{\geq 0} \end{aligned}$$

- original problem

Dantzig-Wolfe reformulation for IPs

$$\begin{aligned} \min \quad & c^T x \\ \text{s.t.} \quad & Ax \geq b \\ & Dx \geq d \\ & x \in \mathbb{Z}^n_{\geq 0} \end{aligned}$$

- original problem
- "discretize" $Dx \geq d$:

$$\{x \in \mathbb{Z}^n_{\geq 0} : Dx \geq d\} = \bigcup_p \{x^p\}$$

- substitute x-variables with λ-variables

$$\begin{aligned} \textstyle\sum_p x^p \lambda_p &= x \\ \textstyle\sum_p \lambda_p &= 1 \\ \lambda &\in \{0,1\}^q \end{aligned}$$

Dantzig-Wolfe reformulation for IPs

$$\begin{aligned} \min \quad & c^T x \\ \text{s.t.} \quad & Ax \geq b \\ & Dx \geq d \\ & x \in \mathbb{Z}^n_{\geq 0} \end{aligned} \quad \xrightarrow[\text{reformulation}]{\text{Dantzig-Wolfe}} \quad \begin{aligned} \min \quad & \textstyle\sum_p c^T x^p \lambda_p \\ \text{s.t.} \quad & \textstyle\sum_p A x^p \lambda_p \geq b \\ & \textstyle\sum_p \lambda_p = 1 \\ & \lambda \in \{0,1\}^q \end{aligned}$$

- original problem
- master IP

- "discretize" $Dx \geq d$:

$$\{x \in \mathbb{Z}^n_{\geq 0} : Dx \geq d\} = \textstyle\bigcup_p \{x^p\}$$

- substitute x-variables with λ-variables

$$\begin{aligned} \textstyle\sum_p x^p \lambda_p &= x \\ \textstyle\sum_p \lambda_p &= 1 \\ \lambda &\in \{0,1\}^q \end{aligned}$$

Operations Research | RWTH AACHEN UNIVERSITY

Dantzig-Wolfe reformulation for IPs

$$\begin{aligned} \min\ & c^T x \\ \text{s.t. } & Ax \geq b \\ & Dx \geq d \\ & x \in \mathbb{Z}^n_{\geq 0} \end{aligned} \quad \xrightarrow{\text{Dantzig-Wolfe reformulation}} \quad \begin{aligned} \min\ & \textstyle\sum_p c^T x^p \lambda_p \\ \text{s.t. } & \textstyle\sum_p A x^p \lambda_p \geq b \\ & \textstyle\sum_p \lambda_p = 1 \\ & \lambda \in \{0,1\}^q \end{aligned}$$

- original problem
- "discretize" $Dx \geq d$:

$$\{x \in \mathbb{Z}^n_{\geq 0} : Dx \geq d\} = \textstyle\bigcup_p \{x^p\}$$

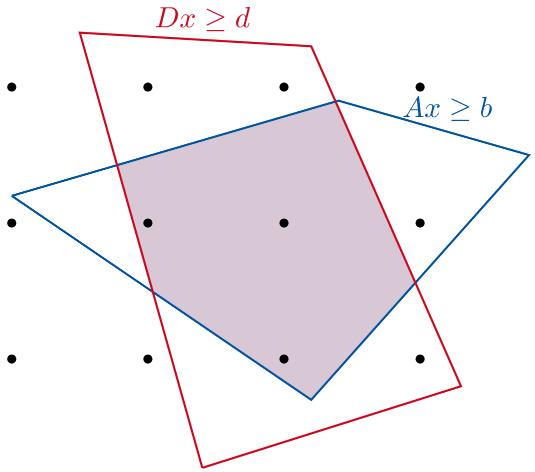
- substitute x-variables with λ-variables

$$\begin{aligned} \textstyle\sum_p x^p \lambda_p &= x \\ \textstyle\sum_p \lambda_p &= 1 \\ \lambda &\in \{0,1\}^q \end{aligned}$$

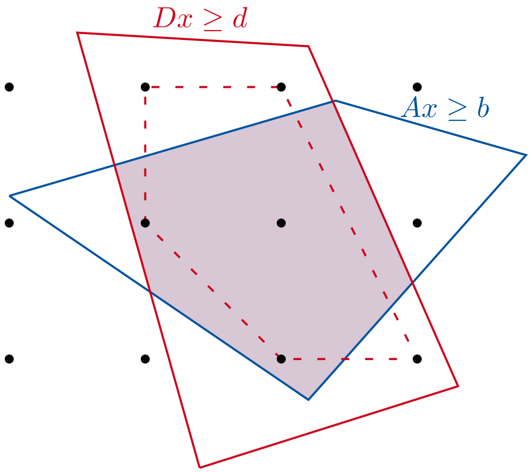
- master IP
- solve master LP with col.gen.
- col.gen. subproblem

$$\begin{aligned} \min\ & \texttt{redcost}(x) \\ \text{s.t. } & Dx \geq d \\ & x \in \mathbb{Z}^n_{\geq 0} \end{aligned}$$

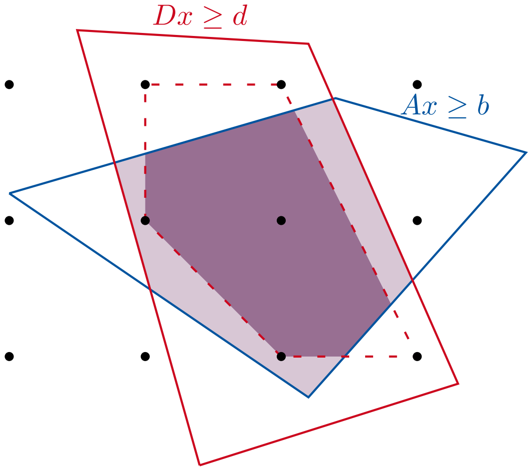
Dantzig-Wolfe reformulation for IPs



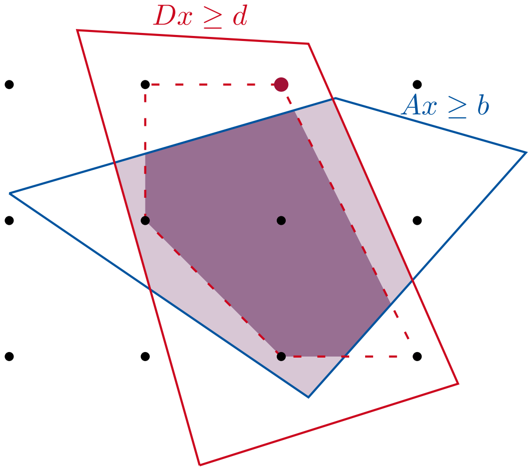
Dantzig-Wolfe reformulation for IPs



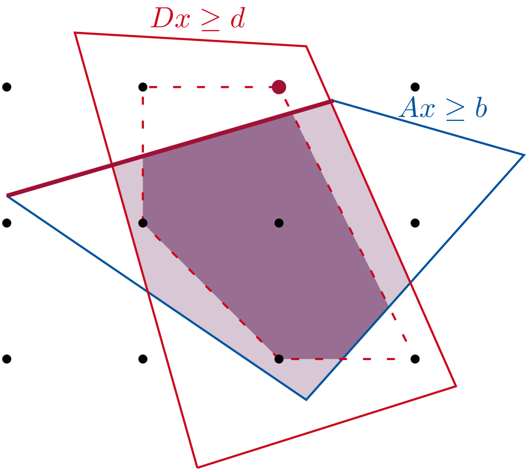
Dantzig-Wolfe reformulation for IPs



Dantzig-Wolfe reformulation for IPs



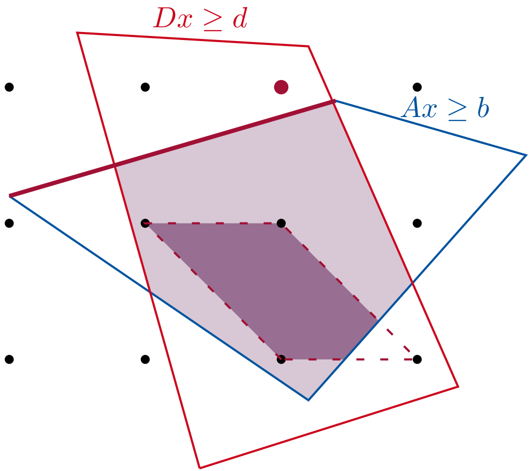
Dantzig-Wolfe reformulation for IPs



Operations Research

RWTH AACHEN UNIVERSITY

Dantzig-Wolfe reformulation for IPs



Literature

- "*A [column] is redundant when the [master IP] admits an optimal solution that can be expressed without this [column].*"
  Vanderbeck and Savelsbergh (2006)

Literature

- "*A [column] is redundant when the [master IP] admits an optimal solution that can be expressed without this [column].*" Vanderbeck and Savelsbergh (2006)
- refine subproblem to eliminate (some) redundant columns
  - until now: only domain propagation for tighter variable bounds in subproblems Vanderbeck and Savelsbergh (2006); Gamrath and Lübbecke (2010)
  - this talk: add inequalities/cuts to subproblems

Literature

- "*A [column] is redundant when the [master IP] admits an optimal solution that can be expressed without this [column].*" Vanderbeck and Savelsbergh (2006)
- refine subproblem to eliminate (some) redundant columns
  - until now: only domain propagation for tighter variable bounds in subproblems Vanderbeck and Savelsbergh (2006); Gamrath and Lübbecke (2010)
  - this talk: add inequalities/cuts to subproblems
- column is *strongly redundant* if it is not part of any optimal solution to the master IP

Redundant columns

$$
\begin{array}{rlclcll}
z^* = \min & c_1^T x^1 & + & c_2^T x^2 & & & \\
\text{s. t.} & A_1 x^1 & + & A_2 x^2 & \geq & b & \\
& D_1 x^1 & & & \geq & d_1 & \\
& & & D_2 x^2 & \geq & d_2 & \\
& & & x^k & \in & \mathbb{Z}_{\geq 0}^{n_k} & \forall k \in \{1, 2\}
\end{array}
$$

- set F of feasible solutions
- set F^k of feasible solution to subproblem $k \in \{1, 2\}$

$$F^k = \{x^k \in \mathbb{Z}_{\geq 0}^{n_k} : D_k x^k \geq d_k\}$$

Redundant columns

$$
\begin{array}{rccccll}
z^* = \min & c_1^T x^1 & + & c_2^T x^2 & & & \\
\text{s.t.} & A_1 x^1 & + & A_2 x^2 & \geq & b & \\
& D_1 x^1 & & & \geq & d_1 & \\
& & & D_2 x^2 & \geq & d_2 & \\
& & & x^k & \in & \mathbb{Z}_{\geq 0}^{n_k} & \forall k \in \{1,2\}
\end{array}
$$

- set F of feasible solutions
- set F^k of feasible solution to subproblem $k \in \{1,2\}$

$$F^k = \{x^k \in \mathbb{Z}_{\geq 0}^{n_k} : D_k x^k \geq d_k\}$$

- how do we check if a column $\bar{x}^1 \in F^1$ is strongly redundant?

Redundant columns

$$
\begin{array}{rlclcll}
z^* = \min & c_1^T x^1 & + & c_2^T x^2 & & & \\
\text{s.t.} & A_1 x^1 & + & A_2 x^2 & \geq & b & \\
& D_1 x^1 & & & \geq & d_1 & \\
& & & D_2 x^2 & \geq & d_2 & \\
& & & x^k & \in & \mathbb{Z}^{n_k}_{\geq 0} & \forall k \in \{1,2\}
\end{array}
$$

- set F of feasible solutions
- set F^k of feasible solution to subproblem $k \in \{1,2\}$

$$F^k = \{x^k \in \mathbb{Z}^{n_k}_{\geq 0} : D_k x^k \geq d_k\}$$

- how do we check if a column $\bar{x}^1 \in F^1$ is strongly redundant?

$\rightarrow$ check if $\exists \bar{x}^2 \in F^2$ with

1. $(\bar{x}^1, \bar{x}^2) \in F$
2. $c_1^T \bar{x}^1 + c_2^T \bar{x}^2 \leq z^*$

Redundant columns

- check strong redundancy of $\bar{x}^1$ with feasibility problem:

$$
\begin{array}{llll}
\min & 0 & & \\
\text{s.t.} & A_2 x^2 & \geq & b - A_1 \bar{x}^1 \\
 & D_2 x^2 & \geq & d_2 \\
 & c_2^T x^2 & \leq & z^* \quad - c_1^T \bar{x}^1 \\
 & x^2 & \in & \mathbb{Z}_{\geq 0}^{n_2}
\end{array}
$$

Redundant columns

- check strong redundancy of $\bar{x}^1$ with feasibility problem:

$$
\begin{array}{llll}
\min & 0 & & \\
\text{s.t.} & A_2 x^2 & \geq & b - A_1 \bar{x}^1 \\
 & D_2 x^2 & \geq & d_2 \\
 & c_2^T x^2 & \leq & z^* \quad - c_1^T \bar{x}^1 \\
 & x^2 & \in & \mathbb{Z}^{n_2}_{\geq 0}
\end{array}
$$

- problem is hard to solve
- how subproblem should be refined to eliminate redundant $\bar{x}^1$ is not clear

Redundant columns

- check strong redundancy of $\bar{x}^1$ with feasibility problem:

$$\begin{array}{lrcl} \min & 0 & & \\ \text{s.t.} & A_2 x^2 & \geq & b - A_1 \bar{x}^1 \\ & D_2 x^2 & \geq & d_2 \\ & c_2^T x^2 & \leq & z^{UB} - c_1^T \bar{x}^1 \\ & x^2 & \in & \mathbb{Z}_{\geq 0}^{n_2} \end{array}$$

- problem is hard to solve
- how subproblem should be refined to eliminate redundant $\bar{x}^1$ is not clear

$\rightarrow$ upper bound z^{UB} instead of z^*

Redundant columns

- check strong redundancy of $\bar{x}^1$ with feasibility problem:

$$\begin{array}{llcl} \min & 0 & & \\ \text{s.t.} & A_2 x^2 & \geq & b - A_1 \bar{x}^1 \\ & D_2 x^2 & \geq & d_2 \\ & c_2^T x^2 & \leq & z^{UB} - c_1^T \bar{x}^1 \\ & x^2 & \in & \mathbb{Q}^{n_2}_{\geq 0} \end{array}$$

- problem is hard to solve
- how subproblem should be refined to eliminate redundant $\bar{x}^1$ is not clear

$\rightarrow$ upper bound z^{UB} instead of z^*

$\rightarrow$ relax integrality of x^2

Redundant columns

- check strong redundancy of $\bar{x}^1$ with feasibility problem:

$$\begin{array}{lrcl} \min & 0 & & \\ \text{s.t.} & A_2 x^2 & \geq & b - A_1 \bar{x}^1 \\ & D_2 x^2 & \geq & d_2 \\ & c_2^T x^2 & \leq & z^{UB} - c_1^T \bar{x}^1 \\ & x^2 & \in & \mathbb{Q}_{\geq 0}^{n_2} \end{array}$$

- problem is hard to solve
- how subproblem should be refined to eliminate redundant $\bar{x}^1$ is not clear

→ upper bound z^{UB} instead of z^*

→ relax integrality of x^2

→ classical Benders' feasibility cuts to refine subproblem

Operations Research

RWTH AACHEN UNIVERSITY

Subproblem refining inequalities

$$\begin{array}{llcl}
\min & 0 & & \\
\text{s. t.} & A_2 x^2 & \geq & b - A_1 \bar{x}^1 \\
 & D_2 x^2 & \geq & d_2 \\
 & c_2^T x^2 & \leq & z^{UB} - c_1^T \bar{x}^1 \\
 & x^2 & \geq & 0
\end{array}$$

Subproblem refining inequalities

$$\begin{array}{lrcl} \min & 0 & & \\ \text{s. t.} & A_2 x^2 & \geq & b - A_1 \bar{x}^1 \\ & D_2 x^2 & \geq & d_2 \\ & c_2^T x^2 & \leq & z^{UB} - c_1^T \bar{x}^1 \\ & x^2 & \geq & 0 \end{array}$$

$$\begin{array}{lrcl} \max & \texttt{dualobj}(\pi, \bar{x}^1) & & \\ \text{s. t.} & (*)\ \pi & \leq & 0 \\ & \underbrace{\pi_A, \pi_D, -\pi_c}_{\pi} & \geq & 0 \end{array}$$

- dual polyhedron independent of $\bar{x}^1$

Operations Research | RWTH AACHEN UNIVERSITY

Subproblem refining inequalities

$$\begin{array}{llcl} \min & 0 & & \\ \text{s.t.} & A_2 x^2 & \geq & b - A_1 \bar{x}^1 \\ & D_2 x^2 & \geq & d_2 \\ & c_2^T x^2 & \leq & z^{UB} - c_1^T \bar{x}^1 \\ & x^2 & \geq & 0 \end{array}$$

$$\begin{array}{lrcl} \max & \texttt{dualobj}(\pi, \bar{x}^1) & & \\ \text{s.t.} & (*)\ \pi & \leq & 0 \\ & \underbrace{\pi_A, \pi_D, -\pi_c}_{\pi} & \geq & 0 \end{array}$$

- dual polyhedron independent of $\bar{x}^1$
- if primal infeasible, $\exists$ Farkas proof (dual ray) $\bar{\pi}$ with:

$$\texttt{dualobj}(\bar{\pi}, \bar{x}^1) > 0$$

Subproblem refining inequalities

$$\begin{array}{llll} \min & 0 & & \\ \text{s.t.} & A_2 x^2 & \geq & b - A_1 \bar{x}^1 \\ & D_2 x^2 & \geq & d_2 \\ & c_2^T x^2 & \leq & z^{UB} - c_1^T \bar{x}^1 \\ & x^2 & \geq & 0 \end{array}$$

$$\begin{array}{llll} \max & \texttt{dualobj}(\pi, \bar{x}^1) & & \\ \text{s.t.} & (*)\ \pi & \leq & 0 \\ & \underbrace{\pi_A, \pi_D, -\pi_c}_{\pi} & \geq & 0 \end{array}$$

- dual polyhedron independent of $\bar{x}^1$
- if primal infeasible, $\exists$ Farkas proof (dual ray) $\bar{\pi}$ with:

$$\texttt{dualobj}(\bar{\pi}, \bar{x}^1) > 0$$

$\rightarrow$ valid inequality for all x^1, not strongly redundant

$$\texttt{dualobj}(\bar{\pi}, x^1) \leq 0$$

Operations Research RWTH AACHEN UNIVERSITY

Subproblem refining inequalities

$$\overbrace{\pi_A^T(b - A_1\bar{x}^1) + \pi_D^T d_2 + \pi_c \cdot (z^{UB} - c_1^T \bar{x}^1)}^{\texttt{dualobj}(\pi, x^1)} \leq 0$$

Operations Research

RWTH AACHEN UNIVERSITY

Subproblem refining inequalities

$$\overbrace{\pi_A^T(b - A_1\bar{x}^1) + \pi_D^T d_2 + \pi_c \cdot (z^{UB} - c_1^T\bar{x}^1)}^{\texttt{dualobj}(\pi, x^1)} \leq 0$$

1. if $\pi_c = 0$:

$$\pi^T b + \mu^T d_2 \leq \pi^T A_1 x^1$$

feasibility subproblem cut

Subproblem refining inequalities

$$\overbrace{\pi_A^T(b - A_1\bar{x}^1) + \pi_D^T d_2 + \pi_c \cdot (z^{UB} - c_1^T\bar{x}^1)}^{\texttt{dualobj}(\pi, x^1)} \leq 0$$

1. if $\pi_c = 0$:

$$\pi^T b + \mu^T d_2 \leq \pi^T A_1 x^1$$

feasibility subproblem cut

2. if $\pi_c < 0$, normalize $\pi_c = -1$:

$$\pi_A^T b + \pi_D^T d_2 - z^{UB} \leq (\pi_A^T A_1 - c_1^T)x^1$$

optimality subproblem cut

Subproblem refining inequalities

$$\overbrace{\pi_A^T(b - A_1\bar{x}^1) + \pi_D^T d_2 + \pi_c \cdot (z^{UB} - c_1^T\bar{x}^1)}^{\texttt{dualobj}(\pi, x^1)} \leq 0$$

1. if $\pi_c = 0$:

$$\pi^T b + \mu^T d_2 \leq \pi^T A_1 x^1$$

feasibility subproblem cut

2. if $\pi_c < 0$, normalize $\pi_c = -1$:

$$\pi_A^T b + \pi_D^T d_2 - z^{UB} \leq (\pi_A^T A_1 - c_1^T)x^1$$

optimality subproblem cut

- alternative way without objective constraints:
  use dual instead of Farkas values, maximizing violation

Elimination of redundant columns

- pricing iteration with redundancy check:
  1. solve col.gen. subproblems
  2. for each subproblem solution
     - check redundancy with LP
     - if redundant, add subproblem cut
  3. if all found columns are redundant $\rightarrow$ 1

Elimination of redundant columns

- pricing iteration with redundancy check:
  1. solve col.gen. subproblems
  2. for each subproblem solution
     - check redundancy with LP
     - if redundant, add subproblem cut
  3. if all found columns are redundant $\rightarrow$ 1

+ potentially stronger dual bound with master LP
+ possibly "better" columns for the master IP

Elimination of redundant columns

- pricing iteration with redundancy check:
  1. solve col.gen. subproblems
  2. for each subproblem solution
     - check redundancy with LP
     - if redundant, add subproblem cut
  3. if all found columns are redundant $\rightarrow$ 1

\+ potentially stronger dual bound with master LP

\+ possibly "better" columns for the master IP

– solve LP for each subproblem solution

– subproblems can become more difficult to solve

Implementation

- implemention in generic BP&C solver GCG based on SCIP
- separator in the master problem
  - callbacks `SEPAEXECLP` and `SEPAEXECSOL` reduce the domain

    ```
    *result = (SCIP_RESULT) SCIP_REDUCEDDOM
    ```

  - cuts are by default only separated in the pricing loop

  $\rightarrow$ public methods for separation called in pricer

  - propagation of domain reductions

Implementation

- implemention in generic BP&C solver GCG based on SCIP
- separator in the master problem
  - callbacks SEPAEXECLP and SEPAEXECSOL reduce the domain

```
*result = (SCIP_RESULT) SCIP_REDUCEDDOM
```

  - cuts are by default only separated in the pricing loop

  $\rightarrow$ public methods for separation called in pricer
  - propagation of domain reductions
- separation applied in probing mode of the original problem
- variable bounds and objective of original LP changed

Implementation

- implemention in generic BP&C solver GCG based on SCIP
- separator in the master problem
  - callbacks SEPAEXECLP and SEPAEXECSOL reduce the domain

    ```
    *result = (SCIP_RESULT) SCIP_REDUCEDDOM
    ```

  - cuts are by default only separated in the pricing loop

  $\rightarrow$ public methods for separation called in pricer
  - propagation of domain reductions
- separation applied in probing mode of the original problem
- variable bounds and objective of original LP changed
- add parameter forcing SCIP to allways compute dual solution

  ```
  SCIP_CALL( SCIPaddBoolParam(..., "misc/alwaysgetduals", ...) );
  ```

Computational tests

- other separators and stabilization disabled for computations
- redundancy check in all pricing iterations at root node

Computational tests

- other separators and stabilization disabled for computations
- redundancy check in all pricing iterations at root node

- preliminary tests:
  - no cuts on binpacking, capacitated p-median, coloring etc.

Computational tests

- other separators and stabilization disabled for computations
- redundancy check in all pricing iterations at root node

- preliminary tests:
  - no cuts on binpacking, capacitated p-median, coloring etc.

- no feasibility cuts because of problem structure

Computational tests

- other separators and stabilization disabled for computations
- redundancy check in all pricing iterations at root node

- preliminary tests:
  - no cuts on binpacking, capacitated p-median, coloring etc.

- no feasibility cuts because of problem structure

- information from original LP relaxation is used
- weak original LP relaxations

Other instances

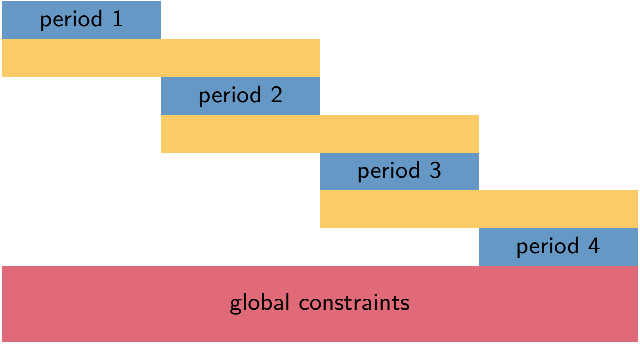
- capacitated lot sizing problems
  Tempelmeier and Derstroff (1996); Trigeiro et al. (1989)
  - period decomposition
    Pimentel et al. (2010); de Araujo et al. (2015)
  - horizon decompositions
    Fragkos et al. (2016)

- linearized thermal unit commitment instances
  Frangioni et al. (2009)
  - period, horizon decompositions
    Kim et al. (2017)

- subset of MIPLIB instances Bergner et al. (2015)
  - automatic detection with GCG
  - max. white score

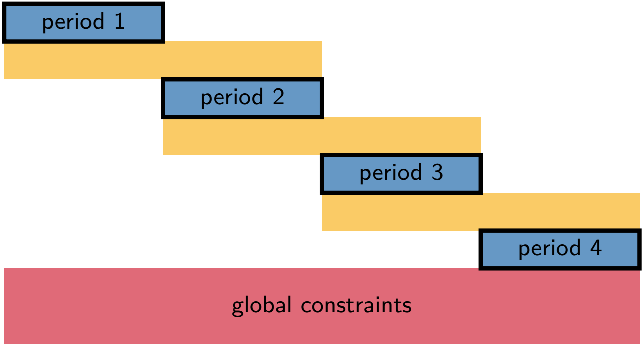
Time decompositions

- "time-dependent" structure in lot sizing and unit commitment:



Time decompositions

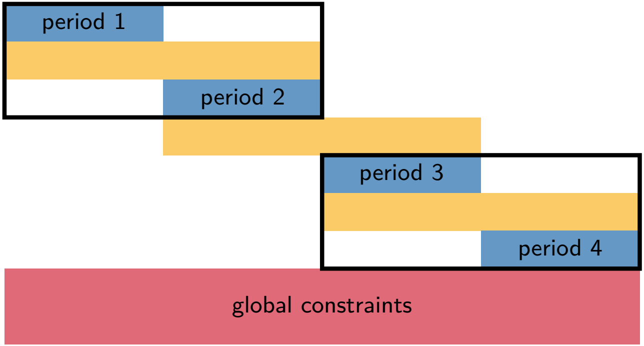
- "time-dependent" structure in lot sizing and unit commitment:



- period decomposition

Time decompositions

- “time-dependent” structure in lot sizing and unit commitment:



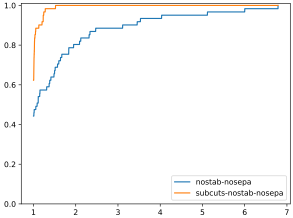
- horizon decomposition with horizons of 2 periods

Overview

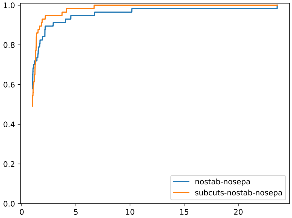
	overall		subcuts		default	
	ninstances	naffected	nsolved	gmeantime	nsolved	gmeantime
ls-derstroff-period	98	75	61	174.48	62	201.55
ls-derstroff-horizon2	98	69	62	117.17	57	157.80
ls-trigeiro-horizon2	69	61	0	3600.00	0	3600.00
ls-trigeiro-horizon4	69	65	9	3079.39	9	3320.52
ls-trigeiro-horizon8	69	50	67	158.08	66	150.11
uc-period	42	30	7	2161.37	7	2067.80
uc-horizon2	42	42	6	2273.87	5	2473.20
uc-horizon4	42	42	6	2119.09	6	2336.34
uc-horizon8	42	31	6	1873.31	6	2200.36
uc-horizon12	42	29	8	1493.45	8	1585.63
miplib-maxwhite	39	13	7	1355.43	7	1343.56

- time to solve root usually increases with subcuts
- behavior of number of col.gen. iterations unpredictable

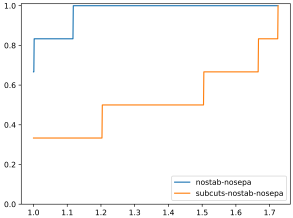
Lot sizing - performance profiles



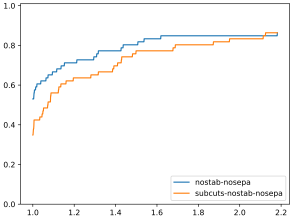
(a) Derstroff period



(b) Derstroff horizon 2



(c) Trigeiro horizon 4

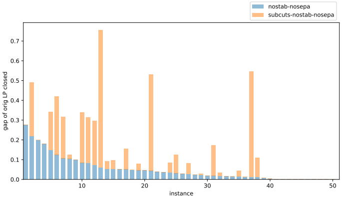


(d) Trigeiro horizon 8

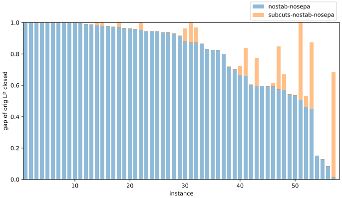
Operations Research

RWTH AACHEN UNIVERSITY

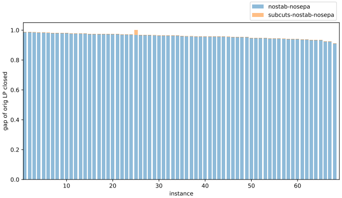
Lot sizing - root gap closed



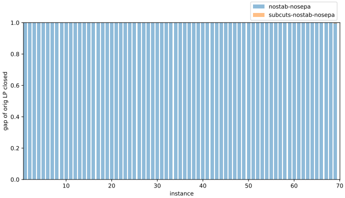
(e) Derstroff period



(f) Derstroff horizon 2



(g) Trigeiro horizon 4



(h) Trigeiro horizon 8

Lot sizing

- one subproblem for each period
- column $\leftrightarrow$ production plan for period t
- periods are linked by demand/balance constraints

Lot sizing

- one subproblem for each period
- column $\leftrightarrow$ production plan for period t
- periods are linked by demand/balance constraints

- original problem contains variables

$$y_{it} = \begin{cases} 1, \text{if product } i \text{ is produced in period } t \\ 0, \text{otherwise} \end{cases}$$

- with setup times and costs

Lot sizing

- one subproblem for each period
- column $\leftrightarrow$ production plan for period t
- periods are linked by demand/balance constraints

- original problem contains variables

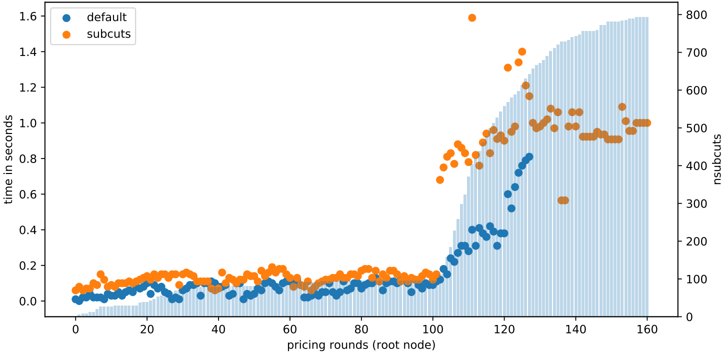
$$y_{it} = \begin{cases} 1, \text{if product } i \text{ is produced in period } t \\ 0, \text{otherwise} \end{cases}$$

- with setup times and costs

- subproblem cuts for period t and subset I' of products

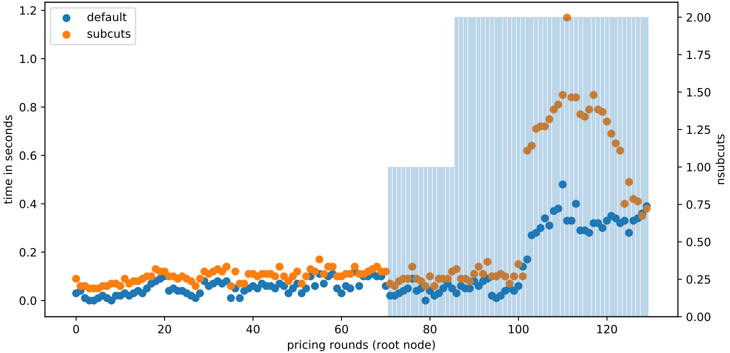
$$\sum_{i \in I'} y_{it} \geq 1$$

Effect on pricing



(i) K8021551-period (Derstroff)

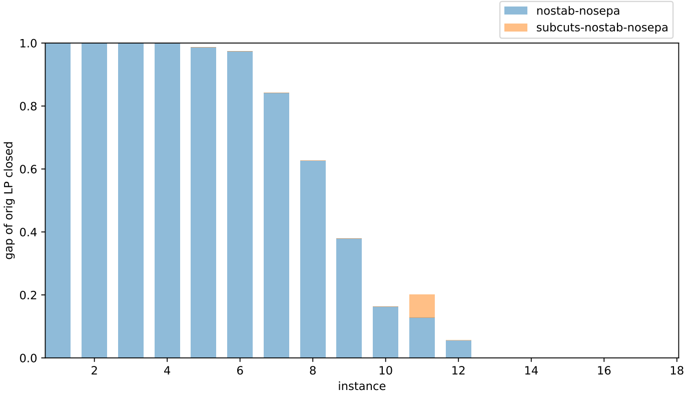
Effect on pricing II



(j) G0065252-period (Derstroff)

MIPLIB instances

- automatic detection (max. white)



Operations Research

RWTH AACHEN UNIVERSITY

Future work

- control of pricing difficulty?
- more problem classes with "time-dependent" structure?
- other decompositions for MIPLIB instances
- effect on price-and-branch heuristic

References

Bergner, M., Caprara, A., Ceselli, A., Furini, F., Lübbecke, M., Malaguti, E., and Traversi, E. (2015). Automatic Dantzig-Wolfe reformulation of mixed integer programs. *Math. Prog.*, 149(1–2):391–424.

de Araujo, S. A., Reyck, B. D., Degraeve, Z., Fragkos, I., and Jans, R. (2015). Period decompositions for the capacitated lot sizing problem with setup times. *INFORMS Journal on Computing*, 27(3):431–448.

Fragkos, I., Degraeve, Z., and Reyck, B. D. (2016). A horizon decomposition approach for the capacitated lot-sizing problem with setup times. *INFORMS Journal on Computing*, 28(3):465–482.

Frangioni, A., Gentile, C., and Lacalandra, F. (2009). Tighter approximated MILP formulations for unit commitment problems. *IEEE Transactions on Power Systems*, 24(1):105–113.

Gamrath, G. and Lübbecke, M. (2010). Experiments with a generic Dantzig-Wolfe decomposition for integer programs. In Festa, P., editor, *Experimental Algorithms*, volume 6049, pages 239–252, Berlin. Springer-Verlag.

Kim, K., Botterud, A., and Qiu, F. (2017). Temporal decomposition for improved unit commitment in power system production cost modeling. Technical report.

References (cont.)

Pimentel, C. M. O., Alvelos, F. P. e., and Valério de Carvalho, J. M. (2010). Comparing dantzig–wolfe decompositions and branch-and-price algorithms for the multi-item capacitated lot sizing problem. *Optimization Methods & Software*, 25(2):299–319.

Tempelmeier, H. and Derstroff, M. (1996). A Lagrangean-based heuristic for dynamic multilevel multiitem constrained lotsizing with setup times. *Management Science*, 42(5):738–757.

Trigeiro, W. W., Thomas, L. J., and McClain, J. O. (1989). Capacitated lot sizing with setup times. *Management Science*, 35(3):353–366.

Vanderbeck, F. and Savelsbergh, M. W. (2006). A generic view of Dantzig-Wolfe decomposition in mixed integer programming. *Operations Research Letters*, 34(3):296 – 306.