Disjunctive cuts in branch-and-but-and-price algorithms

Application to the capacitated vehicle routing problem

Stefan Ropke

Technical University of Denmark, Department of Transport
(DTU Transport)

Column generation 2008, Aussois, France

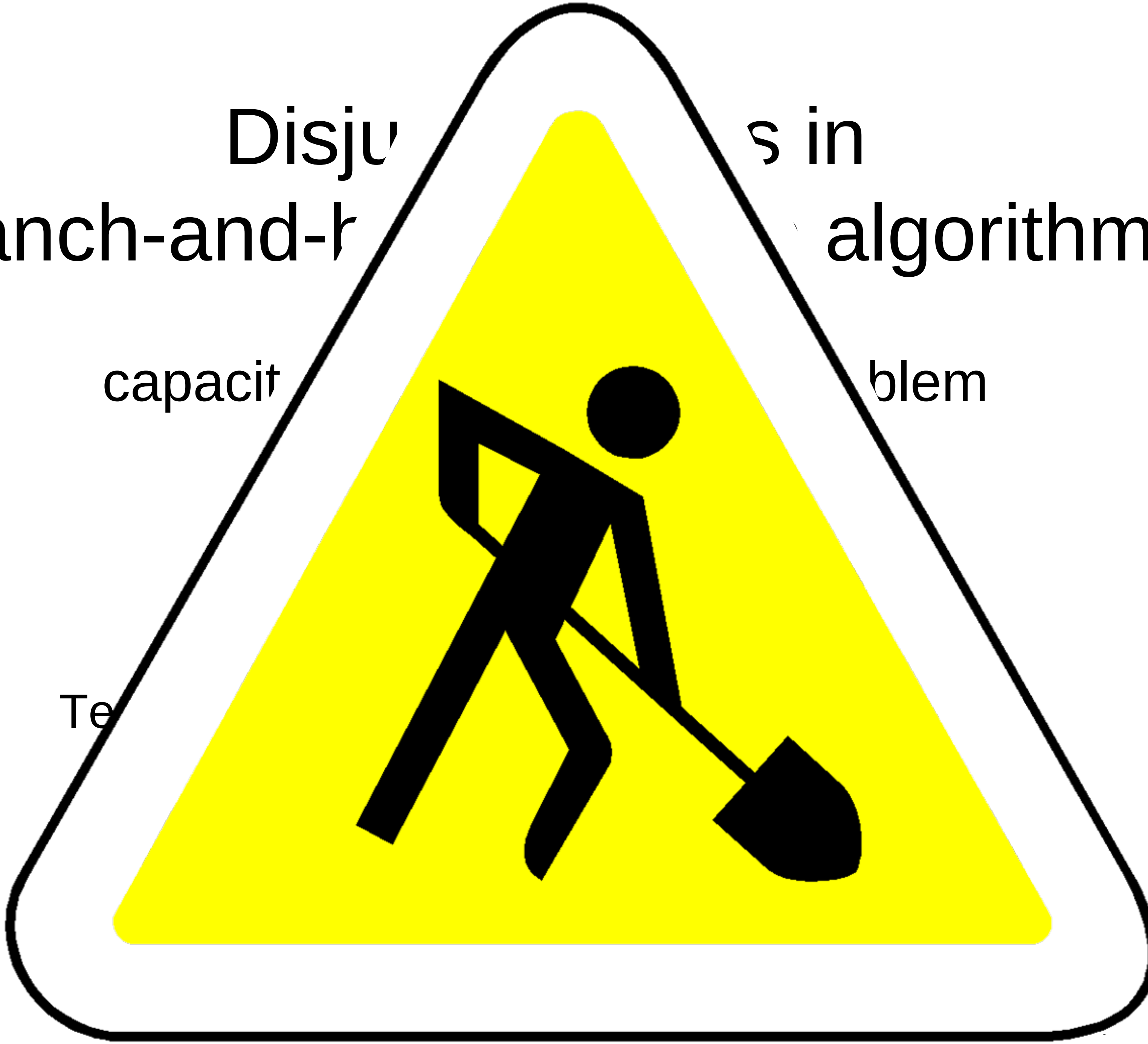
Disju… s in

branch-and-b… algorithms

capacit… blem

Te…

…nce



Motivation

- Our goal: Solve difficult IP problems.
- Our approach: Branch-and-price (BAP).
- A major reason for using BAP: *reduced gap between LP bound and optimal solution (integrality-gap).*

Motivation

- Initially (Eigthies and first half of the nineties): Happy with improved IP-gap from Dantzig-Wolfe decomposition.

- ”Recently” (last half of the nineties till now) Increased research activity focussed on decreasing the integrality-gap by combining branch-and-price with cutting planes (Branch-and-cut-and-price (BCP)).

Motivation: Branch-cut-and-price

- Main research trend:
  - Problem specific cuts in the variables of the compact formulation.
- Many succesful applications:
  - **Vehicle routing problem with time windows**: Kohl, Desrosiers, Madsen, Solomon , Soumis (1999)
  - **Multicommodity flow problems**: Barnhart, Hane, Vance (2000)
  - **Capacitated vehicle routing problem**: Fukasawa, Longo, Lysgaard, Poggi de Aragão, Reis, Uchoa, Werneck (2006)
  - **Multiple depot vehicle scheduling:** Hadjar, Marcotte, Soumis (2006)
  - **Capacitated minimum spanning tree**: Uchoa, Fukasawa, Lysgaard, Pessoa, Poggi de Aragão, Andrade (2007).
- Some research on generic cuts on the variables of the extended formulation:
  - Petersen, Pisinger, Spoorendonk (2007)

How about generic cuts on the variables of the compact formulation?

- We have a pretty good idea about such cuts should be handeled in a BAP algorithm (well studied in the literature).
- Wouldn’t it be nice if we could ”flip a switch” and our integrality gap is decreased by 30 or 40%?
- ... at the cost of more rows in the master problem and increased separation time?

Our candidate: Disjunctive cuts in the variables of the compact formulation

- Dates back to a technical report by E. Balas from 1974.
- Was shown to be useful in practice by Balas, Ceria, Cornuéjols (1993).
- Includes or is equivalent to a number of other generic cutting planes: *Lift-and-project cuts, split cuts, intersection cuts, gomory mixed integer cuts*
- Is included in some form in both CPLEX and Xpress-MP.

Our IP problem (compact formulation)

$$\min cx$$

subject to

$$Ax \leq b$$

$$x \in \mathbb{Z}^n_+$$

Our IP problem (compact formulation)

$$\min cx$$

subject to

$$Ax \leq \quad b$$

$$x \in \mathbb{Z}^n_+$$

Disjunction:

$$\alpha x \leq \beta \quad \vee \quad \alpha x \geq \beta + 1$$

$$\alpha \in \mathbb{Z}^n, \beta \in \mathbb{Z}$$

Our IP problem (compact formulation)

$$\min cx$$

subject to

$$Ax \leq b$$
$$x \in \mathbb{Z}^n_+$$

Here We only consider disjunctions with two terms of the stated form (split cuts)

Disjunction:

$$\alpha x \leq \beta \quad \vee \quad \alpha x \geq \beta + 1$$

$$\alpha \in \mathbb{Z}^n, \beta \in \mathbb{Z}$$

Our IP problem (compact formulation)

$$\min cx$$

subject to

$$Ax \leq b$$
$$x \in \mathbb{Z}^n_+$$

Disjunction:

$$x_i \leq q \quad \vee \quad x_i \geq q + 1$$

$$q \in \mathbb{Z}$$

Disjunctive cuts

We are interested in approximating the polyhedron

$$P^I = \text{conv}\left\{ \begin{array}{rcr} Ax & \leq & b \\ x & \in & \mathbb{Z}^n_+ \end{array} \right\}$$

We are interested in approximating the polyhedron

$$P^I = \text{conv}\left\{ \begin{array}{rcr} Ax & \leq & b \\ x & \in & \mathbb{Z}^n_+ \end{array} \right\}$$

$$P_L = \left\{ \begin{array}{rcr} Ax & \leq & b \\ \alpha x & \leq & \beta \\ x & \in & \mathbb{R}^n_+ \end{array} \right\}$$

$$P_R = \left\{ \begin{array}{rcr} Ax & \leq & b \\ \alpha x & \geq & \beta + 1 \\ x & \in & \mathbb{R}^n_+ \end{array} \right\}$$

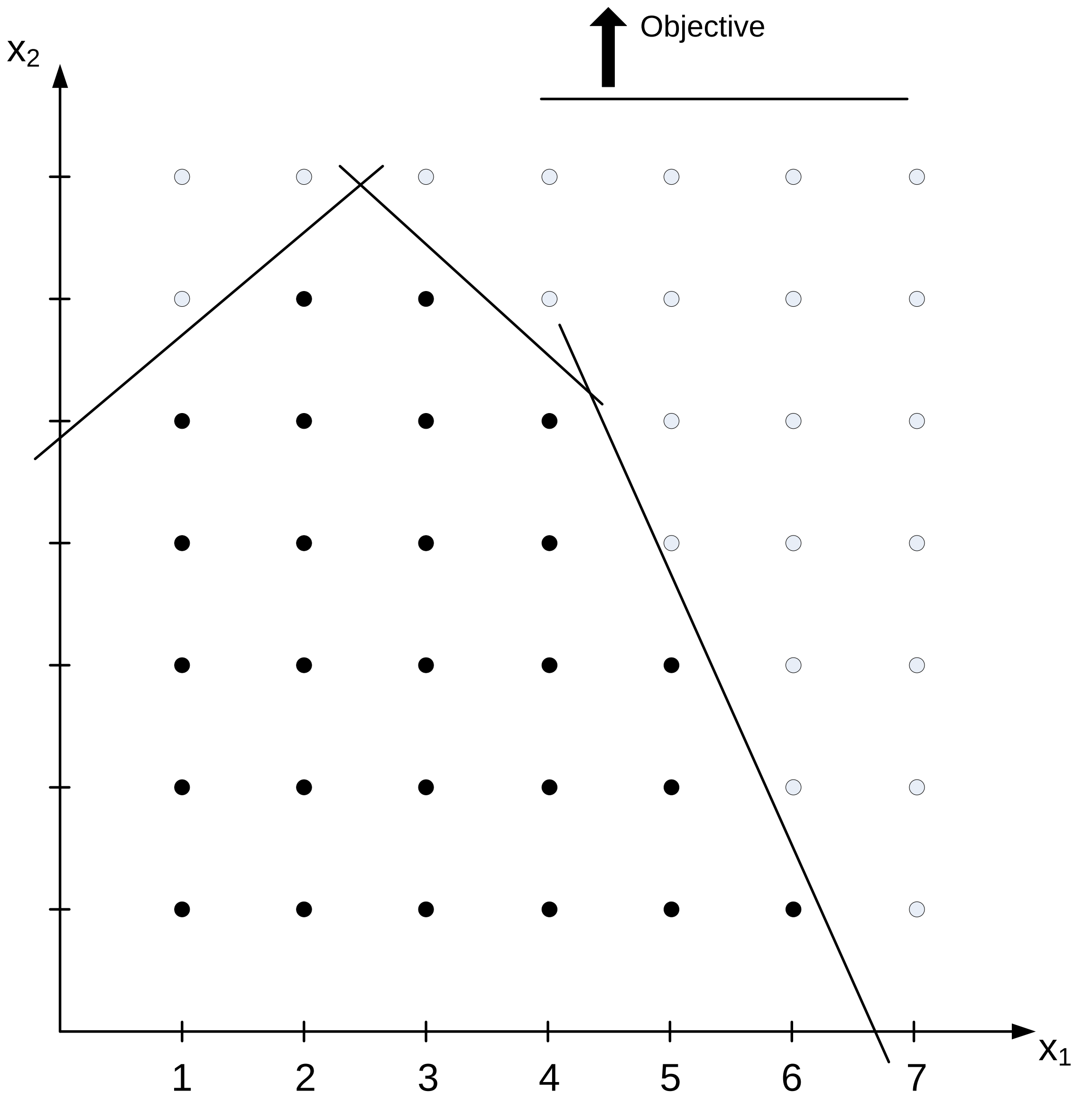
$$P_L = \left\{ \begin{array}{rcr} Ax & \leq & b \\ \alpha x & \leq & \beta \\ x & \in & \mathbb{R}^n_+ \end{array} \right\}$$

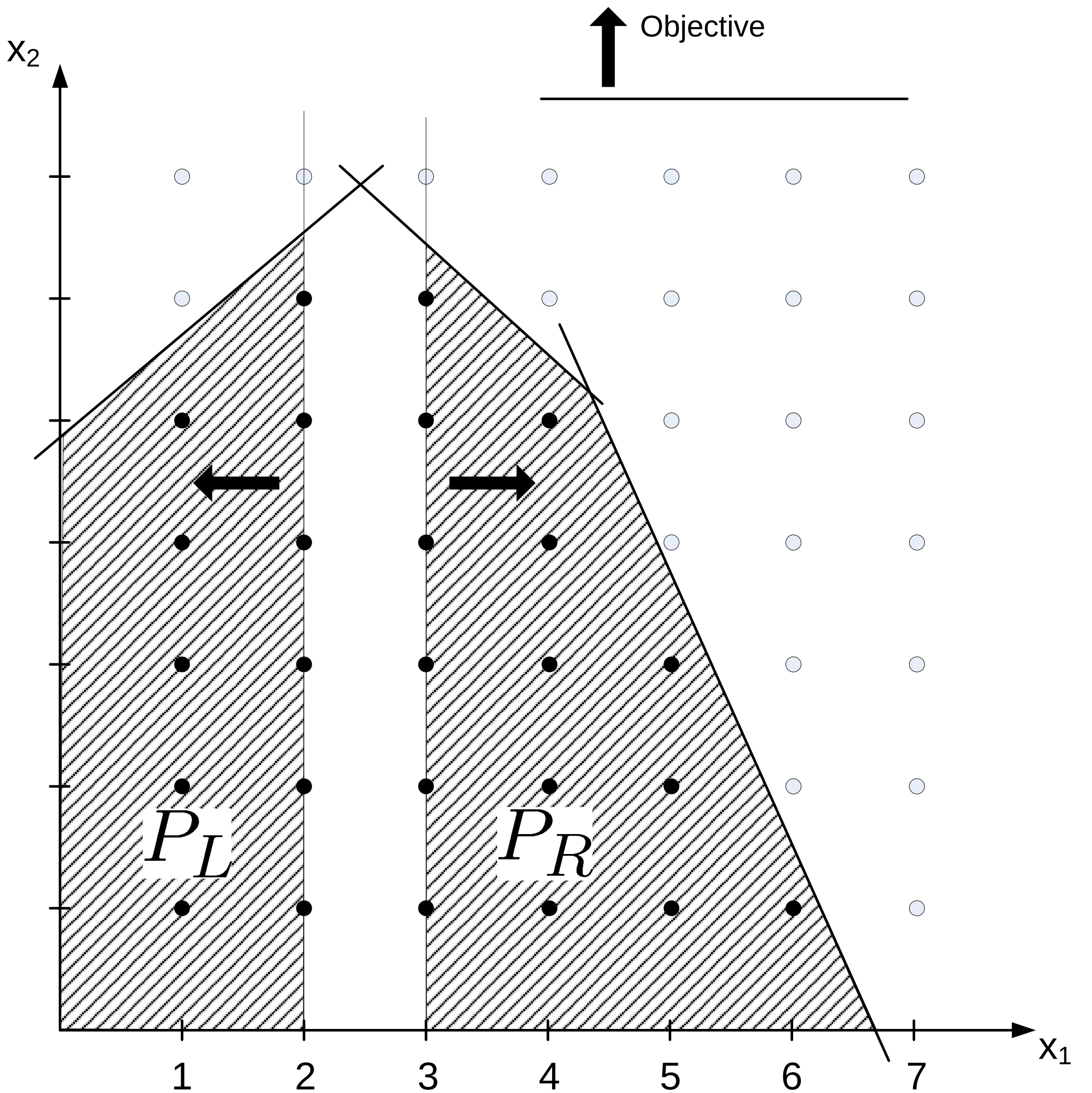
$$P_R = \left\{ \begin{array}{rcr} Ax & \leq & b \\ \alpha x & \geq & \beta + 1 \\ x & \in & \mathbb{R}^n_+ \end{array} \right\}$$

We look for inequalities that are valid for both P_L and P_R. Such inequalities are also valid for P^I, but not necessarily for P.

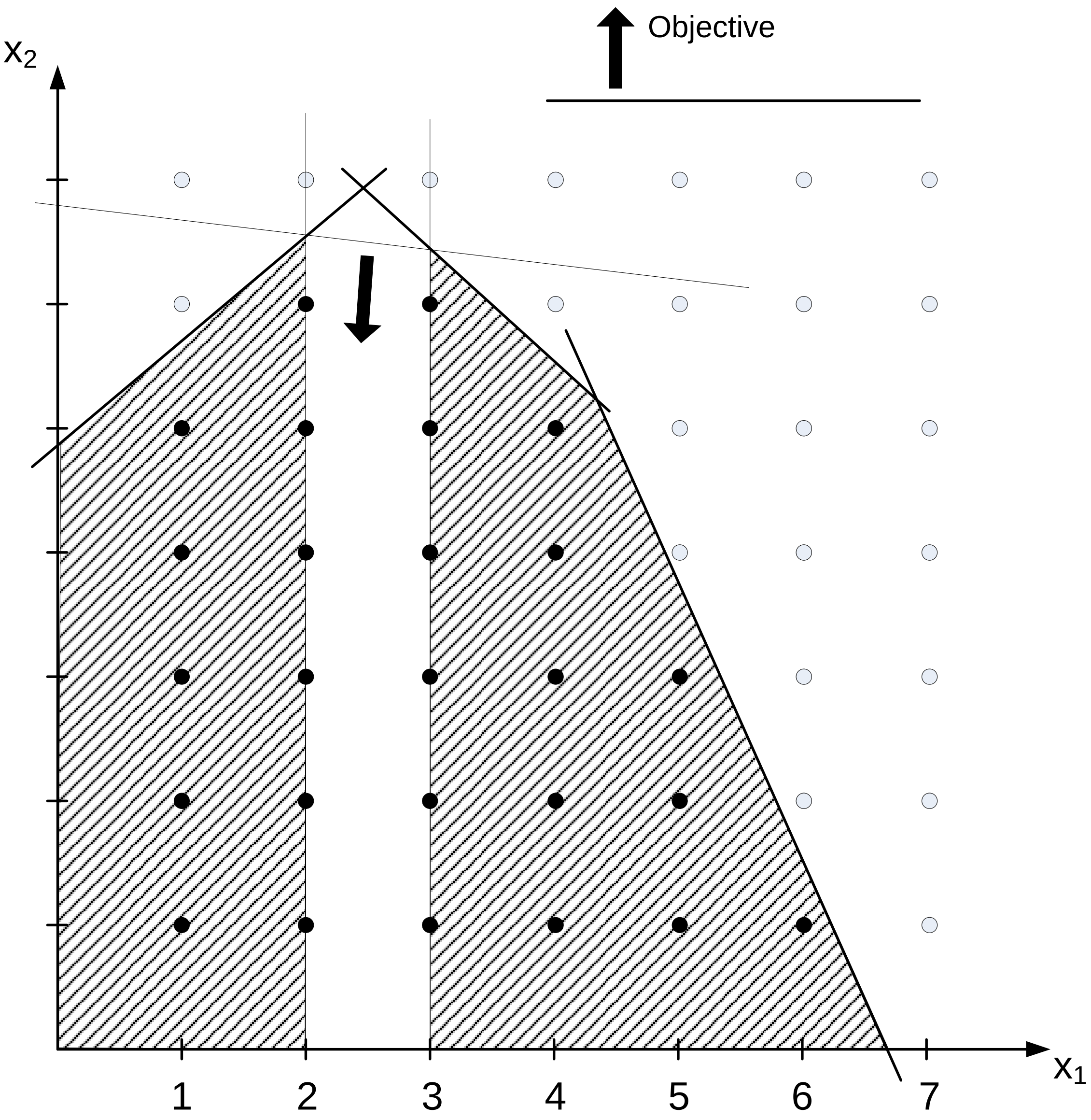
$$P_L = \left\{ \begin{array}{rcr} Ax & \leq & b \\ \alpha x & \leq & \beta \\ x & \in & \mathbb{R}^n_+ \end{array} \right\}$$

$$P_R = \left\{ \begin{array}{rcr} Ax & \leq & b \\ \alpha x & \geq & \beta + 1 \\ x & \in & \mathbb{R}^n_+ \end{array} \right\}$$


Objective
x2
x1
1
2
3
4
5
6
7



Disjunction: $x_1 \leq 2 \vee x_1 \geq 3$.



Objective
x2
x1
1
2
3
4
5
6
7

Recall

$$P_L = \left\{ \begin{array}{rcr} Ax & \leq & b \\ \alpha x & \leq & \beta \\ x & \in & \mathbb{R}^n_+ \end{array} \right\}, P_R = \left\{ \begin{array}{rcr} Ax & \leq & b \\ \alpha x & \geq & \beta + 1 \\ x & \in & \mathbb{R}^n_+ \end{array} \right\}$$

Rewriten

$$P_L = \left\{ \begin{array}{rcr} A^L x & \leq & b^L \\ x & \in & \mathbb{R}^n_+ \end{array} \right\}, P_R = \left\{ \begin{array}{rcr} A^R x & \leq & b^R \\ x & \in & \mathbb{R}^n_+ \end{array} \right\}$$

$uA^L x \leq ub^L$ is a valid inequality for P^L for any multiplier vector $u \in \mathbb{R}_+^{m+1}$.

Similarly, $vA^R x \leq vb^R$ is a valid inequality for P^R for any multiplier vector $v \in \mathbb{R}_+^{m+1}$.

is valid for P_R then

$$\sum_{i=1}^{n} \min\{\pi_i^L, \pi_i^R\} x_i \le \max\{\pi_0^L, \pi_0^R\}$$

is valid for both P_L and P_R (because $P \subset \mathbb{R}^n_+$).

is valid for P_R then

$$\sum_{i=1}^{n} \min\{\pi_i^L, \pi_i^R\} x_i \le \max\{\pi_0^L, \pi_0^R\}$$

is valid for both P_L and P_R (because $P \subset \mathbb{R}^n_+$).

$$
\begin{aligned}
\pi_0 &\le ub \\
\pi_i &\le vA_i^R \quad \forall i \in N \\
\pi_0 &\ge vb^R \\
\pi_i &\in \mathbb{R} \quad \forall i \in N \\
u &\in \mathbb{R}_+^{m+1} \\
v &\in \mathbb{R}_+^{m+1}
\end{aligned}
$$

$N = \{1, \ldots, n\}$

$$
\begin{aligned}
\pi_0 &\le ub \\
\pi_i &\le vA_i^R \quad \forall i \in N \\
\pi_0 &\ge vb^R \\
\pi_i &\in \mathbb{R} \quad \forall i \in N \\
u &\in \mathbb{R}_+^{m+1} \\
v &\in \mathbb{R}_+^{m+1}
\end{aligned}
$$

$N = \{1, \ldots, n\}$

$2n + 2$ rows

$2(n + m + 2)$ columns

Normalization:

$$\max \sum_{i=1}^{n} \pi_i x_i^* - \pi_0$$

subject to

$$\begin{aligned}
\pi_i &\leq u A_i^L \quad \forall i \in N \\
\pi_0 &\geq u b^L \\
\pi_i &\leq v A_i^R \quad \forall i \in N \\
\pi_0 &\geq v b^R \\
\sum_{j=1}^{m+1} u_j + \sum_{j=1}^{m+1} v_j &= 1 \\
\pi_i &\in \mathbb{R} \quad \forall i \in N \\
u &\in \mathbb{R}_+^{m+1} \\
v &\in \mathbb{R}_+^{m+1}
\end{aligned}$$

Separation algorithm

- Select disjunction.

- Solve cut-finding LP for this disjunction.

- What if P has an exponential number of rows (think of the CVRP modeled using *capacity inequalities*)?

Separation algorithm

ford (2002)].

- Not tested in practice to the best of my knowledge.

ford (2002)].

- Not tested in practice to the best of my knowledge.

Handling an exponential number of rows in P by column generation.

$$\max \sum_{i=1}^{n} \pi_i x_i^* - \pi_0$$

subject to

$$\begin{array}{llll}
\pi_i - uA_i^L & \leq 0 & \forall i \in N & (y^L \geq 0) \\
\pi_0 - ub^L & \geq 0 & & (y_0^L \leq 0) \\
\pi_i \quad - vA_i^R & \leq 0 & \forall i \in N & (y^R \geq 0) \\
\pi_0 \quad - vb^R & \geq 0 & & (y_0^R \leq 0) \\
u\mathbf{1} + v\mathbf{1} & = 1 & & (y_N) \\
\pi_i & \in \mathbb{R} & \forall i \in N & \\
u & \in \mathbb{R}_+^{m+1} & & \\
v & \in \mathbb{R}_+^{m+1} & &
\end{array}$$

The reduced cost of a "u" column corresponding to an inequality $\gamma x \leq \gamma_0$ is

$$y^L \gamma + y_0^L \gamma_0 - y_N$$

$$\max \sum_{i=1}^{n} \pi_i x_i^* - \pi_0$$

subject to

$$\begin{array}{llllll}
\pi_i - uA_i^L & & \leq 0 & \forall i \in N & (y^L \geq 0) \\
\pi_0 - ub^L & & \geq 0 & & (y_0^L \leq 0) \\
\pi_i & -vA_i^R & \leq 0 & \forall i \in N & (y^R \geq 0) \\
\pi_0 & -vb^R & \geq 0 & & (y_0^R \leq 0) \\
u\mathbf{1} & +v\mathbf{1} & = 1 & & (y_N) \\
\pi_i & & \in \mathbb{R} & \forall i \in N & \\
u & & \in \mathbb{R}_+^{m+1} & & \\
v & & \in \mathbb{R}_+^{m+1} & &
\end{array}$$

The reduced cost of a $''u''$ column corresponding to an inequality $\gamma x \leq \gamma_0$ is

$$y^L\gamma + y_0^L\gamma_0 - y_N$$

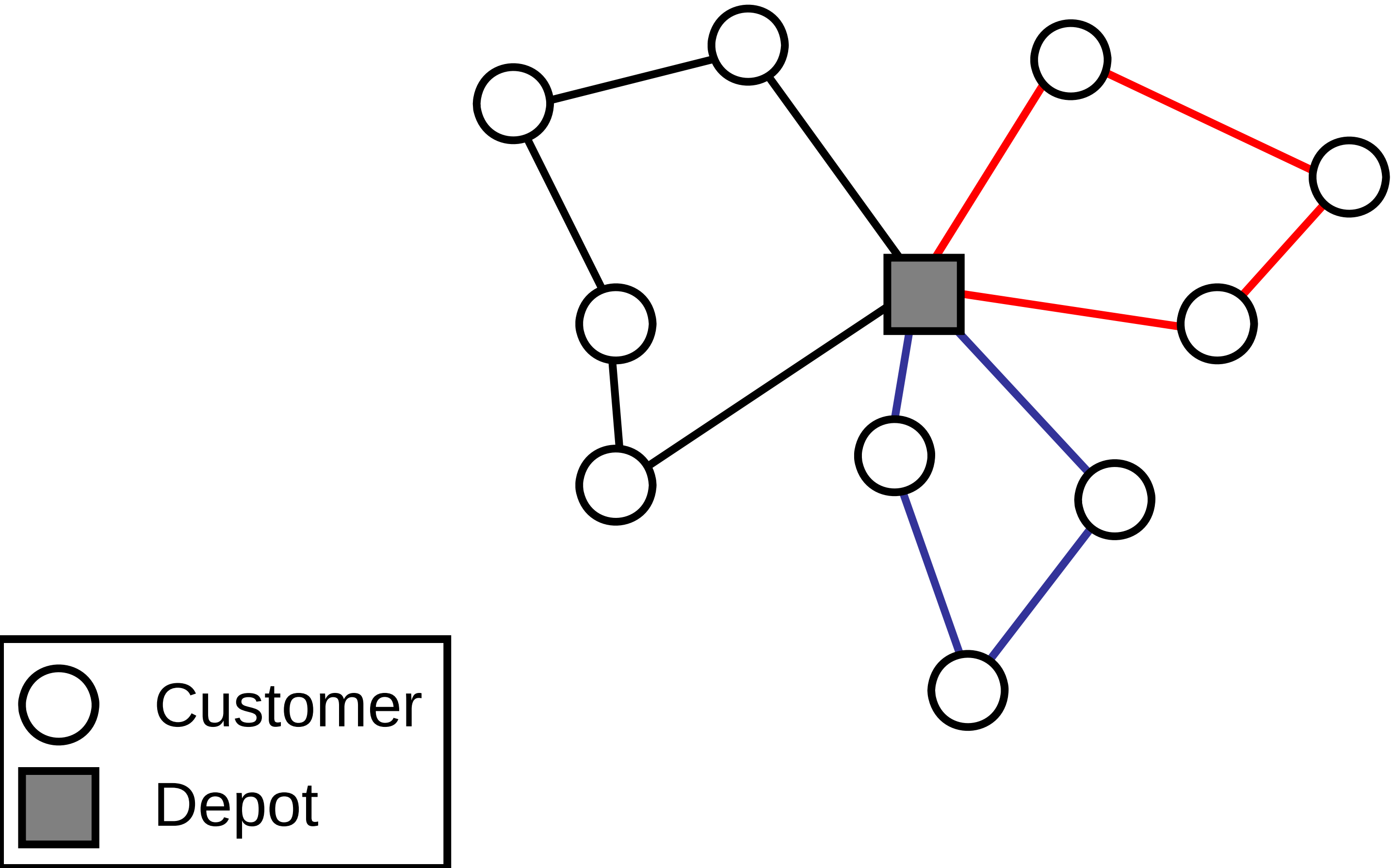
Taking advantage of the sign of the dual variables we obtain that a column has positive reduced cost if

$$\frac{y^L}{\left|y_0^L\right|}\gamma > \gamma_0 + \frac{y_N}{\left|y_0^L\right|}$$

Assuming that $\left|y_0^L\right| > 0$, we can find a column with reduced cost by finding an inequality $\gamma x \leq \gamma_0$ that is violated by at least $y_N / \left|y_0^L\right|$ for the point $x' = y_L / \left|y_0^L\right|$.

y_N has the value of the objective of the cut-finding LP. Normalization constraint makes pricing more difficult.

The capacitated vehicle routing problem (CVRP)



Compact formulation

$$\min \sum_{e \in E} c_e x_e$$

subject to

$$\begin{aligned}
x(\delta(i)) &= 2 && \forall i \in V \setminus \{0\} \\
x(\delta(0)) &= 2K && \\
x(\delta(S)) &\geq 2 \left\lceil \frac{d(S)}{Q} \right\rceil && \forall S \subseteq V \setminus \{0\} \\
x_e &\leq 2 && \forall e \in \delta(0) \\
x_e &\leq 1 && \forall e \in E \setminus \delta(0) \\
x_e &\in \mathbb{Z}_+ && \forall e \in E
\end{aligned}$$

Solving the CVRP - cuts

- Cuts – all from Lysgaard, Letchford & Eglese (2004):
  - Capacity, framed capacity, strengthened comb, 2 edges hypotour, homogeneous multistar
- A new family of inequalities:
  - subtour-depot

Disjunctive cuts for the CVRP

- Step 1: all ”normal cuts” are separated. Repeat as long as violated cuts are identified.
- Step 2: Construct cut-finding LP based on compact formulation and already identified cuts. Solve.
- Step 3: Optional: Improve cut-finding LP by column generation. Generate columns from capacity inequalities.

Disjunctive cuts for the CVRP

- Disjunctions:
  - $x_e \leq \lfloor x_e^* \rfloor \vee x_e \geq \lceil x_e^* \rceil$
  - $x(S) \leq |S| - 2 \vee x(S) \geq |S| - 1 \quad (S \subset V)$
- Disjunctive cuts only generated at root node.
- Only rank-1 inequalities are generated.

Solving the CVRP - Pricing

- Shortest paths with 2-cycle elimination
  - Exact: Bidirectional label setting
  - Heuristic: Truncated label setting
  - Heuristic: Construction heuristic
- Elementary shortest path
  - Exact: simple branch & cut (Jepsen, Petersen, Spoorendonk, 2008)
  - Heuristic: Truncated label setting
  - Heuristic: Construction heuristic
  - Heuristic: LNS

Solving the CVRP - details

- Branching – branch on node sets:

  $x(S) \le |S|-2$ OR $x(S) \ge |S|-1$
- Strong branching
- Column pool, cut pool

Results – SPPCC-2CE

	Gap (%)	Gap closed Relative to Cap cuts (%)
Cap cuts	0.86	

Results – SPPCC-2CE

	Gap (%)	Gap closed Relative to Cap cuts (%)
Cap cuts	0.86	
Cap cuts+Disj. Cuts	0.60	29.9

Results – SPPCC-2CE

	Gap (%)	Gap closed Relative to Cap cuts (%)
Cap cuts	0.86	
Cap cuts+Disj. Cuts	0.60	29.9
All std cuts.	0.76	10.9

Results – SPPCC-2CE

	Gap (%)	Gap closed Relative to Cap cuts (%)
Cap cuts	0.86	
Cap cuts+Disj. Cuts	0.60	29.9
All std cuts.	0.76	10.9
All cuts	0.56	34.9

Results - ESPPCC

	Gap (%)	Gap closed Relative to Cap cuts (%)
Cap cuts	0.84	
Cap cuts+Disj. Cuts	0.57	31.5
All std cuts	0.74	11.1
All cuts	0.53	36.3
ESPPCC - All std cuts	**0.48**	**43.0**
ESPPCC - All cuts	**0.36**	**56.4**

Reduced set of instances

Results

Time limit: 2 hour
83 of the instances
Have been solved in the literature (however, not in two hours)

	Solved (of 86)
Cap cuts	67
Cap cuts+Disj. Cuts	68
All std cuts	68
All cuts (+DC)	68
ESPPCC - All std cuts	-
ESPPCC - All cuts (+DC)	61

Effects of parameters

	Gap (%)	Gap closed (%)	Time sep. (s)
Cap only	0.94		
Std. disj cuts.	0.57	40	77
No strength	0.62	34	54
No col. Gen.	0.72	23	14
no CG no str.	0.77	18	12
No node set disj.	0.57	40	71
Aggressive	0.46	51	243

Reduced set of instances (A-set)

Best methods for the CVRP currently

- Fukasawa, Longo, Lysgaard, Poggi de Aragão, Reis, Uchoa, Werneck (2005)
  - Solves most instances to optimality
- Baldacci, Christofides, Mingozzi (2008)
  - Solves fewer instances to optimality but is faster on the instances that both methods solve.

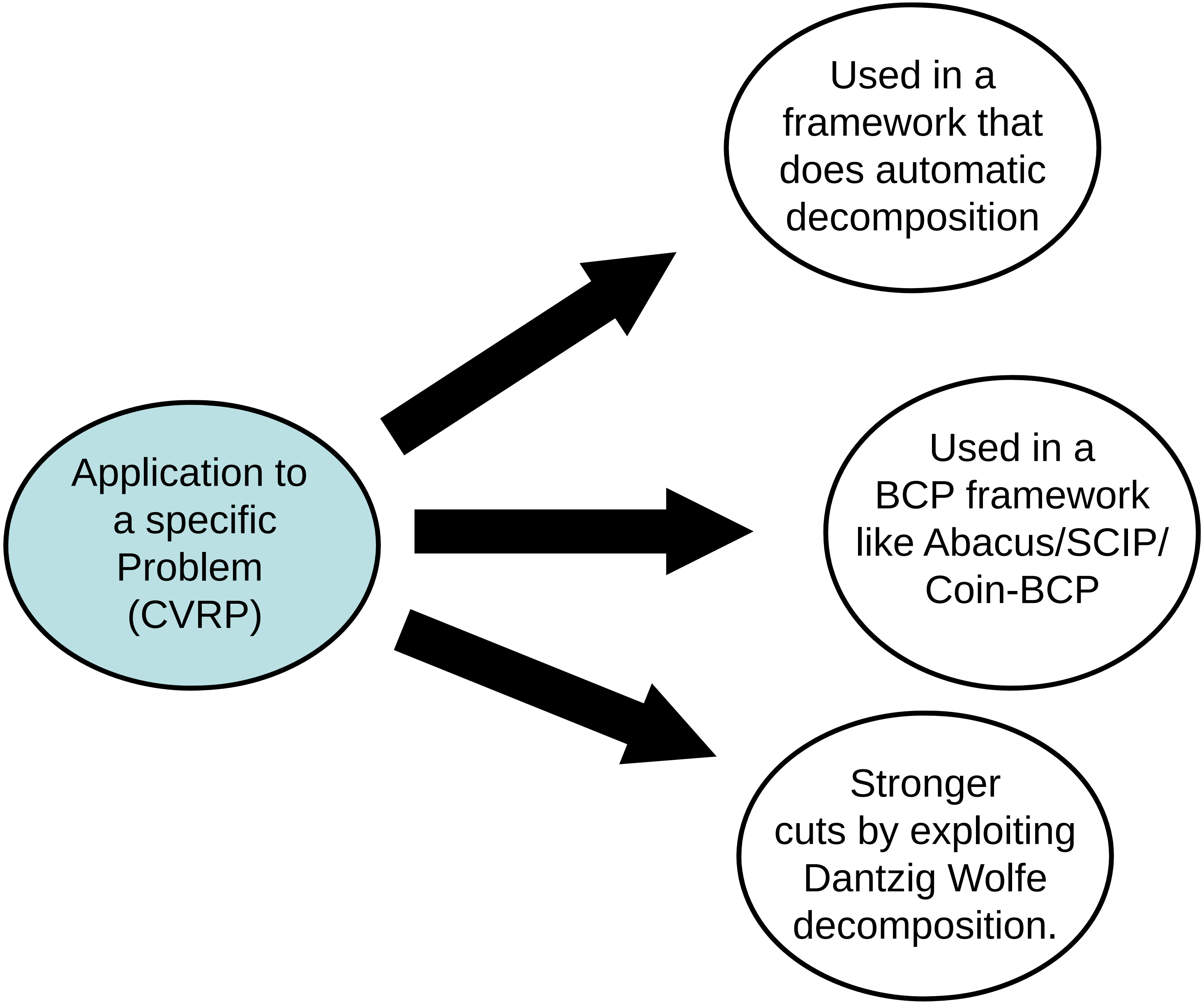
Best methods for the CVRP

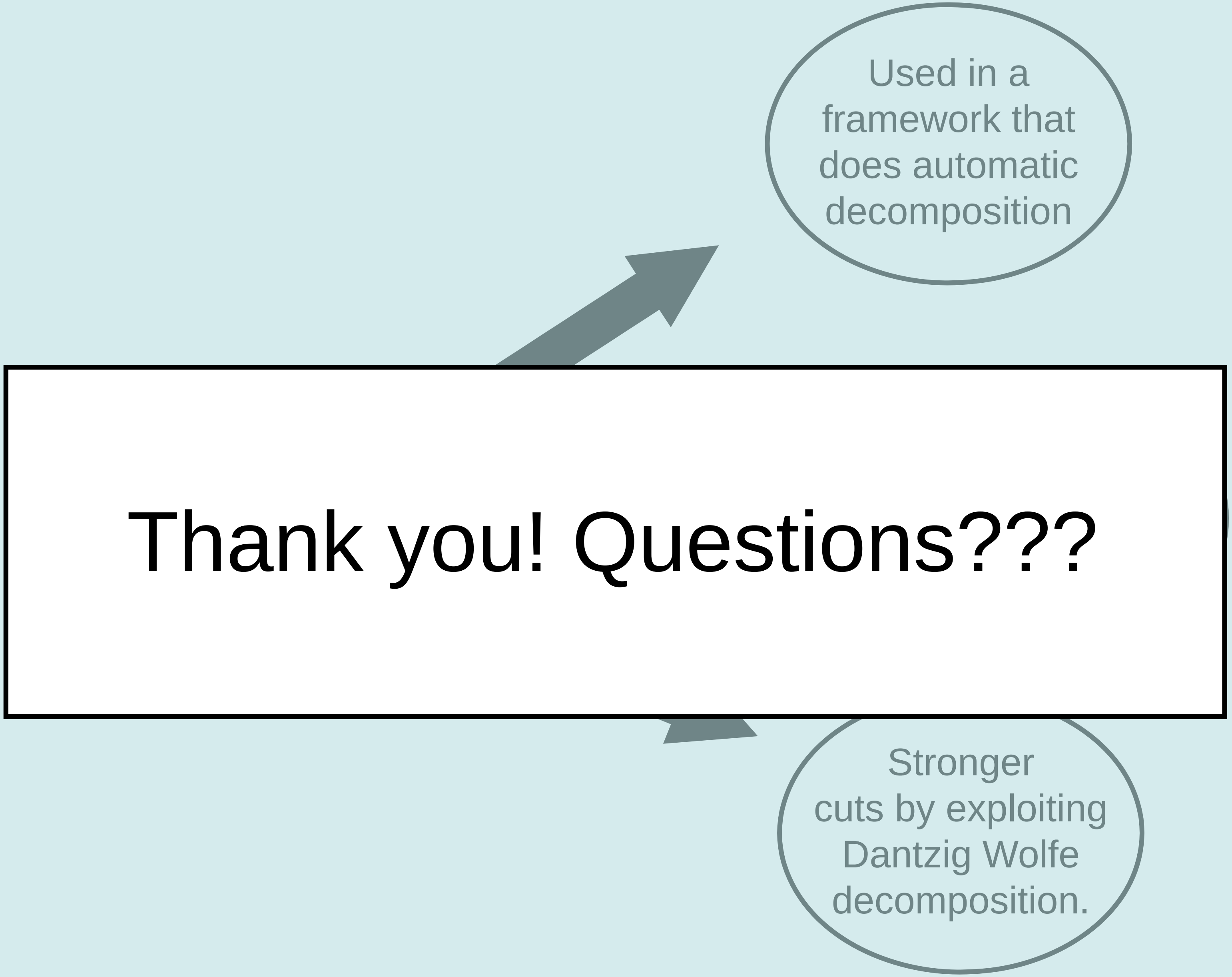
- FLLPRUW05 = Fukasawa, Longo, Lysgaard, Poggi de Aragão, Reis, Uchoa, Werneck (2005), Pentium IV (2.4 GHz)
- Disjunctive cuts code run on AMD Opteron (2.4 GHz)

	FLLPRUW05		Disj. cuts	
	BB Nodes	time (s)	BB Nodes	time (s)
A-set (average)	115	2050	146	1549
E-n76-k7	1712	46520	1280	22117
E-n76-k8	1031	22891	980	22685
E-n76-k10	4292	80722	2644	30451
E-n76-k14	6678	48637	-	-

Conclusion

- Disjunctive cuts are useful in a BCP algorithm for the CVRP. Closes a significant amount of the integrality gap.
- Application to other problems would be interesting.
- Still a lot of work to do on tuning the separation algorithm.
  - Should cuts be added outside the root node?
  - When should the column generation algorithm be applied?
  - Should we use another normalization?


Application to a specific Problem (CVRP)
Used in a framework that does automatic decomposition
Used in a BCP framework like Abacus/SCIP/ Coin-BCP
Stronger cuts by exploiting Dantzig Wolfe decomposition.


Used in a framework that does automatic decomposition
Thank you! Questions???
Stronger cuts by exploiting Dantzig Wolfe decomposition.