Branch and Price for Hub Location Problems with Single Assignment

Ivan Contreras[1], Elena Fernández[2]

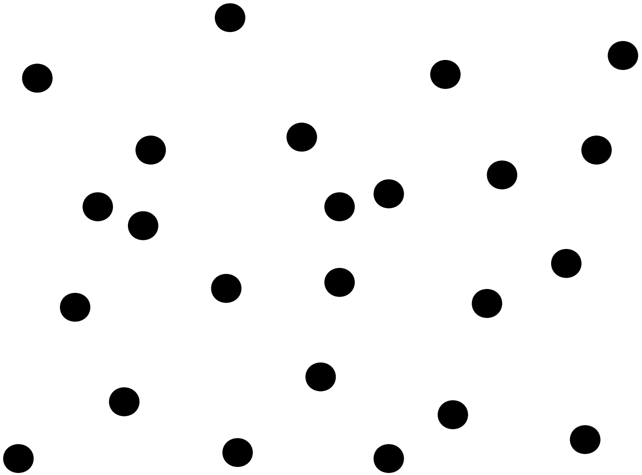
[1]Concordia University and CIRRELT, Montreal, Canada
[2]Technical University of Catalonia, Barcelona, Spain

International Workshop in Column Generation, June 10-13, 2012, Bromont, Canada

Outline

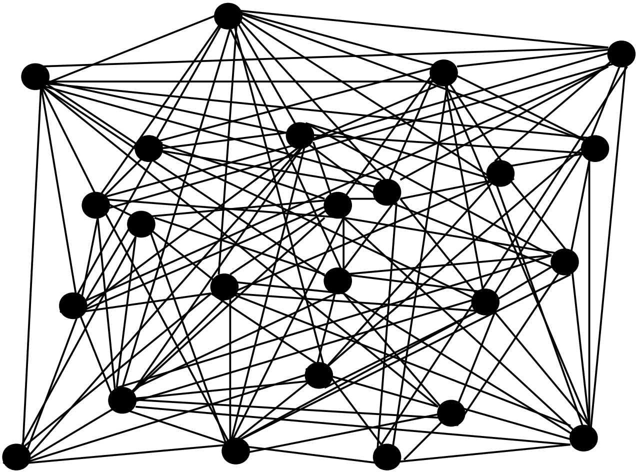
Outline

Hub and Spoke Networks



The key feature of these networks is the way in which commodities are routed from numerous points of origin to specific destinations.

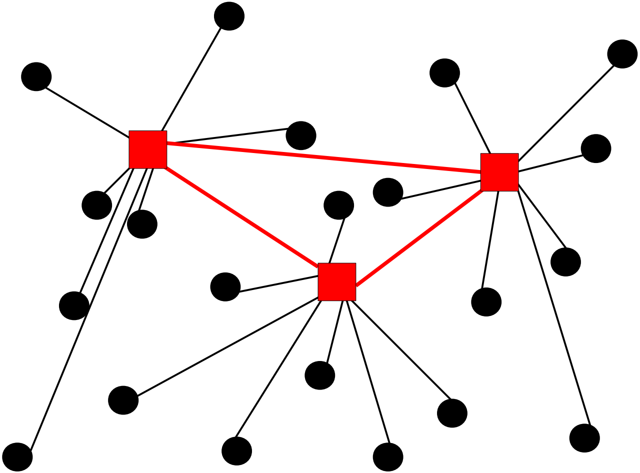
Hub and Spoke Networks



25 demand points (origins and destinations)

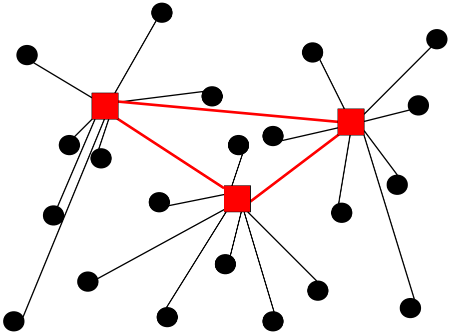
$(25 \times 24)/2 = 300$ direct connections

Hub and Spoke Networks



Hub networks: Concentrate flows to exploit economies of scale in transportation. (3 hub arcs and 21 access arcs).

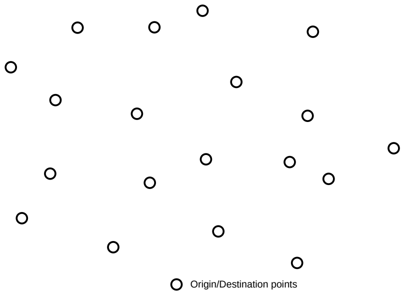
Classical Hub Location Problems



- Hub location problems deal with:
  - location of hub nodes
  - routing of commodities through the network
- Assume that hubs are fully interconnected at no cost
- O/D paths must contain at least one hub and at most two

Outline

Capacitated Hub Location with Single Assignment



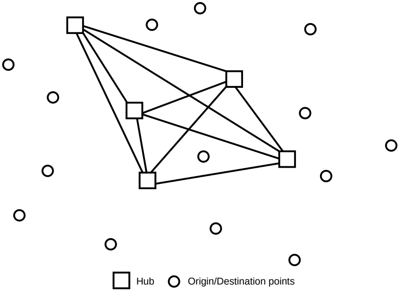
Notation

- N: set of nodes
- W_{ij}: flow between i and j
- $O_i = \sum_{j \in N} W_{ij}$: outgoing flow
- $D = \sum_{i \in N} O_i$: total flow
- b_i: capacities
- f_i: set-up costs
- d_{ij}: transportation costs

The CHLPSA consist of:

- Locate a set of hub nodes
- Single assignment patter on the flows is considered
- Minimize the total set-up and transportation costs
- Transportation costs: $F_{ijkm} = W_{ij} \left(\chi d_{ik} + \alpha d_{km} + \delta d_{mj} \right)$.

Capacitated Hub Location with Single Assignment



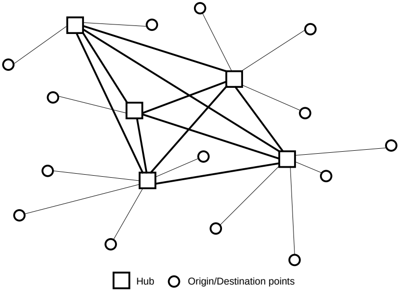
Notation

- N: set of nodes
- W_{ij}: flow between i and j
- $O_i = \sum_{j \in N} W_{ij}$: outgoing flow
- $D = \sum_{i \in N} O_i$: total flow
- b_i: capacities
- f_i: set-up costs
- d_{ij}: transportation costs

The CHLPSA consist of:

- Locate a set of hub nodes
- Single assignment patter on the flows is considered
- Minimize the total set-up and transportation costs
- Transportation costs: $F_{ijkm} = W_{ij}\left(\chi d_{ik} + \alpha d_{km} + \delta d_{mj}\right)$.

Capacitated Hub Location with Single Assignment



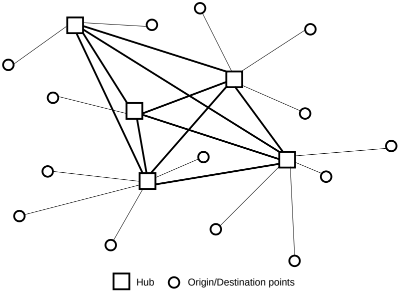
Notation

- N: set of nodes
- W_{ij}: flow between i and j
- $O_i = \sum_{j \in N} W_{ij}$: outgoing flow
- $D = \sum_{i \in N} O_i$: total flow
- b_i: capacities
- f_i: set-up costs
- d_{ij}: transportation costs

The CHLPSA consist of:

- Locate a set of hub nodes
- Single assignment patter on the flows is considered
- Minimize the total set-up and transportation costs
- Transportation costs: $F_{ijkm} = W_{ij} (\chi d_{ik} + \alpha d_{km} + \delta d_{mj})$.

Capacitated Hub Location with Single Assignment



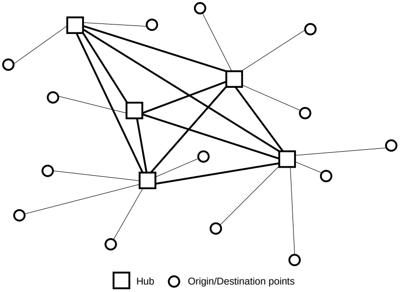
Notation

- N: set of nodes
- W_{ij}: flow between i and j
- $O_i = \sum_{j \in N} W_{ij}$: outgoing flow
- $D = \sum_{i \in N} O_i$: total flow
- b_i: capacities
- f_i: set-up costs
- d_{ij}: transportation costs

The CHLPSA consist of:

- Locate a set of hub nodes
- Single assignment patter on the flows is considered
- Minimize the total set-up and transportation costs
- Transportation costs: $F_{ijkm} = W_{ij} \left(\chi d_{ik} + \alpha d_{km} + \delta d_{mj} \right)$.

Capacitated Hub Location with Single Assignment



Notation

- N: set of nodes
- W_{ij}: flow between i and j
- $O_i = \sum_{j \in N} W_{ij}$: outgoing flow
- $D = \sum_{i \in N} O_i$: total flow
- b_i: capacities
- f_i: set-up costs
- d_{ij}: transportation costs

The CHLPSA consist of:

- Locate a set of hub nodes
- Single assignment patter on the flows is considered
- Minimize the total set-up and transportation costs
- Transportation costs: $F_{ijkm} = W_{ij} \left(\chi d_{ik} + \alpha d_{km} + \delta d_{mj} \right)$.

Literature Review

Capacitated, Single Assignment

- Campbell, 1994, *EJOR*
- Ernst and Krishnamoorthy, 1999, *AOR*
- Labbé, Yaman, Gourdin, 2005, *Mathematical Programming*
- Contreras, Díaz, Fernández, 2009, *OR Spectrum*
- Contreras, Díaz, Fernández, 2011, *Journal on Computing*

Capacitated, Multiple Assignment

- Ebery et al., 2000, *EJOR*
- Boland et al., 2004, *EJOR*
- Marín, 2005, *C&OR*
- Elhedhli and Wu, 2010, *Journal on Computing*
- Contreras, Cordeau, Laporte, 2012, *Transportation Science*

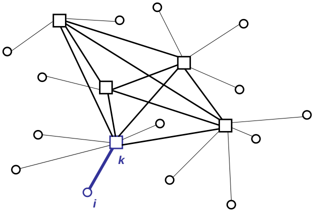
CHLPSA: MIP Formulation

Location/Assignment Variables

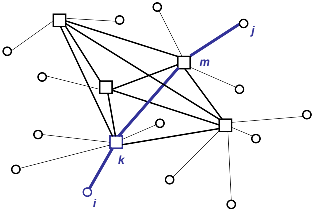
$$z_{ik} = \begin{cases} 1 & \text{if node } i \text{ is assigned to hub } k; \\ 0 & \text{otherwise.} \end{cases}$$

Path Variables

$$x_{ijkm} = \begin{cases} 1 & \text{if flow from } i \text{ to } j \text{ goes via hubs } k \text{ and } m; \\ 0 & \text{otherwise.} \end{cases}$$



z_{ik} variables (binary)



x_{ijkm} variables (binary)

CHLPSA: MIP Formulation

Integer Programming Formulation:

$$\text{minimize} \quad \sum_{k \in N} f_k z_{kk} + \sum_{i \in N} \sum_{j \in N} \sum_{k \in N} \sum_{m \in N} F_{ijkm} x_{ijkm}$$

$$\text{subject to} \quad \sum_{k \in N} \sum_{m \in N} x_{ijkm} = 1 \qquad \forall\, i, j \in N$$

$$z_{ik} \leq z_{kk} \qquad \forall\, i, k \in N$$

$$\sum_{m \in N} x_{ijkm} = z_{ik} \qquad \forall\, i, j, k \in N$$

$$\sum_{k \in N} x_{ijkm} = z_{jm} \qquad \forall\, i, j, m \in N$$

$$\sum_{i \in N} O_i z_{ik} \leq b_k z_{kk} \qquad \forall\, k \in N$$

$$\sum_{k \in N} b_k z_{kk} \geq D$$

$$x_{ijkm} \geq 0 \qquad \forall\, i, j, k, m \in N$$

$$z_{ik} \in \{0, 1\} \qquad \forall\, i, k \in N$$

Outline

Column Generation: Restricted Master Problem

$$(RMP) \quad z(S) = \min \quad \sum_{k \in N} f_k z_{kk} + \sum_{i \in N} \sum_{j \in N} \sum_{k \in S_{ij}} \sum_{m \in S_{ij}} F_{ijkm} x_{ijkm}$$

subject to

$$\sum_{k \in N} \sum_{m \in N} x_{ijkm} = 1 \qquad \forall\, i, j \in N$$

$$z_{ik} \leq z_{kk} \qquad \forall\, i, k \in N$$

$$\sum_{m \in S_{ij}} x_{ijkm} = z_{ik} \qquad \forall\, i, j, k \in N$$

$$\sum_{k \in S_{ij}} x_{ijkm} = z_{jm} \qquad \forall\, i, j, m \in N$$

$$\sum_{i \in N} O_i z_{ik} \leq b_k z_{kk} \qquad \forall\, k \in N$$

$$\sum_{k \in N} b_k z_{kk} \geq D$$

$$x_{ijkm} \geq 0 \qquad \forall\, i, j, k, m \in N$$

$$z_{ik} \in \{0, 1\} \qquad \forall\, i, k \in N$$

Let $S_{ij} \subseteq N$ be the set of *admissible hubs* for sending flow from i to j.

Column Generation: Restricted Master Problem

$$(RMP) \quad z(S) = \min \sum_{k\in N} f_k z_{kk} + \sum_{i\in N}\sum_{j\in N}\sum_{k\in S_{ij}}\sum_{m\in S_{ij}} F_{ijkm} x_{ijkm}$$

subject to

$$\sum_{k\in N}\sum_{m\in N} x_{ijkm} = 1 \qquad \forall\, i,j \in N$$

$$z_{ik} \le z_{kk} \qquad \forall\, i,k \in N$$

$$\sum_{m\in S_{ij}} x_{ijkm} = z_{ik} \qquad \forall\, i,j,k \in N$$

$$\sum_{k\in S_{ij}} x_{ijkm} = z_{jm} \qquad \forall\, i,j,m \in N$$

$$\sum_{i\in N} O_i z_{ik} \le b_k z_{kk} \qquad \forall\, k \in N$$

$$\sum_{k\in N} b_k z_{kk} \ge D$$

$$x_{ijkm} \ge 0 \qquad \forall\, i,j,k,m \in N$$

$$z_{ik} \in \{0,1\} \qquad \forall\, i,k \in N$$

Let $S_{ij} \subseteq N$ be the set of *admissible hubs* for sending flow from i to j.

Column Generation: Lagrangean Relaxation for RMP

Lagrangean Function:

$$L^S(u,v) = L_z(u,v) + L_x^S(u,v)$$

$$\begin{aligned} L_z(u,v) = \min \quad & \sum_{k \in N} \left(f_k - \sum_{j \in N} (u_{kjk} + v_{jkk}) - \xi_k(u,v) \right) z_{kk} \\ \text{s.t.} \quad & \sum_{k \in N} b_k z_{kk} \geq D \\ & z_{kk} \in \{0,1\} \qquad \forall\ k \in N, \end{aligned}$$

where,

$$\begin{aligned} \xi_k(u,v) = \min \quad & \sum_{i \in N: i \neq k} \left(\sum_{j \in N} (u_{ijk} + v_{jik}) \right) z_{ik} \\ \text{s.t.} \quad & \sum_{i \in N: i \neq k} O_i z_{ik} \leq (b_k - O_k) \\ & z_{ik} \in \{0,1\} \qquad \forall\ i \in N, i \neq k. \end{aligned}$$

Column Generation: Lagrangean Relaxation for RMP

Lagrangean Function:

$$L^S(u,v) = L_z(u,v) + L_x^S(u,v)$$

$$
\begin{aligned}
L_x^S(u,v) = \min \quad & \sum_{i\in N}\sum_{j\in N}\sum_{k\in S_{ij}}\sum_{m\in S_{ij}} (F_{ijkm} + u_{ijk} + v_{ijm})\, x_{ijkm} \\
\text{s.t.} \quad & \sum_{k\in S_{ij}}\sum_{m\in S_{ij}} x_{ijkm} = 1 \qquad \forall\, i,j \in N \\
& x_{ijkm} \geq 0 \qquad \forall\, i,j \in N, \forall\, k,m \in S_{ij}.
\end{aligned}
$$

A lower bound for the RMP is:

$$z_D^S = L^S(u^S, v^S) = \max_{u,v} \quad L^S(u,v)$$

Column Generation: Pricing Problem

For $i, j \in N$, the route $i - k - m - j$ with minimum

$$\bar{c}_{ijkm} = F_{ijkm} + u^S_{ijk} + v^S_{ijm} - w^S_{ij}(u^S, v^S)$$

can be determined by solving the *semi-assignment problem*:

$$\begin{aligned} q_{ij}(u^S, v^S, w^S) = \min \quad & \sum_{k \in N} \sum_{m \in N} \bar{c}_{ijkm} x_{ijkm} \\ \text{s.t.} \quad & \sum_{k \in N} \sum_{m \in N} x_{ijkm} = 1 \\ & x_{ijkm} \geq 0 \qquad \forall\, k, m \in N, \end{aligned}$$

- When $q_{ij}(u^S, v^S, w^S) \geq 0, \forall i, j \in N$, then z^s_D is a valid lower bound
- Otherwise, for each $i, j \in N$ such that $q_{ij}(u^S, v^S, w^S) < 0$ we enlarge $S_{ij} := S_{ij} \cup \{\widehat{k}, \widehat{m}\}$

Column Generation: Valid Lower Bounds

A valid lower bound for *MP* is given by

$$LB^1(S) = z_D^S + \sum_{i,j \in N} \min\left\{0, q_{ij}(u^S, v^S, w^S)\right\}$$

We can use a valid lower bound to:

- Apply an **early termination criteria**:

$$z_D^S - LB^1(S) < \epsilon$$

- Apply **reduction tests**: eliminate variables and constraints

Column Generation: Valid Lower Bounds

Consider $LB^2(S) = L(\widehat{u}, \widehat{v})$ obtained by **perturbing** (u^S, v^S)

Define $\widehat{u} = u^S + \alpha$ and $\widehat{v} = v^S + \beta$ such that

$$\min_{\alpha,\beta} \quad z_D^S - L(\widehat{u}, \widehat{v})$$

subject to

$$F_{ijkm} + (u_{ijk}^S + \alpha_{ijk}) + (v_{ijm}^S + \beta_{ijm}) - w_{ij}^S(u^S + \alpha, v^S + \beta) \geq 0, \quad \forall i, j, k, m \in N$$

Observe that

$$z_D^S - L(\widehat{u}, \widehat{v}) = L_z(u^S, v^S) + L_x^S(u^S, v^S) - L_z(\widehat{u}, \widehat{v}) - L_x(\widehat{u}, \widehat{v})$$

Column Generation: Valid Lower Bounds

Consider $LB^2(S) = L(\widehat{u}, \widehat{v})$ obtained by **perturbing** (u^S, v^S)

Define $\widehat{u} = u^S + \alpha$ and $\widehat{v} = v^S + \beta$ such that

$$\min_{\alpha,\beta} \quad z_D^S - L(\widehat{u}, \widehat{v})$$

subject to

$$F_{ijkm} + (u_{ijk}^S + \alpha_{ijk}) + (v_{ijm}^S + \beta_{ijm}) - w_{ij}^S(u^S + \alpha, v^S + \beta) \geq 0, \quad \forall i, j, k, m \in N$$

Observe that

$$z_D^S - L(\widehat{u}, \widehat{v}) = L_z(u^S, v^S) + L_x^S(u^S, v^S) - L_z(\widehat{u}, \widehat{v}) - L_x(\widehat{u}, \widehat{v})$$

Column Generation: Valid Lower Bounds

Consider $LB^2(S) = L(\widehat{u}, \widehat{v})$ obtained by **perturbing** (u^S, v^S)

Define $\widehat{u} = u^S + \alpha$ and $\widehat{v} = v^S + \beta$ such that

$$\min_{\alpha,\beta} \quad z_D^S - L(\widehat{u}, \widehat{v})$$

subject to

$$F_{ijkm} + (u_{ijk}^S + \alpha_{ijk}) + (v_{ijm}^S + \beta_{ijm}) - w_{ij}^S(u^S + \alpha, v^S + \beta) \geq 0, \quad \forall i, j, k, m \in N$$

Observe that

$$z_D^S - L(\widehat{u}, \widehat{v}) = L_z(u^S, v^S) + L_x^S(u^S, v^S) - L_z(\widehat{u}, \widehat{v}) - L_x(\widehat{u}, \widehat{v})$$

Column Generation: Valid Lower Bounds

To approximately solve the problem, we impose that

$$L_x^S(u^S, v^S) = L_x(\widehat{u}, \widehat{v})$$

and thus

$$z_D^S - L(\widehat{u}, \widehat{v}) = L_z(u^S, v^S) - L_z(\widehat{u}, \widehat{v})$$

An indirect way of solving the problem is by minimizing the sum of perturbations, i.e.,

$$\min \quad \sum_{i \in N} \sum_{j \in N} \left(\sum_{k \in N \setminus \widehat{k}} \alpha_{ijk} + \sum_{m \in N \setminus \widehat{m}} \beta_{ijm} \right) \quad (1)$$

$$\text{s.t.} \quad \alpha_{ijk} + \beta_{ijm} \geq -\bar{c}_{ijkm} \qquad \forall\, i, j, k, m \in N \quad (2)$$

$$\alpha_{ijk} \geq 0 \qquad \forall i, j \in N, \forall k \in N \setminus \widehat{k} \quad (3)$$

$$\beta_{ijm} \geq 0 \qquad \forall i, j \in N, \forall m \in N \setminus \widehat{m} \quad (4)$$

Column Generation: Valid Lower Bounds

To approximately solve the problem, we impose that

$$L_x^S(u^S, v^S) = L_x(\widehat{u}, \widehat{v})$$

and thus

$$z_D^S - L(\widehat{u}, \widehat{v}) = L_z(u^S, v^S) - L_z(\widehat{u}, \widehat{v})$$

An indirect way of solving the problem is by minimizing the sum of perturbations, i.e.,

$$\min \quad \sum_{i \in N} \sum_{j \in N} \left(\sum_{k \in N \setminus \widehat{k}} \alpha_{ijk} + \sum_{m \in N \setminus \widehat{m}} \beta_{ijm} \right) \qquad (1)$$

$$\text{s.t.} \quad \alpha_{ijk} + \beta_{ijm} \geq -\bar{c}_{ijkm} \qquad \forall\, i, j, k, m \in N \qquad (2)$$

$$\alpha_{ijk} \geq 0 \qquad \forall i, j \in N, \forall k \in N \setminus \widehat{k} \qquad (3)$$

$$\beta_{ijm} \geq 0 \qquad \forall i, j \in N, \forall m \in N \setminus \widehat{m} \qquad (4)$$

Column Generation: Valid Lower Bounds

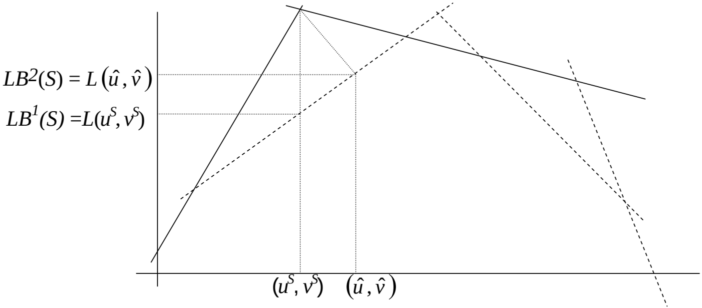


Figure 2: Comparison between $LB^1(S)$ and $LB^2(S)$

Column Generation: Quality of Bounds

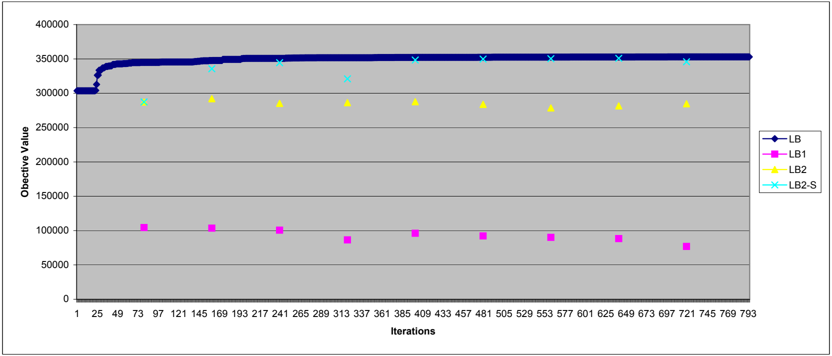


Figure 2: Comparison between $LB^1(S)$ and $LB^2(S)$ using the 70TT instance of the AP data set

Comparison of Standard and Stabilized CG

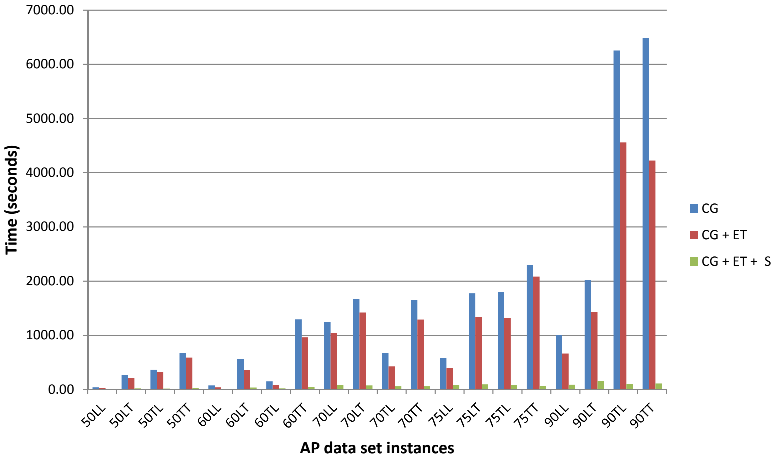


Figure 3: Comparison of CPU times between different versions of CG

Comparison of Standard and Stabilized CG

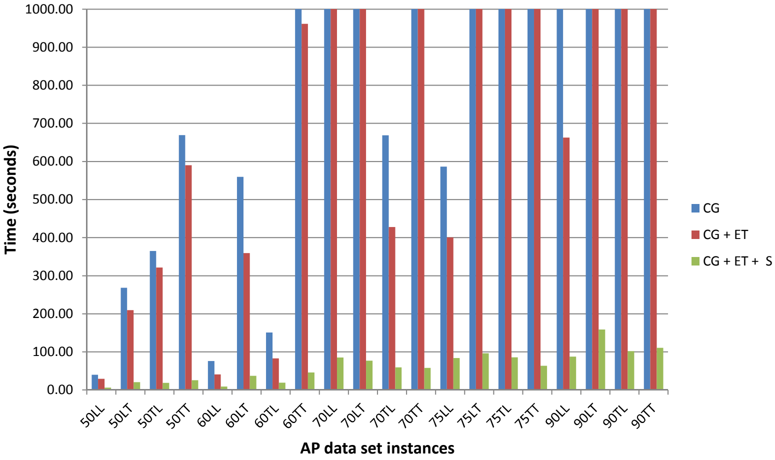


Figure 4: Comparison of CPU times between different versions of CG

Branch and Price for CHLPSA

- The CG procedure is embedded into a branch and bound
- The enumeration algorithm consist of three phases:
  - Partial enumeration
  - Branch in location variables
  - Branch in assignment variables

Outline

Computational Results for the Branch and Price

	Root Node		Totaltime	
Prob	*Dev*	*time*	*BP*	*CPLEX*
70LL	0.80	29.26	214.23	170.84
70LT	0.84	28.75	245.76	1122.67
70TL	0.00	9.80	9.47	43.27
70TT	0.29	8.16	48.55	292.44
75LL	0.20	20.73	48.60	43.61
75LT	0.33	22.88	33.90	726.91
75TL	0.27	13.64	14.86	89.39
75TT	2.50	13.31	24.83	14400.00
90LL	0.10	39.28	39.28	79.11
90LT	0.18	33.56	80.87	4182.70
90TL	0.85	25.92	228.87	971.57
90TT	0.52	27.12	213.90	1220.51
100LL	0.69	138.18	459.89	862.93
100LT	0.30	83.10	347.95	1068.98
100TL	3.43	41.20	124.92	1571.71
100TT	0.79	40.92	328.11	14400.00

Table: Branch and Price results

Computational Results for the Branch and Price

Prob	Root Node Dev	Root Node time	Totaltime BP	Totaltime CPLEX
125LL	1.70	338.89	1650.57	memory
125LT	0.46	294.60	552.99	memory
125TL	0.10	37.57	41.22	memory
125TT	0.29	51.68	322.73	memory
150LL	0.39	738.17	3347.21	memory
150LT	1.63	990.73	11818.19	memory
150TL	0.99	191.52	1114.95	memory
150TT	1.73	174.94	4299.28	memory
175LL	0.32	1349.70	3418.10	memory
175LT	1.23	1632.84	12408.50	memory
175TL	0.30	232.80	256.60	memory
175TT	1.34	307.60	4886.88	memory
200LL	0.49	1606.28	5813.00	memory
200LT	1.36	1984.47	45874.73	memory
200TL	0.34	706.52	869.67	memory
200TT	0.46	447.92	3211.40	memory

Table: Branch and Price results

Outline

Conclusions and Future Research

- Bounding procedure: combination of column generation and lagrangean relaxation methods
- Constraint stabilization method improves the performance of standard CG
- Branch and price optimally solves large-scale instances with up to 200 nondes
- Generalize and extend the stabilization method to other classes of optimization problems