# Some heuristic methods for the $p$-median problem with maximum distance constraints

VIII International Workshop on Locational Analysis and Related Problems 2017, Segovia

Adrián Esteban Pérez and Jesús Sáez Aguado

Dpto. Estadistica e Investigacion Operativa, Universidad de Valladolid

# The $p$-median problem with maximum distance constraints (PMPDC)

# The $p$-median problem with maximum distance constraints (PMPDC)

- It is a variant of the classic $p$-median problem (PMP).

# The $p$-median problem with maximum distance constraints (PMPDC)

- It is a variant of the classic $p$-median problem (PMP).
- It is a NP-hard problem.

# The $p$-median problem with maximum distance constraints (PMPDC)

- It is a variant of the classic $p$-median problem (PMP).
- It is a NP-hard problem.
- The last work is of the '90s. In addition, there are several approaches: the *indirect approach*, introducing a sufficiently large value $M$ in the distance matrix if the distance (or time) between a demand point and a facility site exceeds a distance or time), and the *direct approach*, introducing a maximum distance (or time) constraints.

# The $p$-median problem with maximum distance constraints (PMPDC)

- It is a variant of the classic $p$-median problem (PMP).
- It is a NP-hard problem.
- The last work is of the '90s. In addition, there are several approaches: the *indirect approach*, introducing a sufficiently large value $M$ in the distance matrix if the distance (or time) between a demand point and a facility site exceeds a distance or time), and the *direct approach*, introducing a maximum distance (or time) constraints.
- So we can transform (PMPDC) in a (PMP) with distance matrix modified. In a first look, (PMPDC) can be seen like a (PMP) but this formulation has a big problem: the heuristic methods for (PMP) frequently provide infeasible solutions and the quality of solutions depends of the value of big-M.

- Let $M = \{1, 2, \ldots, m\}$ be a set of demand points of a certain service, with demands $h_i$, $i = 1, \ldots, m$, and a set $N = \{1, 2, \ldots, n\}$ of points where it is possible to locate or open a facility (service points). We will assume that the distance matrix of each demand point $i$ at each point of facility $j$, $(d_{ij})$ is known. Moreover, for every $i \in M$, let $s_i$ be the maximum distance limit between the demand point $i$ and any facility.

- Let $M = \{1, 2, \ldots, m\}$ be a set of demand points of a certain service, with demands $h_i$, $i = 1, \ldots, m$, and a set $N = \{1, 2, \ldots, n\}$ of points where it is possible to locate or open a facility (service points). We will assume that the distance matrix of each demand point $i$ at each point of facility $j$, $(d_{ij})$ is known. Moreover, for every $i \in M$, let $s_i$ be the maximum distance limit between the demand point $i$ and any facility.
- Given a total number of $p$ facilities that are to be opened, the problem of the $p$-median with maximum distance constraints consists in deciding at which points the facilities should be opened and assigning to each demand point an open facility that complies with maximum distance constraints, so that the total distance (averaged with the population) covered to serve the total demand is minimal.

# Dense formulation

$$(\textbf{PMPDC1}) \text{ Minimize} \sum_{i=1}^{m} \sum_{j=1}^{n} h_i d_{ij} y_{ij} \tag{1}$$

$$\text{s.t.} \sum_{j=1}^{n} y_{ij} = 1, \; i = 1, \ldots, m \tag{2}$$

$$\sum_{j=1}^{n} d_{ij} y_{ij} \leqslant s_i, \; i = 1, \ldots, m \tag{3}$$

$$y_{ij} \leqslant x_j, \; i = 1, \ldots, m, \; j = 1, \ldots, n \tag{4}$$

$$\sum_{j=1}^{n} x_j = p, \tag{5}$$

$$x_j \in \{0, 1\}, \; j = 1, \ldots, n \tag{6}$$

$$y_{ij} \in \{0, 1\}, \; i = 1, \ldots, m, \; j = 1, \ldots, n \tag{7}$$

In addition, to the notation discussed above, we add the following, defining $N_i$ as the set of facility sites within $s_i$ distance units from the demand point $i \in M$.

$$\textbf{(PMPDC2)} \text{ Minimize} \sum_{i=1}^{m} \sum_{j \in N_i} h_i d_{ij} y_{ij} \tag{8}$$

$$\text{s.t.} \sum_{j \in N_i} y_{ij} = 1, \; i = 1, \ldots, m \tag{9}$$

$$y_{ij} \leqslant x_j, \; i = 1, \ldots, m, \; j \in N_i \tag{10}$$

$$\sum_{j=1}^{n} x_j = p, \tag{11}$$

$$x_j \in \{0, 1\}, \; j = 1, \ldots, n \tag{12}$$

$$y_{ij} \in \{0, 1\}, \; i = 1, \ldots, m, \; j = 1, \ldots, n \tag{13}$$

- It is important to note that, for some $(p, \mathbf{s})$, $\mathbf{s} = (s_1, \ldots, s_m)$, there may be no feasible solution. In addition, having fixed the value of $p$ , if the values of the maximum distances limits are increased, there exists a limit value $\bar{\mathbf{s}}$ in which the problem solution (**PMPDC2**) will not differ from the solution of the $p$-median classic problem (**PMP**). In practice, it is usual to consider that $s_i = s$ for all $i = 1, \ldots, m$. This is why this hypothesis will be assumed later.

- It is important to note that, for some $(p, \mathbf{s})$, $\mathbf{s} = (s_1, \ldots, s_m)$, there may be no feasible solution. In addition, having fixed the value of $p$ , if the values of the maximum distances limits are increased, there exists a limit value $\bar{\mathbf{s}}$ in which the problem solution (**PMPDC2**) will not differ from the solution of the $p$-median classic problem (**PMP**). In practice, it is usual to consider that $s_i = s$ for all $i = 1, \ldots, m$. This is why this hypothesis will be assumed later.
- Fenchel cutting planes are very slow in comparison with other procedures like GRASP.

- It is important to note that, for some $(p, \mathbf{s})$, $\mathbf{s} = (s_1, \ldots, s_m)$, there may be no feasible solution. In addition, having fixed the value of $p$ , if the values of the maximum distances limits are increased, there exists a limit value $\bar{\mathbf{s}}$ in which the problem solution (**PMPDC2**) will not differ from the solution of the $p$-median classic problem (**PMP**). In practice, it is usual to consider that $s_i = s$ for all $i = 1, \ldots, m$. This is why this hypothesis will be assumed later.
- Fenchel cutting planes are very slow in comparison with other procedures like GRASP.
- The Lagrangian relaxation procedure we apply here is new, and differs from that in Choi and Chaudry (1993), since they fall into a knapsack subproblem and we fall into a trivial problem. The reason for this is that we have been able to adapt the Lagrangian relaxation procedure for the classic $p$-median problem (**PMP**) (Daskin (2013)) to the $p$-median problem with maximum distance constraints (**PMPDC2**).

- **Step 1** (Initialization): Initialize $UB := +\infty$ (upper bound), $LB := -\infty$ (lower bound), $k := 0$ and take initial values for the multipliers.

- **Step 1** (Initialization): Initialize $UB := +\infty$ (upper bound), $LB := -\infty$ (lower bound), $k := 0$ and take initial values for the multipliers.
- **Step 2** (Resolution of subproblems) : In the $k$-th iteration, having fixed $\lambda^k$, we compute

$$V_j = \sum_{i \in M / j \in N_i} \min\left\{0, h_i d_{ij} - \lambda_i\right\}, \;\; j \in N.$$

Find the $p$ lowest values of $V_j$ and let $x_j(\lambda^k) = 1$ in the corresponding facility sites and set $x_j(\lambda^k) = 0$ otherwise. Let $y_{ij} = 1$ if $x_j(\lambda^k) = 1$ and $h_i d_{ij} - \lambda_i < 0$ and $y_{ij} = 0$ otherwise. Calculate

$$V(\lambda^k) = \sum_{i=1}^{m} \sum_{j \in N_i} (h_i d_{ij} - \lambda_i)\, y_{ij} + \sum_{i \in I} \lambda_i^k \tag{14}$$

If $V(\lambda^k) > LB$ then $LB := V(\lambda^k)$.

**Step 3** (Obtaining feasible solutions: Lagrangian Heuristic): Having fixed the $p$ values of the variables $x_j(\lambda^k)$, for all $i = 1, \ldots, m$ find

$$\widehat{j_i} = \arg\min \left\{ d_{ij} \Big/ x_j = 1,\ j \in N \right\}$$

Let $y_{ij}(\lambda^k) = 1$ if $j = \widehat{j_i}$ and $y_{ij}(\lambda^k) = 0$ otherwise.Calculate $\overline{V}(\lambda^k) = \sum_{i=1}^{m} h_i d_{i\widehat{j_i}}$ If $\overline{V}(\lambda^k) < UB$, then $UB := \overline{V}(\lambda^k)$.

**Step 3** (Obtaining feasible solutions: Lagrangian Heuristic): Having fixed the $p$ values of the variables $x_j(\lambda^k)$, for all $i = 1, \ldots, m$ find

$$\widehat{j_i} = \arg\min \left\{ d_{ij} \Big/ x_j = 1,\ j \in N \right\}$$

Let $y_{ij}(\lambda^k) = 1$ if $j = \widehat{j_i}$ and $y_{ij}(\lambda^k) = 0$ otherwise.Calculate $\overline{V}(\lambda^k) = \sum_{i=1}^{m} h_i d_{i\widehat{j_i}}$ If $\overline{V}(\lambda^k) < UB$, then $UB := \overline{V}(\lambda^k)$.**Step 4** (Subgradient iteration): Calculate the subgradient $\xi_i$ using $1 - \sum_{j \in N_i} y_{ij},\ \ i \in M$ Calculate the step size

$$T_k := \frac{\theta(UB - LB)}{\sum_{i=1}^{m} \left(1 - \sum_{j \in N_i} y_{ij}\right)^2}$$

- Update the multipliers:

$$\lambda^{k+1} := \max\left\{0,\ \lambda^k + T_k \xi^k\right\}$$

Moreover, if more than some number of prefixed iterations of the sugradient algorithm are performed without an increment of $LB$, then halve the step length parameter by setting $\theta := \theta/2$ (see Held *et al.* (1974)). In our numerical experiments, we have established 18 iterations.

- Update the multipliers:

$$\lambda^{k+1} := \max\left\{0,\ \lambda^k + T_k \xi^k\right\}$$

Moreover, if more than some number of prefixed iterations of the sugradient algorithm are performed without an increment of $LB$, then halve the step length parameter by setting $\theta := \theta/2$ (see Held *et al.* (1974)). In our numerical experiments, we have established 18 iterations.

- **Step 5** (Stopping criterion): If the duality gap (GAP$=\frac{(UB-LB)}{UB}$) is less than a prefixed tolerance, or if the number of maximum iterations has been reached, or if the norm of the subgradient or $\theta$ are less than a prefixed tolerance, STOP and set $UB$ as the objective function value. Set $x_j(\lambda^k)$ and $y_{ij}(\lambda^k)$ as the final solution obtained. If not, $k := k+1$ and go to step 2.

# A heuristic method based on the GRASP methodology

- GRASP (*Greedy Randomized Adaptive Search Procedure*) is a metaheuristic that works quite well for the classic $p$-median problem. Its adaptation to our $p$-median problem with maximum distance constraints is not entirely immediate, since the maximum distance contraints in the GRASP construction phase have an influence and non-feasible solutions can be obtained. So it is necessary to implement two phases in the construction phase of GRASP.

- GRASP (*Greedy Randomized Adaptive Search Procedure*) is a metaheuristic that works quite well for the classic $p$-median problem. Its adaptation to our $p$-median problem with maximum distance constraints is not entirely immediate, since the maximum distance contraints in the GRASP construction phase have an influence and non-feasible solutions can be obtained. So it is necessary to implement two phases in the construction phase of GRASP.
- Note that the GRASP implemented here requires two parameters: KRCL (cardinal of the restricted candidate list) and `iter_max` (number of maximum iterations of the procedure) (see Resende and Ribeiro (2016) for other aspects of GRASP). In addition, we assume the rest of the parameters of the $p$-median problem with maximum distance constraints and we also assume that the maximum distance is $s_i = s$ for all $i = 1, \dots, m$.

- **Constructive phase:** We randomize (using a restricted candidate list of cardinal KRCL) the following greedy procedure decomposed into two clearly differentiated phases:

  Phase 1 : The greedy heuristic for the set covering is applied (see Murty (1995) or Chvatal (1979)), where the number of facilities is minimized, taking $dc = s$ as the coverage distance. At the end of this heuristic, we get a $k$ number of facilities. The following cases may occur:

  - If $k > p$, predictably the problem is not feasible. In our numerical experiments, whenever this case has been given, the problem was really not feasible.
  - If $k = p$, a feasible solution is already obtained.
  - If $k < p$, go to phase 2.

Phase 2 : Given $k < p$ facilities, we have $p$ complete facilities, with the original objective (1). More specifically, in any iteration of this phase 2, in order to complete up to $p$ facilities, we apply the methodology of the greedy procedure of the $p$-median problem. This procedure is repeated until $p$ facilities are completed, in which case STOP.

- **Local search:** We adapt to our problem the *Whitaker's fast heuristics* with the implementation of Resende and Werneck (2003).

- The first set of data consists of test problems whose parameters come from real data about a certain health service: aint11, aint12 and aint13 with $m = 134$, $n = 121$; $m = 266, n = 21$, and $m = 430, n = 22$ respectively, where $m$ is the number of demand points and $n$ is the number of service points. In addition, we will take $s_i = s$ for all $i = 1, \ldots, m$.
- The second set of data has been generated randomly, to obtain test problems of greater size than the previous ones. More specifically, they are data with $n = m$ of sizes $500, 800$ and 1000. We will denote these test problems as s500_1, s800_1 and s1000_1. In addition, the demands have been randomly generated following a uniform distribution between 10 and 100 and the distances have been obtained by rounding the Euclidean distances calculated from the uniformly generated coordinates between 0 and 100. In addition, we take $s_i = s$ for all $i = 1, \ldots, m$.

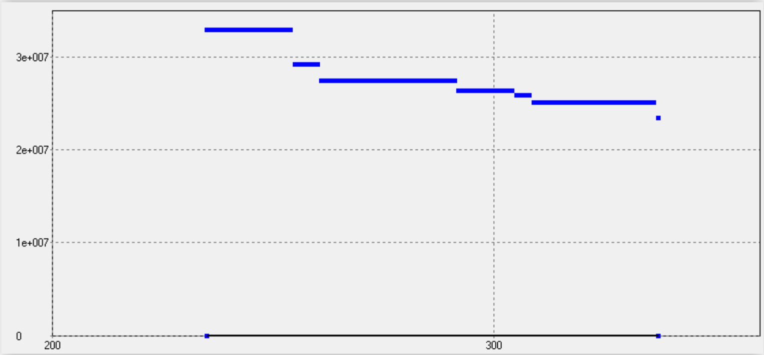


Figure: $z$ (objective value, total distance) versus maximum distance, corresponding to the data set aint12 with $p = 7$. The maximum distance feasibility interval is $[235, 337]$

# Computational results:

| | XPRESS | | | Lagrangian Relaxation | | | GRASP | | |
|---|---|---|---|---|---|---|---|---|---|
| | $z$ | $t$ | $d^*_{max}$ | $z^{LR}$ | $t^{LR}$ | $d^{LR}$ | $z^G$ | $t^G$ | $d^G$ |
| aint13 | | | | | | | | | |
| $(p,s)$ | | | | | | | | | |
| $(7,380)$ | 80.94 | 0.33 | 369 | 80.94 | 0.265 | 0 | 80.94 | 0.01244 | 0 |
| $(7,400)$ | 75.49 | 0.37 | 383 | 75.49 | 0.578 | 0 | 75.49 | 0.01253 | 0 |
| $(10,300)$ | 63.68 | 0.31 | 295 | 63.68 | 0.53 | 0 | 63.68 | 0.01656 | 0 |
| $(10,350)$ | 61.77 | 0.30 | 320 | 61.77 | 0.546 | 0 | 61.77 | 0.01408 | 0 |
| $(12,300)$ | 54.45 | 0.28 | 295 | 54.45 | 0.39 | 0 | 54.45 | 0.01467 | 0 |
| $(12,400)$ | 51.95 | 0.28 | 383 | 51.95 | 0.39 | 0 | 51.95 | 0.0154 | 0 |
| s1000_1 | | | | | | | | | |
| $(p,s)$ | | | | | | | | | |
| $(15,21)$ | 9.55 | 389.85 | 20 | 10.11 | 213.97 | 5.53 | 9.55 | 23.34 | 0 |
| $(15,50)$ | 9.55 | 544.34 | 22 | 9.67 | 234.98 | 1.24 | 9.55 | 27.08 | 0 |
| $(20,21)$ | 8.12 | 841.61 | 19 | 8.53 | 194.53 | 4.81 | 8.12 | 31.65 | 0 |
| $(20,50)$ | 8.12 | 1540.75 | 19 | 8.56 | 219.50 | 5.14 | 8.12 | 35.74 | 0 |

Thanks for the attention!!!