Outer Approximation and Submodular Cuts for Maximum Capture Facility Location Problems with Random Utilities

Ivana Ljubic (ESSEC, Paris, France)

Joint work with
Eduardo Moreno (Universidad Adolfo Ibañez, Santiago, Chile)

Optimization 2017,
Lisbon, September 6-8, 2017

Outline

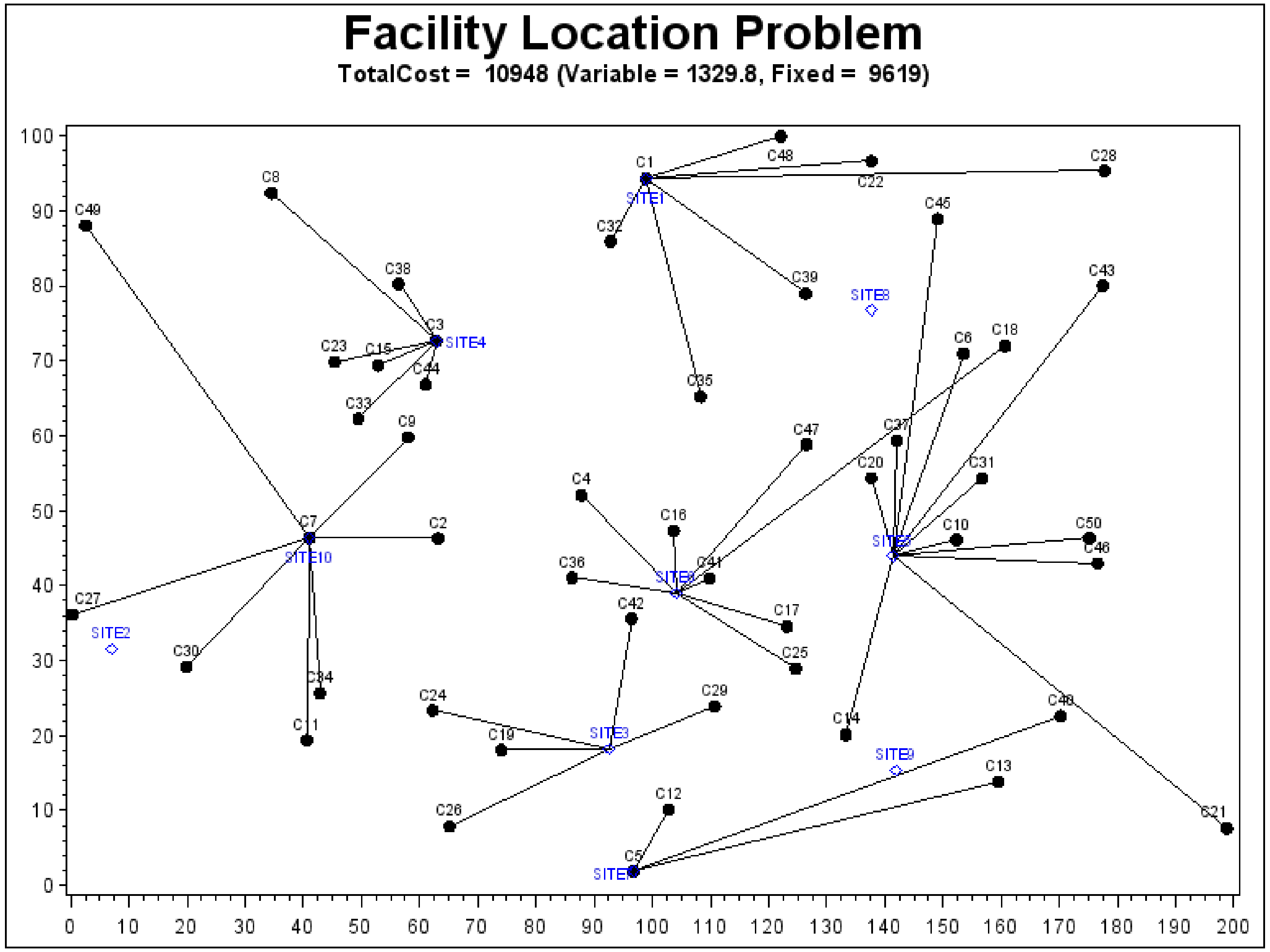
- Max-Capture problem with random utilities
- Methodological approaches to solve the problem
- Proposed branch-and-cut method
- Computational comparison

Facility Location

- One of the most classical problems in Operations Research/Management

- To choose a point in the plane that minimize the weighted sum of distance to *n* existing points (de Fermat 1643, Weber 1909)

- Classical discrete case:
  - Installation costs of facilities
  - transportation cost from clients to facilities
  - minimize the total cost



Competitive Facility Location

- Ice-cream vendor problem (Hotelling ’29)
  - homogeneous product → maximize market share.
  - Clients choose based on distance.

- 70’s: extension to other networks, Nash equilibriums
- Slater (75) and Hakimi (83) formulated the problem as a Facility Location problem.

MAX-Capture Facility Location Model

- **Given:**
  - Set of potential locations (L), clients (S) with demand $\mathbf{d_s}$, and a “cost” (distance) from each client to each location $\mathbf{c_{s,l}}$.
  - A competitor with costs $\mathbf{c_{s,a}}$.
    Remark: w.l.o.g. we can assume that only one competitor exists.
- **Goal:**
  - choose where to locate **k** new facilities so as to **maximize the captured demand** (market share).

$$\max \sum_{l \in L} \sum_{s \in S} d_s p_{s,l}$$

$$p_{s,l} \leq x_l$$

$$p_{s,l} = 0 \quad \text{if } c_{s,l} > c_{s,a}$$

$$\sum_{l \in L} p_{s,l} \leq 1$$

$$\sum_{l \in L} x_l \leq k$$

$$x_l \in \{0, 1\},$$

$$0 \leq p_{s,l} \leq 1$$

x_l = 1 if location l is constructed
$p_{s,l}$= % of demand of s captured by l

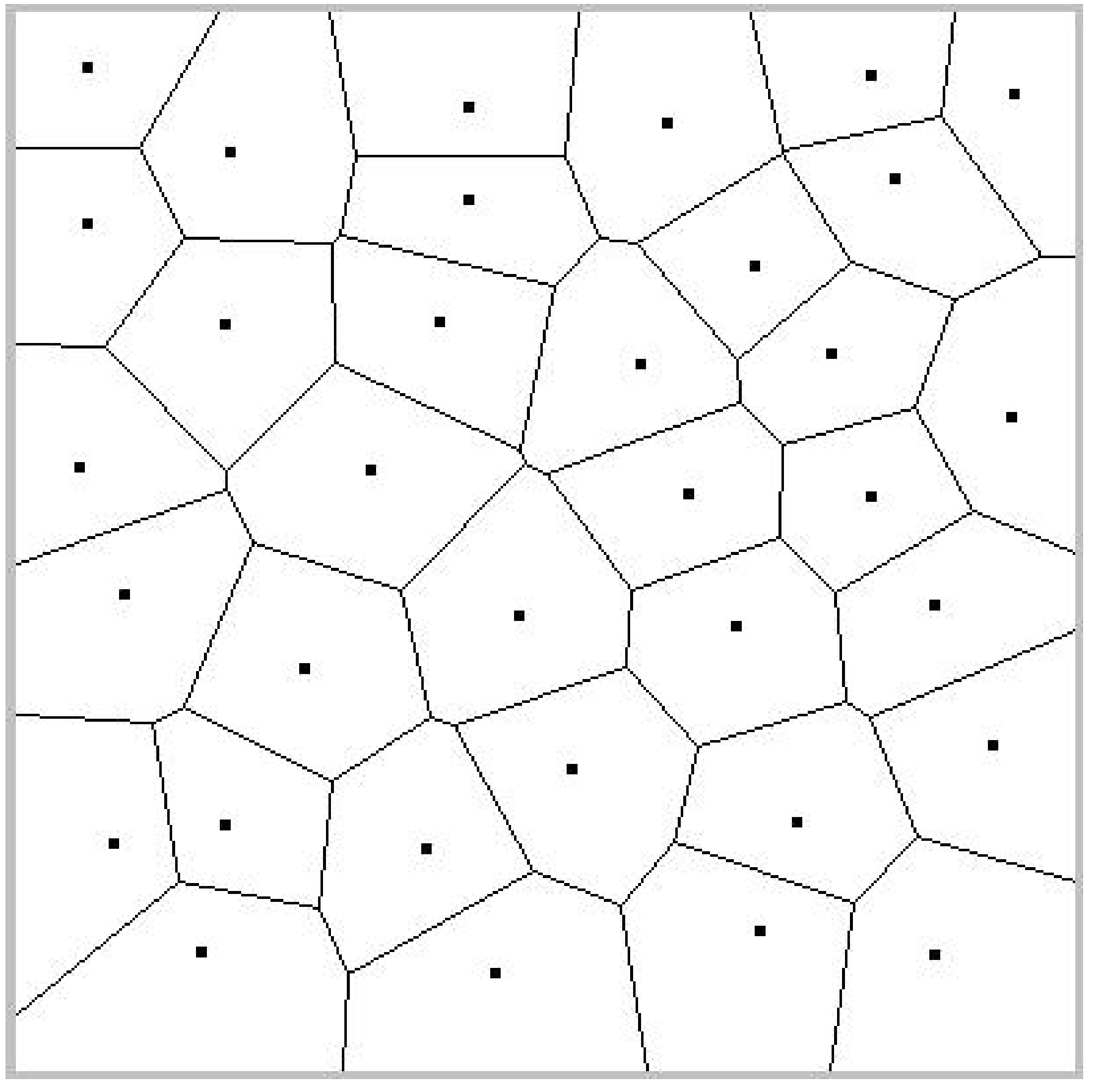
MAX-CAP model

- Result: “all-or-nothing” assignment to the closest facility (Voronoi diagram)

- Unrealistic! Customers do not always prefer the closest facility!

- How to integrate customer behaviour /preferences into an optimization model?

- One possibility: discrete choice models



Random Utility Model (e.g., McFadden, 1973)

- Each customer s has its own utility function $\tilde{u}_{s,l}$ for choosing location l. It will choose location l if

$$\tilde{u}_{s,l} \geq \tilde{u}_{s,h} \; \forall h \in L$$

- The utility function has a deterministic part (observable attributes) and a random term (non-observable attributes).

$$\tilde{u}_{s,l} = v_{s,l} + \epsilon_{s,l}$$

- Random distribution of $\{\epsilon_{s,1}, \ldots, \epsilon_{s,l}\}$ allows to compute the choice probabilities.

- If $\epsilon_{s,l}$ are iid and if they follow a Gumbel distribution, then the probability that a user s selects location l is given by

$$p_{s,l} = \frac{e^{\nu_{s,l}}}{\sum_{h \in L} e^{\nu_{s,h}}}$$

(Multinomial Logit)

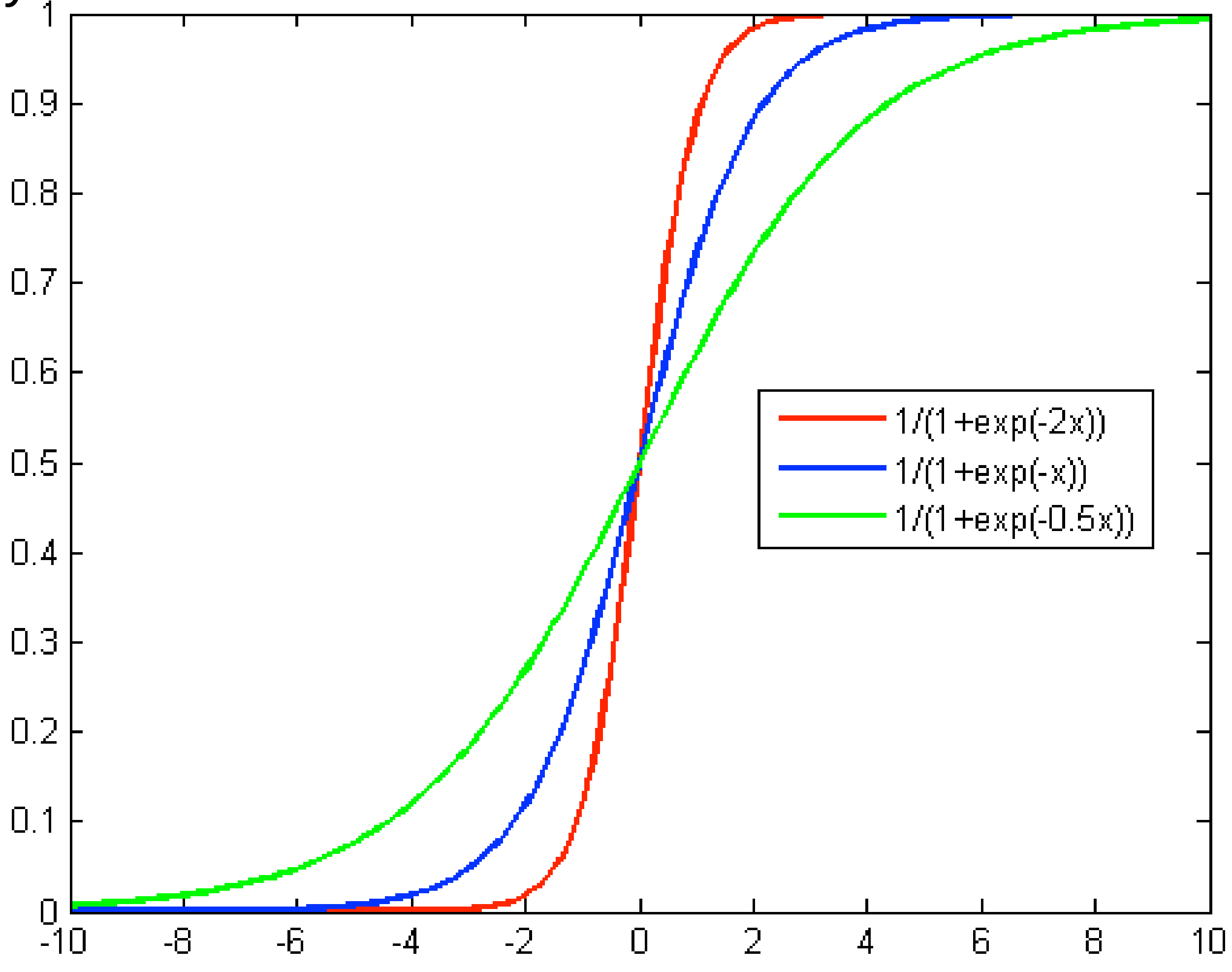
Discrete choice models

- Random utility model (McFadden, 1973). Facility location utility model (Drezner, 1994).

- Users have an “utility” function, and they split between options according to a logit function.

Cost using facility located in l

$$p_{s,l} = \frac{e^{-\theta c_{s,l}}}{\sum_{h \in L} e^{-\theta c_{s,h}}}$$



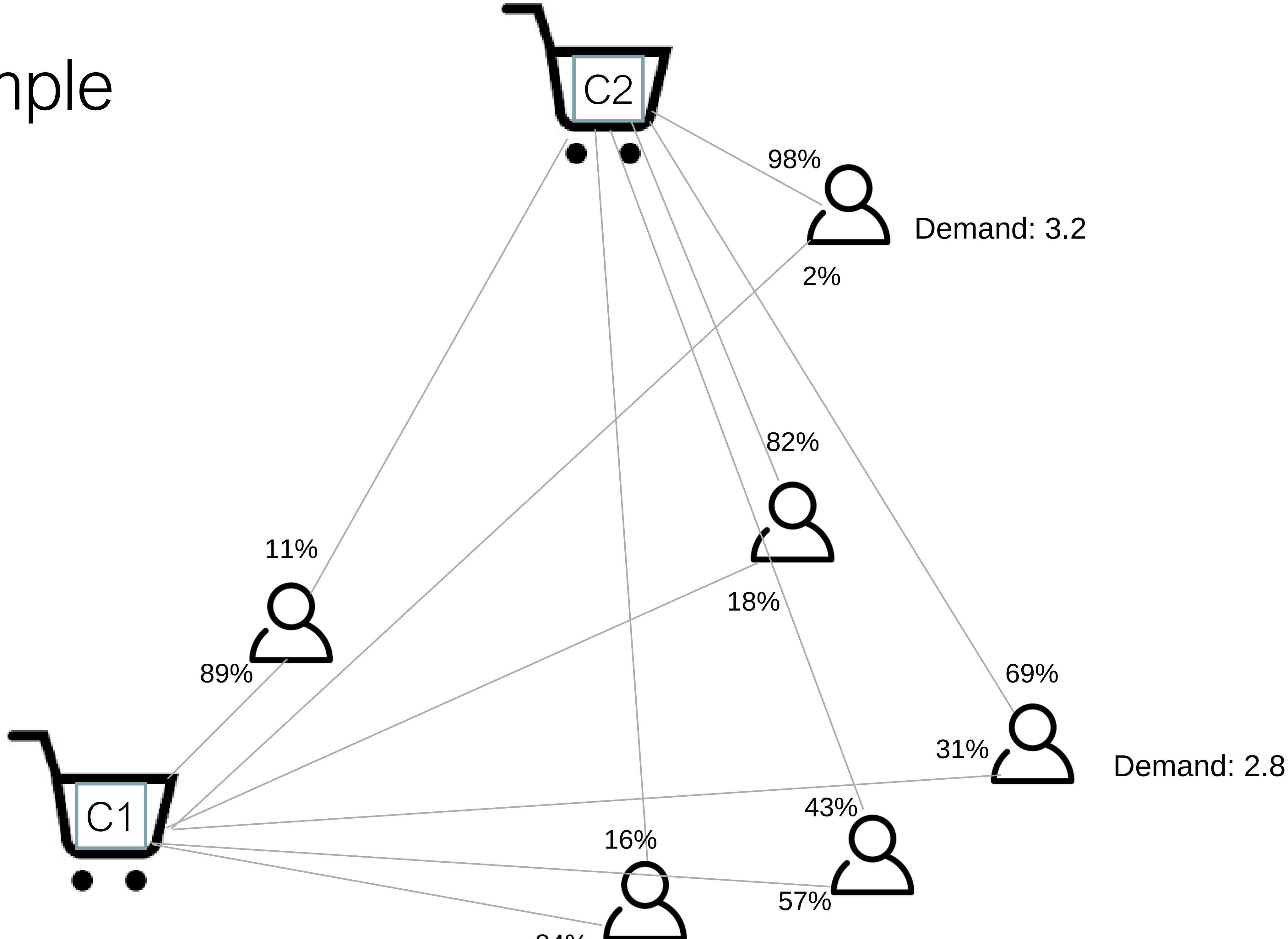
- θ represents the uncertainty of the users.

Example

$$p_1 = \frac{e^{-1.02}}{e^{-1.02} + e^{-3.12}} = 0.89$$

$$p_2 = \frac{e^{-3.12}}{e^{-1.02} + e^{-3.12}} = 0.11$$

Example



$$p_1 = \frac{e^{-1.02}}{e^{-1.02} + e^{-3.12}} = 0.89$$

$$p_2 = \frac{e^{-3.12}}{e^{-1.02} + e^{-3.12}} = 0.11$$

MAX-Capture Facility Location with Random Utilities

Which k facilites to open so as to max the capured demand?

Facility Location with Random Utilities

- Given a set of potential locations, to choose where to locate k new facilities to maximize the captured demand.
- Set of potential locations (L), customers (S) with demand d_s
- Generalized costs for customer s using facility l : $c_{s,l}$.
- A generalized costs for the competitor facility $c_{s,a}$.
  Remark: w.l.o.g. we can assume that only one competitor exists.

$$\max \sum_{l \in L} \sum_{s \in S} d_s p_{s,l}$$

$$p_{s,l} = \frac{x_l \cdot e^{-\theta c_{s,l}}}{e^{-\theta c_{s,a}} + \sum_{h \in L} x_h \cdot e^{-\theta c_{s,h}}}$$

$$\sum_{l \in L} x_l = k$$

$x_l \in \{0, 1\}$ (Facility located in site l)

$p_{s,l} \in [0, 1]$ (Probability of using facility l for customer s)

Max-Capture problem with random utilities

Facility Location with Random Utilities

$$\max \sum_{l \in L} \sum_{s \in S} d_s p_{s,l}$$

$$p_{s,l} = \frac{x_l \cdot e^{-\theta c_{s,l}}}{e^{-\theta c_{s,a}} + \sum_{h \in L} x_h \cdot e^{-\theta c_{s,h}}}$$

$$\sum_{l \in L} x_l = k$$

$$x_l \in \{0, 1\}$$ (Facility located in site l)

$$p_{s,l} \in [0, 1]$$ (Probability of using facility l for customer s)

Max-Capture problem with random utilities

Solving the problem

Method 1: Non-linear model (Benati & Hansen’ 02)

$$\max \sum_{s \in S} d_s \cdot \underbrace{\frac{\sum_{l \in L} x_l \cdot e^{-\theta c_{s,l}}}{e^{-\theta c_{s,a}} + \sum_{h \in L} x_l \cdot e^{-\theta c_{s,h}}}}_{w_s(x)}$$

$$\sum_{l \in L} x_l = k$$

$$x_l \in \{0, 1\}$$

- $w_s(x)$ is a concave function (Benati & Hansen 2002)
  Proof: Composition of concave non-decreasing function f(y)=y/(1+y) with a linear function.

- Can be solved using a branch-and-bound algorithm.

Method 2: MIP reformulation (Haase '09)

$$p_{s,l} = \frac{x_l \cdot e^{-\theta c_{s,l}}}{e^{-\theta c_{s,a}} + \sum_{h \in L} x_l \cdot e^{-\theta c_{s,h}}} \quad \Leftrightarrow \quad \begin{aligned} & \boxed{p_{s,l} \leq x_l} \\ & p_{s,a} \geq \frac{e^{-\theta c_{s,a}}}{e^{-\theta c_{s,l}}} \cdot p_{s,l} \\ & p_{s,a} + \sum_{l \in L} p_{s,l} = 1 \end{aligned}$$

- Aros-Vera et al (2013): $p_{s,l} \leq x_l$
- Haase (2009): $p_{s,l} \leq \frac{e^{-\theta c_{s,l}}}{e^{-\theta c_{s,a}} + e^{-\theta c_{s,l}}} x_l$
- Freire et al (2016): $p_{s,l} \leq \frac{e^{-\theta c_{s,l}}}{e^{-\theta c_{s,a}} + \sum_{h \in L(s,l)} e^{-\theta c_{s,h}}} x_l$

Proposed: Branch-and-cut (Outer Approximation)

- Idea: to approximate the concave function by its first-order approximation in a given point x*, but in a cutting-plane approach.

$$\max \sum_{s\in S} d_s \cdot z_s$$

$$w_s(x)$$

$$z_s \le w_s(x^*) + \sum_{l\in L} \left(\frac{\partial w_s}{\partial x_l}(x^*) \right) \cdot (x - x^*) \quad \forall s \in S, \forall x^* \in X$$

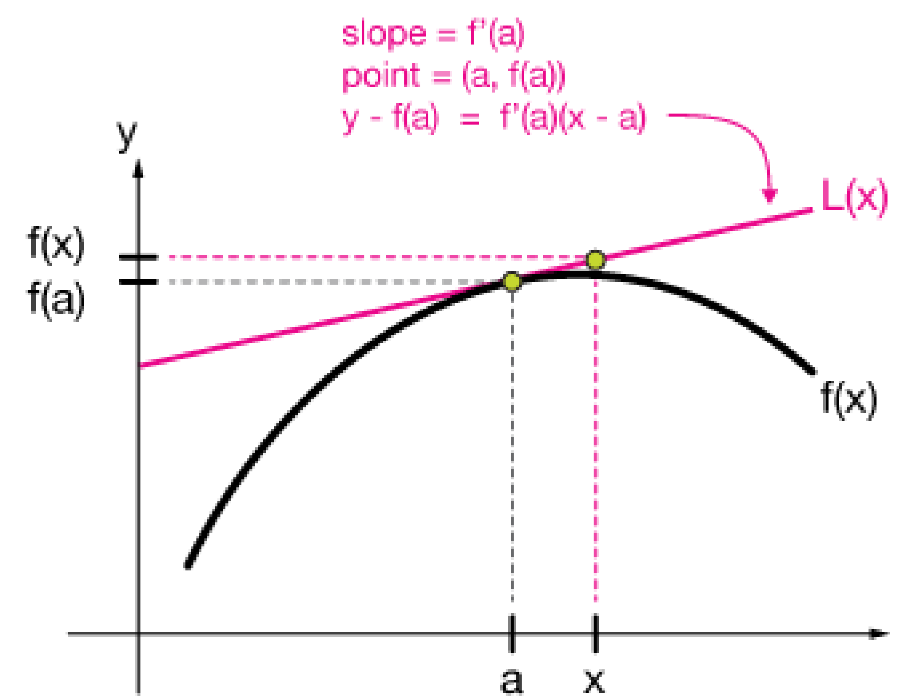
$$\frac{e^{-\theta c_{sl}}}{(1+\sum_{h\in L} e^{-\theta x_h^* c_{sh}})^2}$$

$$\sum_{l\in L} x_l = k$$

$$x_l \in \{0,1\}$$

$$z_s \in [0,1]$$

- Original idea from Quesada & Grossman (1992)



Proposed: Branch-and-cut (Outer Approximation)

$$\max \sum_{s \in S} d_s \cdot z_s$$

$$z_s \leq w_s^2(x^*) + \sum_{l \in L} x_l \cdot \frac{e^{-\theta c_{sl}}}{(1 + \sum_{h \in L} e^{-\theta x_h^* c_{sh}})^2} \quad \forall s \in S, \forall x^* \in X$$

$$\sum_{l \in L} x_l = k$$

$$x_l \in \{0, 1\}$$

$$z_s \in [0, 1]$$

- Implementation details: In the branch & bound tree, if a solution x* is integer, we check if this constraint is violated for some s, and we add the cut to the problem (lazy-cut callback in CPLEX/GUROBI)

- It can also be applied to a fractional solution (user-cut callback)

Proposed: Branch-and-cut (Submodular cuts)

- Submodular function: marginal gain of adding a new location decreases with the size of the already included locations.

$$f(X \cup \{x\}) - f(X) \geq f(Y \cup \{x\}) - f(Y) \quad \text{for } X \subseteq Y$$

- The fraction of demand of a client s captured by a set of locations given by x is a non-decreasing submodular function (Benati, 1997)

$$w_s(x) := \frac{\sum_{l \in L} x_l \cdot e^{-\theta c_{s,l}}}{e^{-\theta c_{s,a}} + \sum_{h \in L} x_l \cdot e^{-\theta c_{s,h}}}$$

- Nemhauser and Wolsey (1981) provide a MIP valid cut for maximizing non-decreasing sub modular functions.

Proposed: Branch-and-cut (Submodular cuts)

$$\max \sum_{s \in S} d_s \cdot z_s$$

$$z_s \leq w_s(K) + \sum_{l \in L \setminus K} \frac{e^{-\theta(c_{s,l} - c_{s,a})}}{(1 + Z_K^s)(1 + Z_{K+l}^s)} x_l$$

$$- \frac{1}{1 + Z_L^s} \sum_{l \in K} \frac{e^{-\theta(c_{s,l} - c_{s,a})}}{1 + Z_{L-l}^s}(1 - x_l) \qquad \forall K \subseteq K_X$$

$$\sum_{l \in L} x_l = k$$

$$x_l \in \{0, 1\}$$

$$z_s \in [0, 1]$$

$$K = \{l \in L : x_l^* = 1\}$$

$$Z_K^s = \frac{\sum_{l \in K} e^{-\theta c_{s,l}}}{e^{-\theta c_{s,a}}}$$

- In general, is NP-hard to separate violated cut, but it can be proven that we only need to separate these cuts at integer solutions x* of the branch-and bound, which can be done efficiently.

Some important properties:

1. Outer-approximation cuts and Submodular cuts do not dominate each other. We can apply both cuts simultaneously.

2. All results previous results for these cuts also applied to more general sets of constraints imposed on the possible locations (e.g., tree, tour, 2-connectivity…).

$$X = \{x \in \{0,1\}^{|L|} : Ax + By \leq b, y \in Y\}$$

3. The model is very sparse (only linear instead of quadratic number of variables)

Computational results

Implementation and Dataset

- MIP formulation and cutting planes solved using CPLEX 12.6 under default settings. Nonlinear relaxation solved using method-of-moving-asymptotes (MMA) implemented in NLopt v2.4.

- **Dataset HM14:** Haase & Müller (2014). Clients and candidate locations uniformly distributed in a rectangular region with unit demand. Client cost are distances to each facility. **50 to 400 customers, 25 to 100 locations**.
- **Dataset ORlib:** Hoefer (2003). Classic facility location problems where a competitor is created selecting a subset of locations and fixing the cost to the minimum among them with up to **1000 clients and 100 locations**.
- P&R NYC Dataset (Aros-Vera, 2013): **82341 clients and 59 locations (almost 5 Mio of p_{sl} variables!)**

- Utilities: $v_{sl} = -\theta \cdot c_{sl}$ $v_{sa} = -\theta \cdot \alpha \cdot c_{sl}$
  θ: Uncertainty of customers, α: Competitiveness of incumbent location

- 81 configurations (3 values of θ, 3 values of α, and 9 values of k)

Computational Results

A smaller and faster subproblems

Good approximation of the integer polytope

Table 1: Results for ORLIB (up) and HM14 (down) datasets, grouped by problem name (81 instances per row). Time limit set to one hour.

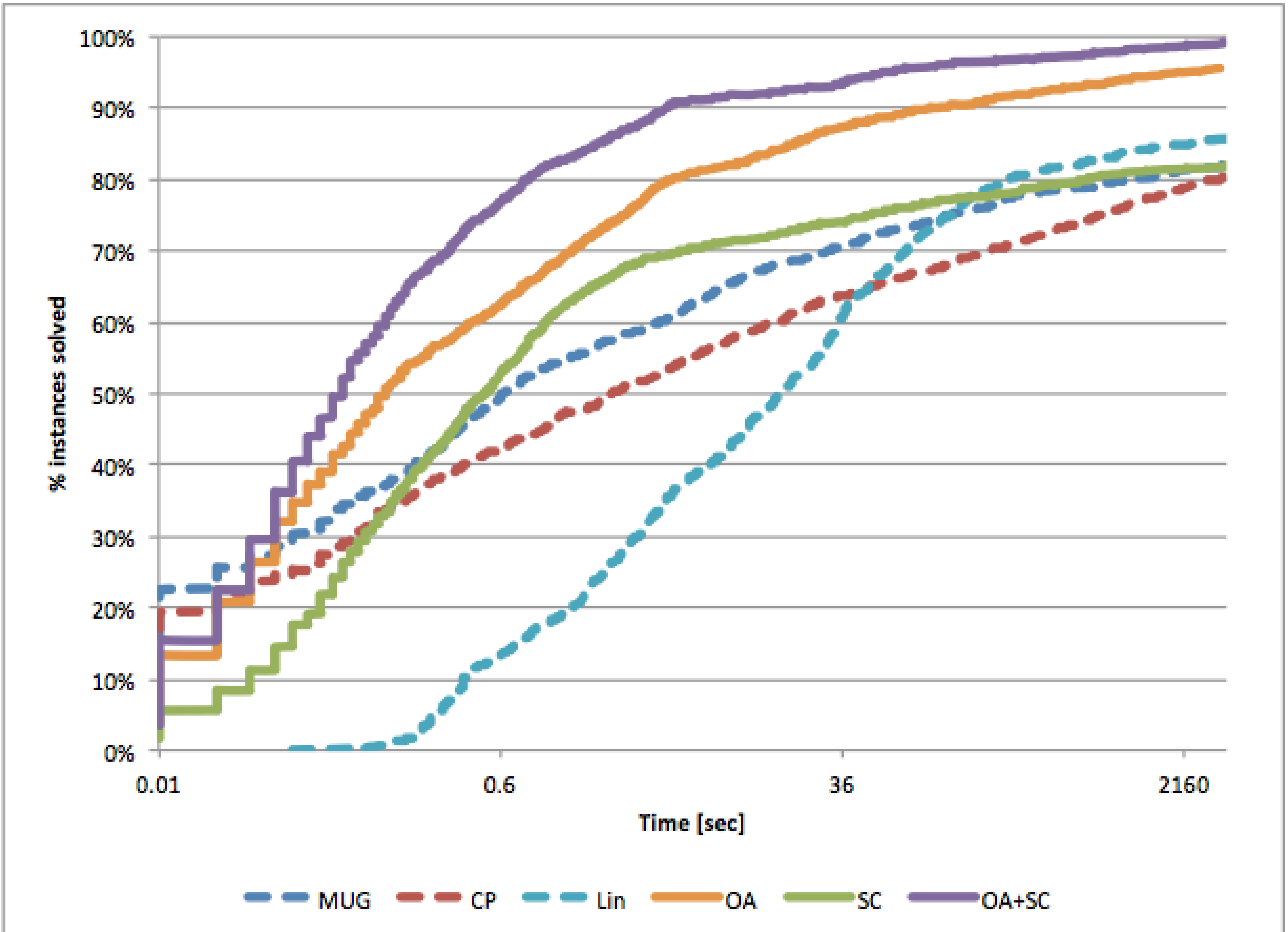
	#(Solved Instances)						Computing Time [s]*						B&B Nodes*						Root gap*					
Name	Lin	CP	MUG	OA	SC	OA+SC	Lin	CP	MUG	OA	SC	OA+SC	Lin	CP	MUG	OA	SC	OA+SC	Lin	CP	MUG	OA	SC	OA+SC
cap101	81	81	81	81	75	81	13.8	0.4	0.2	0.0	100.9	0.0	4111	34	9057	6	1279	2	10.2	0.3	8.4	0.5	5.1	0.1
cap102	81	81	81	81	75	81	14.1	0.9	0.2	0.0	116.3	0.0	4840	170	11596	6	1460	2	10.3	0.4	8.6	0.7	5.1	0.1
cap103	81	81	81	81	81	81	6.6	0.7	0.1	0.0	199.7	0.0	1387	86	7559	4	1845	1	10.3	0.5	8.9	0.7	5.0	0.1
cap104	81	81	81	81	78	81	8.3	0.1	0.2	0.0	151.2	0.0	1862	7	10026	4	1495	1	10.2	0.2	8.5	0.5	5.1	0.1
cap131	78	81	81	81	61	81	253.1	1.6	7.5	0.1	94.6	0.1	9303	59	296281	7	997	2	11.9	0.5	10.5	0.9	6.5	0.2
cap132	79	81	81	81	62	81	213.2	0.5	5.9	0.1	145.0	0.1	7362	15	225039	4	855	2	12.0	0.5	11.0	0.8	6.4	0.1
cap133	78	81	81	81	62	81	199.6	0.3	14.2	0.1	219.2	0.1	4694	8	543304	2	1404	1	12.3	0.3	10.9	0.7	6.4	0.1
cap134	79	81	81	81	60	81	218.3	0.9	13.9	0.1	97.2	0.1	9494	36	525487	3	982	2	12.2	0.5	11.1	0.7	6.3	0.1
capa	–	48	21	81	–	74	–	737.5	356.3	298.4	–	229.9	–	112	308778	2016	–	888	–	0.2	31.0	1.4	–	0.9
capb	–	49	23	81	1	78	–	665.6	471.4	120.8	3039.9	193.4	–	110	393240	1143	1245	760	–	0.1	30.2	1.3	16.3	0.9
capc	–	53	21	80	–	75	–	477.0	296.5	271.8	–	413.3	–	61	225427	1812	–	1051	–	0.1	30.1	1.4	–	1.0

		#(Solved Instances)						Computing Time [s]*						B&B Nodes*						Root gap*					
$\|S\|$	$\|L\|$	Lin	CP	MUG	OA	SC	OA+SC	Lin	CP	MUG	OA	SC	OA+SC	Lin	CP	MUG	OA	SC	OA+SC	Lin	CP	MUG	OA	SC	OA+SC
50	25	81	69	81	81	81	81	28.1	13.8	0.2	0.4	0.0	0.0	1	451	11070	1761	1	0	0.6	9.4	19.3	12.2	0.2	0.1
50	50	81	67	79	81	81	81	26.6	211.1	106.3	0.7	0.1	0.1	7	5375	4141023	3219	3	1	0.8	9.1	22.4	12.4	0.2	0.1
50	100	81	48	61	70	81	81	270.3	272.5	167.1	94.1	0.1	0.1	33	461	4480988	262864	8	7	0.5	5.0	27.7	15.0	0.5	0.3
100	25	81	67	81	81	81	81	19.8	55.3	1.7	3.8	0.0	0.0	0	2573	40582	12740	1	0	0.5	8.4	22.8	13.7	0.6	0.5
100	50	81	58	72	80	81	81	22.9	162.6	207.9	127.4	0.1	0.1	5	1696	4778935	208127	1	1	0.3	8.6	32.9	17.0	0.3	0.2
100	100	81	49	58	68	81	81	52.7	289.1	200.3	60.4	0.7	0.5	70	368	2959530	86653	70	18	1.1	5.3	28.6	14.0	0.7	0.5
200	25	81	74	81	81	81	81	14.3	142.7	9.4	1.4	0.1	0.1	2	2922	110327	3175	1	0	0.4	11.4	27.9	13.6	0.2	0.1
200	50	81	57	67	73	81	81	39.6	254.7	211.1	57.8	0.2	0.2	2	1316	2400039	86289	2	2	0.4	10.2	33.1	16.5	0.5	0.4
200	100	81	46	46	63	81	81	653.2	404.5	112.7	74.4	2.0	1.2	54	228	686808	35141	56	26	0.9	5.4	32.2	17.7	0.5	0.3
400	25	81	77	81	81	81	81	34.3	133.0	11.7	2.8	0.1	0.2	1	1367	49637	4808	2	1	0.4	10.9	29.3	13.4	0.2	0.2
400	50	81	52	62	72	81	81	234.4	388.3	259.9	116.9	0.5	0.6	11	970	1044952	72659	3	2	0.5	9.7	35.1	17.7	0.3	0.4
400	100	76	36	45	60	81	81	552.2	355.7	299.2	34.4	4.0	2.5	14	172	758270	6168	62	21	0.7	5.7	32.7	15.4	0.6	0.5

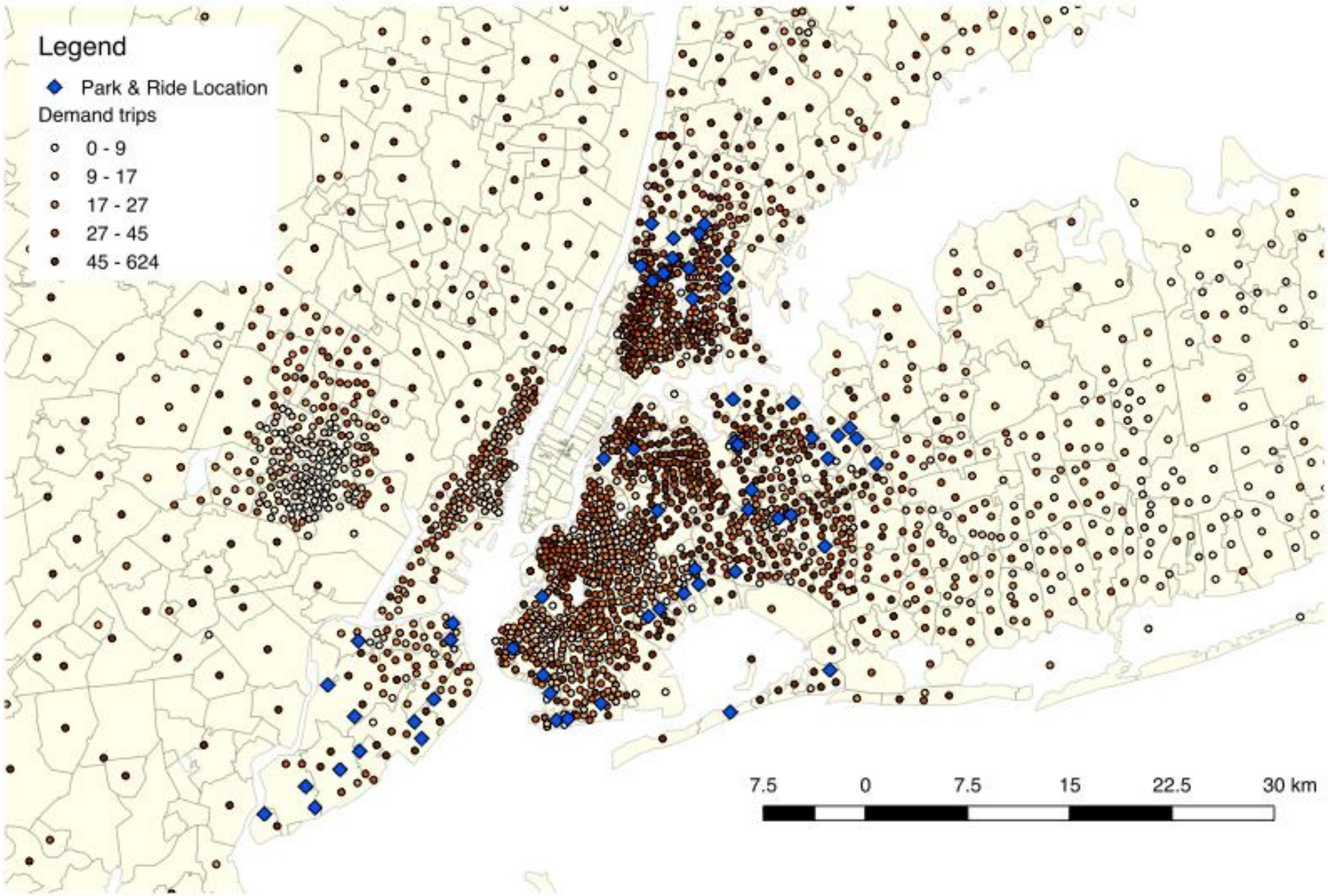
B&C solves more instances

(*) Average values between solved instances

Computational Results



Large-scale Instance : P&R locations in NY

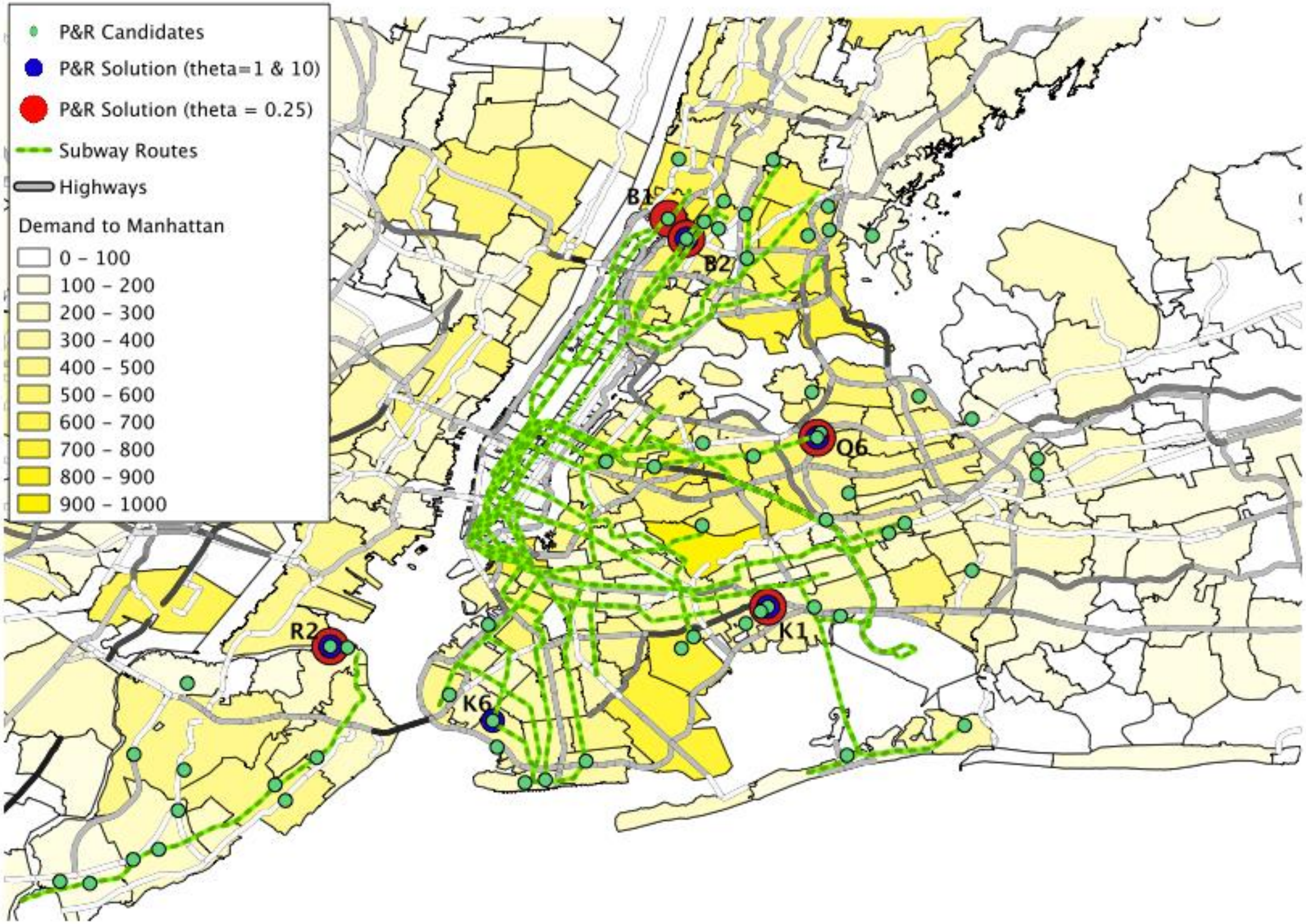


82341 “clients”, **59** locations

Computational Results (P&R NYC instances)

	Inst. Solved					Time [s] (*)				
	CP	MUG	OA	SC	OA+SC	CP	MUG	OA	SC	OA+SC
2	6	9	9	9	9	3727	69	1363	456	971
3	6	9	9	9	9	2485	170	2177	514	573
4	5	9	9	9	9	2338	411	2950	603	674
5	5	9	9	9	9	1813	1303	783	504	570
6	7	9	9	9	9	4707	3187	464	430	596
7	6	9	9	9	9	1169	6562	418	422	510
8	6	9	9	9	9	2441	10157	391	603	538
9	6	6	9	9	9	4025	2995	397	429	512
10	5	6	9	9	9	1469	3843	414	412	503

(*) Among solved instances within time-limit of 4 hrs.



Results | Transit Network

Conclusions

- A Branch-and-cut method that exploits the structure of the captured demand function (concave, submodular, non-decreasing)

- Very robust, suitable for more general facility location problems
  - Cardinality or budget constraints
  - Simultaneous facility location and design decisions
  - Infrastructure requirements (e.g., connectivity between facilities)
  - Other (convex, non-decreasing and submodular) utility functions

- Further improvements can be obtained by strengthening the submodular cuts (Yu & Ahmed, 2017)

- Remains to be exploited for other discrete choice models with similar properties

Thanks