OPERATIONS RESEARCH CENTER

working paper

MASSACHUSETTS INSTITUTE OF TECHNOLOGY 1861 MENS ET MANUS

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

OPTIMAL SERVER LOCATION ON A NETWORK OPERATING AS AN M/G/1 QUEUE

by

Oded Berman

Richard C. Larson

OR 100-80 July 1980

ABSTRACT

This paper extends Hakimi's one-median problem by embedding it in a general queueing context. Demands for service that arise solely on the nodes of a network G occur in time as a Poisson process. A single mobile server resides at a facility located on G. The server, when available, is dispatched immediately to any demand that occurs. When a demand finds the server busy with a previous demand, it is either rejected (model 1) or entered into a queue that is depleted in a first-come, first-served manner (model 2). It is desired to locate the facility on G so as to minimize average cost of response, which is either a weighted sum of mean travel time and cost of rejection (model 1) or the sum of mean queueing delay and mean travel time. For model 1, one finds that the optimal location reduces to Hakimi's familiar nodal result. For model 2, nonlinearities in the objective function can yield an optimal solution that is either at a node or on a link. Properties of the objective function for model 2 are utilized to develop efficient finite-step procedures for finding the optimal location.

Ever since Hakimi's work in 1964[1] and 1965,[2] there has been considerable interest in the problem of optimally locating one or more facilities on a network. Consider an undirected network G(N,L), where N is the set of nodes ($|N|=n$) and L is the set of links, having a fraction h_i of all service demands originate at node $i\varepsilon N$. (No demand originates on the links). If $d(x,i)$ is the distance between the facility at $x\varepsilon G$ and node $i\varepsilon N$, then the average travel distance associated with a random service demand is

$$\overline{J}(x) = \sum_{i=1}^{n} h_i \, d(x,i).$$

Hakimi's "1-median" problem is to locate a facility at a point $x^* \varepsilon G$ such that for all $x\varepsilon G$, $\overline{J}(x^*) \leq \overline{J}(x)$. Hakimi showed that an optimal location existed in the node set N, thus reducing a continuous search to a simple finite one. An analogous result regarding nodal locations was given for the multi-median problem.

While the median problem exhibits certain mathematically appealing properties, its implied operational assumptions can be somewhat limiting in practice. In particular, the median problem incorporates only one of two types of probabilistic behaviors often seen in applications: it does include the probabilistic spatial nature of service demands, using h_i as the probability that a random service demand originates at node i; it does not include the probabilistic temporal nature of service demands, which in certain operating systems can result in service demands either being rejected ("lost") or placed in queue due to unavailability of the server associated with the facility. The probability of being rejected or placed in queue is often far from insignificant: if the server is busy servicing demands 50 percent of the time, and if service demands arrive in time in a Poisson manner, then 50 percent of the arriving service demands find the server busy and are either rejected or placed in queue. With the queueing option, the mean in-queue waiting time is often much larger than the mean travel time, the quantity emphasized in the median problem. Thus one is motivated to formulate and analyze location problems in which temporal as well as spatial uncertainties are incorporated.

In this paper we consider two formulations that add temporal uncertainty to the Hakimi model in a general and, we think, natural way. We consider the location on a network of a single facility that garages a mobile server. Service demands occur at nodes in a random (homogeneous Poisson) manner, and in response to each demand, the server (if available) travels to the demand to provide on-scene and perhaps off-scene service. If the server is unavailable at the time of a service demand, the demand is either lost or entered into a queue that is depleted in a first-in, first-out (FIFO) manner. From a queueing point of view, the system is an M/G/1 system (meaning Poisson input, general [independent] service times, and a single server)[3] operating in steady state, with either zero queue capacity (when demands can become lost) or infinite queue capacity.

For the infinite queue capacity case, the objective is to locate the facility so that the sum of the mean in-queue delay and mean travel time is minimized. For the zero queue capacity case, the objective is to minimize an appropriately weighted sum of mean travel time (for those demands that are serviced) and cost of rejection (for those that are lost). For both extremes of queue capacity, we find the optimal location of the facility. For the case of zero queue capacity, we find that the optimal facility location reduces to Hakimi's familiar nodal result. For the case of infinite queue capacity, nonlinearities in the objective function can yield an optimal solution that is either at a node or on a link. Exact finite-step procedures for finding the optimal location are developed.

I. Problem Definition

Let G(N,L) be an undirected network with node set N ($|N|=n$) and link set L. Service demands occur exclusively at the nodes, with each node i generating an independent Poisson stream with rate λh_i ($\sum_{i=1}^{n} h_i = 1$). Travel distance from point

$x\varepsilon G$ to node $i\varepsilon N$ is $d(x,i)$. Travel distance on link (i,j) is d_{ij}. The distance required to travel a fraction θ of link (i,j) is assumed to be θd_{ij}. In all cases travel time is equal to travel distance divided by travel speed v.

A single mobile server is stationed at a facility located at $x\varepsilon G$. The server is free or available whenever it is located at x and immediately ready to service a demand. Given a service demand from node $i\varepsilon N$, and given that the server is free, the server is immediately dispatched to node i, incurring a travel time or travel cost $d(x,i)/v$. At node i there is an on-scene service time R_i, having mean $\overline{R}_i$ and second moment $\overline{R_i^2} < \infty$. Following the on-scene service time, there is an additional travel time $(\beta-1)\ d(x,i)/v$, where $\beta \geq 2$, followed by an additional off-scene service time W_i, having mean $\overline{W}_i$ and second moment $\overline{W_i^2} < \infty$. The total service time associated with a serviced demand from node i is

$$S_i = d(x,i)/v + R_i + (\beta-1)\ d(x,i)/v + W_i = \beta\ d(x,i)/v + R_i + W_i \qquad (1)$$

The server is busy during any of the four phases of service (see Figure 1). Whenever a demand is generated and the server is busy servicing a previous demand, the new demand is either lost (which usually implies service by a back-up service system), incurring a travel cost $\gamma > 0$, or it is entered into a queue that is depleted in a FIFO manner.

As an example, if $\beta=2$ the model could represent an ambulance garaged at a hospital located at $x\varepsilon G$; $d(x,i)/v$ is the travel time to a patient at node i; R_i is the time to stabilize the patient and place him (her) in the ambulance; $d(i,x)/v = d(x,i)/v$ is the travel time back to the hospital; and W_i is the time to deliver the patient to physicians and to prepare the ambulance for the next service demand. If the system has zero queue capacity, here γ might represent the travel time required for a back-up server (perhaps in an adjacent

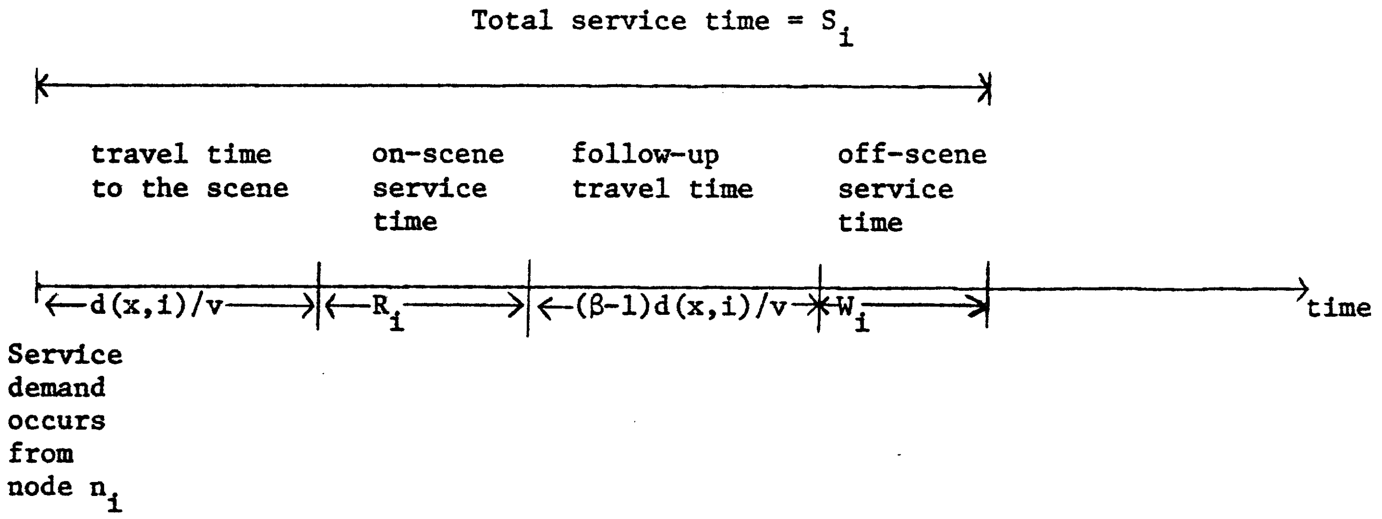

Figure 1: Time Sequence for a Demand for Service

community) to reach any patient who demanded service while the primary ambulance was busy. Values of β greater than 2 could result if speed back to the hospital were necessarily slower than the rapid speed of initial response and/or if the return route followed other than a minimum distance path, but a path proportional in length to d(x,i).

To simplify notation, we define α and σ^2_{R+W} as the mean and variance, respectively, of the nontravel-related service time. Clearly,

$$\alpha = \sum_{i=1}^{n} h_i \, (\overline{R}_i + \overline{W}_i) \tag{2}$$

$$\sigma^2_{R+W} = \sum_{j=1}^{n} h_j \, (\overline{R_j^2} + \overline{W_j^2}) - (\sum_{j=1}^{n} h_j \overline{R}_j)^2 - (\sum_{j=1}^{n} h_j \overline{W}_j)^2 . \tag{3}$$

In the following we utilize Little's queueing formula[4], which when applied to a single server, states that $\overline{N}_c = \lambda' \overline{S}$, where

$\overline{N}_c \equiv$ average number of customers (i.e., service demands) being served by the server at a random time

$\overline{S} \equiv$ average service time

$\lambda' \equiv$ time-average rate at which potential customers are accepted into service (λ' excludes rejected customers).

Since only 0 or 1 customer can be with the server at any time, $\overline{N}_c = \rho =$ fraction of time that the server is busy = system utilization factor < 1. Hence,

$$\rho = \lambda' \overline{S}. \tag{4}$$

II. The Case of Lost Demands (Model 1)

We consider first the relatively easy situation in which no queueing is allowed. Define

$\rho(x)$ = average fraction of time that the server is busy, given that it is located at $x \varepsilon G$ when free.

Since demands are Poisson, a fraction $(1-\rho(x))$ of demands find the server free and are thus serviced by the server, and a fraction $\rho(x)$ find it busy and are thus lost, incurring a cost $\gamma > 0$. The expected cost of travel for a random demand is

$$\overline{J}(x) = (1-\rho(x)) \sum_{i=1}^{n} h_i \, d(x,i)/v + \rho(x) \, \gamma \tag{5}$$

We wish to find $x^* \varepsilon G$ such that for all $x \varepsilon G$, $\overline{J}(x^*) \leq \overline{J}(x)$. The location x^* could be called a stochastic loss median. The term "loss" is appropriate since the service system is an M/G/1 loss queue, i.e., customers who arrive when the server is busy are lost and handled by a back-up system.

Theorem 1 There exists at least one node of G which is a stochastic loss median, and that node corresponds to the Hakimi median.

Proof Applying Little's formula to the server located at $x \varepsilon G$, $\rho(x) = \lambda'(x) \, \overline{S}(x)$, where $\lambda'(x)$ = average rate at which the server accepts service demands and $\overline{S}(x)$ = expected total service time of a random serviced demand. Due to Poisson arrivals, $\lambda'(x) = \lambda(1-\rho(x))$. Hence, $\rho(x) = \lambda\overline{S}(x)/[1+\lambda\overline{S}(x)]$. Now, $\overline{S}(x) = \beta\ \overline{t}(x) + \alpha$, where $\overline{t}(x) = \frac{1}{v} \sum_{i=1}^{n} h_i\ d(x,i)$ = average travel time to a random service demand and $\alpha > 0$ is given in (2). Simple substitution into (5) yields $\overline{J}(x) = [\lambda\gamma\alpha + \overline{t}(x)\ (1 + \lambda\gamma\beta)]/[1 + \alpha\gamma + \overline{t}(x)\beta\lambda]$. It is easily verified that $\partial\overline{J}(x)/\partial(\overline{t}(x)) > 0$ for all $\overline{t}(x) \geq 0$ and thus $\overline{J}(x)$ increases strictly monotonically with $\overline{t}(x)$. Hence $\overline{J}(x)$ is minimized by minimizing $\overline{t}(x)$. But by Hakimi's proof [1], $\overline{t}(x)$ is minimized at a node and that node is the Hakimi median. ■

III. The Case of Queued Demands (Model 2)

We now consider the more difficult case in which demands that occur when the server is busy are entered into a queue that is depleted in a FIFO manner. We use the same notation as in Sections I and II with the additional convention that the facility is assumed to be located on a link connecting nodes a and b at a distance x from node a (Figure 2).

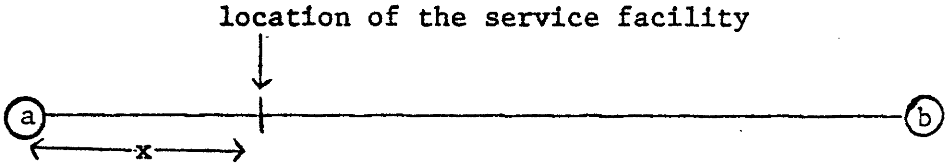

Figure 2: Possible Link Location of the Facility

Let ℓ be the length of link (a,b) and let d(i,j) be the shortest distance between nodes $i,j \in N$. The mean and the variance of the service time are readily computed,

$$E[S(x)] \equiv \overline{S}(x) = \alpha + \frac{\beta}{v} [\sum_{j=1}^{n} h_j \min \{x+d(a,j);\ (\ell-x) + d(b,j)\}] \qquad 6(a)$$

$$\mathrm{VAR}[S(x)] \equiv \sigma^2_{S(x)} = \frac{\beta^2}{v^2} \{ [\sum_{j=1}^{n} h_j \min \{(x+d(a,j))^2; ((\ell-x) + d(b,j))^2] \tag{6(b)}$$

$$- [\sum_{j=1}^{n} h_j \min \{x+d(a,j); (\ell-x) + d(b,j)\}]^2\} + \sigma^2_{R+W}$$

where σ^2_{R+W} is given in (3) and where we have assumed that the travel time and the two nontravel time components of service time are statistically independent. In Equation (6) we have taken into account the fact that, given a service demand from node j, there are two alternatives for the service unit to travel to node j: (i) travel first to node a and then proceed to node j; (ii) travel first to node b and then proceed to node j.

Given facility location x, the expected response time $\overline{T}_R(x)$ associated with a random service demand is the sum of the mean in-queue delay $\overline{W}_q(x)$ and the expected travel time $\overline{t}(x)$. Since the stochastic system is a single server queue having Poisson input and general independent service times (i.e., an M/G/1 queue), it is well known that

$$\overline{W}_q(x) = \begin{cases} \dfrac{\lambda\overline{S}(x)^2 + \lambda\sigma^2_{S(x)}}{2(1 - \lambda\overline{S}(x))} = \dfrac{\lambda\overline{S^2(x)}}{2(1 - \lambda\overline{S}(x))} & \text{for } \lambda\overline{S(x)} < 1 \\ + \infty & \text{for } \lambda\overline{S(x)} \geq 1 \end{cases} \tag{7}$$

Hence, for $\lambda\overline{S(x)} < 1$,

$$\overline{T}_R(x) = \overline{W}_q(x) + \overline{t}_T(x)$$

$$= \frac{\lambda\overline{S}(x)^2 + \sigma^2_{S(x)}}{2(1 - \lambda\overline{S}(x))} + \frac{1}{v} [\sum_{j=1}^{n} h_j \min \{x+d(a,j); (\ell-x) + d(b,j)\}] \tag{8}$$

The objective is to find $x^* \in [a,b]$, $[a,b] \in L$, such that

$$\overline{T}_R(x^*) \leq \overline{T}_R(x) \ \forall\ x \in (a',b'),\ (a',b') \in L \tag{9}$$

Here location x^* could be called a stochastic queue median.

3.1. The Expected Response Time $\overline{T}_R(x)$

We start by simplifying the expression for $\overline{T}_R(x)$ in (8). Let us partition the node set N into two disjoint sets A and B:

$$A = \{j;\ x+d(a,j) \le (\ell-x) + d(b,j)\};\ B = N-A,$$

where x is again the distance of the facility from node a on link (a,b). Using these sets we can rewrite $\overline{S}(x)$ in (2) as

$$\overline{S}(x) = \alpha + \frac{\beta}{v}\left[\sum_{j\varepsilon A} h_j\ (x+d(a,j)) + \sum_{j\varepsilon B} h_j\ ((\ell-x) + d(b,j))\right].$$

In a similar manner we can rewrite $\sigma^2_{S(x)}$ and $\overline{T}_R(x)$. After some algebraic manipulations, $\overline{T}_R(x)$, when finite, can be rewritten as

$$\overline{T}_R(x) = \left[\frac{\lambda[\alpha + \frac{\beta}{v}(C_1 x + C_2)]^2 + \frac{\beta^2\lambda}{v^2}[(C_3 x^2 + C_4 x + C_5 - (C_1 x + C_2)^2] + \lambda\sigma^2_{R+W}}{2\{1-\lambda\ [\alpha + \frac{\beta}{v}(C_1 x + C_2)]\}}\right] + \frac{1}{v}(C_1 x + C_2) \qquad (10)$$

where

$$C_1 = \sum_{j\varepsilon A} h_j - \sum_{j\varepsilon B} h_j$$

$$C_2 = \sum_{j\varepsilon A} h_j\ d(a,j) + \sum_{j\varepsilon B} h_j\ (\ell+d(b,j))$$

$$C_3 = \sum_{j\varepsilon A} h_j + \sum_{j\varepsilon B} h_j = 1$$

$$C_4 = 2\{\sum_{j\varepsilon A} h_j\ d(a,j) - \sum_{j\varepsilon B} h_j\ (\ell+d(b,j))\}$$

$$C_5 = \sum_{j\varepsilon A} h_j\ (d(a,j))^2 + \sum_{j\varepsilon B} h_j\ (\ell+d(b,j))^2$$

Further simplification of (10) yields

$$\overline{T}_R(x) = \frac{a_1 x^2 + a_2 x + a_3}{a_4 x + a_5} \tag{11}$$

where

$$a_1 = (\beta - 2\ C_1{}^2)\ \frac{\lambda\beta}{v^2}$$

$$a_2 = \frac{-2C_1\ \alpha\ \lambda}{v} + \frac{\beta^2\ C_4\ \lambda}{v^2} - \frac{4\ \beta\ \ C_1\ C_2\ \lambda}{v^2} + \frac{2\ C_1}{v} + \frac{2\lambda\alpha\beta C_1}{v}$$

$$a_3 = \frac{\beta\lambda}{v^2}\ \ [\beta\ C_5\ -\ 2\ C_2^2] + \frac{2\ C_2}{v}\ [1 - \lambda\alpha + \lambda\alpha\beta] + \lambda[\alpha^2 + \sigma^2_{R+W}]$$

$$a_4 = \frac{-2\beta\lambda C_1}{v}$$

$$a_5 = 2 - 2\ \lambda\alpha - \frac{2\ \beta\ \lambda\ C_2}{v}$$

Let us observe again the expression for $\overline{T}_R(x)$ in (11). When changing x along the link (a,b) the sets A and B may change and hence the parameters C_1, C_2, C_4, C_5 and consequently the parameters a_1, a_2, a_3, a_4 and a_5 may change. As an example we can refer to the simple network in Figure 3. The numbers near the links are the lengths of the corresponding links. It is easy to verify by inspection

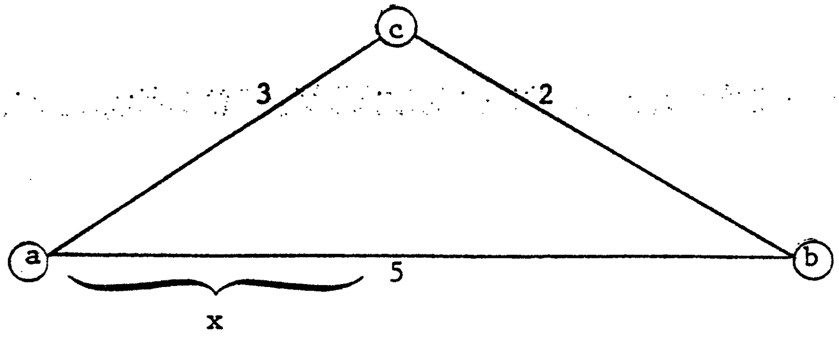

Figure 3: An Example For Showing Changes In The Sets A And B

that as long as $x \leq 2$, $A = \{a,c\}$ and $B = \{b\}$, but when $x > 2$, $A = \{a\}$ and $B = \{b,c\}$.

Let us designate as <u>break points</u> all the points on G(N,L) at which the sets A and B change (e.g. $x = 2$ in Figure 3). We now can state some properties of $\overline{T}_R(x)$.

<u>Property 1.</u> The parameters C_2 and C_5 are non-negative, with $C_5 \geq C_2^2$, since $(C_5 - C_2^2)$ is the variance of the travel time from node <u>a</u> to a random service demand.

<u>Property 2.</u> $a_1 \geq 0$; sgn $(C_1) = -\text{sgn}(a_4)$; $C_1 = 0$ implies $a_4 = 0$.

<u>Property 3.</u> If $\lambda\overline{S}(x) = \rho(x) < 1$, $a_4 x + a_5 > 0$, since $a_4 x + a_5 = 2(1-\rho(x))$.

<u>Property 4.</u> $a_1 x^2 + a_2 x + a_3 > 0$, since x is real and for $\rho(x) < 1$, $a_1 x^2 + a_2 x + a_3 = 2(1-\rho(x))\,\overline{t}(x) + \lambda\,\overline{S^2}(x) > 0$.

<u>Property 5.</u> $a_3 > 0$, since for $\beta \geq 2$ $(\beta C_5 - 2C_2^2) > 0$ [Prop. 1] and the other terms in a_3 are non-negative.

<u>Property 6.</u> As long as $\rho(x) < 1$, $\overline{T}_R(x)$ is a continuous piece-wise differentiable function of x. The only points of nondifferentiability are at the breakpoints (which are finite in number), at which the left and right derivatives exist (and are not equal).

The above properties lead to

<u>Lemma 1.</u> For any interval on link (a, b) on which $\overline{T}_R(x)$ is finite and differentiable with respect to x (but not including the two points that bound the interval), $\overline{T}_R(x)$ is convex.

<u>Proof.</u> If $a_4 = 0$, $\overline{T}_R(x) = a_1 x^2 + a_2 x + a_3$, and since $a_1 \geq 0$ [Prop. 2], $\overline{T}_R(x)$ is clearly convex. If $a_4 \neq 0$, $\overline{T}_R(x)$ can be written as

$$\frac{a_1}{a_4} x + \frac{a_2 a_4 - a_1 a_5}{a_4^2} + \frac{a_3 a_4^2 - a_2 a_4 a_5 + a_1 a_5^2}{a_4^2 (a_4 x + a_5)} .$$

But $a_3 a_4^2 - a_2 a_4 a_5 + a_1 a_5^2 = a_3 [(a_4 - \frac{a_2}{2a_3} a_5)^2 + (\frac{a_1}{a_3} - \frac{a_2^2}{4a_3^2}) a_5^2] > 0$

since [Prop. 5] $a_3 > 0$ and since $(\frac{a_1}{a_3} - \frac{a_2^2}{4a_3^2}) = \frac{4a_1 a_3 - a_2^2}{4a_3^2} > 0$

because [Prop. 4] $a_1 x^2 + a_2 x + a_3$ has no real roots. Hence $\overline{T}_R(x)$ is a sum of convex functions and is therefore convex. ■

The conclusion of lemma 1 is that given any interval $[x_1,x_2]$ where x_1 and x_2 are adjacent breakpoints [$\overline{T}_R(x)$ is finite and differentiable on (x_1,x_2)], if the right derivative of $\overline{T}_R(x)$ at $x = x_1$ is negative and the left derivative at $x = x_2$ is positive then $\overline{T}_R(x)$ has a local minimum over (x_1, x_2); otherwise the minimum of $\overline{T}_R(x)$ over $[x_1,x_2]$ is either at x_1 or at x_2. One minor complication involves the possibility that $\overline{T}_R(x) = +\infty$ for some or all $x \varepsilon [x_1,x_2]$, where x_1 and x_2 are adjacent breakpoints. Recall that $\overline{T}_R(x) = +\infty$ only if $\lambda \overline{S}(x) \geq 1$. Concavity of $\overline{S}(x)$ along a link implies that the set $\{x \varepsilon [x_1,x_2]: \lambda \overline{S}(x) \geq 1\}$ is compact and contains either x_1 or x_2 or both. These are all key facts for the algorithm given on Section 3.4.

3.2. Finding a Local Optimum

When $\overline{T}_R(x)$ has a local minimum over an interval (x_1,x_2), that minimum can be calculated analytically. There are two cases:

Case 1: $C_1 \neq 0$. Then

$$x_{min} = \frac{- b_2 + \sqrt{b_2^2 - 4b_1 b_3}}{2b_1} \qquad (12)(a)$$

where

$$b_1 = a_1 a_4$$

$$b_2 = 2a_1 a_5$$

$$b_3 = a_2 a_5 - a_3 a_4$$

Case 2: $C_1 = 0$. Then

$$x_{min} = - a_2/2a_1 \qquad (12)(b)$$

3.3 Finding the Breakpoints

The algorithm to be presented in Section 3.4 requires identification of all the breakpoints for each link (a, b) ϵL. If we consider again the mobile server located a distance x from node a on link (a, b) and a service demand at node $j \epsilon N$, obviously the server will travel to node j via node a as long as

$$x < \frac{d(b,j) - d(a,j) + \ell}{2} \qquad (13)$$

A breakpoint occurs at that value of x for which (13) becomes an equality.

We now describe a method to identify all the breakpoints for some link (a, b) ϵL.

Step 1. For each $j \epsilon N$ calculate

$$c(j) = \frac{d(b,j) - d(a,j) + \ell}{2}$$

Step 2. Sort in ascending order the vector $c \equiv (c(1), c(2),\ldots,c(n))$. Call the sorted vector cc.

Step 3. The set of all breakpoints, denoted BP (ordered by their distance from node a), is the set composed of all the distinct components of the vector cc. [If the triangle inequality holds, BP will always include 0 and ℓ].

As an example we can use the method above for link (2,3) in Figure 4 (the numbers near the links are the link lengths). Here (a,b) = (2,3) and $\ell = 3$. Following Step 1 we obtain c = (2,3,0,2,0), so cc = (0,0,2,2,3),

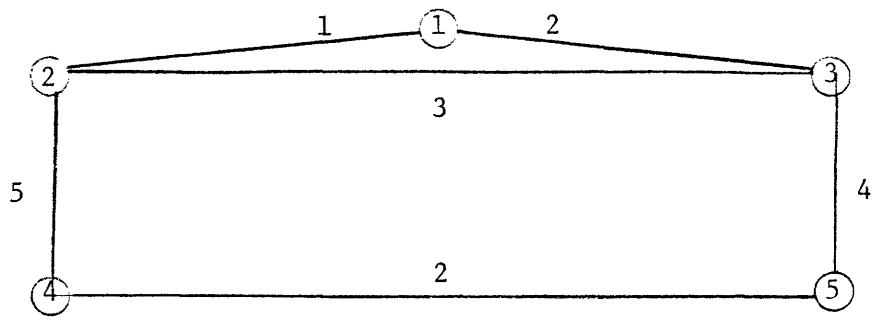

Figure 4: An Example for Finding Breakpoints

and the set of all breakpoints is BP = { 0,2,3 }.

3.4. An Algorithm for Finding the Optimal Location

Building on the local convexity of $\overline{T}_R(x)$ and the method for finding breakpoints, we are now ready to specify a finite-step algorithm for finding the optimal location x^*. For any differentiable function f(x) define the right derivative of f(x) as

$$\dot{f}_{\leftarrow}(x) \equiv \lim_{\Delta x \to 0} \frac{f(x) - f(x + |\Delta x|)}{|\Delta x|}$$

and the left derivative of f(x) as

$$\dot{f}_{\rightarrow}(x) \equiv \lim_{\Delta x \to 0} \frac{f(x - |\Delta x|) - f(x)}{|\Delta x|} \quad .$$

In the following algorithm, $\overline{T}_R^*$ is a running value for minimum mean response time, and $(a,b)^*$ and x^* denote the link and location on the link that yield that value. The algorithm is as follows:

Step 1. Set $\overline{T}_R^* = M$ (M very large)

Step 2. Take any link $(a,b) \varepsilon L$ and calculate the set of all breakpoints. Say that the power of this set BP is m+1, so that there are m intervals in which $\overline{T}_R(x)$ is differentiable.

Step 3. Set $I = 1$.

Step 4. Set $y = I^{th}$ entry in BP

Set $z = I + 1^{st}$ entry in BP

Calculate $\overline{T}_R(y)$, $\overline{T}_R(z)$, $\dot{\overline{T}}_{R\leftarrow}(y)$, $\dot{\overline{T}}_{R\rightarrow}(z)$.

If $\overline{T}_R(y) = +\infty$ and $\overline{T}_R(z) = +\infty$, $I \leftarrow I + 1$ and return to the beginning of Step 4.

If $\overline{T}_R(y) = +\infty$ and $\overline{T}_{R\rightarrow}(z) > 0$, Go to Step 5.

If $\overline{T}_{R\leftarrow}(y) < 0$ and $\overline{T}_R(z) = +\infty$, go to Step 5.

If $\dot{\overline{T}}_{R\leftarrow}(y) < 0$ and $\dot{\overline{T}}_{R\rightarrow}(z) > 0$, go to Step 5.

Otherwise compare $\overline{T}_R(y)$ and $\overline{T}_R(z)$ to $\overline{T}_R^*$. If either $\overline{T}_R(y)$ or $\overline{T}_R(z)$ is less than $\overline{T}_R^*$, update $\overline{T}_R^*$ with new minimum and set $x^* = y$ or z (whichever yields the lower $\overline{T}_R$) and $(a,b)^* = (a,b)$.

Step 5. Calculate the local minimum x_{min} of $\overline{T}_R$ over (y,z) using Equation (12). If $\overline{T}_R(x_{min}) < \overline{T}_R^\circ$ update $\overline{T}_R^\circ$ and record new incumbent $x^* = x_{min}$, $(a,b)^* = (a,b)$.

Step 6. If $I < m$, $I \leftarrow I + 1$ and go to Step 4. Otherwise remove (a,b) from L; if there are links remaining in L go to Step 2. Otherwise FINISH. The optimal location is x^* on link $(a,b)^*$, yielding a minimum mean travel time $\overline{T}_R(x^*)$.

4. Trajectory of the Optimal Location as a function of λ

In this section we examine how x^* varies as we vary the total demand rate λ continuously from 0 to a maximum possible value. The properties of this trajectory of optimal locations can be used to make the algorithm for finding x^* much more efficient.

Lemma 2. (a) when $\lambda = 0+$ x^* is the Hakimi median of $G(N,L)$.

(b) when $\lambda \to \lambda_{\max}$, $x^* \to$ the Hakimi median of $G(N,L)$, where $\lambda_{\max}$ is such that for some $x \varepsilon G(N,L)$, $\lambda\overline{S}(x) = \rho = 1$, and for all $x' \varepsilon G(N,L)$, $\lambda\overline{S}(x') \geq 1$. (i.e., $\lambda_{\max}$ is the smallest value of λ for which the queue explodes for all possible server locations)

Proof (a) when $\lambda = 0+$, $\overline{W}_q = 0$, so that $\overline{T}_R$ is the expected travel time to a random service demand, given by the weighted sum in (8), which is the objective function to the Hakimi median problem. Thus $x^*|_{\lambda = 0+}$ = median of $G(N,L)$.

(b) $\lambda_{\max}$ is the largest λ such that $\exists x \varepsilon G(N,L)$ such that for this x, call it x°, $\lambda_{\max}\,\overline{S}(x^\circ) = 1$. Regardless of server location, any higher values of λ would yield $\rho \geq 1$. By definition of $\lambda_{\max}$, for any $\lambda = \lambda_{\max} - \varepsilon$ $(\varepsilon > 0)$, $\exists x^* \varepsilon G(N,L)$ with $\lambda\overline{S}(x^*) < 1$ and thus $\overline{T}_R(x^*) < \infty$. It is sufficient to show that for ε small, x^* = Hakimi median of $G(N,L)$. But minimization of $\overline{T}_R(x)$ for values of λ near $\lambda_{\max}$ $(\lambda < \lambda_{\max})$ corresponds to maximization of the (positive) denominator of (11) (which equals $a_4\,x + a_5 = 2[1 - \lambda\overline{S}(x)]$), or equivalently to the minimization of $\overline{S}(x)$. But

$$\overline{S}(x) = \alpha + \frac{\beta}{v} \min_{\substack{(a,b)\varepsilon L \\ x\varepsilon(a,b)}} \sum_{j=1}^{m} h_j \{x+d(a,j);\ (\ell-x) + d(b,j)\},$$

which is minimized at x^* = Hakimi median of $G(N,L)$. ■

This lemma says that the trajectory of the optimal location $x^*(\lambda)$ starts at the median when $\lambda = 0$ and eventually returns to the median as λ approaches λ_{max}. Examining again the expression for $\overline{T}_R(x)$ in (8), we have seen that mean travel time $\overline{t}(x)$ dominates the solution for low values of λ and the denominator of (7) dominates for high values of λ.

Both intuition and computational experience have verified that for intermediate values of λ the numerator of $\overline{W}_q(x)$, which equals $\lambda\overline{S}(x)^2 + \lambda\sigma_S^2(x) = \lambda\overline{S^2}(x)$, can play a dominant role in determining x^*. In other words, the second moment of the service time becomes an important factor for intermediate λ values, whereas the mean service time is much more important for extreme λ values.

While we will formally investigate properties of $\overline{S^2}(x)$ in the next section, it is instructive here to illustrate typical trajectories of x^*. Example a utilizes the network presented earlier in Figure 4 with $h_1 = 0.1$, $h_2 = 0.35$, $h_3 = 0.1$, $h_4 = 0.35$, $h_5 = 0.1$, $\alpha = v = 1$, $\sigma_R^2 = \sigma_W^2 = 0$. For each possible nodal location of the facility, the associated expected travel time and second moment of the service time is shown in Table 1.

Node i (Location of facility)	1	2	3	4	5
$\overline{t}$	3.25	2.85	3.75	3.15	4.15
$E[S^2]$	81.7	71	87	79	112.6

Table 1. Expected Travel Times and Second Moments for Example a

The computed trajectory of optimal facility locations is shown in Table 2. In this example, $x^*(\lambda)$ starts at the median for small λ and then moves continuously toward node 4 on link (2,4) [$\overline{S^2}$ becomes smaller as one moves away from node 2 in

(2.4), although at some intermediate point it begins increasing again]. The maximum value for $x^*(\lambda)$ in (2.4) is approximately $x^*(0.07) \simeq 1.63$, and for λ values greater than 0.07 $x^*(\lambda)$ moves continuously back toward to the median along the same path.

λ	Optimal Location, $x^*(\lambda)$	$\overline{T}_R$
0	Node 2	2.85
0.01	Node 2	3.23
0.02	x = 0.8871 on (a,b) = (2,4)	3.63
0.03	x = 1.286 on (a,b) = (2,4)	4.05
0.04	x = 1.471 on (a,b) = (2,4)	4.555
0.05	x = 1.568 on (a,b) = (2,4)	5.153
0.06	x = 1.614 on (a,b) = (2,4)	5.893
0.07	x = 1.627 on (a,b) = (2,4)	6.838
0.08	x = 1.609 on (a,b) = (2,4)	8.086
0.09	x = 1.557 on (a,b) = (2,4)	9.809
0.10	x = 1.457 on (a,b) = (2,4)	12.332
0.11	x = 1.278 on (a,b) = (2,4)	16.344
0.12	x = 0.934 on (a,b) = (2,4)	23.525
0.13	x = 0.172 on (a,b) = (2,4)	38.551
0.14	Node 2	83.011

Table 2 Trajectory of Optimal Facility Locations for Example a

Example b utilizes the same network as Example a with only the h_j's changed: $h_1 = 0.35$, $h_2 = 0.1$, $h_3 = 0.3$, $h_4 = 0.125$ and $h_5 = 0.125$. Table 3 contains the expected travel times and second moments of service times for the five possible nodal facility locations. The computed optimal trajectory $x^*(\lambda)$ is shown in Table 4.

Node i (Location of Facility)	1	2	3	4	5
$\bar{t}_r$	2.2	2.75	2.25	4.65	4.25
$E[S^2]$	51	61.2	45.2	125.2	109.2

Table 3. Expected Travel Times and Second Moments for Example b

λ	Optimal Location, $x^{*}(\lambda)$	T_R
0.01	Node 1	2.464
0.015	Node 1	2.616
0.02	Node 3	2.757
0.05	Node 3	3.808
0.06	Node 3	4.273
0.08	Node 3	5.478
0.11	Node 3	8.543
0.13	Node 3	12.558
0.15	Node 3	21.621
0.158	Node 3	29.508
0.160	Node 1	32.2

Table 4 Trajectory of Optimal Facility Locations for Example *b*

As indicated in Table 4, $x^{*}(\lambda)$ starts at the median (node 1), then jumps to node 3 for intermediate values of λ then jumps back to node 1 for λ near λ_{max}.

Examples *a* and *b* are typical of our computational experience: either continuous movement of x^{*} along a link or discontinuous jumps from node to node. We have also generated examples having both features: a discontinuous jump to another node, followed by continuous movement away from that node along an adjoining link; in such a case, x^{*} reaches a maximum value along the link, then moves continuously back to the node, then discontinuously back along the earlier node-to-node path, eventually returning to the median for λ near λ_{max}. Computationally we have observed that (1) the trajectory of the optimal solution is unique in the sense that the optimal solution moves to a certain point and returns in exactly the same way; (2) the trajectory away from the median always goes through nodes with decreasing second moments of the service time. In the next section we use these observations to develop an efficient heuristic to solve the problem.

5. A Heuristic for Finding the Optimal Location

The heuristic we outline here has one major advantage over the exact algorithm presented in Section 3.4: with the heuristic we do not have to consider all the links of the network but only those links that lie on an "assumed feasible trajectory" of the optimal solution. We note that in all the numerical examples we have examined so far, the solution obtained by the heuristic and the optimal solution obtained by the exact algorithm are identical.

Before presenting the heuristic, it is useful to note some relationships pertaining to the computation of $\overline{S(x)^2}$. We can simplify the expression for $\overline{S(x)^2}$ given in the numerator of (10) as follows:

$$\overline{S(x)^2} = \frac{\beta^2}{v^2} x^2 + [\frac{\beta^2 C_4}{v^2} + \frac{2\beta\alpha C_1}{v}] x + [\frac{\beta^2 C_5}{v^2} + \frac{2\beta\alpha C_2}{v} + \alpha^2 + \sigma^2_{R+W}] \qquad (14)$$

For x = 0, or equivalently, for the facility at node $\underline{a}$ εN,

$$\overline{S(0)^2} = [\frac{\beta^2 C_5}{v^2} + \frac{2\beta\alpha C_2}{v} + \alpha^2 + \sigma^2_{R+W}] \qquad (15)$$

Also for x = 0, we have

$$C_2 = \sum_{j=1}^{m} h_j \, d(a,j) \qquad \text{(a)}$$

$$C_5 \quad \sum_{j=1}^{m} h_j \, d(a,j)^2 \qquad \text{(b)} \qquad (16)$$

Hence, for x = 0, C_2 and C_5 are respectively the expectation and second moment of the travel time from node a. When it exists, the derivative of $\overline{S(x)^2}$ with respect to x is readily computed,

$$\frac{d\overline{S(x)^2}}{dx} = \frac{2\beta^2 x}{v^2} + \frac{\beta^2 C_4}{v^2} + \frac{2\beta\alpha C_1}{v} . \tag{17}$$

The heuristic is as follows:

Step 1 Start at the Hakimi median of G(N,L). Using (15) calculate the second moment of the service time at the median, denoting it σ and labelling the median.

Step 2. For all unlabelled nodes i connected directly by a link to a labelled node, compute $\overline{S(i)^2}$ (i.e., the second moment of the service time evaluated at node i). If $\overline{S(i)^2} > \sigma \ \forall$ i go to Step 3. If $\exists i^*$ with $\overline{S(i^*)^2} \leq \sigma$ label node i^*, set $\sigma = \overline{S(i^*)^2}$ and repeat Step 2.

Step 3. Call the last labelled node i^*. Examine the set NL of all the unlabelled nodes i connected directly by a link to node i^*. Apply the exact algorithm of Section 3.4 to the sub-network that includes: all links in the path to i^* that goes through labelled nodes; all the nodes in the set NL and all the links that connect directly the nodes of NL with the last labelled node.

As an example of the heuristic let us consider again Example b. Inspection of Table 3 implies that the sub-network for the heuristic is that shown in Figure 5. Hence, the exact algorithm need be applied only to this 2-link, 3-node sub-network.

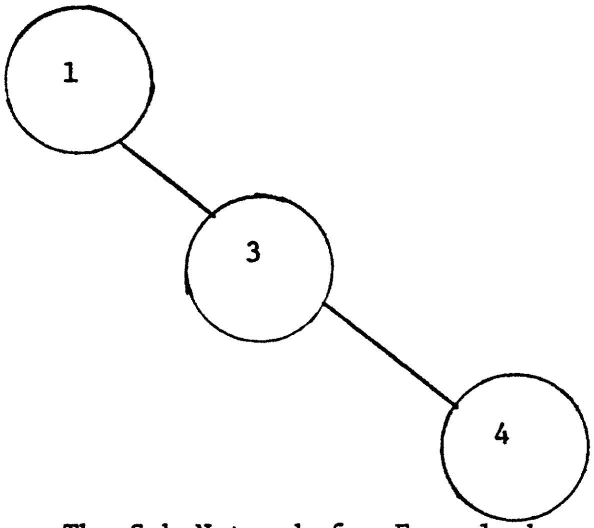

Figure 5. The Sub-Network for Example b

6. An Efficient Exact Algorithm for a Tree Network

In this section we show that when the network is a tree, a modified version of the heuristic of the previous section is in fact an exact algorithm.

We start with very simple

Lemma 3. When G(N,L) is a tree, for each link (a,b) there are only two breakpoints which are a and b (or, equivalently, $x = 0$ and $x = \ell$).

The proof is trivial, but the implication is that breakpoints do not have to be calculated and the parameters C_1, C_2, C_4 and C_5 remain constant for all $x \varepsilon$ (a,b).

The basis for the efficient tree algorithm is given by

Theorem 2. Suppose i and j are two nodes connected directly by a link, and suppose $\overline{S(i)^2} < \overline{S(j)^2}$. Then $\forall\, x \;\varepsilon\; (j,k),\; k \neq i$, where x is a point on link (j,k) a distance x from node j, $\overline{S(j)^2} < \overline{S(x)^2}$.

Proof. Letting d_{ij} be the length of link (i,j) and using (14), (15) and Lemma 3, $\overline{S(i)^2} < \overline{S(j)^2}$ implies

$$\frac{\beta^2 C_5'}{v^2} + \frac{2\beta\alpha C_2'}{v} + \alpha^2 + \sigma^2_{R+W} < \frac{\beta^2}{v^2}\, d_{ij}^{\;2} + \left[\frac{\beta^2 C_4'}{v^2} + \frac{2\beta\alpha C_1'}{v}\right] d_{ij}$$
$$+ \left[\frac{\beta^2 C_5'}{v^2} + \frac{2\beta\alpha C_2'}{v} + \alpha^2 + \sigma^2_{R+W}\right]$$

which implies that a test quantity be positive:

$$\frac{\beta d_{ij}}{v^2} + \frac{\beta C_4'}{v^2} + \frac{2\alpha C_1'}{v} > 0,$$

where C_4' and C_1' are the relevant parameters C_4 and C_1 for (i,j). Let x be a point on (j,k), $x \neq j$ and suppose by contradiction $\overline{S(j)^2} \geq \overline{S(x)^2}$. But then in the same manner as above

$$\frac{\beta x}{v^2} + \frac{\beta C''_4}{v^2} + \frac{2\alpha C''_1}{v} > 0,$$

where C'_4 and C'_1 are the relevant parameters C_1 and C_4 for (j,k). Let $A(i,j) = \{\ell \varepsilon N: \ d(i,\ell) + d_{ij} \leq d(j,\ell)\}$ and let $B(i,j) = N - A(i,j)$. Given $G(N,L)$ is a tree, it is easy to verify that for a facility at node $j, A(i,j)$ and $B(i,j)$ are the sets A and B defined in Section 3.1. For link (j,k) with length d_{jk} we define for a facility at node $k, A(j,k) = \{\ell \varepsilon N: \ d(j,\ell) + d_{jk} \leq d(k,\ell\}$ and $B(j,k) = N - A(j,k)$. Recalling (10) we can write

$$C''_4 = 2 \sum_{\ell \varepsilon A(j,k)} h_\ell \, d(j,\ell) - 2 \sum_{\ell \varepsilon B(j,k)} h_\ell \, [d(k,\ell) + d_{jk}].$$

But for a tree $A(j,k) - \{j\} = A(i,j)$ and $B(j,k) \cup \{j\} = B(i,j)$, and for $\ell \varepsilon B(j,k)$, $d(k,\ell) + d_{jk} = d(j,\ell)$, so that $C''_4 = 2[\sum_{\ell \varepsilon A(i,j)} h_\ell \, d(j,\ell) + h_j \, d(j,j)]$ $-2[\sum_{\ell \varepsilon B(i,j)} h_\ell \, d(j,\ell) - h_j \, d(j,j)]$. Also for $\ell \varepsilon A(i,j)$, $d(j,\ell) = d_{ij} + d(i,\ell)$ so we can write $C''_4 = C'_4 + 2 \sum_{\ell \varepsilon A(i,j)} h_\ell \, d_{ij} + 2 \sum_{\ell \varepsilon B(i,j)} h_\ell \, d_{ij} = C'_4 + 2 \, d_{ij}$.

Also, $C''_1 = \sum_{\ell \varepsilon A(j,k)} h_\ell - \sum_{\ell \varepsilon B(j,k)} h_\ell = \sum_{\ell \varepsilon A(i,j)} h_\ell + h_j - (\sum_{\ell \varepsilon B(i,j)} h_\ell - h_j)$

$= C'_1 + 2h_j$.

Therefore the test quantity can be written

$$\frac{\beta x}{v^2} + \frac{\beta C''_4}{v^2} + \frac{2\alpha C''_1}{v} = \frac{\beta x}{v^2} + \frac{\beta(C''_4 + 2d_{ij})}{v^2} + \frac{2\alpha(C'_1 + 2h_j)}{v} =$$

$$\left(\frac{\beta x}{v^2} + \frac{\beta d_{ij}}{v^2} + \frac{4\alpha h_j}{v}\right) + \left(\frac{\beta d_{ij}}{v^2} + \frac{\beta C'_4}{v^2} + \frac{2\alpha C'_1}{v}\right),$$

which must be positive since the first expression in parameters contains only positive quantities and the second is a test quantity already proved positive. But this is a contradiction to $\overline{S(j)^2} \geq \overline{S(x)^2}$. ■

This theorem provides us with valuable information about the trajectory of optimal facility locations on a tree. For any two nodes i,j of link (i,j) such that $\overline{S(i)^2} < \overline{S(j)^2}$, any trajectory that enters j with increasing λ must exit j along link (i,j) toward node i.

Thus the heuristic presented in the previous section is an exact algorithm for the tree. In other words, a sub-tree containing the exact trajectory of optimal facility locations is obtained. Step 3 of the heuristic (now the algorithm) can be modified:

> Call the last labelled node i^*. Examine the set NL of all the unlabelled nodes i connected directly by a link to node i^*. Compute the test quantity $(\frac{\beta^2 C_4}{v^2} + \frac{2\beta\alpha C_{\bar{1}}}{v})$, which is the right derivative of $\overline{S(x)^2}$ evaluated at node i (or x = 0) [Eq. (17)]. If the test quantity is positive remove node i from the set NL. If the quantity is negative label node i and apply the algorithm of Section 3.4 to the path that starts at the median and goes through all the labelled nodes.

Let us apply this new algorithm to the single tree shown in Figure 6, where the numbers near the nodes are the weights (h_j) and the numbers near the links are lengths. The expected travel times and second moments of the service times for each possible nodal location are shown in Table 5. ($v = \alpha = 1$; $\sigma^2_{R+W} = 0$).

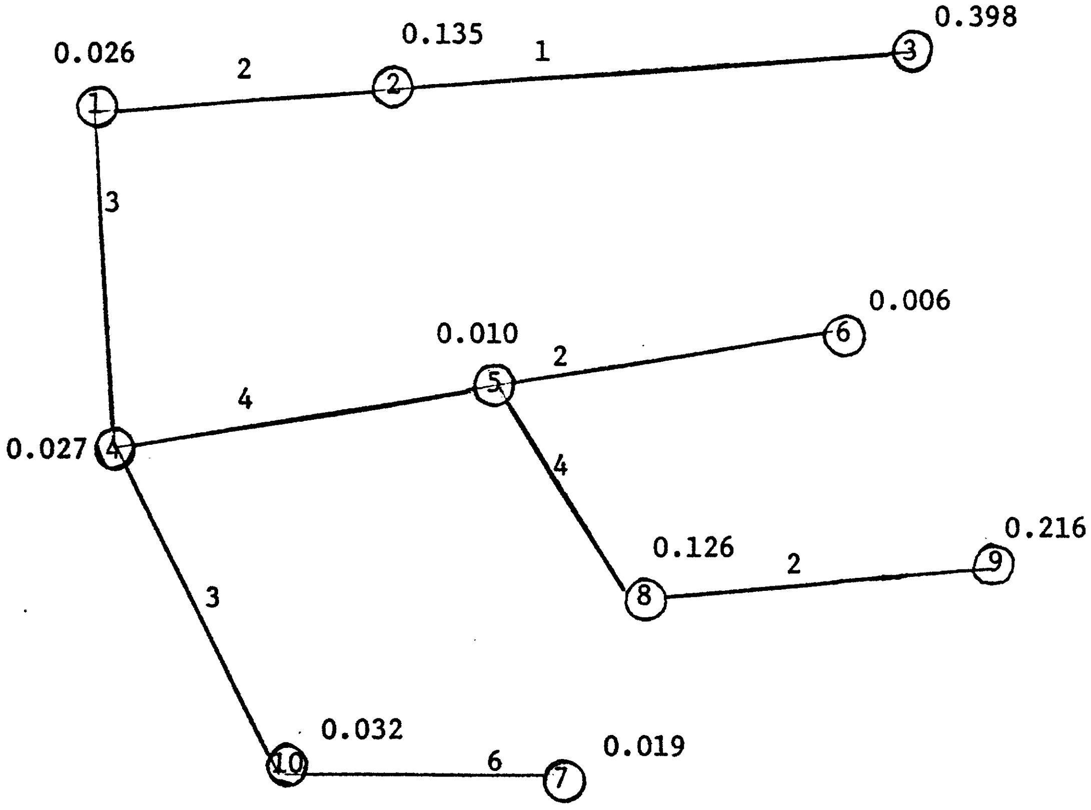

Figure 6. A Tree Example

Node i (Location of Facility)	1	2	3	4	5	6	7	8	9	10
$\overline{t}_T$	6.30	6.17	6.37	6.67	7.80	9.77	15.13	9.05	10.19	9.36
$E[S^2]$	67.50	84.86	96.81	56.62	75.10	112.18	256.45	125.92	163.84	105.108

Table 5. Expected Travel Times and Second Moments for Tree Example

The algorithm operates as follows:

Step 1. The median is node 2. $\overline{S(2)^2} = 84.86 \equiv \sigma$. Node 2 is labelled.

Step 2. $\overline{S(3)^2} = 96.81$; $\overline{S(1)^2} = 67.50$. Node 1 is labelled and $\sigma = 67.50$.

Step 2. $\overline{S(4)^2} = 56.62$. Node 4 is labelled and $\sigma = 56.62$.

Step 2. $\overline{S(10)^2} = 105.108$. $\overline{S(5)^2} = 75.10$.

Step 3. For link (4,5) $c_1 = 0.277$, $c_4 = 3.192$ so that

$(\frac{\beta^2 C_4}{v^2} + \frac{2\beta\alpha C_1}{v}) > 0$. For link (4,10) $C_1 = 0.888$, $C_4 = 12.54$ and

again

$$(\frac{\beta^2 C_4}{v^2} + \frac{2\beta\alpha C_1}{v}) > 0.$$

Therefore we need to apply the algorithm of Section 3.4 only to the path 2-1-4. When $\lambda = 0.01$, for example, the optimal location is $x^* = 0.584$ on link (1,4) and $\overline{T}_R = 7.869$.

References

1. Hakimi, S.L., "Optimum Locations of Switching Centers and the Absolute Centers and Medians of a Graph," Operations Research, 12, 450-459 (1964).

2. Hakimi, S.L., "Optimum Distribution of Switching Centers on a Communications Network and Some Related Graph Theoretic Problems," Operations Research, 13, 462-475 (1965).

3. Kleinrock, L. Queueing Systems I, Theory, Wiley, New York, Chapter 5 (1975).

4. Little, J.D.C., "A Proof of the Queueing Formula L = λW" Operations Research, 9, 383-387 (1961).