# A Maximum Expected Covering Location Model: Formulation, Properties and Heuristic Solution

MARK S. DASKIN

*Department of Civil Engineering and The Transportation Center, Northwestern University, Evanston, Illinois*

*The maximum covering location model has been used extensively in analyzing locations for public service facilities. The model is extended to account for the chance that when a demand arrives at the system it will not be covered since all facilities capable of covering the demand are engaged serving other demands. An integer programming formulation of the new problem is presented. Several properties of the formulation are proven. A heuristic solution algorithm is presented and computational results with the algorithm are discussed. Directions for future study are also discussed.*

## INTRODUCTION

Early papers on location theory, and in fact most of the location literature to date, deal with the problem of locating facilities under deterministic conditions. Examples include the set covering problem,[6] the maximum covering location problem[8] and the $P$-median and $P$-center problems.[14, 15, 19] Much literature has been devoted to the solution of these problems either exactly or in a heuristic manner as well as to the application of these models to real problems.

More recent work has concentrated on location problems in which elements of the problem description are stochastic. MIRCHANDANI AND ODONI[20] consider the $P$-median problem when the travel times on the links are random variables. They demonstrate that using the mean value of the link travel times can result in locations that differ from the locations one would identify if the entire distribution of link travel times is considered. They also show that at least one optimal solution consists of locating the facilities at the nodes of the network. MIRCHANDANI AND OUDJIT[21] review other recent extensions of deterministic location theory



Transportation Science
Vol. 17, No. 1, February 1983

0041-1655/83/1701-0048 $01.25

that consider stochastic problem elements. In their review, as in this paper, we are concerned with location on a network.

Two location problems are explicitly concerned with the notion of coverage: the set covering and the maximum covering location problems. Let $d_{ki}$ be the distance from node $i$ to node $k$. We assume that $d_{ki}$ is known for all $k$, $i$ and is deterministic. In both the set covering location problem and the maximum covering location problem, demands are assumed to be generated at the nodes only, and we say that demands generated at node $k$ are covered by a facility at node $i$ if $d_{ki} \leq D$. The set covering problem is to find the locations of the minimum number of facilities, located on the nodes, which cover all the demands. The maximum covering location problem is to find a set of $M$ nodes which maximize the total demand that is covered. In the maximum covering location problem, nodes are weighted according to the demands generated at the nodes; in the set covering problem all nodes are weighted equally. Several authors have used the set covering problem[4, 17, 22, 24, 25] and the maximum covering problem[3, 11, 12] in locating facilities.

It is worth noting that restricting facility locations to nodes on the network results from a simplifying assumption designed to make these problems computationally feasible. Location on the nodes only, as opposed to locating anywhere along the arcs of the network, may not be optimal. Consider Figure 1 in which all distances are assumed to be symmetric. If the critical distance is 5 units, the optimal solution to the set covering problem is to locate two facilities: one at each node. The solution to the maximum covering problem, if we are to locate only one facility, is to locate at node 1. However, if we relax the assumption that facilities must be located at the nodes, the optimal solution to both problems is to locate a single facility at point $A$, midway between nodes 1 and 2. In short, location on the nodes, which can be shown to be optimal for the $P$-median problem under a broad set of assumptions,[16] is not necessarily optimal for the set covering and maximum covering problems. Restricting the choice of facility sites to the nodes for these problems results either from a more limited problem statement or from the demands of mathematical tractibility. Throughout the remainder of the paper, when discussing these problems, we restrict facility locations to the nodes on the network. We note that the suboptimality introduced by restricting facilities to be located on the nodes of the network may be resolved through the use of network intersect points as suggested by CHURCH AND MEADOWS.[7]

The classical formulations of these problems assume that demands at node $k$ need to be covered by only one facility. Alternatively, these models assume that a facility at node $i$ will be able to respond to all demands at all nodes $k$ such that $d_{ki} \leq D$. However, in many real world

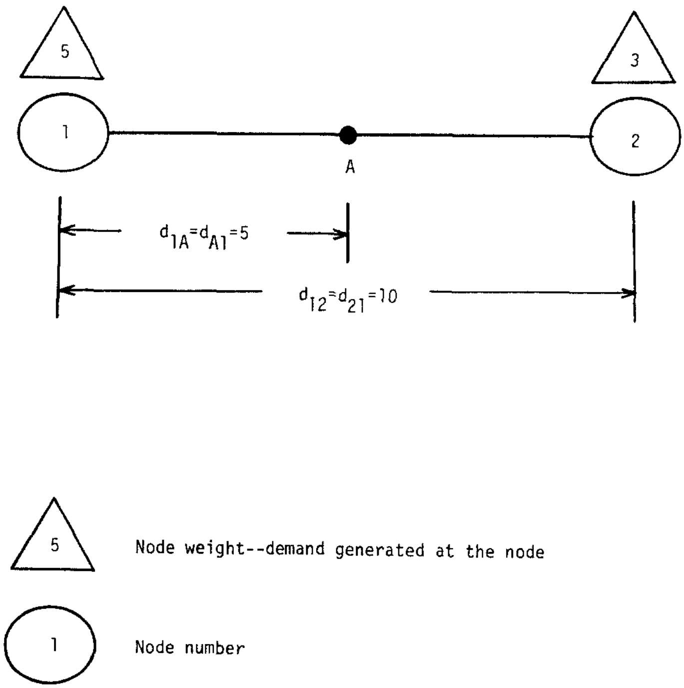


**Fig. 1.** Simple network.

situations, the facility that covers demands at some node $k$ may not be able to respond to demands at that node. For example, in emergency medical service systems, the nearest vehicle to a call for service may be responding to another call. In these cases a more remote vehicle will have to respond to the service request. In Austin, Texas, roughly one-third of the service requests are handled by a vehicle other than the closest unit.[12] SWOVELAND et al.[23] report a similar finding from a simulation of ambulance operations in Vancouver, Canada. In industrial contexts, facilities may be unable to respond to demands due to inclement weather, labor conditions or facility maintenance needs. We would therefore like to have more than one facility capable of covering demands at the nodes, particularly those nodes that generate large numbers of demands.

In a recent paper[10] we addressed this problem in the context of the set covering problem by using a hierarchical objective function. The primary objective was to minimize the number of facilities required to cover all the demands. The second objective was to maximize a measure of multiple coverate. The measure valued covering one node by 5 extra units equally with covering 5 nodes by one extra unit each. BENEDICT[2] has extended these results to the maximum covering problem.

MASOG[18] also attempts to find a set of locations that perform well when only a subset of the facilities are working. He defines a cascade solution, formulates an integer programming problem and extends CHURCH AND REVELLE'S[8] greedy adding and substitution algorithm to solve this problem. KOLESAR AND WALKER[17] develop a set of models to relocate fire companies to ensure adequate coverage when several companies are busy. CHAPMAN AND WHITE[5] present several probabilistic extensions of the set covering and $P$-center problems. Their formulations account for both vehicle availability and stochastic travel times. ALY AND WHITE[1] also examine a probabilistic set covering model.

Our approach, which is similar to that of Chapman and White, is to recognize explicitly that if $M$ facilities are located on the network, not all facilities will be able to respond to demands at all times. We therefore need to compute the probability that a zone will be covered if a particular configuration of facilities is chosen.

In the following section we restate the maximum covering location model and extend it to incorporate the stochastic considerations outlined above. Section 2 develops an example of the use of the new formulation and demonstrates the need to consider the *probability* of a zone being covered. Several model properties are proven in Section 3. Section 4 presents a heuristic algorithm for solving the model of Section 2. Computational experience with the heuristic is discussed in Section 5. In Section 6 we conclude with recommendations for future study.

## 1. MODEL FORMULATION

THE TRADITIONAL maximum covering location model may be formulated as follows:

$$\text{Maximize } \textstyle\sum_k h_k y_k \tag{1}$$

$$\text{Subject to } y_k - \textstyle\sum_i a_{ki} X_i \le 0, \qquad k = 1, \cdots, N \tag{2}$$

$$\textstyle\sum_i X_i \le M, \tag{3}$$

$$\left.\begin{aligned} X_i &= 0, 1, \quad i = 1, \cdots, N \\ y_k &= 0, 1, \quad k = 1, \cdots, N \end{aligned}\right\} \tag{4}$$

where

$h_k$ = demand generated at node $k$

$y_k = \begin{cases} 0 \text{ if node } k \text{ is not covered} \\ 1 \text{ if node } k \text{ is covered} \end{cases}$

$X_i = \begin{cases} 0 \text{ if a facility is not located at node } i \\ 1 \text{ if a facility is located at node } i \end{cases}$

$$a_{ki} = \begin{cases} 0 \text{ if } d_{ki} > D\text{—a facility at } i \text{ does not cover demands at } k \\ 1 \text{ if } d_{ki} \le D\text{—a facility at } i \text{ covers demands at } k \end{cases}$$

$M$ = number of facilities to be located

$N$ = number of nodes in the network.

The objective function (1) maximizes the demand that is covered. Constraint (2) states that node $k$ cannot be covered unless at least one facility is located at one of the nodes $i$ which cover node $k$. Constraint (3) states that at most $M$ facilities are to be located. In general, this constraint will be binding.

We recognize, however, that not all facilities will be able to respond to demands at all times. If a facility is capable of responding to demands, we say the facility is "working," otherwise the facility is said to be "not working" or "broken down." Let $p$ be the probability that a facility is not working. We assume that $p$ is known and the same for all facilities. We further assume that the probability that facility $i$ is working is independent of the probability that facility $j$ is working for all $i \neq j$. While these may at first appear to be restrictive assumptions, they are certainly more realistic than the traditional assumption that all facilities are working at all times. The implications of the assumptions in the model results are discussed in Section 6 and in reference [9].

Under these assumptions the number of working facilities at any time follows a binomial distribution:

Prob ($j$ facilities working out of $M$ located)

$$= \binom{M}{j}(1-p)^j p^{M-j}, \qquad j = 0, 1, \cdots M. \quad (5)$$

We can also compute the probability that node $k$ is covered by a working facility given $m$ facilities cover node $k$. This is simply

Prob (node $k$ covered by a working facility given $m$ facilities are capable of covering the node)

$$\begin{aligned} &= 1 - \text{Prob } (m \text{ facilities are not working}) \\ &= 1 - p^m. \end{aligned} \quad (6)$$

Let $H_{k,m}$ be a random variable equal to the number of demands at node $k$ covered by a working facility given $m$ facilities are capable of covering node $k$. We have

$$H_{k,m} = \begin{matrix} h_k \text{ with probability } 1 - p^m \\ 0 \text{ with probability } p^m \end{matrix} \quad (7)$$

and $$E(H_{k,m}) = h_k(1 - p^m), \forall k, m. \quad (8)$$

The increase in expected coverage at node $k$ that results from increasing the number of facilities that cover node $k$ from $m - 1$ to $m$, for $m = 1, \cdots, M$ is given by

$$\Delta E(H_{k,m}) = E(H_{k,m}) - E(H_{k,m-1}) \qquad (9)$$
$$= h_k p^{m-1}(1-p), \qquad m = 1, 2, \cdots M.$$

Finally, we note that the number of facilities that are capable of covering node $k$ is given by $\sum_i a_{ki}X_i$ in the integer programming problem (1)–(4).

Using the definitions given above we formulate the *maximum expected covering location problem* (MEXCLP) as follows:

$$\text{Maximize } \sum_{k=1}^{N} \sum_{j=1}^{M} (1-p)p^{j-1}h_k y_{jk} = \sum_k \sum_j w_j h_k y_{jk} \qquad (10)$$

$$\text{Subject to } \sum_{j=1}^{M} y_{jk} - \sum_{i=1}^{N} a_{ki}X_i \le 0, \qquad \forall k \qquad (11)$$

$$\sum_{i=1}^{N} X_i \le M \qquad (12)$$

$$X_i = 0, 1, \cdots, M, \ \forall i \qquad (13)$$

$$y_{jk} = 0, 1, \qquad \forall j, k \qquad (14)$$

where

$$y_{jk} = \begin{cases} 1 \text{ if node } k \text{ is covered by at least } j \text{ facilities} \\ 0 \text{ if node } k \text{ is covered by less than } j \text{ facilities} \end{cases}$$

and $X_i$ = number of facilities located at node $i$

$$w_j = (1-p)p^{j-1}, \qquad j = 1, \cdots, M.$$

Note that in this formulation we allow more than one facility to be located at a given node. In Section 2 we demonstrate that doing so may be optimal under certain conditions. We also observe that the objective function is concave in $j$ for each $k$. This implies that if $y_{jk} = 1$ then $y_{1k} = y_{2k} = \cdots = y_{jk} = 1$ and if $y_{lk} = 0$ then $y_{lk} = y_{l+1,k} = \cdots = y_{Mk} = 0$.

Finally we note that the objective function may be written as

$$\sum_{j=1}^{M} (1-p)p^{j=1} \sum_{k=1}^{N} h_k y_{jk}.$$

The inner summation represents the number of demands that are covered by at least $j$ facilities for all values of $j$. Th‹ coefficient of each such term, $(1 - p)p^{j-1}$, may be interpreted as the weight associated with the objective of maximizing the number of dema‹ ‹ds covered at least $j$ times. Thus the problem may be viewed as a m‹‹tiobjective programming problem with $M$ objectives. The single parameter $p$ specifies the relative weights associated with the objectives. As $p$ changes, the set of facility locations change; however, we will only be interested in those sets of facility locations that are *noninferior* in a multiobjective programming sense. A set of facility locations $\underline{X}_0$, is noninferior if there does not exist some other set $\underline{X}_1$, that covers more demand at least $j$ times without

TABLE I

*Example of Inferior and Noninferior Location Sets*

| Location Set | No. of Demands Covered | | Inferior/Noninferior |
|---|---|---|---|
| | At least once | Twice | |
| $A$ | 100 | 20 | Noninferior |
| $B$ | 80 | 20 | Inferior |
| $C$ | 75 | 25 | Noninferior |

covering less demand at least $l$ times for any value of $j$ and of $l$. For example, consider the case in which we are locating two facilities ($M = 2$). Consider three candidate sets of locations: $A$, $B$, and $C$. The hypothetical number of demands covered at least once and at least twice by the three candidate location sets are shown in Table I. Location sets $A$ and $C$ are noninferior. Location set $B$ is inferior to $A$, however, since we can select set $A$ and thereby increase the number of demands covered at least once without decreasing the number of demands covered twice. We use the notion of noninferior sets of candidate sites in the small numerical example shown below.

## 2. EXAMPLE NETWORK AND MODEL APPLICATION

IN THIS section we discuss an example of the model formulated above in Section 1. The example illustrates the sensitivity of facility siting decisions to the probability that a facility will not be working, $p$. The example also demonstrates that locating more than one facility at a node may be optimal in the maximum expected location problem while it is never optimal to do so in the traditional maximum covering model.

Consider the network shown in Figure 2. Let the critical distance $D$, be 9 units and assume we are to locate 2 facilities ($M = 2$). Under these conditions, the 6 pairs of sites listed in Table II are noninferior. Table III lists the values of the objective function for $p$ equal to 0.05, 0.15, 0.25, and 0.35. Figure 3 plots the objective function for each set of sites as a function of $p$. The upper envelope of the four curves is the objective function as a function of $p$.

Note that the optimal solution depends critically on the value of $p$. In this example, the solution that maximizes coverage by at least one unit—the solution one would obtain from the traditional maximum covering model—is only optimal for values $p$ between 0.0 and 0.1. Between $p = 0.1$ and $p = 0.2$, combination $D$ and $E$ is optimal; for $p = 0.2$ through $p = \frac{1}{3}$ locations $D$ and $F$ constitute the optimal solution. Finally, for $p \geq \frac{1}{3}$ it is optimal to locate both facilities at node $F$. We also observe that for any given set of facility sites, the objective function is concave in $p$, the probability that a facility will not be working.

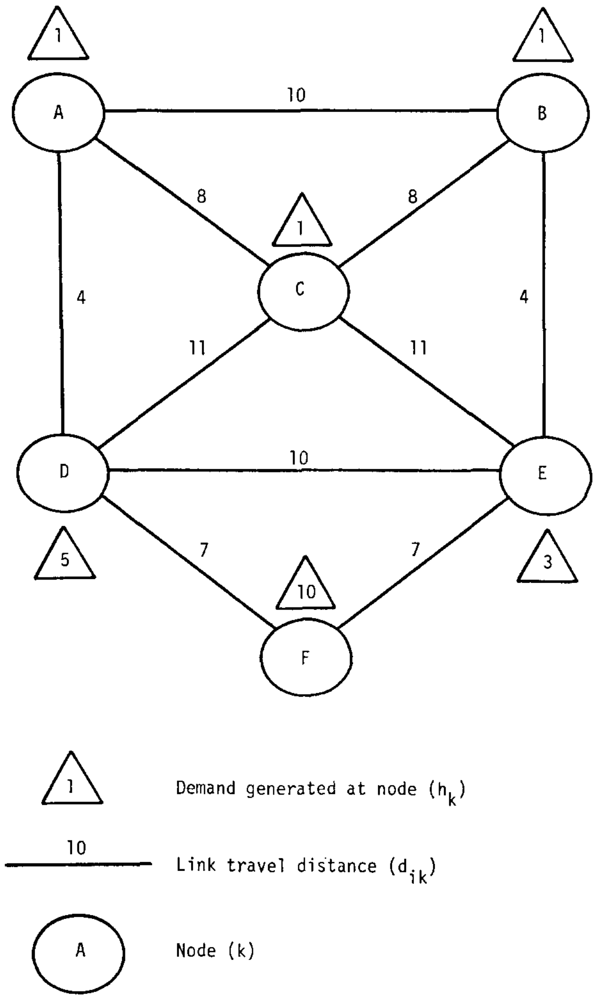


**Fig. 2.** Example network.

## 3. MODEL PROPERTIES

We now outline two properties of the MEXCLP which apply for $0 < p < 1$.

Property 1. *At the optimal solution to the MEXCLP*

a. $\sum_j y_{jk} = \sum_i a_{ki} X_i$

b. $\sum_i X_i = M$.

*That is, inequalities* (11) *and* (12) *will be satisfied by strict equality in the optimal solution.*

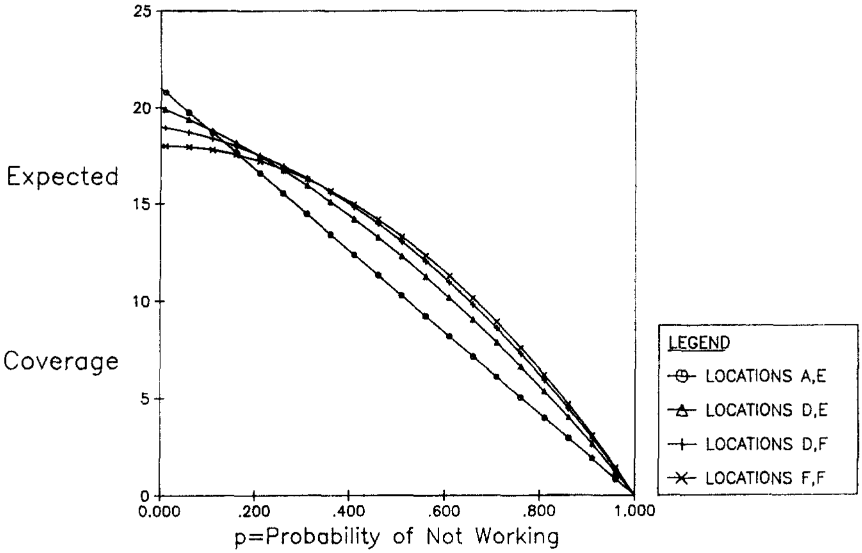


**Fig. 3.** Objective function versus the probability of a facility not working.

TABLE II

*Noninferior Facility Sites*

| Locations | No. of Demands Covered | |
|---|---|---|
| | At least once | Twice |
| *A, E* / *B, D* / *C, F* | 21 | 0 |
| *D, E* | 20 | 10 |
| *D, F* | 19 | 15 |
| *F, F* | 18 | 18 |

TABLE III

*Objective Function Values for Noninferior Sites and Various Values of p*

| Locations | Objective Function Values[a] | | | |
|---|---|---|---|---|
| | $p = 0.05$ | $p = 0.15$ | $p = 0.25$ | $p = 0.35$ |
| *A, E* / *B, D* / *C, F* | 19.95 | 17.85 | 15.75 | 13.65 |
| *DE* | 19.475 | 18.275 | 16.875 | 15.275 |
| *DF* | 18.7625 | 18.0652 | 17.0625 | 15.7625 |
| *FF* | 17.9550 | 17.595 | 16.875 | 15.795 |

[a] Underlined values indicate optimal locations for each value of $p$.

*Proof.* (a) Assume $\sum_j y_{jk} < \sum_i a_{ki}X_i$ for some $k$. Select some $y_{jk} < 1$ and increase it. Since $w_j h_k > 0$, $y_{jk} < 1$ could not be optimal.

*Note*: Some $y_{jk} < 1$ must exist if $\sum_k y_{jk} < \sum_i a_{ki}X_i$ since $\sum_i a_{ki}X_i \leq \sum_i X_i \leq M$ (by constraint (12) and the definition of $a_{ki}$). If $\sum_j y_{jk} < \sum_i a_{ki}X_i$, we have $\sum_j y_{jk} < M$ which implies that there exists at least one $y_{jk} < 1$ under these conditions.

(b) Assume $\sum_i X_i < M$. Select any $X_i$ and increase it by $M - \sum X_i$. This forces $\sum_j y_{jk} < \sum a_{ki}X_i$ for those inequalities with $a_{ki} = 1$. (Note that at least one such inequality exists since we assume $a_{ii} = 1$ for all $i$.) But part (a) showed that $\sum_j y_{jk} < \sum a_{ki}X_i$ is suboptimal. Therefore $\sum_i X_i = M$ in the optimal solution.

This property implies that, in solving the MEXCLP as a optimization problem, constraints (11) and (12) may be replaced by equality constraints, if it is advantageous to do so from the perspective of the optimization algorithm being used. Doing so will not affect either the optimal solution or the objective function. Note that this is not true of constraint (2) in the traditional maximum covering model which must remain as an inequality. The proof of the property also implies that adding facilities, or increasing $M$, will always increase the objective function. Note that in the traditional maximum covering model, increasing the number of facilities beyond the minimum number needed to cover each node at least once will not result in an improvement in the objective function.

The next property demonstrates that the optimal solution to the MEXCLP will automatically exclude dominated nodes. Node $j$ dominates node $m$ if node $j$ covers every node covered by node $m$ plus at least one additional node. Formally, node $j$ dominates node $m$ if

$$a_{kj} \geq a_{km} \text{ for all } k$$

and

$$a_{kj} > a_{km} \text{ for at least one value of } k.$$

PROPERTY 2. *If node $m$ is dominated by some node $j$, then $X_m = 0$ in the optimal solution.*

*Proof.* Let $X_m = \delta > 0$ and $X_j = \gamma$. Consider changing the solution to $X_m' = 0$, $X_j' = \gamma + \delta$

(a) Feasibility of $X_m$, $X_j$ implies feasibility of $X_m'$, $X_j'$.

(b) Let $k^*$ be the index of a node such that $a_{kj} > a_{km}$. For $k^*$, $\sum_j y_{jk^*} < \sum_i a_{k^*i}X_i'$ which, by Property 1, is suboptimal. That is, the value of the objective function under the primed solution $(X_i')$ exceeds the value for the unprimed solution. Therefore, dominated nodes cannot be in the solution to the MEXCLP.

## 4. HEURISTIC SOLUTION ALGORITHM

THE MEXCLP (Equations 10 through 13) is a fairly large integer programming problem. For a system with $N$ zones and $M$ facilities to be located, the model has $N + 1$ constraints (excluding the integer constraints (13) and (14)) and $N(M + 1)$ integer variables. For a 55-node problem with 5 facilities, we have 56 constraints and 330 integer variables. Furthermore, we would like to be able to solve the problem for a range of values of $p$, the probability that a facility is not working. Instead of solving the problem directly for a given value of $p$, we develop a heuristic procedure which finds good solutions for *all* values of $p$. We note at this point that, in most cases, we guarantee that the heuristic will find the optimal solution for $p \to 1$. Also, in most test cases the algorithm has found an optimal solution to the traditional maximum covering location problem (the $p = 0$ case). However, we should point out that the heuristic does not guarantee an optimal solution.

### 4.1. Qualitative Description of the Algorithm

The algorithm is based, in part on the observation that, for values of $p$ close to 1.0, the optimal solution is to place all $M$ facilities at the node which covers the greatest demand, assuming there is a unique node with this property. To see this, consider Figure 4. Assume that $M - K$ facilities are located at node $A$, which is the node that covers the greatest demand. Let this demand be $C_j^{\max}$. That is,

$$C_j^{\max} = C_A = \max_j \left(\textstyle\sum_k a_{kj} h_k\right).$$

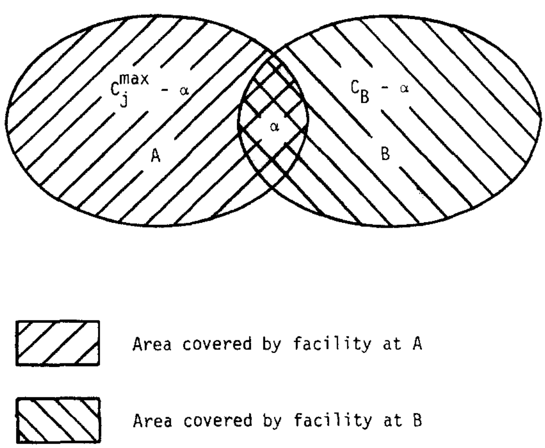


**Fig. 4.** Schematic coverage areas.

The remaining $K$ facilities are located at node $B$ which covers $C_B$ demands ($C_B < C_j^{max}$). Assume $\alpha$ demands are covered by both facilities. The objective function for this configuration of facilities is

$$(C_B - \alpha)(1 - p^K) + (C_j^{max} - \alpha)(1 - p^{M-K}) + \alpha(1 - p^M). \quad (15)$$

Consider moving one facility from $B$ to $A$. The objective function becomes

$$(C_B - \alpha)(1 - p^{K-1}) + (C_j^{max} - \alpha)(1 - p^{M-K+1}) + \alpha(1 - p^M). \quad (16)$$

Expression (16) exceeds expression (15) if

$$p > [(C_B - \alpha)/(C_j^{max} - \alpha)]^{1/(M+1-2K)}.$$

Thus, for sufficiently large values of $p$, it is optimal to place all the facilities at the node which covers the greatest number of demands if this node is unique. If this node is not unique, the algorithm will attempt single node exchanges in an effort to improve upon the solution for $p$ very close to 1.0. While single node exchanges will not guarantee that the optimal configuration will be identified for large values of $p$, the procedure is likely to work quite well.

Assuming the node which covers the maximum demand is unique, the heuristic begins with all facilities located at this node, with $p^*$, an indicator, set to 1.0, and $p^{**}$, a second indicator, set to 0.0. The heuristic then considers single node substitutions to this set of locations. At any stage in the algorithm, we refer to the set of facility sites for which single node substitutions are being considered as the *current* location set. Initially the current location set has all facilities at the node which covers the maximum number of demands. Each set of locations which may be formed by removing one facility site from the current solution and replacing it with another location—performing a single node substitution—is termed a *trial* solution. In evaluating a trial solution, three results may occur:

a. The trial solution is inferior to the current solution for all values of $p$ between $p^{**}$ and $p^*$
b. The trial solution is better than the current solution for some value of $p$ in the interval $[p^{**}, p^*)$.
c. The trial solution is better than the current solution for $p$ equal to $p^*$.

In case (a), the trial solution is discarded and a new trial solution is generated, if any remain to be generated. In case (b) the trial solution becomes the *tentative* solution, replacing the previous tentative solution if one existed. In addition, $p^{**}$ is updated and set equal to the value of $p$ at which the trial solution becomes better than the current solution.

Again, a new trial solution is generated, if any remain to be generated. Note that, at any step of the heuristic, $p^{**}$ is the smallest value of $p$ at which the current solution is the best the heuristic can find. For values of $p$ less than $p^{**}$, the tentative solution is better than the current solution. In case (c), the trial solution immediately replaces the current solution, $p^{**}$ is reset to 0.0, and a new series of single node substitutions is initiated.

Once all possible single node substitutions that can be generated based on the current solution have been evaluated, if a tentative solution has been identified, it becomes the new current solution. In this case, $p^*$ is set equal to $p^{**}$ (which will be smaller than $p^*$), $p^{**}$ is then reset to 0.0 and a new series of single node substitutions is initiated. If a tentative solution does not exist, in which case $p^{**} = 0.0$, the algorithm stops.

## 4.2. Technical Description of the Algorithm

In this section, we describe the operation of the algorithm in detail, focusing on the manner in which trial solutions are evaluated. Step numbers refer to the steps of the formal algorithm presented in the Appendix.

The algorithm begins by ordering the nodes in descending order of the number of demands covered by each node. This step (Step 1A) is optional but is recommended because examining nodes in this order should reduce the execution time of the algorithm. Next, the algorithm locates all facilities at the node which covers the greatest number of demands and initializes $p^*$ equal to 1.0 (Step 1B). Single node substitutions or trial solutions are now examined (Step 2).

The evaluation of each trial solution proceeds as follows. Let $\{X_i^*\}$ be the current solution and $\{\hat{X}_i\}$ be a trial solution to be evaluated. Since the algorithm considers only single node substitutions to the current solution, exactly two elements of $\{X_i^*\}$ and $\{\hat{X}_i\}$ will be different. The number of times node $k$ is covered in the current solution, $q(k)$, is

$$q(k) = \sum_i a_{ki} X_i^*. \tag{17}$$

Thus, the objective function for the current solution as a function of $p$ is given by

$$\sum_k h_k(1-p) \sum_{j=1}^{q(k)} p^{j-1} \tag{18}$$

where the inner summation is taken to be zero if $q(k) = 0$.

Let $\hat{q}(k)$ be the number of times node $k$ is covered in the trial solution. We have

$$\hat{q}(k) = \sum_i a_{ki} \hat{X}_i. \tag{19}$$

The objective function for the trial solution as a function of $p$ is

$$\sum_k h_k(1-p)\sum_{j=1}^{\hat{q}(k)} p^{j-1} \quad (20)$$

where again the inner summation is taken to be zero if $\hat{q}(k)=0$.

The change in the objective function as a result of substituting $\{\hat{X}_i\}$ for $\{X_i^*\}$ as a function of $p$ is given by the difference between (20) and (18)

$$\sum_k h_k(1-p)\Big[\sum_{j=1}^{\hat{q}(k)} p^{j-1} - \sum_{j=1}^{q(k)} p^{j-1}\Big]. \quad (21a)$$

Since we are primarily interested in whether the change is positive or negative for any value of $p$ and since $(1-p)$ is strictly positive for $0 \le p < 1$ we can eliminate the $(1-p)$ term to obtain

$$I(p) = \sum_k h_k\Big[\sum_{j=1}^{\hat{q}(k)} p^{j-1} - \sum_{j=1}^{q(k)} p^{j-1}\Big] \quad (21b)$$

where $I(p)$ denotes the improvement in the objective function as a function of $p$ due to substituting the trial solution for the current solution, excluding the $(1-p)$ term.

We note that since a node may be covered at most $M$ times, $I(p)$ is an $M-1$ order polynomial in $p$ which we may write as:

$$I(p) = \sum_{j=1}^{M} \delta_j p^{j-1}.$$

Evaluation of $I(p)$ using (21b) directly is not very efficient, since as we show below, any node $k$ will contribute to at most one of the coefficients $\delta_j$. The coefficients $\delta_j$ may be determined by noting that $\hat{q}(k)$ differs from $q(k)$ by at most 1. Therefore, we may write

$$\hat{q}(k) = q(k) + \Delta_k \quad (22)$$

where

$$\Delta_k = \sum_i a_{ki}(\hat{X}_i - X_i^*). \quad (23)$$

Combining Equations 22 and 21b, we note that the contribution of node $k$ to $I(p)$ may be written as

$$\gamma_k = \begin{cases} h_k p^{q(k)} & \text{if } \Delta_k = 1 \\ 0 & \text{if } \Delta_k = 0 \\ -h_k p^{q(k)-1} & \text{if } \Delta_k = -1. \end{cases} \quad (24)$$

Clearly, node $k$ contributes to only one of the $\delta_j$ terms. In particular, if $\Delta_k = 1$, node $k$ increases $\delta_{q(k)+1}$ (the coefficient of $p^{q(k)}$) by $h_k$; if $\Delta_k = -1$, node $k$ decreases $\delta_{q(k)}$ by $h_k$; and if $\Delta_k = 0$, node $k$ does not contribute to $I(p)$.

The algorithm begins evaluating trial solutions by setting $p^{**} = 0.0$ and initializing a number of other indicators including variables used to store the tentative solution (Step 2A). Next, the algorithm computes the coefficients $\delta_j$ in the improvement function $I(p)$ for a specific trial solution (Steps 2B, 2C, and 2D) using Equations 23 and 24. Next, the improvement function at $p^*$ is evaluated (Step 2E). If $I(p^*) > 0$, the current solution is inferior to the trial solution at $p = p^*$ and for values of $p = p^* + \epsilon$ for sufficiently small $\epsilon$ values. The heuristic indicates this suboptimality with a warning message, replaces the current solution with the trial solution (Step 3B) and the single node substitution process is reinitiated (Step 2A). If the trial and current solutions have identical objective functions at $p^*$, the slope of the improvement function at $p^*$ is evaluated (Step 2F). If the slope is negative the trial solution again replaces the current solution and the single node substitution process is reinitiated.

If the value of the improvement function at $p^*$ is negative or if its value is zero and its slope is non-negative, the algorithm finds the largest root of $I(p)$ in the interval $[p^{**}, p^*)$, if such a root exists (Step 2G). If no root exists, a new trial solution is generated (Steps 2J and 2K). If a root exceeding $p^{**}$ is found, $p^{**}$ is set equal to the root (Step 2G), the trial solution is saved as the tentative solution (Step 2I) and a new trial solution is generated (Steps 2J and 2K). Finally, if the root of $I(p)$ equals $p^{**}$, the slopes of the improvement functions for the trial and tentative solutions at $p^{**}$ are compared (Step 2H). If the slope of the trial solution is greater than the slope of the tentative solution at $p^{**}$, the trial solution replaces the tentative solution (Step 2I). In either case a new trial solution is generated (Steps 2J and 2K).

If all trial solutions have been tested, the algorithm checks the value of $p^{**}$ (Step 3A). If $p^{**} > 0.0$, a tentative solution was identified; $p^*$ is set equal to $p^{**}$, the tentative solution becomes the current solution (Step 3B) and the single node substitution process is reinitiated (Step 2A). If $p^{**} = 0.0$, no tentative solution was identified and the algorithm stops.

One final observation is worth noting. Property 2 states that dominated nodes are not included in the optimal solution to MEXCLP. The heuristic preserves this property despite the fact that it does not guarantee optimality in general. To see this, we note that nodes can be inserted into the solution in three ways:

a. As part of the initial solution for large $p$ (Step 1B);
b. As an improved solution to the current solution at $p^*$ (when Step 3B is reached from either Step 2E or Step 2F);
c. When a tentative solution is substituted for the current solution (when Step 3 is executed following Step 2K).

A dominated node cannot enter and remain in a current solution in any of these three ways under the rules of the heuristic. Clearly a dominated node cannot be part of the initial solution. If Step 1A is executed, a dominated node will never enter an improved solution to the current solution. This is true because the heuristic will try to substitute the *dominating* node into the current solution before it tries the *dominated* node. Since the objective function using the dominated node must be less than that obtained using the dominating node, the dominated node will not enter the current solution in this way. If Step 1A is not executed, a dominated node may enter the current solution for a few calculations but will be removed as soon as the single node substitution part of the heuristic checks the dominating node. Finally, a dominated node cannot be part of the final tentative solution to be substituted for the current solution (Step 3) since a solution with a dominated node will result in a smaller value of $p^{**}$ than will the same solution with the dominating node substituted for the dominated node. Step 2G thus prevents the dominated node from being in the final tentative solution.

## 5. COMPUTATIONAL EXPERIENCE

THE HEURISTIC was tested using a frequency used 55-node test problem[11, 18] shown in Figure 5. Euclidean distances were computed and the

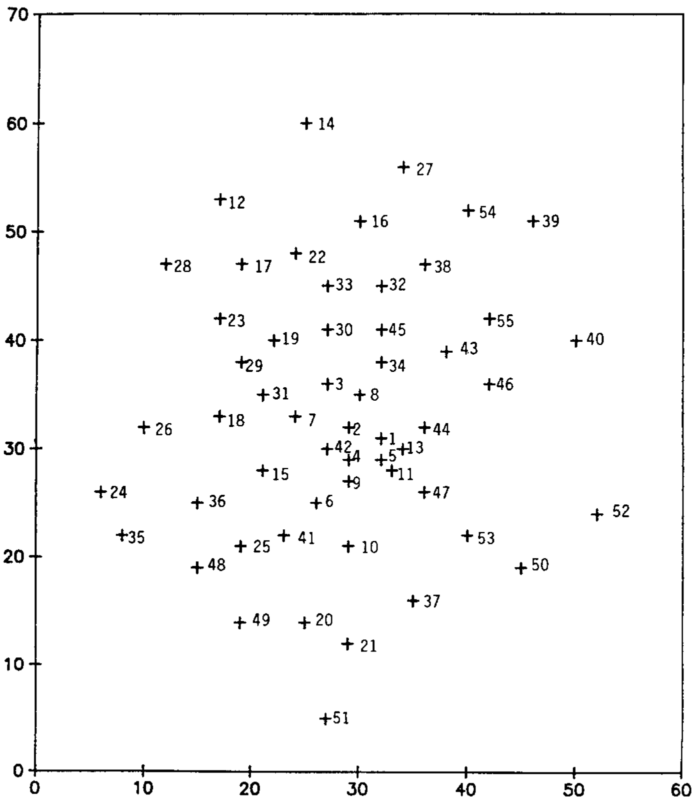


**Fig. 5.** The 55-node test problem. + = candidate sites.

critical distance used to define coverage was taken to be 15 units. Table IV lists the results obtained when 3 facilities are to be located.

In one of the seven solutions identified, the heuristic detected suboptimality. That is, in searching for a new tentative solution (Step 2), the algorithm identified a better solution at $p^*$ and the improved solution was substituted for the current solution (Step 3B was executed after 2E). Note that when suboptimality is detected, the objective function identified by the heuristic is discontinuous. For example, the second solution is suboptimal at 0.0339. The objective function at that point is 6062.0 using the second solution and 6123.9 using the first solution at $p = 0.0339$. In fact, the first solution is better than the second solution throughout the range $0.0339 < p < 0.0634$. That is, the second solution is completely dominated by the first solution. The first solution is inferior to the third for $p > 0.0699$. Table V presents the revised results based on these post-heuristic analyses.

Several points are worth noting. First, while the heuristic does not guarantee optimal solutions, it can detect certain suboptimalities. Second, the differences between the improved and suboptimal objective functions are usually small. The largest difference found to date was less than 1.4%

TABLE IV

*Sample Heuristic Results*

| Range of $p$ | | Solution | Objective Function Coefficients | | | Objective Function at | | Suboptimality Detected? |
|---|---|---|---|---|---|---|---|---|
| Low value | High value | | $1-p$ | $1-p^2$ | $1-p^3$ | Low value of $p$ | High value of $p$ | |
| 0.0 | 0.0339 | 22, 25, 43 | 3630 | 2620 | 0 | 6250.0 | 6123.9 | No |
| 0.0339 | 0.0634 | 15, 22, 53 | 1860 | 4270 | 0 | 6062.0 | 5994.8 | Yes |
| 0.0634 | 0.2830 | 9, 15, 22 | 1270 | 3550 | 1270 | 5994.8 | 5417.6 | No |
| 0.2830 | 0.4042 | 9, 15, 19 | 980 | 1120 | 3770 | 5417.6 | 5041.9 | No |
| 0.4042 | 0.5898 | 7, 9, 19 | 610 | 1350 | 3800 | 5041.9 | 4151.2 | No |
| 0.5898 | 0.8212 | 7, 7, 9 | 290 | 430 | 4720 | 4151.2 | 2297.7 | No |
| 0.8212 | 1.0 | 7, 7, 7 | 0 | 0 | 5150 | 2297.7 | 0.0 | No |

TABLE V

*Results of Post-Heuristic Analysis*

| Range of $p$ | | Solution | Objective Function Coefficients | | | Objective Function at | |
|---|---|---|---|---|---|---|---|
| Low value | High value | | $1-p$ | $1-p^2$ | $1-p^3$ | Low value of $p$ | High value of $p$ |
| 0.0 | 0.0699 | 22, 25, 43 | 3630 | 2620 | 0 | 6250.0 | 5983.4 |
| 0.0699 | 0.2830 | 9, 15, 22 | 1270 | 3550 | 1270 | 5983.4 | 5417.6 |
| 0.2830 | 0.4042 | 9, 15, 19 | 980 | 1120 | 3770 | 5417.6 | 5041.9 |
| 0.4042 | 0.5898 | 7, 9, 19 | 610 | 1350 | 3800 | 5041.9 | 4151.2 |
| 0.5898 | 0.8212 | 7, 7, 9 | 290 | 430 | 4720 | 4151.2 | 2297.7 |
| 0.8212 | 1.0 | 7, 7, 7 | 0 | 0 | 5150 | 2297.7 | 0.0 |

of the value of the improved solution. The difference identified in Table IV is only 1.0% of the improved solution. Third, simple post-heuristic analyses may be performed when suboptimality is detected to improve the heuristic solution. These calculations involve finding the roots of an $M$th order polynomial and can be readily performed on a programmable calculator. In most cases, the range in which a solution remains the best of those identified by the heuristic changes little as a result of the post-heuristic analyses. In a few cases, an entire solution is eliminated as a result of these calculations. This happened in Table V.

Finally, we note that the solution identified at $p = 0.0$ corresponds with a known solution to the maximum covering location problem. It is worth noting that in this solution only 150 of the 6400 demands in the system are not covered. However, this solution is only optimal over a small range of $p$ $(0.0 \leq p \leq 0.0699)$. For $p > 0.0699$ it is optimal to decrease the number of demands that are covered in favor of increasing the multiple coverage of demands. This result has been found in all test cases to date. The traditional maximum covering solution remains optimal for only a small range of $p$.

## 6. SUMMARY AND RECOMMENDATIONS FOR FUTURE WORK

A VARIANT of the maximum covering location problem that accounts for the possibility that facilities may be unable to respond to demands was formulated. The model is called the maximum *expected* covering location problem (MEXCLP). Several properties of the MEXCLP were proven. The most important of these is that for $0 < p < 1$, the solution to the MEXCLP will not include dominated nodes. A single node substitution heuristic algorithm was proposed. The algorithm finds good solutions for the full range of values of $p$. The heuristic also does not allow dominated nodes to enter and remain in any solution. Computational experience with the algorithm using a 55-node test problem was discussed. The heuristic identified a known optimal solution to the maximum covering location problem (the $p = 0$ case.)

The model assumes that the probability $p$ of a facility being busy is identical for all facilities. In addition, the model assumes that the probability of a facility being busy is independent of whether or not the other facilities are busy. Both assumptions are clearly not strictly justified. The model may be reformulated as an integer nonlinear programming model that allows the probability of a facility being busy to depend on the facility site, while retaining the independence assumption. Work on this model is currently underway and will be reported on in a later paper. Relaxing the independence assumption may be more difficult. However, a related study compared the expected coverage predicted by the

MEXCLP model which assumes independence with that computed using Fitzsimmon's CALL model,[13] which relaxes the independence assumption. The differences were not significant. Future research should attempt to relax the independence assumption and should examine its effect on location decisions as well as the expected coverage.

Several additional directions for future study may also be identified. First, an efficient optimal solution algorithm to the MEXCLP would clearly be desirable. Two approaches are currently under study: Bender's decomposition is being tested to solve a mixed integer formulation of the problem, and nonlinear programming approaches are being applied to the extension of the model discussed above that allows for location-specific probabilities of a facility being busy. The results of these studies will be reported in a subsequent paper. Second, additional computational experience with either heuristic or exact solutions to the model will enable us to strengthen the qualitative conclusions discussed in Section 5. Comparison of these results with those of simulation or queueing theory based models will also allow us to test the effects on both the objective function value and the location decisions of assuming that the probability of facilities being busy is the same for all facilities and that it is independent of whether or not other facilities are busy. Finally, optimization based models that relax these assumptions should also be developed.

## APPENDIX—FORMAL STATEMENT OF THE HEURISTIC

### *STEP 1. INITIALIZATION*

*Step 1A. Ordering Candidate Nodes:* For each node, compute

$$C_j = \textstyle\sum_k a_{kj} h_k = \text{demand covered by node } j.$$

Create a new index set $i$, such that

$$C_1 \geq C_2 \geq \cdots \geq C_i \geq \cdots \geq C_N.$$

(Note: Renumbering the candidate locations in this way is optional.)

*Step 1B. Determine Initial Current Solution:*

Let $j^*$ be the index of a node which maximizes $C_j$.

Set $L(m) \leftarrow j^*$, $m = 1, \cdots, M$ ($L(m)$ is the current solution).

Set $p^* = 1.0$.

### *STEP 2. EVALUATION OF SINGLE NODE SUBSTITUTIONS*

*Step 2A. Initialization for Single Node Substitutions:*

Set $p^{**} = 0.0$, $i = 1$, $m = 1$, $j = L(m)$, $i' = 0$, $m' = 0$, and $\delta_q' = 0$ ($q = 1, 2, \cdots, M$) where

$p^{**}$ = the smallest value of $p$ for which the current solution is the best solution the heuristic can find

$m$ = a counter for the number of nodes in the current solution that have been or are being tested for substitution

$j$ = node number of the node in the current solution which is being tested for substitution

$i$ = node number of the node being tested to replace node $j$

$m'$ = pointer to the position in vector $L(m)$ of the node to be removed to obtain the tentative solution

$i'$ = node to replace node $L(m')$ in the current solution to obtain tentative solution

$\delta_q'$ = vector used to store coefficients of the $I(p)$ function for the tentative solution.

*Step 2B. Initialize I(p) Coefficient Vector*:

Set $\delta_q = 0$ $(q = 1, 2, \cdots, M)$ where $\delta_q$ is the coefficient of $p^{q-1}$ in $I(p)$.

*Step 2C. Compute Change in Node Coverage*:

Compute $\Delta_n = a_{ni} - a_{nj}$ for all $n$ where

$\Delta_n$ = change in the number of times node $n$ is covered as a result of substituting the trial solution (node $i$) for the current solution (node $j$).

*Step 2D. Compute Changes in Objective Function Coefficients*:

For all zones $n (n = 1, 2, \cdots, N)$

— if $\Delta_n = 1$ increase $\delta_{q(n)+1}$ by $h_n$

— if $\Delta_n = 0$ leave $\delta_q$ unchanged for all $q$

— if $\Delta_n = -1$ decrease $\delta_{q(n)}$ by $h_n$

where

$$q(n) = \begin{cases} \text{number of times node } n \text{ is covered by the current solution} \\ \sum_{m=1}^{M} a_{nL(m)}. \end{cases}$$

*Step 2E. Evaluate Trial Solution at $p^*$*:

Evaluate $I(p^*) = \sum_{q=1}^{M} \delta_q (p^*)^{q-1}$.

If $I(p^*) > 0$, replace $i'$ by $i$ and $m'$ by $m$; go to Step 3B

If $I(p^*) = 0$, go to Step 2F

If $I(p^*) < 0$, go to Step 2G

*Step 2F. Test Slope of Improvement Function at $p^*$*:

Evaluate $S(p^*) = \sum_{q=1}^{M} \delta_q (q-1)(p^*)^{q-2}$.

If $S(p^*) \geq 0.0$, go to Step 2G

If $S(p^*) < 0.0$, replace $i'$ by $i$ and $m'$ by $m$; go to Step 3B.

*Step 2G. Find Root of I(p)*:

Find the largest root of $I(p) = \sum_{q=1}^{M} \delta_q p^{q-1}$ in the interval $(p^{**}, p^*)$. If no root is found, go to Step 2J. If the root exceeds $p^{**}$, set $p^{**}$ equal to the root and go to Step 2I. If the root equals $p^{**}$, go to Step 2H.

*Step 2H. Compare Slopes of Trial and Tentative Solutions at $p^{**}$*:

Evaluate $S(p^{**}) = \sum_{q=1}^{M} \delta_q(q-1)(p^{**})^{q-2}$

Evaluate $S'(p^{**}) = \sum_{q=1}^{M} \delta_q'(q-1)(p^{**})^{q-2}$

If $S(p^{**}) > S'(p^{**})$, go to Step 2I; otherwise go to Step 2J.

*Step 2I. Update Tentative Solution*:

Replace $i'$ by $i$, $m'$ by $m$ and $\delta_q'$ by $\delta_q (q = 1, \cdots, M)$.

*Step 2J. Try Substituting Next Mode*:

Increment $i$ by 1. If $i$ exceeds $N$ go to Step 2K; otherwise go to Step 2B.

*Step 2K. Change Node Being Deleted from Current Solution*:

Set $i = 1$. Increment $m$ by 1. If $m$ exceeds $M$, go to Step 3A; otherwise set $j = L(m)$ and go to Step 2B.

*STEP 3. UPDATING CURRENT SOLUTION*

*Step 3A. Update $p^*$ and Check for Termination*:

If $p^{**} > 0.0$, replace $p^*$ with $p^{**}$ and go to Step 3B.

If $p^{**} = 0$, *STOP*, final solution identified.

*Step 3B. Replace Current Solution with Tentative Solution*:

Replace $L(m')$ by $i'$. Go to Step 2A.

ACKNOWLEDGMENTS

THE AUTHOR would like to acknowledge Professors Richard Church and David Eaton of the University of California at Santa Barbara and the University of Texas at Austin respectively for numerous discussions from which this work has benefited.

This research was supported by a grant CME-800677, "Location Techniques for Finite Capacity Service Centers Subject to Random Breakdowns," from the National Science Foundation. This support is gratefully acknowledged.

REFERENCES

1. A. A. ALY AND J. A. WHITE, "Probabilistic Formulation of the Emergency Service Location Problem," *J. Opnl. Res. Soc.* **29**, 1167–1179 (1978).
2. J. M. BENEDICT, "Three Hierarchical Objective Models which Incorporate the Concept of Excess Coverage to Locate EMS Vehicles or Hospitals," forthcoming M.S. thesis, Department of Civil Engineering, Northwestern University, Evanston, Ill., 1982.
3. V. L. BENNETT, D. J. EATON, AND R. L. CHURCH, "Selecting Sites for Rural Health Workers," *Soc. Sci. Med.* **16**, 63–72 (1982).
4. G. N. BERLIN AND J. C. LIEBMAN, "Mathematical Analysis of Emergency Ambulance Location," *Socio-Econ. Plan. Sci.* **8**, 323–328 (1971).

5. S. C. CHAPMAN AND J. A. WHITE, "Probabilistic Formulations of Emergency Service Facilities Location Problems," paper presented at the 1974 ORSA/ TIMS Conference, San Juan, Puerto Rico, 1974.
6. N. CHRISTOFIDES, *Graph Theory: An Algorithmic Approach*, Academic Press, New York, 1975.
7. R. L. CHURCH AND M. E. MEADOWS, "Location Modeling Using Maximum Service Distance Criteria," *Geograph. Anal.* **11,** 358–373 (1979).
8. R. L. CHURCH AND C. REVELLE, "The Maximal Covering Location Problem," *Papers Regional Sci. Assoc.* **32,** 101–118 (1974).
9. M. S. DASKIN, "Application of an Expected Covering Model to Emergency Medical Service System Design," *Decision Sci.* **13,** 416–439 (1982).
10. M. S. DASKIN AND E. H. STERN, "A Hierarchical Objective Set Covering Model for EMS Vehicle Deployment," *Trans. Sci.* **15,** 137–152 (1981).
11. D. EATON, R. CHURCH AND C. REVELLE, "Location Analysis: A New Tool for Health Planners," Methodological Working Document 53, Sector Analysis Division, Bureau for Latin America, Agency for International Development, 1977.
12. EMERGENCY MEDICAL SERVICES POLICY RESEARCH PROJECT, "Location Techniques for Emergency Medical Service Vehicles," Vol. I–IV, Policy Research Project Report Number 34, LBJ School of Public Affairs, University of Texas at Austin, Austin, Tex., 1979.
13. J. A. FITZSIMMONS, "A Methodology for Emergency Ambulance Deployment," *Mgmt. Sci* **19,** 608–627 (1973).
14. S. L. HAKIMI, "Optimum Locations of Switching Centers and the Absolute Centers and Medians of a Graph," *Opns. Res.* **12,** 450–459 (1964).
15. S. L. HAKIMI, "Optimum Distribution of Switching Centers in a Communication Network and Some Related Graph Theoretic Problems," *Opns. Res.* **13,** 462–475 (1965).
16. G. Y. HANDLER AND P. B. MIRCHANDANI, *Location on Networks: Theory and Algorithms*, M.I.T. Press, Cambridge, Mass., 1979.
17. P. KOLESAR AND W. E. WALKER, "An Algorithm for the Dynamic Relocation of Fire Companies," *Opns. Res.* **22,** 249–274 (1974).
18. T. G. MASOG, "Modelling Techniques for a Flexible Emergency Aid Location System," unpublished M.S. thesis, Department of Civil Engineering and the LBJ School of Public Affairs, University of Texas at Austin, Austin, Tex., 1980.
19. E. MINIEKA, "The Centers and Medians of a Graph," *Opns. Res.* **25,** 641–650 (1977).
20. P. B. MIRCHANDANI AND A. R. ODONI, "Locations of Medians on Stochastic Networks," *Trans. Sci.* **13,** 85–97 (1979).
21. P. B. MIRCHANDANI AND A. OUDJIT, "Probabilistic Demands and Costs in Facility Location Problems," paper presented at the Regional Science Association 27th North American Meeting, Milwaukee, Wisc., 1980.
22. D. R. PLANE AND T. E. HENDRICK, "Mathematical Programming and the Location of Fire Companies for the Denver Fire Department," *Opns. Res.* **25,** 563–578 (1977).

23. C. SWOVELAND, D. UYENO, I. VERTINSKY AND R. VICKSON, "Ambulance Location: A Probabilistic Enumeration Approach," *Mgmt. Sci.* **20**, 686–698 (1973).
24. C. TOREGAS, R. SWAIN, C. REVELLE AND L. BERGMAN, "The Location of Emergency Service Facilities," *Opns. Res.* **19**, 1363–1373 (1971).
25. W. E. WALKER, "Using the Set-Covering Problem to Assign Fire Companies to Fire Stations," *Opns. Res.* **22**, 275–277 (1974).

Received February 1981