JOURNAL OF RESEARCH of the National Bureau of Standards — B. Mathematical Sciences
Vol. 79B, Nos. 3 and 4, July–December 1975

Maximin Facility Location*

Peter J. Slater

Institute for Basic Standards, National Bureau of Standards, Washington, D.C. 20234

(March 4, 1975)

Using the criterion that vertex u of a graph G is "more central" than vertex v if there are more vertices closer to u than to v, the security center and security centroid of a graph are defined. Several results, including the fact that the security center of a tree is the centroid, are presented. A simple algorithm to find the security center (centroid) of a tree is presented. The problem of finding the security center of G is shown reducible to a search over a single block of G.

Key words: Center; centroid; facility location; graph; tree.

1. Introduction

Consider a finite, connected, undirected graph G with vertex set V and edge set E. If $e \epsilon E$, then $L(e)$ will denote the length of edge e ($L(e) > 0$), and if P is a path, then $L(P) = \Sigma_{e \epsilon P} L(e)$ denotes the length of P. The distance between vertices u and v, denoted $d(u, v)$, is the length of the shortest path from u to v.

Several mathematical problems arise from different instances of the question of what is an optimal location for a facility in a graph. Hakimi [3][1] posed and considered two such optimization problems. The first problem is to determine a vertex $u \epsilon V$ so as to minimize

$$D_1(u) = \Sigma_{v \epsilon V} w(v) d(v, u) \tag{1}$$

where $w(v)$ is a nonnegative weight given for each $v \epsilon V$. The second problem is to determine a vertex $u \epsilon V$ so as to minimize

$$D_2(u) = \max\{w(v) d(u, v) : v \epsilon V\}. \tag{2}$$

Interpretations of these problems [2] include minimizing transportation cost to a central facility for (1), and minimizing a maximum response time from an emergency facility, such as a hospital, for (2). While a more general treatment is presented in the third section, it is assumed for now that $w(v) = 1$ for every $v \epsilon V$. Until section four, it is assumed that each edge has length equal to one.

For each $u \epsilon V$, let S_u denote the family of distances from u to the other vertices in V. That is, $S_u = \{d(u, v) : v \epsilon V - \{u\}\}$. Note that one allows repetitions in S_u. For example, for the graph G_1 of figure 1, $S_{u_1} = \{1, 1, 2, 2, 3\}$ and $S_{u_2} = \{1, 1, 1, 1, 2\}$.

Hakimi's first problem involves finding a vertex u for which the sum of all the elements of S_u is a minimum. This sum will be called the *distance of u in G*, denoted $d(u)$. The set of all vertices

AMS-MOS Subject Classification: 05C05.

*This work was done while the author was a National Academy of Sciences-National Research Council Postdoctoral Research Associate at the National Bureau of Standards, Washington, D.C. 20234.

[1] Figures in brackets indicate the literature references at the end of this paper.

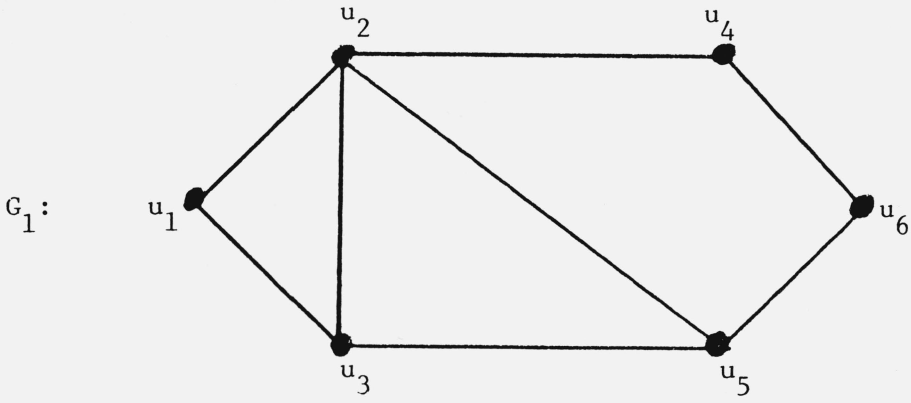


FIGURE 1. *Graph* G_1.

with minimum distance in G is called the *centroid*, and Zelinka [6] has shown that for trees this agrees with the "branch weight" definition of centroids (see sec. 2). The maximum value in S_u will be called the eccentricity of u, denoted $e(u)$, and Hakimi's second problem is to find a point of minimum eccentricity. The set of all vertices with minimum eccentricity is called the *center*.

Another set of numbers will now be associated with u, and paralleling the previous definitions, the security centroid and security center will be defined for G. Consider two players X and Y (e.g., two companies or two countries), each of which is to locate a facility at a vertex of G (e.g., a store or a military base), say at vertices x and $y (x \neq y)$. If $d(v, x) < d(v, y)$ then x will be said to have more "influence" over v than y does (customers usually shop at the nearer of two otherwise equal stores, a location is more easily protected by the closer of two bases). The player who must choose a location first desires to have more influence than his competitor over as many vertices as possible.

If u and v are distinct vertices in V, let V_{uv} be the set of vertices which are closer to u than to v. That is,

$$V_{uv} = \{s \epsilon V : d(s, u) < d(s, v)\}.$$

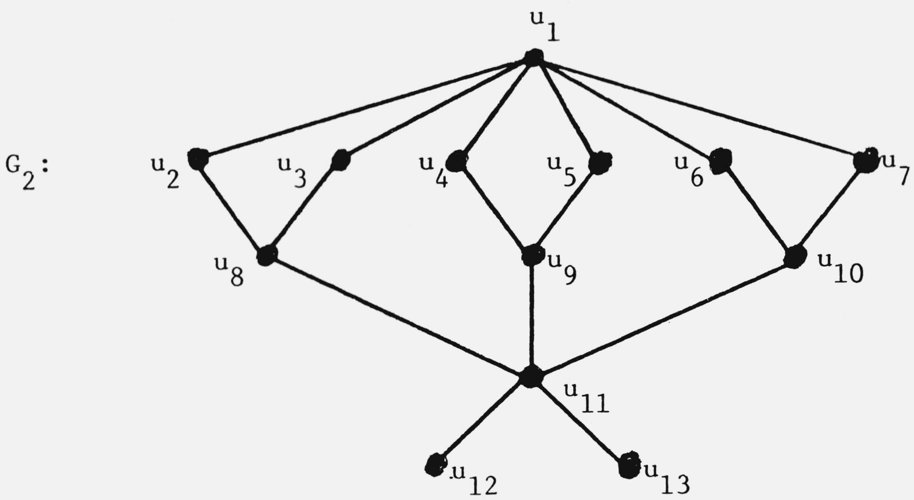
Note that $u \epsilon V_{uv}$. Let $|S|$ denote the cardinality of a set S, and let $f(u, v) = |V_{uv}| - |V_{vu}|$. That is, $f(u, v)$ equals the number of vertices closer to u than to v minus the number of vertices closer to v than to u. One is now interested in maximizing the functions

$$f(u) = \min\{f(u, v) : v \epsilon V - u\}, \tag{3}$$

$$g(u) = \Sigma_{v \epsilon V - u} f(u, v). \tag{4}$$

Label the vertices of G as $u_1, u_2, \ldots, u_p$. One can form an array M with $m_{ij} = f(u_i, u_j)$ $(1 \leq i, j \leq p, i \neq j)$. Now $f(u_i)$ is the smallest element in the ith row, and $g(u_i)$ is the sum of the ith row. Examples of graphs and some of the rows from the corresponding arrays are given in figures 2, 3, and 4.

If there is a point u such that $f(u) \geq 1$, then, since $f(v, u) = -f(u, v) \leq -1, f(v) \leq -1$ for every $v \epsilon V - u$ (as in fig. 2), and so u is the unique maximizer of f. There may be several points $u_1, \ldots, u_k$ such that $f(u_i) = 0$ for $1 \leq i \leq k$ (as in fig. 3); then these points are the maximizers of f.



u \ v	u_1	u_2	u_3	u_4	u_5	u_6	u_7	u_8	u_9	u_{10}	u_{11}	u_{12}	u_{13}
u_1	—	3	3	3	3	3	3	1	1	1	1	4	4
u_2	−3	—	0	0	0	0	0	−1	−1	−1	−3	3	3
u_8	−1	1	1	1	1	1	1	—	0	0	−5	3	3
u_{11}	−1	3	3	3	3	3	3	5	5	5	—	11	11
u_{12}	−4	−3	−3	−3	−3	−3	−3	−3	−3	−3	−11	—	0

FIGURE 2. *Graph* G_2 *with* $f(u_1) > 0$.

Finally (as in fig. 4) it is possible that $f(u) \leq -1$ for every $u \epsilon V$. For this last case, "follow the leader is optimal," that is, the player who chooses a location after the other player has chosen has the advantage. In the first case, the first moving player has the advantage, while in the second case, the order of moving is immaterial.

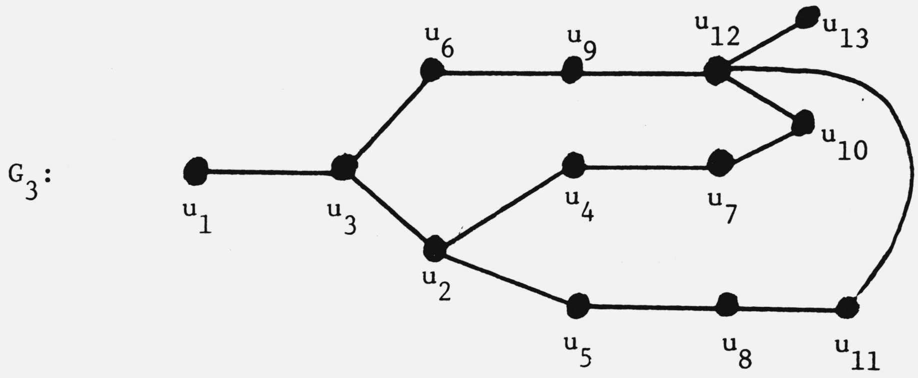
Let $r(G)$, the radius of G, and $s(G)$ be defined as follows.

$$r(G) = \min_{u \epsilon V} (\max_{v \epsilon V} d(u, v)) \tag{5}$$

$$s(G) = \max_{u \epsilon V} (\min_{v \epsilon V - u} f(u, v)) \tag{6}$$

$$= \max_{\mu \epsilon V} f(\mu)$$

The center of G consists of all vertices u for which $e(u)$ equals $r(G)$. The *security center* of G is the set of all vertices u for which $f(u)$ equals $s(G)$, and it will be denoted by $C(G)$. The *security centroid* of G, denoted $C_1(G)$, is the set of all vertices at which function g is a maximum. Referring



u \ v	u_1	u_2	u_3	u_4	u_5	u_6	u_7	u_8	u_9	u_{10}	u_{11}	u_{12}	u_{13}
u_1	—	−6	−11	−5	−5	−5	−3	−3	−3	−3	−3	−4	−1
u_2	6	—	1	3	3	1	2	2	1	1	1	0	3
u_{12}	4	0	1	1	1	2	3	3	3	5	5	—	11
u_{13}	1	−3	−2	−2	−2	−1	−1	−1	−4	−3	−3	−11	—

FIGURE 3. *Graph* G_3 *with* $f(u_2)=f(u_{12})=0$.

to figures 1, 2, 3, and 4, $C(G_1)=C_1(G_1)=\{u_2\}$, $C(G_2)=\{u_1\}$, $C_1(G_2)=\{u_{11}\}$, $C(G_3)=\{u_2, u_{12}\}$, $C_1(G_3)=\{u_{12}\}$, $C(G_4)=\{u_1, u_4, u_5, u_6, u_7, u_{11}, u_{12}, u_{13}\}$, and $C_1(G_4)=\{u_7\}$.

2. Security Centers and Security Centroids of Trees

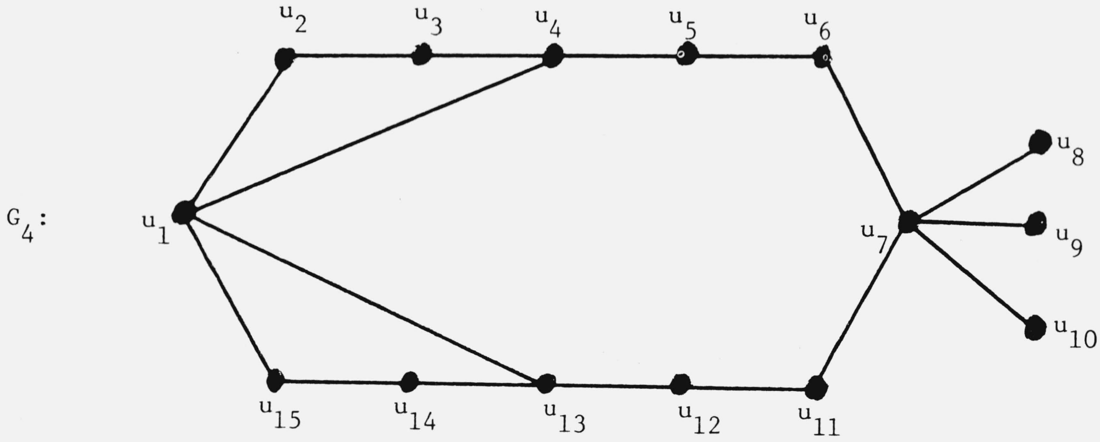
It is reasonable to think that if u is closer to more vertices of G than v is, then u is "more central" than v. However, as for G_4 (fig. 4), it is possible to find a set of vertices $\{u_1, u_2, \ldots, u_k\}$, for example vertices u_4, u_5, u_{13}, and u_{12} in G_4, such that u_i is "more central" than u_{i+1} $(1 \leq i \leq k-1)$ and u_k is "more central" than u_1. Despite this, the next theorem is a strong indication that the security center truly is a set of "central" vertices.

Suppose G is a tree and $v \in V$. Label the components of $G-v$ as $B_1, B_2, \ldots, B_t$ (where the degree of v is t). The *weight at point* v is the maximum cardinality of a V_i where V_i is the set of vertices in $B_i (1 \leq i \leq t)$. A vertex v is called a *branch weight centroid vertex* of the tree G if v has minimum weight, and the *branch weight centroid* of G consists of all such vertices. As noted previously, Zelinka has shown the branch weight centroid of a tree to be identical with its centroid.

THEOREM 1: *If* G *is a tree, then the security center of* G *is the centroid.*

PROOF: Jordan [5] has shown that the centroid of a tree consists of one vertex or two adjacent vertices.

First, suppose the centroid consists of one point, say v, with branches $B_1, B_2, \ldots, B_t$ whose vertex sets satisfy $|V_1| \leq |V_2| \leq \ldots \leq |V_t|$. Let u_t be the vertex of V_t adjacent to v. If $|V_t| > |V_1|$



u \ v	u_1	u_2	u_3	u_4	u_5	u_6	u_7	u_8	u_9	u_{10}	u_{11}	u_{12}	u_{13}	u_{14}	u_{15}
u_1	–	11	5	−1	0	1	1	3	3	3	1	0	−1	5	11
u_4	1	6	11	–	1	1	1	3	3	3	0	−1	0	5	6
u_5	0	1	5	−1	–	1	0	3	3	3	−1	0	1	4	3
u_{12}	0	3	4	1	0	−1	0	3	3	3	1	–	−1	5	1
u_{13}	1	6	5	0	−1	0	1	3	3	3	1	1	–	11	6

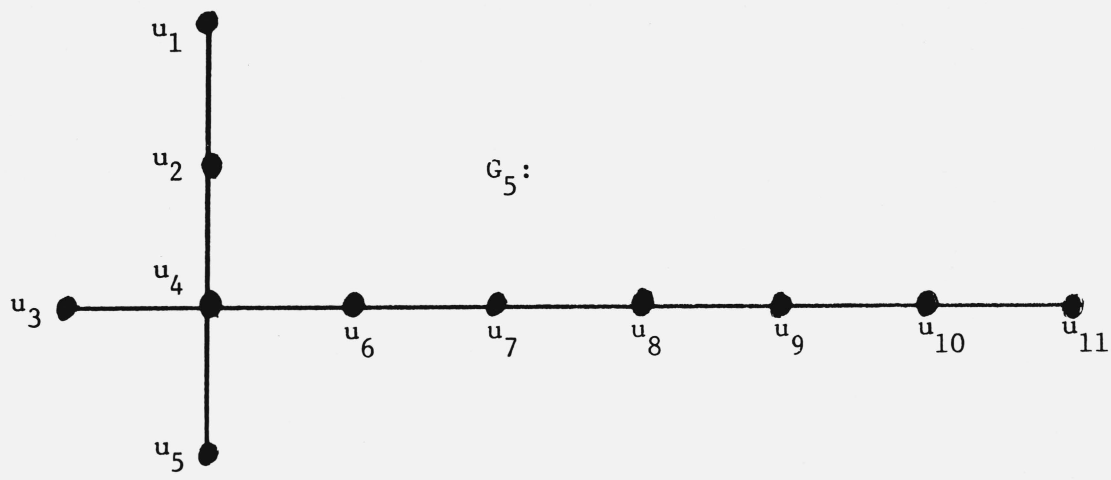
FIGURE 4. *Graph G_4 with $f(u) \leq -1$ for every vertex u.*

$+|V_2|+ \ldots +|V_{t-1}|$, then u_t has a weight bounded above by max $(|V_1|+ \ldots +|V_{t-1}|, |V_t|-1) \leq |V_t|$. Since the weight of u_t is less than or equal to the weight $|V_t|$ of v, the centroid of G could not be $\{v\}$. Thus $|V_t| \leq |V_1|+ \ldots +|V_{t-1}|$. Now let $u \epsilon V_i$. Then $f(v, u) \geq (\Sigma_{j=1, j\neq i}^{t} |V_j|) - (|V_i|-1) \geq 1$. Thus $f(v) \geq 1$ and $C(G)=\{v\}$ is the centroid.

Next, suppose the centroid of G consists of two adjacent vertices, v_1 and v_2. Let B_1 and B_2 be the components of $G-v_1v_2$, with $v_i \epsilon B_i$ and V_i the vertex set of $B_i (i=1, 2)$. If $|V_1| > |V_2|$ then the weight of v_2 is $|V_1|$, and the weight of v_1 is bounded above by max $(|V_2|, |V_1|-1) < |V_1|$. Thus V_1 and V_2 would have different weights, so the centroid of G could not be $\{v_1, v_2\}$. Similarly $|V_2| \leq |V_1|$. Thus $|V_1| = |V_2|$. If $u \epsilon V_i - \{v_i\}$ $(i=1$ or $2)$, then $f(u, v_i) \leq -2$, which implies $f(u) \leq -2$. It is easy to see that $f(v_1)=f(v_2)=0$, and so $C(G)=\{v_1, v_2\}$ is the centroid.

COROLLARY 1a: *The security center of a tree consists of one point or two adjacent points.*

Graph G_4 (fig. 4) is an example of a graph for which the center, centroid, security center and security centroid are different subsets of V. Even for trees it is possible to have the center, centroid (security center) and security centroid distinct, as in figure 5. Graph G_6 (fig. 6) has a security centroid consisting of three independent points, namely those which are adjacent to the endpoints, when $k \geq 14$.



u \ v	u_1	u_2	u_3	u_4	u_5	u_6	u_7	u_8	u_9	u_{10}	u_{11}
u_4	8	7	9	—	9	−1	0	1	2	3	4
u_6	7	4	5	1	5	—	1	2	3	4	5
u_7	4	1	1	0	1	−1	—	3	4	5	6

FIGURE 5. *Graph* G_5 *with center* $\{u_7\}$, *centroid* $\{u_6\}$ *and security centroid* $\{u_4\}$.

One can modify graph G_6 by using n branches at v (instead of three), each of which is identical to a branch at v in G_6. It can be shown that if k is sufficiently large, then the security centroid is precisely those n points which are adjacent to endpoints. One therefore has the following theorem.

THEOREM 2: *For any natural number* n *there is a tree* T_n *whose security centroid consists of* n *independent vertices (in fact, no three of which are on one path).*

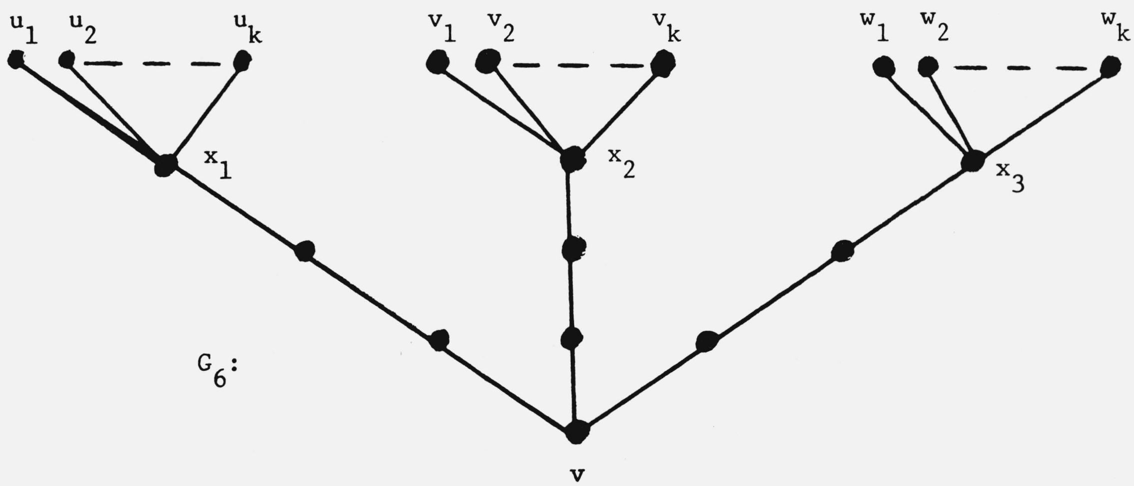


FIGURE 6. *Graph* G_6 *with security centroid* $\{x_1, x_2, x_3\}$.

3. Cutpoints and Bridges in Graphs

As Goldman has done for (2) in [2], it is convenient to put function f from (3) in a more general setting by assigning a nonnegative integer valued addend $a(s)$ to each vertex s. One may think of $a(s)$ in this case as a number of vertices of degree one, each of which is adjacent to s. If $s \epsilon V_{uv}$ then each of these vertices would also be closer to u than to v. Let $\alpha(s)=a(s)+1$, and let $f^*(u, v)$ be defined as follows.

$$f^*(u, v)=(|V_{uv}|+\Sigma_{s \epsilon V_{vu}} a(s))-(|V_{vu}|+\Sigma_{s \epsilon V_{vu}} a(s))$$

$$=\Sigma_{s \epsilon V_{uv}} \alpha(s)-\Sigma_{s \epsilon V_{uv}} \alpha(s). \quad (7)$$

Note that $u \epsilon V_{uv}$. One is now interested in maximizing the function

$$f^*(u)=\min\{f^*(u, v) : v \epsilon V\}. \quad (8)$$

Maximizing f is the same as maximizing f^* when $\alpha(s)=1$ for every vertex s of V. The security center of (G, α) is the set of vertices for which f^* is a maximum, and is denoted $C^*(G, \alpha)$.

Given a connected graph G, a *cutpoint* (*bridge*) is a vertex (edge) whose removal disconnects G. A block B of G is a connected subgraph maximal with respect to the property that it has no cutpoints (and therefore if B has at least three vertices, B will also not have a bridge). In this section it will be shown how maximizing the function (8) can be reduced to an equivalent analogous problem for a single block of G. This will result in a simple algorithm for finding the security center (centroid) of a tree.

Suppose one is given (G, α) where $\alpha(u) \geq 1$ for every vertex u of G. Suppose s is a cutpoint, and let $S_1, S_2, \ldots, S_k (k \geq 2)$ denote the vertex sets of the components of $G-s$. Let $\sigma_i=\Sigma_{u \epsilon S_i} \alpha(u)$, and assume the components are labeled in such a way that $\sigma_1 \leq \sigma_2 \leq \ldots \leq \sigma_K$.

LEMMA 3: *If* $\sigma_k < \alpha(s)+\Sigma_{i=1}^{k-1} \sigma_i$, then $C^*(G, \alpha)=\{s\}$.

PROOF: Assume $1 \leq j \leq k-1$ and $v \epsilon S_j$. Now $f^*(s, v) \geq (\alpha(s)+\Sigma_{i=1, i \neq j}^{k} \sigma_j)-\sigma_j \geq 1$. Therefore $f^*(s) \geq 1$, implying $f^*(u) \leq -1$ for every vertex u of $G-s$. Thus $C^*(G, \alpha)=\{s\}$.

LEMMA 4: *If* $\sigma_k \geq \alpha(s)+\Sigma_{i=1}^{k-1} \sigma_i$, *then* $C^*(G, \alpha)$ *is contained in* $S_k \cup \{s\}$.

PROOF: If $f^*(s) \geq 1$ then the security center is $\{s\}$. Suppose $f^*(s) \leq 0$. Let $S=S_k \cup \{s\}$, and let $\bar{S}=\cup_{i=1}^{k-1} S_i$. Since $\sigma_k \geq \alpha(s)+\Sigma_{i=1}^{k-1} \sigma_i$, if $w \epsilon \bar{S}$ then $f^*(s, w) \geq 1$. Thus there is a point $x \epsilon S_k$ such that $f^*(s, x)=f^*(s) \leq 0$.

Let $w \epsilon \bar{S}$. Since $f^*(w) \leq f^*(w, x)$, to show that $w \notin C^*(G, \alpha)$ it will suffice to show that $f^*(w, x) < f^*(s, x)=f^*(s)$.

If $v \epsilon V_{xs}$ then $v \epsilon S_k$, and so $d(v, s) < d(v, w)$, implying $v \epsilon V_{xw}$. Thus $V_{xs} \subseteq V_{xw}$. Suppose $v \epsilon V_{wx}$. If $v \notin S_k$ then clearly $v \epsilon V_{sx}$, and if $v \epsilon S_k$ then $d(v, s) < d(v, w) < d(v, x)$, which implies $v \epsilon V_{sx}$. Thus $V_{wx} \subseteq V_{sx}$. Note also that $w \epsilon V_{sx}$. It is claimed that at least one of the two inclusions $V_{xs} \subseteq V_{xw}$ and $V_{wx} \subseteq V_{sx}$ must be proper. For, suppose $V_{wx}=V_{sx}$. In particular, one has $d(s, w) < d(s, x)$ since $s \epsilon V_{sx}$. Hence $d(s, x) \geq 2$. Let $s, s_1, s_2, \ldots, s_p, x$ be a shortest path from s to x, and note that $p \geq 1$. Now p cannot be even, else $s_{p/2}$ would be in V_{sx} but not in V_{wx}. Say $p=2j+1$. Now s_{j+1} is in V_{xw} but not V_{xs}, and the claim is proved.

Now we have

$$f^*(w, x)=\Sigma_{u \epsilon V_{wx}} \alpha(u)-\Sigma_{u \epsilon V_{xw}} \alpha(u)$$

$$< \Sigma_{u \epsilon V_{sx}} \alpha(u)-\Sigma_{u \epsilon V_{xs}} \alpha(u)$$

$$=f^*(s, x)=f^*(s).$$

591-673 O - 76 - 2

Consequently $f^*(w) < f^*(s)$, and w is not in the security center.

THEOREM 5: *If* $\sigma_k \geq \alpha(s) + \Sigma_{i=1}^{k-1} \sigma_i$, *then* $C^*(G,\alpha) = C^*(G-\bar{S}, \alpha')$ *where* $\bar{S} = \cup_{i=1}^{k-1} S_i$, $\alpha'(u) = \alpha(u)$ *if* $u \epsilon S_k$, *and* $\alpha'(s) = \alpha(s) + \Sigma_{u \epsilon \bar{S}} \alpha(u)$.

PROOF: Let $S = S_k \cup \{s\}$. By definition of $\alpha'(s)$, if x and y are any two vertices of S then $f^*(x,y)$ in $(G-\bar{S}, \alpha')$ equals $f^*(x,y)$ in (G,α). In particular, if $f^*(s) \geq 1$ in (G,α) then $f^*(s) \geq 1$ in $(G-\bar{S}, \alpha')$, and so $C^*(G,\alpha) = C^*(G-\bar{S}, \alpha') = \{s\}$.

Suppose $f^*(s) \leq 0$ in (G,α). Now $f^*(s) = f^*(s,y)$ in (G,α) for some y in S_k. Thus $f^*(s)$ has the same value in $(G-\bar{S},\alpha')$ as in (G,α). If $x \epsilon S_k$ and $w \epsilon \bar{S}$, then (by the proof of Lemma 4) $f^*(w,x) < f^*(s,x)$, so that $f^*(x,w) = -f^*(w,x) > -f^*(s,x) = f^*(x,s)$. Therefore $f^*(x)$ in (G,α) equals min $\{f^*(x,u)\colon u \epsilon S\}$, which is $f^*(x)$ in $(G-\bar{S}, \alpha')$.

Thus no point of $\bar{S}$ is in $C^*(G,\alpha)$ and every point of $G-\bar{S} = S$ attains the same value of f^* in $(G-\bar{S}, \alpha')$ as in (G,α). Consequently $C^*(G,\alpha) = C^*(G-\bar{S},\alpha')$.

Now suppose edge st is a bridge of G, and let S and T be the vertex sets of the components of $G-st$ with $s \epsilon S$ and $t \epsilon T$. Let $\sigma = \Sigma_{u \epsilon S} \alpha(u)$, and let $\tau = \Sigma_{u \epsilon T} \alpha(u)$. Note that each of s and t is a cutpoint or an endpoint.

COROLLARY 5a: *If* $\sigma = \tau$ *then the security center of* (G, α) *is* $\{s, t\}$.

COROLLARY 5b: *If* $\sigma > \tau$, *then no vertex of* T *can be in the security center.*

COROLLARY 5c: *If* $\sigma > \tau$, *then the security center of* (G, α) *is the security center of* $(G-T, \alpha')$ *with* $\alpha'(u) = \alpha(u)$ *for* $u \epsilon S - \{s\}$ and $\alpha'(s) = \alpha(s) + \Sigma_{v \epsilon T} \alpha(v)$.

The following algorithm, Algorithm *SC*, can be used to reduce the problem of finding $C^*(G, \alpha)$ to the problem of finding $C^*(B, \alpha')$ where B is a single block of G. Given any graph G one finds $C^*(G)$ by setting $\alpha(v) = 1$ for every vertex v.

STEP 1: If G has only one vertex, stop; that vertex is $C^*(G, \alpha)$.

STEP 2: If G has no bridges, then go to Step 3. Otherwise, select any bridge st. Let S and T denote the components of $G-st$ with $s \epsilon S$ and $t \epsilon T$. Compute $\sigma = \Sigma_{u \epsilon S} \alpha(u)$ and $\tau = \Sigma_{u \epsilon T} \alpha(u)$. If $\sigma = \tau$, stop; $C^*(G, \alpha) = \{s, t\}$. If $\sigma \neq \tau$, then one can assume $\sigma > \tau$. Modify G by deleting T and replacing $\alpha(s)$ by $\alpha(s) + \tau$. Go to Step 1.

STEP 3. If G has no cutpoints, stop; G is a block. Otherwise, select any cutpoint s. Let $S_1, S_2, \ldots, S_k$ be the vertex sets of the components of $G-s$, let $\sigma_i = \Sigma_{u \epsilon S_i} \alpha(u)$, and assume $\sigma_i \leq \sigma_k (1 \leq i \leq k-1)$. If $\sigma_k < \alpha(s) + \Sigma_{i=1}^{k-1} \sigma_i$, stop; $C^*(G, \alpha) = \{s\}$. Otherwise modify G by deleting each S_i $(1 \leq i \leq k-1)$ and replacing $\alpha(s)$ by $\alpha(s) + \Sigma_{i=1}^{k-1} \sigma_i$. Repeat Step 3.

Note that if G is modified by Step 3, then $S_k \cup \{s\}$ has at least three points and no bridges are created when one deletes each S_i.

If G is a tree then one never enters Step 3. Steps 1 and 2 therefore comprise an algorithm for finding the security center of a tree. This algorithm is similar to one given in [1] by Goldman.

4. Graphs With Edges of Arbitrary Lengths

If s is a cutpoint of graph G, then it is easy to see that Lemma 3 holds independent of the lengths of the edges. If $\sigma_k \geq \alpha(s) + \Sigma_{i=1}^{k-1} \sigma_i$, it is also clear that $C^*(G, \alpha)$ is not changed if one alters the length of an edge in $\bar{S}$ (as defined in Theorem 5).

THEOREM 6: *If* e *is an edge of graph* G *which is not in the block that contains* $C^*(G, \alpha)$, *then* $C^*(G, \alpha)$ *will not be changed if the length of* e *is altered.*

COROLLARY 7a: *If* T *is a tree, then the security center (the centroid) is determined independent of the lengths of the edges.*

COROLLARY 7b: *Algorithm* SC *can be used to reduce the problem of finding* $C^*(G, \alpha)$ *to the problem of finding* $C^*(B, \alpha')$, *where* B *is a single block of* G, *without considering the lengths of the edges of* G.

Given (G, α), where G is a block, it is necessary to consider the lengths of the edges of G in order to determine $C^*(G, \alpha)$. For example, consider graph G_7 (fig 7) with $\alpha(u_1) = \alpha(u_2) = \alpha(u_3) = 1$

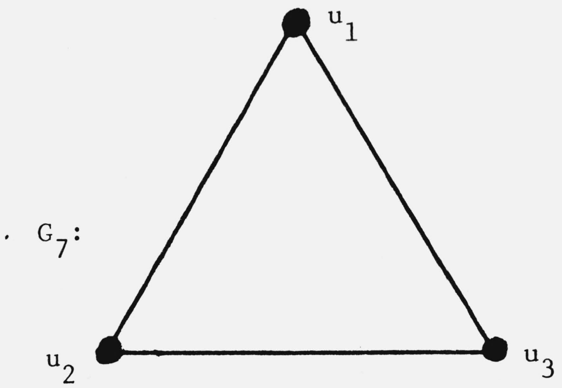


FIGURE 7. *Graph* G_7.

and $L(u_3u_1)=L(u_3u_2)=1$. If $L(u_2u_1)=1$, then $C^*(G, \alpha)=\{u_1, u_2, u_3\}$; if $L(u_2u_1)=2$, then $C^*(G,\alpha)=\{u_3\}$.

5. References

[1] Goldman, A. J., Optimal center location in Simple Networks, Transportation Science **5**, 212–221, (1971).
[2] Goldman, A. J., Minimax location of a facility in a network, Transportation Science **6**, 407–418, (1972).
[3] Hakimi, S. L., Optimum locations of switching centers and the absolute centers and medians of a graph, Operations Research **12**, 450–459 (1964).
[4] Harary, F., Graph Theory (Addison-Wesley, Reading, Mass., 1969).
[5] Jordan, C., Sur les Assemblages de Lignes, J. Reine Angew. Math. **70**, 185–190 (1869).
[6] Zelinka, B., Medians and Peripherians of Trees, Archivum Mathematicum (Brno), 87–95 (1968).

(Paper 79B3&4–426)