Anti-cent-dian on networks

Moreno Pérez, José A. and Rodríguez Martín, Inmaculada

Dpto. de Estadística, I.O. y Computación.

Universidad de La Laguna, 38271 La Laguna, Spain

e-mail: JAMORENO@ULL.ES

Abstract

This paper deals with the problem of locating an undesirable facility on a network using the anti-cent-dian criterion. The objective is the maximization of a convex combination of the average and the minimum distance to the population. The model is formulated and its main properties are studied. Two problems are considered: to find the optimal location given the weights of the convex combination, and to provide the set of optimal locations for all convex combinations. A finite dominant set of points is identified for both problems, and for the first a heuristic algorithm is proposed to reduce the number of computations. Some computational results on randomly generated networks are given.

Keywords: Location, Obnoxious, Networks.

1 Introduction

Location analysis deals with the location of facilities taking into account their effect on the population nearby. People want the (desirable) service facilities as close as possible, but with undesirable facilities the opposite is true.

Most classical criteria take into account the average and the worst effect on population. The objective of the median problem is to minimize the average distance from the facilities to the population and the objective of center problems is to minimize the distance to the farthest population elements. The cent-dian criterion, formulated by Halpern (1976), consists in the minimization of a convex combination of these two objectives. The anti-cent-dian problem arises by considering the corresponding criteria for undesirable facilities. The anti-cent-dian is the location for the facility that maximizes a convex combination of the average and the minimum distance to the population.

The population is assumed to be concentrated at a finite number of points on the network identified with a subset of the vertices, and the facilities can be established anywhere on the network, vertices or in the interior of the edges.

In this paper the anti-cent-dian model is formulated, its main properties studied and a finite dominant set of points identified. Two problems are considered: to find the optimal location given the weights of the convex combination, and to provide the set of optimal locations for all convex combinations.

2 Notation and formulation of the model

Let $N = (V, E)$ be an undirected connected network where $V = \{v_1, v_2, ..., v_n\}$ is the vertex set and E is a finite edge set, each edge $e = [v_i, v_j]$ having a positive length $l(e) = l(v_i, v_j)$. Let $\mid E \mid = m$.

Any edge might be considered to be an infinite linear set of points joining two vertices. A point x on an edge $e = [v_i, v_j]$ is determined by a value θ^e, where $0 \leq \theta^e \leq l(e)$, representing the length of the subedge between v_i and x which will be denoted by $[v_i, x]$. Also, $x^e(\theta^e)$ will be used to denote x. If $0 < \theta^e < l(e)$ then $x^e(\theta^e)$ is an interior point of edge e.

Given two points of the network, x, $y \in N$, a *path* $P(x, y)$ joining x and y is a minimal connected subset of N including x and y. The length of a path is the sum of the lengths of all its edges and subedges. The *distance* $d(x, y)$ between $x \in N$ and $y \in N$ is defined as the length of the shortest path joining x and y.

Each vertex $v_i \in V$ has an associated nonnegative weight w_i representing a measure of the importance of that point. An unweighted vertex ($w_i = 0$) is a vertex without population. Let W be the set of weighted vertices. For a given subset $U \subseteq V$, let $w(U) = \sum\limits_{v_i \in U} w_i$. In particular, $w(V)$ represents the sum of the weights of all vertices of N.

A point x in the interior of an edge $[v_i, v_j]$ is a *bottleneck* point with respect to some vertex $v_k \in W$ if

$$d(v_k, x) = d(v_k, v_i) + l(v_i, x) = d(v_k, v_j) + l(x, v_j).$$

Let B_e be the set of the bottlenecks of a given edge $e \in E$ and $B_E = \bigcup\limits_{e \in E} B_e$ the set of all the bottenecks of the network.

Let B'_e be the set of bottleneck points of edge e with respect to some vertex $v_k \in W$ satisfying

$$d(v_k, x) = \min_{v_r \in W} d(x, v_r),$$

and let $B'_E = \bigcup\limits_{e \in E} B'_e$.

A point x in the interior of an edge $[v_i, v_j]$ is an *equidistant* point with respect to a pair of distinct vertices $v_k, v_l \in W$ if $d(x, v_k) = d(x, v_l)$ and $d(x, v_k)$ and $d(x, v_l)$ do not both decrease when x is perturbed slightly in either direction.

Let Q'_e be the set of equidistant points of edge e with respect some pair of weighted vertices v_k and v_l satisfying

$$d(v_k, x) = d(v_l, x) = \min_{v_r \in W} d(x, v_r),$$

and let $Q'_E = \bigcup\limits_{e \in E} Q'_e$.

The *average weighted distance* between a point $x \in N$ and the weighted vertices $v_i \in W$ is given by

$$z_{sum}(x) = \frac{1}{w(W)} \sum_{v_i \in W} w_i d(x, v_i).$$

A point maximizing $z_{sum}(.)$ in N is called an *antimedian.* The function $z_{sum}(.)$ is known to be concave and piecewise linear along any edge, with breakpoints at bottlenecks (see Hakimi, 1964). The antimedian problem has been studied and solved by Church and Garfinkel (1978) and by Minieka (1983).

The minimum distance between a point $x \in N$ and the weighted vertices $v_i \in W$ is given by

$$z_{\min}(x) = \min_{v_i \in W} d(x, v_i).$$

A point maximizing $z_{\min}(.)$ in N is called an *anticenter* (note that for Minieka (1983) the anticenter is the solution to the maxmax problem, while for us it is the solution to the maxmin problem). The function $z_{\min}(.)$ is concave and piecewise linear along any given edge $e \in E$, with at most two pieces of slopes $+1$ and -1. The only breakpoint x'_e of $z_{\min}(.)$ on edge e, if it exists, is a point of $B'_e \cup Q'_e$.

When all the vertices are uniformly weighted, that is, when $V = W$, the breakpoint x'_e of $z_{\min}(.)$ on each edge $e \in E$ is the midpoint of e. The anticenter problem has then a tivial solution: the midpoint of the largest edge. However, in the general case the breakpoint x'_e of function $z_{\min}(.)$ on a given edge $e = [v_i, v_j]$ is obtained as follows. Let v be the weighted vertex closest to v_i and u the weighted vertex closest to v_j. Then $x'_e = x^e(\theta^e_{uv})$, where

$$\theta^e_{uv} = \frac{d(u, v_j) - d(v, v_i) + l(v_i, v_j)}{2}, \tag{1}$$

and

$$z_{\min}(x'_e) = \frac{d(u, v_j) + d(v, v_i) + l(v_i, v_j)}{2}.$$

If $v \neq u$ then x'_e is an equidistant point with respect to v and u, $x'_e \in Q'_e$. Otherwise, let $p(v) = d(v, v_i) - d(v, v_j)$.
- If $p(v) = l(v_i, v_j)$, then $x'_e = v_i$ ($z_{\min}(.)$ is linearly decreasing in e).
- If $p(v) = -l(v_i, v_j)$, then $x'_e = v_j$ ($z_{\min}(.)$ is linearly increasing in e).
- If $\mid p(v) \mid < l(v_i, v_j)$, then x'_e is the bottleneck point with respect to v.

Note that the anticenter can be determined with complexity $O(n^2)$. Finding the closest vertex to every $v \in V$ requires $O(n^2)$ operations, providing that the distance matrix is known. Then $O(1)$ operations on each edge $e \in E$ are needed to determine x'_e, and the anticenter is the breakpoint with largest value of $z_{sum}(.)$. Since $m \leq n^2$, we have that the total number of operations required to obtain the anticenter is $O(n^2)$.

The objective function of the anti-cent-dian problem, $z_\lambda(.)$, is a convex combination of the objectives functions of the antimedian and anticenter problems, namely

$$z_\lambda(x) = \lambda z_{\min}(x) + (1 - \lambda) z_{sum}(x), \text{ with } 0 \leq \lambda \leq 1.$$

Any point $x \in N$ maximizing $z_\lambda(.)$ is called a λ-*anti-cent-dian* point, and the set of all λ-anti-cent-dian points is denoted $\lambda - ACD$.

In particular, if $\lambda = 0$ any λ-anti-cent-dian is an antimedian and if $\lambda = 1$ it is an anticenter. The value of λ reflects the weight attributed to the minimum distance with respect to the average.

3 Properties of the anti-cent-dian

Combining the properties of $z_{\min}(.)$ and $z_{sum}(.)$ it follows that

Property 1 *For* $x \in e = [v_i, v_j]$ *and a given value for* λ, $0 \leq \lambda \leq 1$, $z_\lambda(.)$ *is a continuous, concave and piecewise linear function*

i) with a finite number of breakpoints, all belonging to $B_e \cup \{x'_e\} = \Lambda_e$*,*

ii) and with a finite number of locally maximum values, all attained at points belonging to $\{v_i, v_j\} \cup \Lambda_e$*.*

If for a given value of λ and for two consecutive points x and y of $\{v_i, v_j\} \cup \Lambda_e$ we have $z_\lambda(x) = z_\lambda(y)$, then all points of the subedge $[x, y]$ are local maxima of $z_\lambda(.)$.

The previous property identifies a finite dominating set for the anti-cent-dian problem.

Property 2 $V \cup B_E \cup Q'_E$ *is a finite dominating set for the anti-cent-dian problem.*

Let $z(\lambda) = \max\limits_{x \in N} z_\lambda(x)$ denote the value of the λ-anti-cent-dian function at a λ-anti-cent-dian point.

Using Property 1, $z(\lambda)$ may be rewritten as

$$z(\lambda) = \max_{x \in N} z_\lambda(x) = \max_{e=[v_i,v_j] \in E} \max_{x \in \{v_i,v_j\} \cup \Lambda_e} z_\lambda(x) = \max_{x \in V \cup B_E \cup Q'_E} z_\lambda(x)$$

Since $V \cup B_E \cup Q'_E$ is a finite set, it follows that

Property 3 *The function* $z(\lambda)$ *is a continuous, convex and piecewise linear function of* λ *for* $0 \leq \lambda \leq 1$*.*

Property 3 implies that if λ_{i-1} and λ_i are adjacent points of discontinuity in the derivative of $z(\lambda)$, then there is some $x_i \in N$ such that $z_\lambda(x_i) \geq z_\lambda(x)$ for all $x \in N$ and $\lambda_{i-1} \leq \lambda \leq \lambda_i$. Therefore,

Property 4 *There exists two finite sets* $\{\lambda_i \in [0,1] : i = 0, ..., r\}$ *and* $\{x_i \in N : i = 1, ...r\}$ *with the following properties:*

(i) $\lambda_i < \lambda_{i+1}$ *for all* i*;* $\lambda_0 = 0$ *and* $\lambda_r = 1$*.*

(ii) $x_1 = a_m$ *(an antimedian) and* $x_r = a_c$ *(an anticenter).*

(iii) $z(\lambda) = z_\lambda(x_{i+1})$ *for* $\lambda_i \leq \lambda \leq \lambda_{i+1}$*,* $i = 0, ..., r-1$*.*

4 λ-anti-cent-dian points for all $\lambda \in [0,1]$

The method proposed in this section for calculating the set $\bigcup\limits_{\lambda=0}^{1} \lambda - ACD$ is based on an idea similar to that given by Hansen et al. (1991) to calculate all the λ-cent-dian points of a network.

Let $h : N \longrightarrow I\!R^2$ be a mapping from the network N into the plane $I\!R^2$ defined as $h(x) = (z_{\min}(x), z_{sum}(x))$ for all $x \in N$. Since N is connected and $z_{\min}(.)$ and $z_{sum}(.)$ are continuous and piecewise linear functions, the image $h(N)$ is a connected set composed of linear segments of $I\!R^2$. The extremes of these linear segments are the images of points of $V \cup B_E \cup Q'_E$.

Given the definition of $z_\lambda(.)$, all λ-anti-cent-dian points $(0 \leq \lambda \leq 1)$ are the points $x \in N$ having an image $h(x)$ which belongs to the upper boundary of the convex hull of $h(N)$. Moreover, this convex hull coincides with the convex hull of $h(V \cup B_E \cup Q'_E)$.

¿From these results derives the following method to find $\bigcup\limits_{\lambda=0}^{1} \lambda - ACD$.

Algorithm 1

STEP 1: For each edge $e = [v_i, v_j]$:

(i) Find the set B_e of bottlenecks.

(ii) Find the breakpoint x'_e of $z_{\min}(.)$ using (1).

(iii) Merge $\{v_i, v_j\}$, B_e and $\{x'_e\}$ and sort the points of the resulting set, S_e, in terms of increasing distance to v_i.

(iv) Determine the set $h(S_e)$.

STEP 2: Determine the convex hull $\mathcal{H}$ of

$$h(S_E) = \bigcup_{e \in E} h(S_e).$$

The set $\bigcup_{\lambda=0}^{1} \lambda - ACD$ is given by the points $x \in N$ satisfying the condition that $h(x)$ is an extreme point, or is on a side of the upper boundary of $\mathcal{H}$.

Proposition 1 *Algorithm 1 determines $\bigcup_{\lambda=0}^{1} \lambda - ACD$ in $O(mn \log n)$ operations.*

Proof. The number of bottlenecks of an edge $e = [v_i, v_j]$ is bounded by $| V | = n$, and they can be determined with complexity $O(n)$. Only one operation is required to calculate x'_e using (1). Merging the sets $\{v_i, v_j\}$, B_e and $\{x'_e\}$ requires at most $O(n)$ operations. Finally, it is possible to calculate the values of $z_{\min}(.)$ and $z_{sum}(.)$ for the elements of the sorted set $\{v_i, v_j\} \cup \Lambda_e$ with complexity $O(n)$.

Therefore, Step 1 requires $O(mn)$ operations, as it is applied to each edge $e \in E$.

The set $h(V \cup B_E \cup Q'_E)$ contains $O(mn)$ points, and the convex hull of a set of n points can be determined with $O(n \log n)$ operations. Then, Step 2 requires $O(mn \log n)$ operations (note that in any network $m \leq n^2$). $\square$

Let $\mathcal{H}^*$ be the set of efficient points of the $\max\{z_{\min}(.), z_{sum}(.)\}$ bicriterion problem, that is, $\mathcal{H}^*$ is the upper boundary of the convex hull $\mathcal{H}$. This curve coincides with the upper boundary of $\mathcal{H}$ when the range of $h(.)$ is $\{x \in N : z_{\min}(x) \in [z_{\min}(a_m), z_{\min}(a_c)]$ and $z_{sum}(x) \in [z_{sum}(a_c), z_{sum}(a_m)]\}$, where a_m is the antimedian with largest value of $z_{\min}(.)$ and a_c is the anticenter with largest value of $z_{sum}(.)$. Since $h(N)$ is composed of a finite number of line segments, the definition of $\mathcal{H}^*$ implies that it is a piecewise concave curve connecting $h(a_m)$ and $h(a_c)$. Let $\mathcal{H}^*$ be called the *anti-cent-dian curve.*

Theorem 1 *The set $\{x_i \in N : i = 1, ..., r\}$ from property 4 coincides with the set of point $x \in N$ satisfying the condition that $h(x)$ is an extreme point of $\mathcal{H}^*$.*

Proof. The first and last elements, a_m and a_c, of both sets coincide by definition. We have to see that x_i maximizes $z_\lambda(.)$ over an interval $[\lambda_{i-1}, \lambda_i]$ if and only if $h(x_i)$ is an extreme point of $\mathcal{H}^*$.

CASE 1:

Suppose $h(x_i)$ is an extreme point of $\mathcal{H}^*$. Then, there exists a line tangent to $\mathcal{H}^*$ at $h(x_i)$, the equation of which might be $\lambda x + (1-\lambda)y = k$, with $\lambda > 0$ and slope $\lambda/(\lambda - 1)$; thus, $(\lambda, 1-\lambda)h(x) \leq (\lambda, 1-\lambda)h(x_i)$, that is, $z_\lambda(x) \leq z_\lambda(x_i)$, for all $x \in N$. In fact, there exists a family of lines tangent to $\mathcal{H}^*$ at

$h(x_i)$, the elements of which have the following correspondence with the set of $\lambda_{i-1} \leq \lambda \leq \lambda_i$: the slope of the line segment of H^* connecting $h(x_{i+1})$ and $h(x_i)$ is equal to $\lambda_i/(\lambda_i - 1)$. Thus, for each $\lambda_{i-1} \leq \lambda \leq \lambda_i$, the function $z_\lambda(.)$ reaches its maximum at x_i.

CASE 2:

Suppose x_i maximizes $z_\lambda(.)$ for all $\lambda \in [\lambda_{i-1}, \lambda_i]$. Then, $h(x_i)$ has to be an extreme point of $\mathcal{H}^*$. Otherwise, $h(x_i)$ is an interior point of a linear segment of $\mathcal{H}^*$, implying that x_i maximizes $z_\lambda(.)$ for only a unique value of λ and not for an interval, as assumed. $\square$

5 $\lambda - ACD$ set for a given λ

In this section we consider the problem of finding all the λ-cent-dian points for a given λ, $0 \leq \lambda \leq 1$. The way to proceed consists in finding first the best points on each edge and then choosing those points where $z_\lambda(.)$ is maximized. This is an adaptation of the method proposed by Church and Garfinkel (1978) to calculate the antimedian of a graph.

Consider edge $e = [v_i, v_j]$ and let all vertices $v_k \in W$ be sorted and reindexed in terms of decreasing

$$p(v_k) = d(v_i, v_k) - d(v_j, v_k).$$

Note that if $\mid p(v_k) \mid < l(v_i, v_j)$ then the point $x^e(\theta_k^e)$ given by

$$\theta_k^e = \frac{l(v_i, v_j) - p(v_k)}{2}$$

is a bottleneck.

The value

$$D(t) = \frac{1}{w(V)} \left(\sum_{k=t+1}^{n} w_k - \sum_{k=1}^{t} w_k \right), t = 1, ..., n,$$

represents the slope of function $z_{sum}(.)$ on the subedge $[x^e(\theta_t^e), x^e(\theta_{t+1}^e)]$, when $\theta_t^e \neq \theta_{t+1}^e$.

The slope of $z_\lambda(.)$ is a convex combination of the slopes of functions $z_{\min}(.)$ and $z_{sum}(.)$, so, it follows that its value on each subedge $[x^e(\theta_t^e), x^e(\theta_{t+1}^e)]$ is

$$\begin{array}{ll} \lambda + (1-\lambda)D(t) & \text{if } x'_e \geq x^e(\theta_{t+1}^e) \\ \left.\begin{array}{l} \lambda + (1-\lambda)D(t), \text{ in } [x^e(\theta_t^e), x'_e] \\ -\lambda + (1-\lambda)D(t), \text{ in } [x'_e, x^e(\theta_{t+1}^e)] \end{array}\right\} & \text{if } x^e(\theta_t^e) < x'_e < x^e(\theta_{t+1}^e) \\ -\lambda + (1-\lambda)D(t) & \text{if } x'_e \leq x^e(\theta_t^e) \end{array}$$

The first breakpoint x^* of $z_\lambda(.)$ along $e \in E$ where the slope of the function is less than or equal to zero is a best point of $z_\lambda(.)$ on e. If the slope to $z_\lambda(.)$ in x^* is exactly equal to zero, then all the points of the subedge delimited by x^* and the next breakpoint of $z_\lambda(.)$ are best points.

Theorem 3 *The set $\lambda - ACD$ of a graph can be determined with complexity $O(mn \log n)$.*

Proof. The proof of this theorem is based on the following ideas: given an edge $e = [v_i, v_j]$, we have that

1) The set B_e of bottlenecks can be determined with complexity $O(n)$, and applying (1) x'_e can be calculated with $O(1)$ operations.

2) The set $\{v_i, v_j\} \cup \Lambda_e$ can be sorted with complexity $O(n \log n)$, since its cardinality is $O(n)$.

3) It is not necessary fo find the slopes of all the linear segments of $z_\lambda(.)$. Since they constitute a sorted and nonincreasing set of numbers, a binary search can be applied to find only the first of these slopes which is nonpositive. The complexity of the binary seach is $O(\log n)$, and calculating each slope requires $O(n)$ operations. Therefore, finding the first nonpositive slope is $O(n \log n)$ complex. □

We can obtain an upper bound for the optimal value of $z_\lambda(.)$ over an edge as follows. Given an edge $e = [v_i, v_j]$ and a vertex v_k,

$$d(x, v_k) \leq \frac{l(v_i, v_j) + d(v_i, v_k) + d(v_j, v_k)}{2}, \text{for all } x \in e.$$

Thus

$$z_{sum}(x) = \frac{1}{w(V)} \sum_{v_k \in V} w_k d(x, v_k)$$

$$\leq \frac{1}{w(V)} \sum_{v_k \in V} w_k \frac{l(v_i, v_j) + d(v_i, v_k) + d(v_j, v_k)}{2}$$

$$= \frac{l(v_i, v_j) + z_{sum}(v_i) + z_{sum}(v_j)}{2} = UB_{z_{sum}}(i, j).$$

and

$$z_{\min}(x) = \min_{v_k \in W} d(x, v_k)$$

$$\leq \min_{v_k \in W} \frac{l(v_i, v_j) + d(v_i, v_k) + d(v_j, v_k)}{2}$$

$$= \frac{l(v_i, v_j) + z_{\min}(v_i) + z_{\min}(v_j)}{2} = UB_{z_{\min}}(i, j).$$

The convex combination of the upper bounds of $z_{\min}(.)$ and $z_{sum}(.)$ constitutes an upper bound for $z_\lambda(.)$. Therefore,

$$UB_\lambda(i, j) = \lambda UB_{z_{\min}}(i, j) + (1 - \lambda) UB_{z_{sum}}(i, j).$$

We propose the following heuristic algorithm to find the λ-anti-cent-dian using these upper bounds.

Algorithm 2

STEP 1: Let v be the vertex with largest value of $z_\lambda(.)$. Let $LB_\lambda = z_\lambda(v)$.

STEP 2: For each edge $[v_i, v_j] \in E$ compute $UB_\lambda(i, j)$.

STEP 3: Apply the method described above to get a candidate to anti-cent-dian point only for those edges with $LB_\lambda < UB_\lambda(i, j)$.

STEP 4: Choose the candidate with largest value of $z_\lambda(.)$.

6 Computational results

The computational results shown in this section have been obtained by applying algorithm 2 to connected and randomly generated networks. The way of generating the networks ensures: first that they are connected by obtaining the minimum spanning tree; then the remaining edges are added randomly. The program was run on a HP 9000 712/80 machine.

For a given network $N(V, E)$ with n vertices and m edges and for a given $\lambda \in [0, 1]$, the table shows the time in seconds (T-Ex) that the exact algorithm needs to solve the problem, the number of edges examined by the heuristic algorithm and the time in seconds (T-He) it takes to give the solution.

(n, m)	λ	**T-Ex**	**Edges Examined**	**T-He**
(100, 150)	0.1	0.45	3	0.06
	0.3	0.45	4	0.06
	0.5	0.46	6	0.07
	0.7	0.46	24	0.13
	0.9	0.47	92	0.33
(100, 250)	0.1	0.77	11	0.09
	0.3	0.78	38	0.19
	0.5	0.77	88	0.34
	0.7	0.77	149	0.54
	0.9	0.77	215	0.73
(100, 700)	0.1	2.16	368	1.31
	0.3	2.15	424	1.47
	0.5	2.15	496	1.7
	0.7	2.14	574	1.93
	0.9	2.14	656	2.19
(500, 1000)	0.1	11.51	2	1.25
	0.3	11.47	4	1.27
	0.5	11.51	28	1.53
	0.7	11.62	308	4.75
	0.9	11.55	742	9.75

References

[1] Church, R.L., and Garfinkel, R.S. (1978), "Locating an obnoxious facility on a network", *Trans. Sci.* 12/2, 107-118.

[2] Hakimi, S.L. (1964), "Optimal locations of switching centers and the absolute centers and medians of a graph", *Oper. Res.* 12, 450-459.

[3] Halpern, J. (1976) "The location of a center-median convex combination on an undirected tree", *Journal of Regional Science* 16/2, 237-245.

[4] Hansen, P., Labbe, M., and Thisse, J.F. (1991) "From the median to the generalized center", *Recherche Operationnelle* 25/1, 73-86.

[5] Minieka, E. (1983) "Anticenters and antimedians of a network", *Networks* 13, 359-364.