Practical results of the p-median problem for hospital localization in Sweden

Pascal Rebreyend *Högskolan Dalarna*
and
Laurent Lemarchand small *lab-STICC, Brest, France*
Reinhardt Euler *lab-STICC, Brest, France*

November, 10th, 2015

p-median: Practical definition

- *Applications*: **localization** (Hospitals, public services,...)
- Given a set of D demand points
- Given a set of N candidate nodes
- Choose p candidates nodes in order to minimize the distance from each demand point to the closest selected candidate node
- *In our context*: points, nodes are located on a **road** network. The distance is the distance via this network.

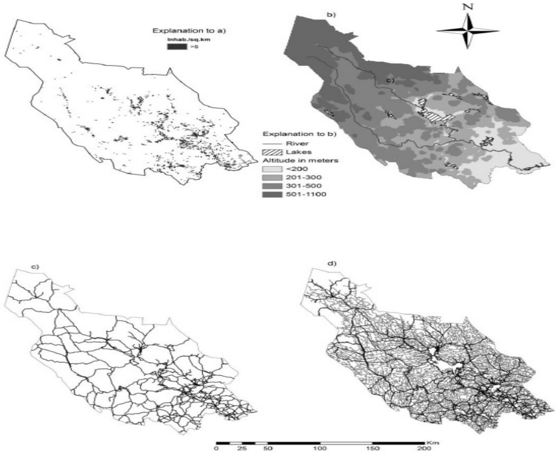
Rules of the game

- *p*-median is NP-hard (*Kariv-Hakimi*).
- *Hakimi*: We need to consider only nodes of the graph as candidate points
- Different methods have been proposed.
- **Goal**: to compare algorithms on different graphs and sizes.
- **Focus**: (Very) large graphs due to **real** problems.
- In short, we want to test the different approaches in practice.

Benchmarks

- Beasley (1990)
  - Synthetic graphs with up to 900 candidates nodes
  - The commonly used benchmark for the p-median problem
  - We compute from the graph the matrix of distances
  - a node is both a candidate node and a demand point (weight=1)
- Real data (Sweden - 2013)
  - Distances computed from Trafiksverket's and SCB data
  - Up to 188,325 demands weighted points (5,411,373 persons)
  - Up to 1938 candidates nodes (All swedish settlements)

Dalarna


Explanation to a)
Inhab./sq.km
>5
b)
N
c)
Explanation to b)
River
Lakes
Altitude in meters
<200
201-300
301-500
501-1100
c)
d)
Km
0
25
50
100
150
200

NVDB: The swedish road database

- Provided by the Swedish Road Administration
- A list of road segments (x,y,z) with extra information (Speed, direction)
- Problem: No description of crossings (Not the purpose of the database): Thus, precomputation are needed.
- We also find islands for particular pre-processing.

Definition 2: Variables

h_i : weight of the demand of the customer i
d_{ij} : distance between customer i and facility j
X_i : decision variable, indicating the facility i is selected or not
Y_{ij} : decision variable, indicating the customer i is served by facility j or not
p : the number of facilities we want to select.

Definition 2: Equations

$$Minimize \sum_{i=1}^{D} \sum_{j=1}^{N} h_i d_{ij} Y_{ij} \tag{1}$$

subject to

$$\sum_{j=1}^{N} Y_{ij} = 1, \forall i, 1 \leq i \leq D \tag{2}$$

$$\sum_{j=1}^{N} X_j = p \tag{3}$$

$$Y_{ij} - X_j \leq 0, \forall i, j, 1 \leq i \leq D, 1 \leq j \leq N \tag{4}$$

$$X_j \in \{0, 1\}, \forall j, 1 \leq j \leq N \tag{5}$$

$$Y_{ij} \in \{0, 1\}, \forall i, j, 1 \leq i \leq D, 1 \leq j \leq N \tag{6}$$

Cplex

- Definition 2: a mixed integer programming problem (MIP)
- ILOG-CPLEX software is used. Can work with mixed integer problems.
- Algorithm used for them: Branch and Cut
- CPLEX: A powerfull tool to deal with integer and mixed integer programming problems. (*easy to interface with C and other languages*)

Cplex : comments

- Branch and Cut and CPLEX: Used a lot in OR community
- CPLEX: Simple to use
- but rather difficult to tune some parameters (*do we need to ?*)
- Caution: *Branch and Cut* and *Branch and Bound* are very different in their spirit

Volume algorithm

- Describes by Barahona and Anbil (2000)
- Used on the p-median by Gay (2011)
- *Idea*: work on the dual problem with the sub-gradient method
- Every iteration: primal variables are computed too
- Clervaux-Bourges Heuristic $0(m\ln(m))$ to find feasible solution from reals values (by sorting edges)
- Thus, **both lower and upper** bounds
- *Details???*
- Thanks to C. Clervaux for the C Code

The combinatorial optimization approach

- We come back to definition 1 (Real problem oriented)
- NP-hard means "*impossible to solve if the problem is big enough*" !
- For us, we **need** to work *at least* with the biggest graph.
- We are searching for approximate solutions.
- Ideas: heuristics and meta-heuristics.

Simulated Annealing (SA)

- Evolutionary method
- Neighbourhood: move one selected point to another candidate node
- In our case: adaptive scheme to decrease the temperature and re-heat when stuck in local minimas
- Stopping criteria: A number of iterations
- Easy to write and run

GA

- We have a population of solutions.
- Here: version proposed by Correa.
- Classical representation: A chromosom is a string of p ID representing the selected nodes.
- **Crossover**: Exchange of some ID between the 2 chromosoms.
- **Mutation**: Between 1 and $1/p$ ID are changed.
- In short, a basic pure GA (thus, not very efficient).
- **plus** ...

Hypermutation - Correa

- Local search method implemented via a specific mutation (**hybrid** approach)
- Every single selected candidate node is replaced by each of all non-selected node
- complexity: $O(n * p)$

Improved-GA (Hybrid)

- Based on Correa but a different local search.
- No hypermutation but a different mutation with a local search approach.
- **Mutation**: We pick at random 5 selected facilities and we try to replace them by each unselected facility.

Experiments

- 32Gb memory available
- Max 2 days of computations
- Stopping criterias: simple, based on previous experiments.

Results

Graph	n	p	Cplex	Volume	SA	GA	Hybrid GA
Sweden	1938	10	MEM	62163	58590	TIME	**57332**
Sweden	1938	50	MEM	**20225**	21661	TIME	20788
Sweden	1938	100	MEM	**13095**	14291	TIME	13689
DalGävle	195	20	10323	**10302**	10502	10323	10323
pmed38	900	5	**11060**	11263	11072	**11060**	**11060**
pmed40	900	90	**5128**	5300	5354	5144	5138
Dev in %			0.07	4.57	3.53	0.2	1.45

Running time

Sequential running time in seconds:

Graph	p	Cplex	Volume	SA	GA	Improved GA
Sweden	10	MEM	16978	8771	TIME	3018
Sweden	50	MEM	10380	25636	TIME	8304
Sweden	100	MEM	8032	41297	TIME	14618
DalGävle	20	4440	56.8	1274	86.2	100
pmed38	5	30000	9.86	26	13.9	15.4
pmed40	90	189	8.8	165	763	789

- **Two main approaches**: exact-abstract and approximate-data based
- **First question**: which problem you want to address? *size, quality of the solution,lower bound,...*
- **Exact algorithms**: work fine until some point
- **Approximate methods**: more stable in quality, able to give "good" approximate solution for very large problems.
- Huge variations of running time and its variation depending on the graph
- Practical results: Administrative borders (counties) are not leading to worst solutions for hospitals localization
- Current Ikea location are rather good.

In short

	Cplex	Volume	SA	GA	Hybrid GA
Small graphs	+	+	-	-	-
Large graphs	-	-	+	-	+
Flexibility	-	-	+	+	+
Coding	Soft	-	++	+	+
Lower bounds	Yes	Yes	No	No	No
$p \nearrow$: running time	$\searrow$	$\searrow$	$\nearrow$	$\nearrow$	$\nearrow$
Distance matrix	Precomp.	Precomp.	Lazy	Lazy	Lazy

Future

- Heuristics ?
- Branch and bound ?
- More candidate nodes ?
- Other objective functions? (P-gravity ?, taken into account distributions of distances ?)
- Do we need to precompute the full distance matrix?
- Multi-objective criteria.