A Capacity Scaling Procedure for the Multi-Commodity Capacitated Network Design Problem

Ryutsu Keizai University

Naoto KATAYAMA

International Seminar on Applied Mathematics for Real World Problems 2006

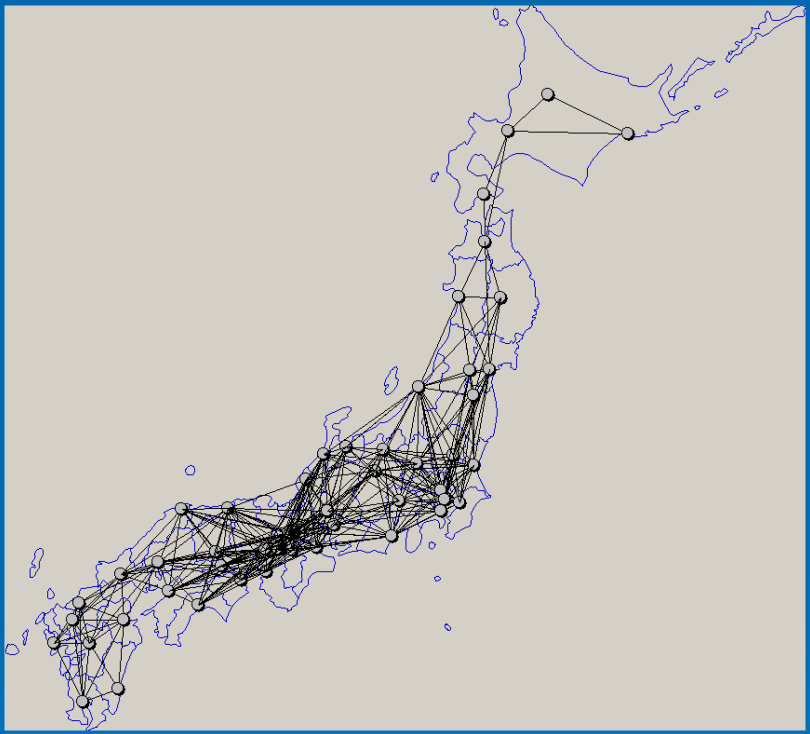
Multi-Commodity Network Design Problem

- The basic model for transportation network design, communication network design, traffic network design and etc.
- Given nodes, arcs, multi-commodities, costs, demands and side constraints,
- Select arcs to be minimized the total cost.
- Find paths for commodities.

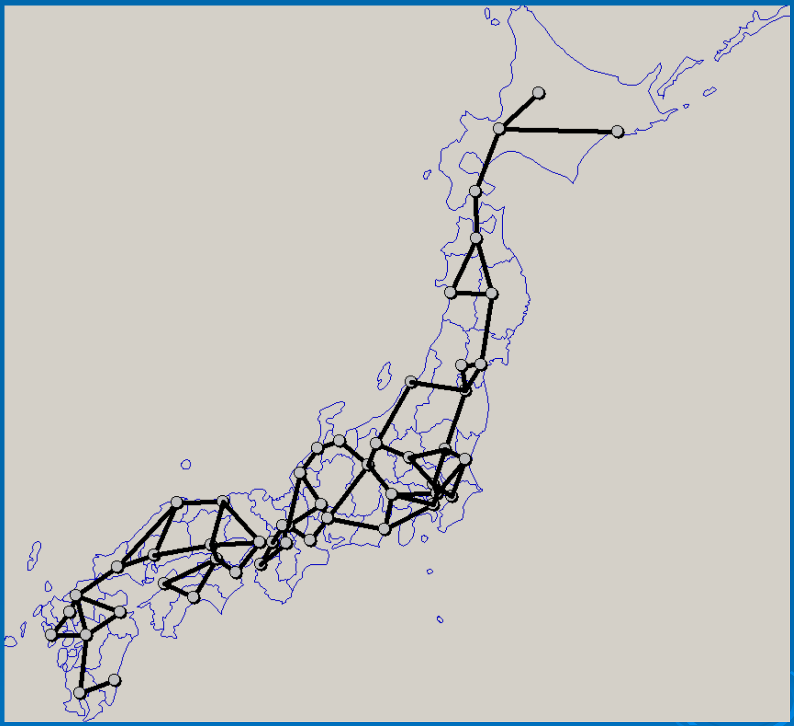
Multi-Commodity Network Design Problem

- Budget network design problem
- Fixed charge network design problem
- Capacitated network design problem
- User equilibrium network design problem
- Hub network design problem
- Less-than-truckload problem

Multi-Commodity Network Design Problem



Select arcs



Japanese express highway design; Budget network design, Total length 9000Km,50 nodes,654 arcs,1225 commodities,1.3million variables (katayama2003)

Capacitated Network Design Problem (CND)

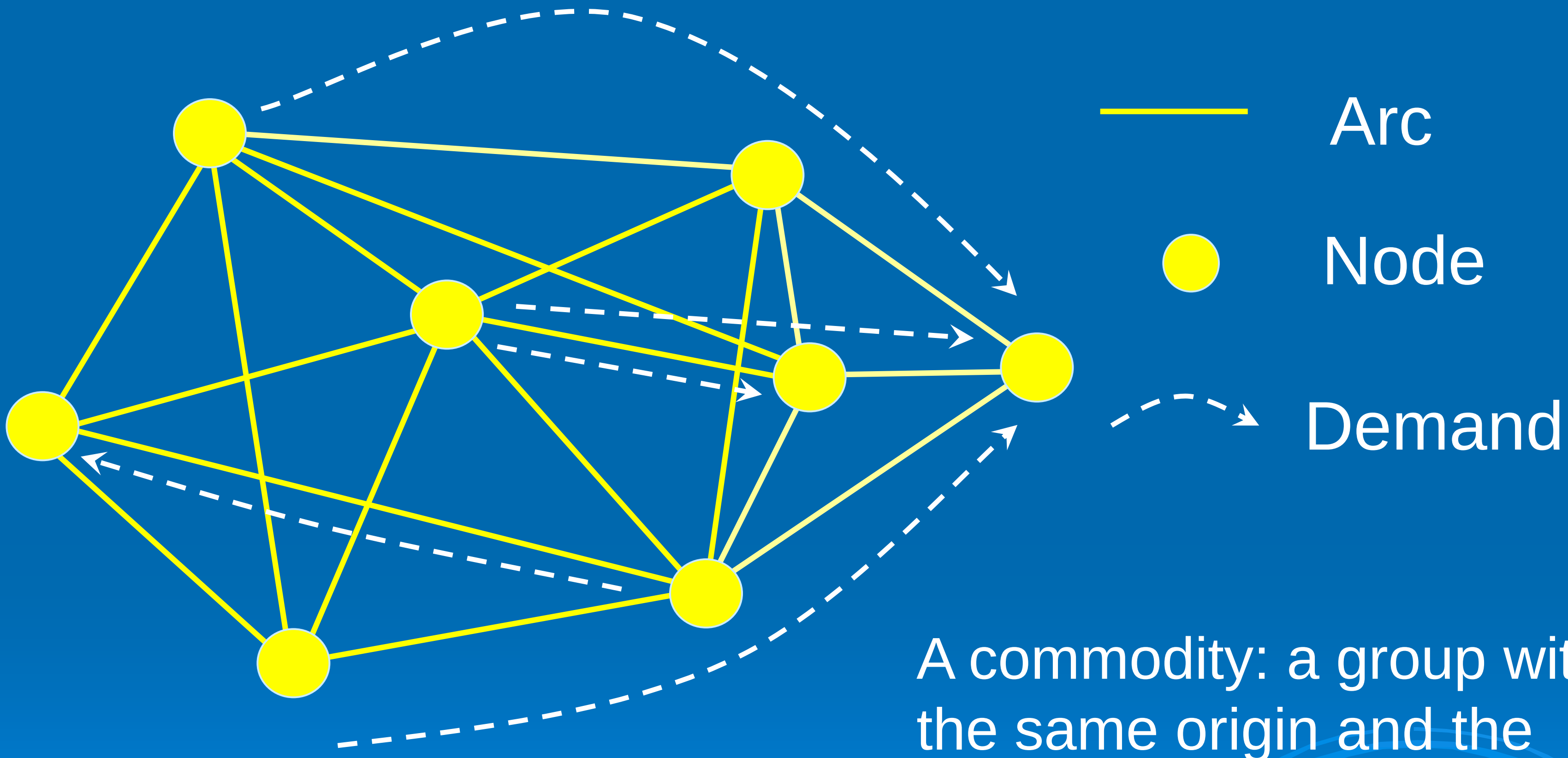
- Capacitated network design problem:
  - Large size mixed integer problem
  - NP-hard
- Need to develop approximation methods to solve CND.

- Propose:
  - For a strong path type formulation,
  - A capacity scaling procedure
  - With column and row generation techniques.

Capacitated Network Design Problem (CND)

- Given:
  - The set of nodes and the set of arcs,
  - The set of multi-commodities and demands,
  - Arc design costs and arc flow costs,
  - Arc capacities.
- Find:
  - A network configuration; selected arcs,
  - Path flows for each commodity
  - The total of design and flow costs should be minimized.

Capacitated Network Design Problem



A commodity: a group with the same origin and the same destination.

Capacitated Network Design Problem

Literatures: Valid inequality

- Katayama(1992)
  Cut-set inequality, detour inequality
- Magnanti-Mirchandan(1993)
  Cut-set inequality, three-partition inequality, arc residual capacity inequality
- Barahona(1996)
  Valid inequalities, multi-cut inequality
- Chouman-Crainic-Gendron(2003)
  Cut-set inequalities and lifting

Lower bound

- Katayama(1993)

  Dual ascent method for integer rounding cuts with weak cut-set formulation

- Gendron-Crainic(1994,1996)

  Lagrange relaxation method for capacity and/or flow conservation constraints

- Herrmann-Ioannou-Minis(1996)

  Dual ascent method

- Crainic-Frangioni-Gendron(1998)

  Bundle-based Lagrange relaxation method

- Katayama(2003)

  Lagrange multiplier adjustment method for a strong and cut-set formulation

- Chouman-Crainic-Gendron(2003)

  Linear relaxation for cut-set inequalities and lifting

Heuristics

- Gendron-Crainic(1994,1996)
  Resource decomposition heuristics
- Holmberg-Yuan(2000)
  A Lagrange heuristic based branch and bound method
- Crainic-Gendreau-Farvolden(2000)
  Simplex-based tabu search heuristics
- Crainic-Gendreau(2002)
  Cooperative parallel tabu search heuristics
- Ghamlouche-Crainic-Gendreau(2003)
  Cycle based neighborhood search heuristics
- Crainic-Gendron-Hernu(2003)
  Slope scaling and Lagrange perturbation heuristics for a weak formulation
- Ghamlouche-Crainic-Gendreau(2002,2004)
  Path relinking heuristics

Notations

- N :set of nodes
- A :set of arcs
- K :set of commodities
- P : set of paths
- P^k :set of paths for commodity k
- P'^k :subset of path for k
- d^k :demand for commodity k
- c_{ij}^k :unit flow cost for commodity on arc (i, j)
- f_{ij} :arc design cost of arc(i, j)
- C_{ij} :arc capacity of arc (i, j)
- δ_{ijp}:1, if arc (i, j) is included in path, otherwise 0
- x_p^k : continuous path flow variable of commodity k on path p
- y_{ij} : binary arc design variable 1, if arc (i, j) is selected, otherwise 0,

Strong Path Flow Formulation

$$(CND) \quad \min \sum_{(i,j)\in A} \sum_{k\in A} c_{ij}^k \sum_{p\in P^k} \delta_{ijp} + \sum_{(i,j)\in A} f_{ij}$$

$$st \quad \sum_{k\in K} \sum_{p\in P^k} \delta_{ijp} x_p^k \le C_{ij} y_{ij} \qquad (i,j)\in A$$

$$\sum_{p\in P^k} x_p^k = d^k \qquad k\in K$$

$$\sum_{p\in P^k} \delta_{ijp} x_p^k \le d^k y_{ij} \quad (i,j)\in A \;\; k\in K$$

$$x_p^k \ge 0 \qquad k\in K \qquad p\in P^k$$

$$y_{ij} \in \{0,1\} \qquad (i,j)\in A$$

Strong Path Flow Formulation

$$(CND) \quad min \sum_{(i,j)\in A} \sum_{k\in A} c_{ij}^{k} \sum_{p\in P^k} \delta_{ijp} x_p^k + \sum_{(i,j)\in A} f_{ij}\, y_{ij}$$

Flow costs

Arc design costs

Strong Path Flow Formulation

Arc capacity constraints

$$st \qquad \sum_{k\in K}\sum_{p\in P^k}\delta_{ijp}x_p^k \le C_{ij}y_{ij} \qquad (i,j)\in A$$

$$\sum_{p\in P^k} x_p^k = d^k \qquad k\in K$$

Flow conservation constrainers

Strong Path Flow Formulation

Forcing constraints for each arc and commodity

$$\sum_{p \in P^k} \delta_{ijp} x_p^k \le d^k y_{ij} \quad (i,j) \in A \quad k \in K$$

Tight and strong valid constrains

$$x_p^k \ge 0 \qquad k \in K \qquad p \in P^k$$

$$y_{ij} \in \{0,1\} \qquad (i,j) \in A$$

Strong Path Flow Formulation

- A large mixed integer problem
  - The number of integer design variables is $O(|A|)$
  - The number of flow variables is $O(|K| \cdot |A|!)$

- Fixed design variables, a tight multi-commodity network flow problem; not so easy problem.
- The linear relaxation problem is not so easy problem.
- The linear relaxation problem without forcing constraints is an easy problem.

Strong Path Flow Formulation

- The number of path flow variables is huge.

  $O(|K| \cdot |A|!)$

  - A column generation technique

- Forcing constraints for each arc and commodity are tight valid inequalities.
- The number of forcing constraints is large.

  $O(|K||A|) = (|N|^4)$

  - A row generation technique

Capacity Scaling Procedure

- An aproximation method for capacitated problems
- Until design variables y converge to 0 or 1.
  - Solve the linear relaxation problem (LP).
  - Change arc capacities based on a solution of LP and previous capacities.
- Solve the multi-commodity flow problem where all design variables y are fixed to the convergent solutions.

Linear Relaxation Problem

$$(LP) \quad min \quad \sum_{(i,j)\in A} \sum_{k\in A} c_{ij}^{k} \sum_{p\in P^k} \delta_{ijp} \quad + \sum_{(i,j)\in A} f_{ij}$$

$$st \quad \sum_{k\in K} \sum_{p\in P^k} \delta_{ijp} x_p^k \leq C_{ij} y_{ij} \qquad (i,j)\in A$$

$$\sum_{p\in P^k} x_p^k = d^k \qquad k\in K$$

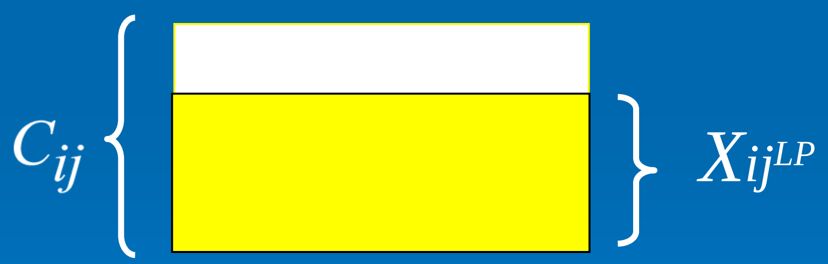
$$\sum_{p\in P^k} \delta_{ijp} x_p^k \leq d^k y_{ij} \quad (i,j)\in A \quad k\in K$$

$$x_p^k \geq 0 \qquad k\in K \qquad p\in P^k$$

$$0 \leq y_{ij} \leq 1 \qquad (i,j)\in A$$

Capacity Scaling Procedure

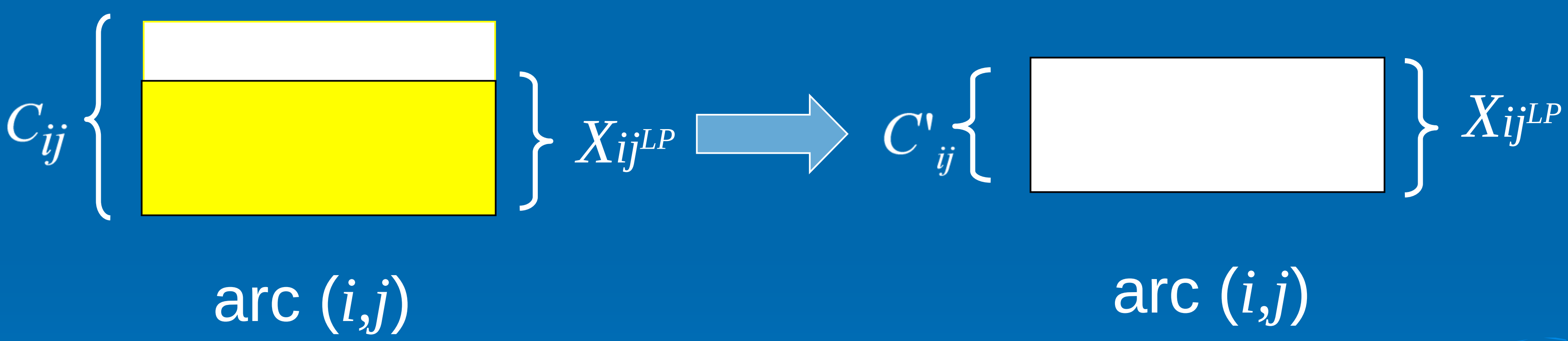
- Solve the linear relaxation problem
- Obtain LP solution y_{ij}^{LP}, X_{ij}^{LP} $(= \sum_{k\in K}\sum_{p\in P'^k} \delta_{ijp} x_p^k$, arc flow)
- y_{ij}^{LP} may be a fraction between 0 and 1.



arc (i,j)

Capacity Scaling Procedure

- Solve the linear relaxation problem
- LP solution y_{ij}^{LP}, X_{ij}^{LP} ($= \sum_{k \in K} \sum_{p \in P'^k} \delta_{ijp} x_p^k$, arc flow)



Change capacities

Capacity Scaling Procedure

- If all flow variables X_{ij}^{LP} are the optimal flows for *CND*, the optimal design variables for the problem changed capacities are equivalent to the optimal design variables for *CND*
- If the optimal capacities (= the optimal flow values) were found, the optimal solution for *CND* could be obtained.
- If capacities can be close to the optimal flow values bit at a time, good approximate solutions may be obtained.

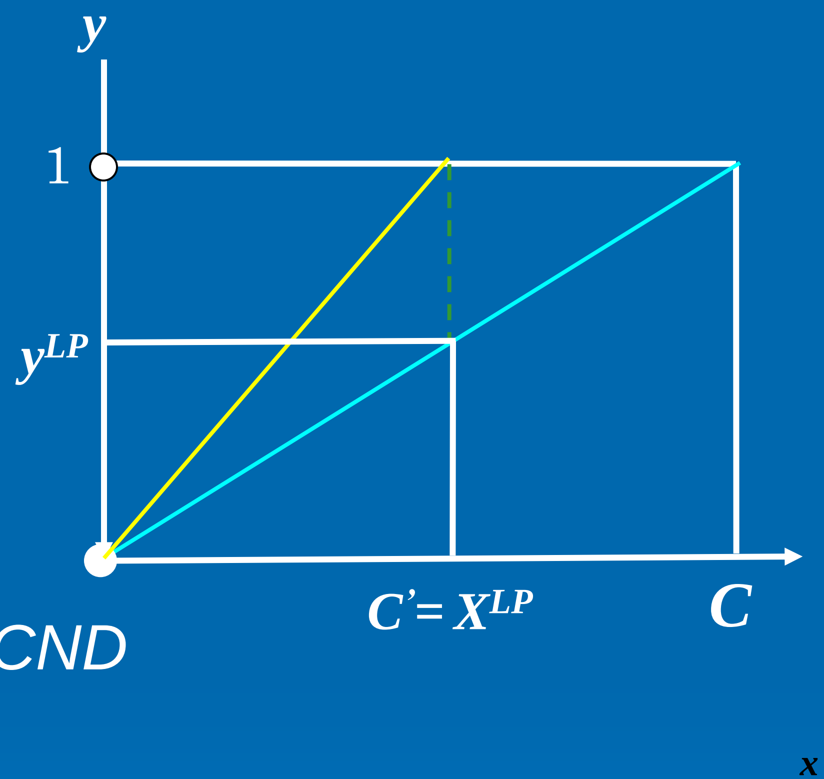
Capacity Scaling Procedure

- Relationship between X and y

$$X_{ij} \le C_{ij} y_{ij}, X_{ij} \ge 0, y_{ij} = \{0,1\}$$

- Relationship between X and y at the optimal solutions for LP

$$X_{ij}^{LP} \approx C_{ij} y_{ij}^{LP}, X_{ij}^{LP} \ge 0, 0 \le y_{ij}^{LP} \le 1$$



- Assume X^{LP} is the optimal flow for *CND*

- Change capacity C to $C'_{ij} = X_{ij}^{LP}$

Capacity Scaling Procedure

- Change capacity *C'* by the smoothing parameter λ
- Prevent large change of *C'* and flows

$$C'_{ij} = \lambda X_{ij}^{LP} + (1-\lambda)C'_{ij}$$

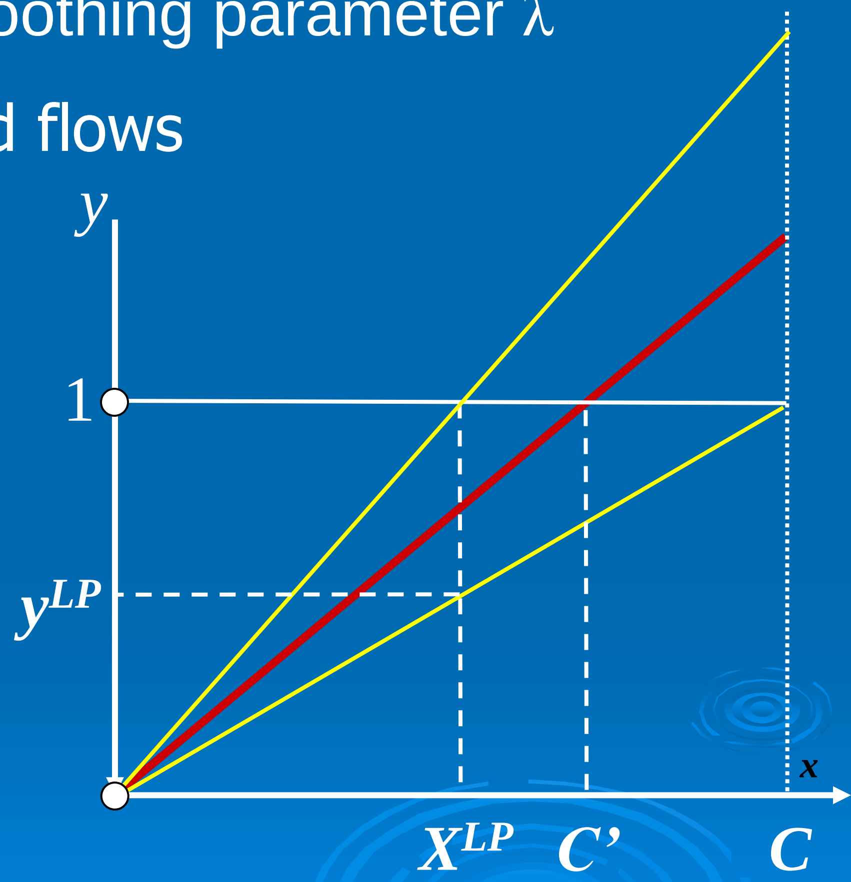
- Change the upper bound of *y*

$$0 \le X_{ij} \le C_{ij}$$

$$0 \le C'_{ij}\, y_{ij} \le C_{ij}$$

↓

$$0 \le y_{ij} \le \frac{C_{ij}}{C'_{ij}}$$



Capacity Scaling Procedure

$$(CND(C')) \quad min \sum_{(i,j)\in A} \sum_{k\in A} c_{ij}^{k} \sum_{p\in P^{k}} \delta_{ijp} x_{p}^{k} + \sum_{(i,j)\in A} f_{ij} y_{ij}$$

$$st \quad \sum_{k\in K} \sum_{p\in P^{k}} \delta_{ijp} x_{p}^{k} \le C'_{ij} y_{ij} \quad (i,j)\in A$$

$$\sum_{p\in P^{k}} \delta_{ijp} x_{p}^{k} \le d^{k} y_{ij} \quad (i,j)\in A, k\in K$$

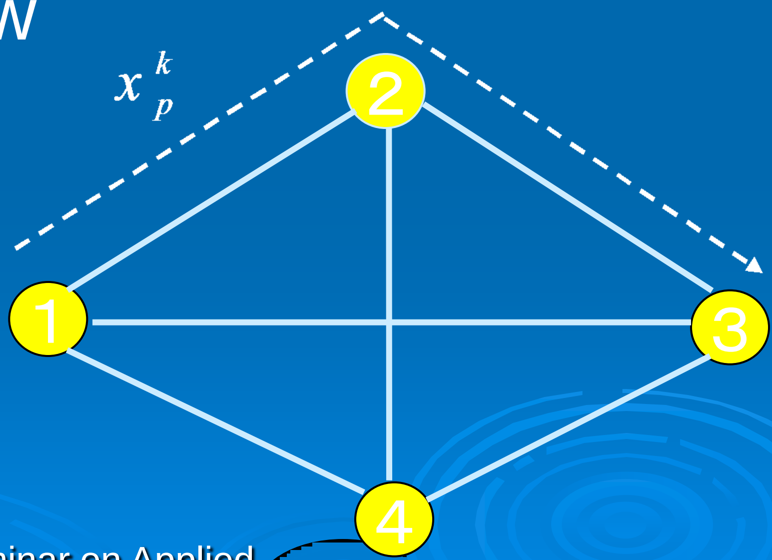
$$\sum_{p\in P^{k}} x_{p}^{k} = d^{k} \quad k\in K$$

$$x_{p}^{k} \ge 0 \quad k\in K, p\in P^{k}$$

$$0 \le y_{ij} \le C_{ij} / C'_{ij} \quad (i,j)\in A$$

Column generation

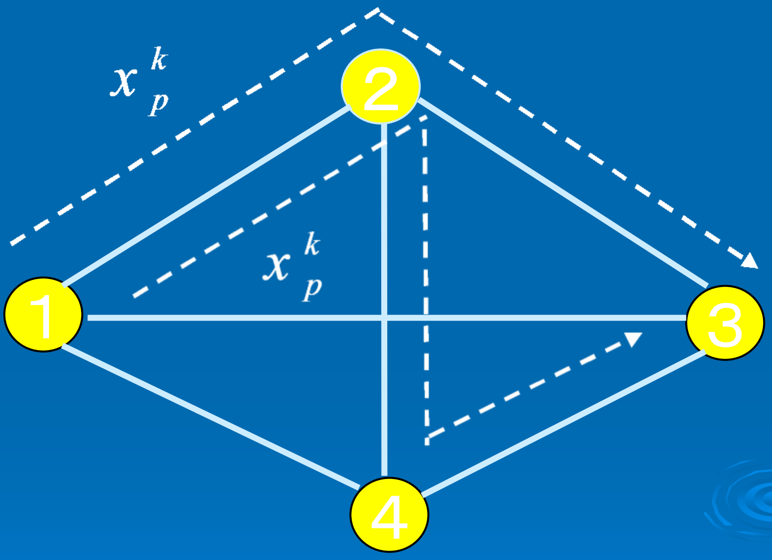
- The number of paths may be huge.
- Solve LP with the subset of paths, P' instead of P.
- If not optimal:
  - Generate new path flow variables (columns),



Column generation

- The number of paths may be huge.
- Solve LP with the subset of paths P' instead of P.

- If not optimal:
  - Generate new path flow variables (columns),
  - Add paths to LP as columns,
  - Solve LP again.



Restricted primal problem

$$(CND(C',P')) \quad \min \sum_{(i,j)\in A} \sum_{k\in A} c_{ij}^k \sum_{p\in P^k} \delta_{ijp} x_p^k + \sum_{(i,j)\in A} f_{ij} y_{ij}$$

$$\pi_{ij} \cdots \qquad st \quad \sum_{k\in K} \sum_{p\in P'^k} \delta_{ijp} x_p^k \le C'_{ij} y_{ij} \quad (i,j)\in A$$

$$\sigma_{ij}^k \cdots \qquad \sum_{p\in P'^k} \delta_{ijp} x_p^k \le d^k y_{ij} \quad (i,j)\in A, k\in K, \sum_{p\in P'^k} \delta_{ijp} > 0$$

$$\mu^k \cdots \qquad \sum_{p\in P'^k} x_p^k = d^k \quad k\in K$$

$$x_p^k \ge 0 \quad k\in K, p\in P'^k$$

$$0 \le y_{ij} \le C_{ij} / C'_{ij} \quad (i,j)\in A$$

Column generation - Reduced cost

- Using dual variables, substitute constraints into the objective function

$$\sum_{k\in K}\sum_{p\in P'^k}\Big[\sum_{(i,j)\in A}(c_{ij}^k+\pi_{ij}+\sigma_{ij}^k)\delta_{ijp}-\mu^k\Big]x_p^k+\sum_{(i,j)\in A}\Big(f_{ij}-\pi_{ij}C_{ij}-\sum_{k\in K}\sigma_{ij}^k d^k\Big)y_{ij}+\sum_{k\in K}\mu^k d^k$$

- All coefficients (reduced costs) of x_p^k should be nonnegative at the optimal solution.
- Solve a pricing problem, find new x_p^k such that a reduced cost is negative.

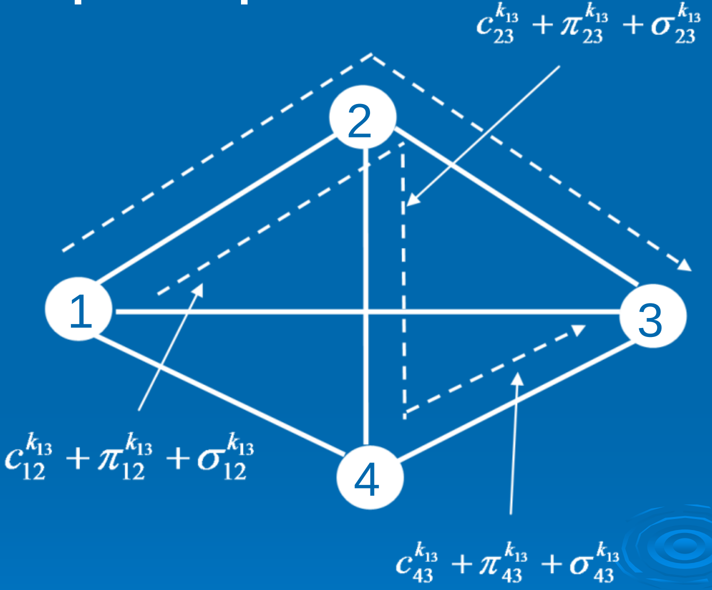
Column generation - Reduced cost

- For each commodity k,
  - A pricing problem is shortest path problem

$$\min \quad \sum_{(i,j)\in A} (c_{ij}^{k} + \pi_{ij} + \sigma_{ij}^{k}) z_{ij} - \mu^{k}$$

$$st \quad \sum_{j\in N} z_{ij} - \sum_{j\in N} z_{ji} = \begin{cases} -1 & i = O^{k} \\ 1 & i = D^{k} \\ 0 & i \in N \setminus \{O^{k}, D^{k}\} \end{cases}$$

$$z_{ij} \in \{0,1\} \quad (i,j) \in A$$



- If the reduced cost of the optimal length is negative, this shortest path is a new column.

Row generation

- The number of forcing constraints is $O(|K||A|)$

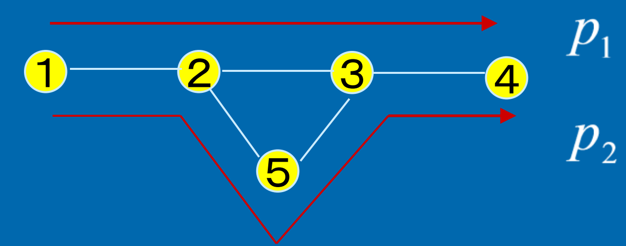
$$\sum_{p \in P'^k} \delta_{ijp} x_p^k \le d^k y_{ij} \quad (i,j) \in A, k \in K$$

- When columns are generated, forcing constraints (rows) corresponding to these columns are only generated.

Row generation

p_1 : Initial path: $1 \to 2 \to 3 \to 4$

Generated path $p_2 : 1 \to 2 \to 5 \to 3 \to 4$



$$\begin{cases} x_{p_1}^k \le d^k y_{12} \\ x_{p_1}^k \le d^k y_{23} \\ x_{p_1}^k \le d^k y_{34} \end{cases}$$

$$\sum_{p \in P'^k} \delta_{ijp} x_p^k \le d^k y_{ij}$$

$$x_{p_1}^k + x_{p_2}^k \le d^k y_{12}$$

$$x_{p_1}^k \le d^k y_{23}$$

$$x_{p_1}^k + x_{p_2}^k \le d^k y_{34}$$

$$x_{p_2}^k \le d^k y_{25}$$

$$x_{p_2}^k \le d^k y_{53}$$

Generated column

Generated rows

Capacity Scaling Procedure

- Even if using column and row generation techniques, it takes long time to solve CND(C',P') .
- As design variables change by capacity scaling procedure, the column and row generation are iterated.
- If all design variables converge to 0 or 1, we solve a multi-commodity flow problem fixed all design variables and an upper bound and an approximate solution are obtained.
- There is no guarantee such that all design variables converge to 0 or 1.
- It takes much iteration until most design variables converge to 0 or 1.

Capacity Scaling Procedure

- If the number of free design variables (not converge to 0 or 1) is less than the certain number, then branch-and-bound method is applied to the problem with free design variables.
- If the number of free design variables is small (e.g. 75-100), the problem can be solve easily by CPLEX.
- After the maximum iteration number, solve the multi-commodity flow problem fixed all design variables by an arc flow formulation exactly and an upper bound is obtained.

Capacity Scaling Procedure

1) Set $\lambda \in (0,1]$, B and an initial path set P', and $C' := C$,
2) Until exceed the maximum iteration number and the number of free design variables is less than B;
   a) Solve CND(C', P') by column and row generation, obtain solutions.
   b) Change capacities C' and upper bounds of y.
   c) When the number of free design variables is less than B, solve the problem with free design variables by a branch-and-bound method and obtain an upper bound.
3) Solve the multi-commodity flow problem fixed all design variables by an arc flow formulation and obtain an upper bound.

Numerical Experiments

- Data: Crainic et al. (2000,2003,2004)

  Bench mark data for CND
- Computer：Pentium 4, 3.2GHz, RAM 1Gb
- Language: Visual Basic.NET
- Software for solving LP and branch-and-bound ：CPLEX 9.0
- Smoothing parameter λ: 0.025,0.050,0.075,0.100,0.125,0.150
- The minimum number of variables when applying a branch-and-bound method: 75
- The maximum iteration number: 40

Numerical Experiments

Table 1. Computational Results (Type C problem)

Node	Arc	Commodity	Fixed Cost	Capacity	Best Upper Bound	Capacity Scaling	Difference	Improvement
20	100	10	V	L	14,712	14,712	0	0.00%
20	100	10	F	L	14,941	15,037	96	−0.64%
20	100	10	F	T	49,899	50,771	872	−1.75%
25	100	30	V	T	365,272	365,272	0	0.00%
25	100	30	F	L	37,515	37,471	-44	0.12%
25	100	30	F	T	86,296	85,801	-495	0.57%
20	230	40	V	L	424,385	424,075	-310	0.07%
20	230	40	V	T	371,475	371,906	431	−0.12%
20	230	40	F	T	644,172	644,483	311	−0.05%
20	300	40	V	L	429,398	429,398	0	0.00%
20	300	40	F	L	589,190	587,800	-1390	0.24%

- Best Upper Bound: the best known bound; Simplex-Based Tabu, Cycle-Based Neighborhoods, Path Relinking, Multilevel Cooperative Tabu

Numerical Experiments

Table 2. Computational Results (Type C problem)

Node	Arc	Commodity	Fixed Cost	Capacity	Best Upper Bound	Capacity Scaling	Difference	Improvement
20	300	40	V	T	464,509	464,569	60	−0.01%
20	300	40	F	T	606,364	604,198	-2166	0.36%
20	230	200	V	L	98,582	94,247	-4335	4.40%
20	230	200	F	L	143,150	137,642	-5508	3.85%
20	230	200	V	T	102,030	97,968	-4062	3.98%
20	230	200	F	T	141,188	136,130	-5059	3.58%
20	300	200	V	L	78,184	74,913	-3271	4.18%
20	300	200	F	L	121,951	115,784	-6168	5.06%
20	300	200	V	T	77,251	75,302	-1949	2.52%
20	300	200	F	T	111,173	107,858	-3316	2.98%
100	400	10	V	L	28,485	28,426	-59	0.21%
100	400	10	F	L	23,949	24,459	510	−2.13%
100	400	10	F	T	65,278	73,566	8288	−12.70%
100	400	30	V	T	384,926	384,883	-43	0.01%
100	400	30	F	L	50,456	51,956	1500	−2.97%
100	400	30	F	T	141,359	144,314	2955	−2.09%

Numerical Experiments

Table 3. Computational Results (Type C problem)

Node	Arc	Commodity	Fixed Cost	Capacity	Best Upper Bound	Capacity Scaling	Difference	Improvement
30	520	100	V	L	54,904	54,088	-816	1.49%
30	520	100	F	L	99,586	94,801	-4785	4.80%
30	520	100	V	T	52,985	52,282	-703	1.33%
30	520	100	F	T	102,477	98,839	-3638	3.55%
30	700	100	V	L	48,398	47,635	-763	1.58%
30	700	100	F	L	62,471	60,194	-2277	3.64%
30	700	100	V	T	47,025	46,169	-856	1.82%
30	700	100	F	T	56,576	55,359	-1217	2.15%
30	520	400	V	L	115,671	112,846	-2825	2.44%
30	520	400	F	L	156,601	149,446	-7155	4.57%
30	520	400	V	T	120,170	114,641	-5530	4.60%
30	520	400	F	T	160,217	152,744	-7474	4.66%
30	700	400	V	L	102,631	97,972	-4659	4.54%
30	700	400	F	L	143,988	135,064	-8924	6.20%
30	700	400	V	T	99,195	95,306	-3889	3.92%
30	700	400	F	T	138,266	130,148	-8118	5.87%
AVERAGE					168,076	166,058	−2,018	1.55%

Numerical Experiments

Table 4. Computational Times (Type C problem; seconds)

Node	Arc	Commodity	Fixed Cost	Capacity	Simplex Tabu	Cycle Based	Path Relinking	Capacity Scaling
20	100	10	V	L	6	19	13	1
20	100	10	F	L	8	22	14	8
20	100	10	F	T	17	34	24	3
25	100	30	V	T	17	142	101	3
25	100	30	F	L	33	112	75	18
25	100	30	F	T	72	133	97	13
20	230	40	V	L	71	214	149	3
20	230	40	V	T	90	241	157	4
20	230	40	F	T	122	260	172	4
20	300	40	V	L	71	305	225	3

- Simplex-Based Tabu, Cycle-Based Neighborhoods, Path Relinking:

  Sun Enterprise 10000 400Mz with 64CPU

- Capacity Scaling: Pentium4 3.2GHz

Numerical Experiments

Table 5. Computational Times (Type C problem ; seconds)

Node	Arc	Commodity	Fixed Cost	Capacity	Simplex Tabu	Cycle Based	Path Relinking	Capacity Scaling
20	300	40	F	L	113	336	228	6
20	300	40	V	T	145	379	248	4
20	300	40	F	T	123	349	214	5
20	230	200	V	L	505	2586	2495	503
20	230	200	F	L	492	3142	2878	1193
20	230	200	V	T	548	2730	2211	931
20	230	200	F	T	890	3634	3386	1122
20	300	200	V	L	982	4086	3566	676
20	300	200	F	L	1317	4210	4013	3691
20	300	200	V	T	938	4204	3924	755
20	300	200	F	T	1066	4855	3857	4163
100	400	10	V	L	33	252	89	6
100	400	10	F	L	33	197	83	78
100	400	10	F	T	81	451	210	32
100	400	30	V	T	278	1200	493	30
100	400	30	F	L	100	717	315	384
100	400	30	F	T	216	1301	481	100

Numerical Experiments

Table 6. Computational Times (Type C problem ; seconds)

Node	Arc	Commodity	Fixed Cost	Capacity	Simplex Tabu	Cycle-Based	Path Relinking	Capacity Scaling
30	520	100	V	L	996	2261	1194	35
30	520	100	F	L	939	2684	1460	324
30	520	100	V	T	1219	2716	1514	47
30	520	100	F	T	670	2892	1523	148
30	700	100	V	L	1265	2960	1861	34
30	700	100	F	L	1480	3182	1838	123
30	700	100	V	T	2426	3746	1894	44
30	700	100	F	T	1736	3547	1706	79
30	520	400	V	L	5789	55771	27477	885
30	520	400	F	L	6407	40070	36669	3087
30	520	400	V	T	6522	4679	23089	260
30	520	400	F	T	8415	49887	52173	1154
30	700	400	V	L	12636	38857	22315	424
30	700	400	F	L	11368	68214	75665	1398
30	700	400	V	T	15880	51764	24289	634
30	700	400	F	T	11660	79053	44936	1573
				AVERAGE	2274	10428	8124	558

Numerical Experiments

Table 7. Gaps (Type R problem, 153 problems)

Capacity Level	Fixed Cost Level	Simplex Tabu	Cycle Based	Path Relinking	Cooperative Tabu	Capacity Scaling	Improvement
1	1	2.49%	1.48%	0.76%	2.48%	0.06%	-0.70%
1	5	12.31%	3.49%	2.43%	7.85%	0.29%	-2.14%
1	10	19.86%	3.55%	3.09%	12.09%	-0.14%	-3.23%
2	1	2.21%	1.31%	0.78%	1.67%	0.09%	-0.69%
2	5	9.16%	3.68%	2.64%	6.76%	0.28%	-2.36%
2	10	14.45%	4.27%	3.04%	8.99%	0.26%	-2.78%
8	1	2.96%	1.74%	1.15%	1.95%	0.33%	-0.82%
8	5	6.33%	4.14%	3.23%	6.41%	0.72%	-2.51%
8	10	8.60%	4.40%	4.11%	4.94%	0.66%	-3.45%

- LB =the optimal value or the best upper bound by CPLEX by Crainic et al.
- Gap= (UB-LB)/LB. Gaps are not real gaps.
- Nodes 10,20; Arcs 25,50,75,100,200,300; Commodities 10,25,40,50,100,200

Conclusion

- For the multi-commodity capacitated network design problem,
  - Present a strong path type formulation,
  - Propose a capacity scaling procedure combined with column and row generation techniques and a branch-and-bound method.

Conclusion

- Test the effectiveness of the capacity scaling procedure by Crainic's data.
  - For type C problem, 31 new best solutions out of 44 problems are obtained.
  - For type R problem, improvement by capacity scaling procedure is 0.69%-3.45%.