Environment and Planning A, 1979, volume 11, pages 373-380

The robustness of two common heuristics for the p-median problem

K E Rosing*, E L Hillsman¶, Hester Rosing-Vogelaar*
* Economisch-Geografisch Instituut, Erasmus Universiteit, Rotterdam, The Netherlands
¶ Energy Division, Oak Ridge National Laboratory†, Oak Ridge, Tennessee 37830, USA
Received 25 September 1978

Abstract. Optimal p-median solutions were computed for six test problems on a network of forty-nine demand nodes and compared with solutions from two heuristic algorithms. Comparison of the optimal solutions with those from the Teitz and Bart heuristic indicates that this heuristic is very robust. Tests of the Maranzana heuristic, however, indicate that it is efficient only for small values of p (numbers of facilities) and that its robustness decreases rapidly as problem size increases.

1 Introduction

The p-median problem is to find locations for p facilities on a network which minimize the aggregate distance of all nodes (demand nodes) in the network from their closest facility. ReVelle and Swain (1970) have shown that this is the mathematical equivalent of minimizing the average distance which individuals in the system would have to travel to obtain service. As such it is an appealing objective in the planning of locations for various sorts of service centers.

The problem is very difficult to solve optimally for realistically large problems, and as a result several heuristics have been developed for it. The most common of these are the ones proposed by Maranzana (1964) and by Teitz and Bart (1968). The purpose of this paper is to compare solutions derived from these heuristics with optimal solutions for the problem.

2 Heuristic methods

Rushton and Kohler (1973) describe both the Maranzana and the Teitz and Bart algorithms, making copies of them available as the computer program ALLOC. The Maranzana heuristic operates by alternately enforcing two requirements. Each demand node must be assigned to its closest facility, and each facility must minimize the aggregate distance between it and the demand nodes assigned to it. A pattern of facilities which meets both conditions simultaneously is termed a stable partition (Ostresh, 1973), and the Maranzana heuristic terminates upon finding one.

The Teitz and Bart heuristic operates by taking each demand node in turn and examining it as a replacement for each facility in turn. If such a replacement reduces the value of the objective function, an exchange of the facility for the demand node is effected. When no facility can be moved to any demand node to reduce the objective, execution terminates. In their discussion of this algorithm, Rushton and Kohler make several corrections to the original description by Teitz and Bart (1968). These corrections speed the algorithm's convergence by correctly calculating the value of the objective function for alternative solutions to the problem.

Once a heuristic is defined, the computational procedure is fixed. From any particular starting point and up to the conclusion, the steps which must be performed are always the same and the same result will obtain. Improvements in efficiency then

† Operated by the Union Carbide Corporation under contract W-7405-eng-26 with the US Department of Energy.

result from the improvement of the method of performing each step and from the internal organization of the data. Hillsman and Rushton (1975) report on a series of problems which were solved by use of ALLOC IV, an improved coding of ALLOC. The solutions obtained here are from ALLOC V, which is even more efficient (Hillsman, 1979). A new version, ALLOC VI, further increases efficiency, this time by improved data storage and handling. The solution arrived at from a particular starting point with any of these versions of ALLOC will be the same because the steps performed remain the same.

3 Optimal methods

ReVelle and Church (1980) note that comparisons of optimally derived and heuristic solutions are generally lacking in the literature. This is probably because large p-median problems cannot at this time be solved optimally, whereas researchers working with heuristics have generally been loathe to solve problems of small or 'trivial' size[(1)]. ReVelle and Swain (1970) developed a linear-programming formulation of the p-median problem which, when solved, may or may not be feasible in terms of the spatial problem. Considerable experience has now been gained with this linear program (ReVelle and Swain, 1970; Rosing and ReVelle, 1978). In approximately 95% of cases the solution is feasible, and a tentative explanation of infeasible solutions has been advanced (Rosing and ReVelle, 1978). Linear-programming codes which can contain more than about fifty demand nodes are not available at this time. The branch-and-bound method has been applied to the problem (Jarvinen et al, 1972; El-Shaieb, 1973; ReVelle and Rosing, 1979; Odell et al, 1976), but thirty nodes seems to be about the maximum which has been solved by this approach. Swain (1974) has developed a decomposition algorithm based on the linear-programming formulation, and it has been used successfully to solve problems with up to fifty-five demand nodes (Swain, 1974; Garfinkel et al, 1974).

It would appear that the technology for solving p-median problems optimally is limited at this time to problems with about fifty demand nodes. Since "heuristics do have positive value ... when they are used on problems of sizes beyond the capacities of the exact methods" (ReVelle and Church, 1980), we have solved a series of problems of moderate size, both optimally and heuristically, to test the quality of the solutions from the Maranzana and Teitz and Bart heuristics. Although care should be exercised in extending these results to problems where current methods cannot verify them, they do provide a baseline for comparison at or near the limit of current technology.

4 The test problems

The data matrix selected was for the Talala area of Junagadh District, Gujarat, India. This matrix has been used extensively in the development of the ALLOC series of programs and in the development of unpublished heuristics (Hillsman, 1979). The provision of optimal solutions could thus complement this ongoing research.

The Talala area contains fifty villages, one of which had a population of zero in 1971. This distance matrix used in all instances consists of the shortest road or cart-track distances between the forty-nine inhabited villages, and it is symmetrical. The shortest distance from each village (i) to each other village (j) was multiplied by the 1971 population of the ith village. This created an asymmetrical matrix of population-weighted distances with zeros on the principal diagonal. All forty-nine demand nodes

(1) There is a considerable difference in the conceptualization of 'large' and 'small' between these two groups. A person developing a heuristic seems to consider the problem involving a few hundred or thousand demand nodes large, whereas a person trying to solve problems optimally would consider fifty large and a hundred as very large.

were eligible to be facility sites, and no maximum-distance constraint was used. Six problems were defined with 2, 4, 5, 6, 9, and 10 facilities.

5 The optimal solutions

The linear-programming formulation developed by ReVelle and Swain (1970) was used: namely,

$$\text{minimize } Z = \sum_{i=1}^{n} \sum_{j=1}^{n} a_{ij} X_{ij} \tag{1}$$

subject to

$$\sum_{j=1}^{n} X_{ij} = 1 , \qquad \text{for all } i , \tag{2}$$

$$X_{jj} - X_{ij} \geqslant 0 , \qquad \text{for all } i, j, \quad i \neq j , \tag{3}$$

$$\sum_{j=1}^{n} X_{jj} = p , \tag{4}$$

and

$$X_{ij} = \begin{cases} 1 & \text{if } i \text{ is served by } j, \\ 0 & \text{otherwise,} \end{cases} \qquad \text{for all } i, j, \tag{5}$$

where

$$a_{ij} = w_i d_{ij} , \qquad \text{for all } i, j, \tag{6}$$

d_{ij} is the distance from demand node i to a potential facility location j,
w_i is the population of place i in 1971,
p is exogenous to the program and set equal to the number of facilities to be sited,
X_{ij} is the decision variable, and
n is the number of demand nodes.

Constraint (2) requires that each demand node be assigned to one and only one facility. Constraint (3) requires that demand nodes be assigned only to potential facility sites which are allocated facilities, and that any potential facility site which accepts an assignment must self-assign and be one of the p selected facility sites. Constraint (4) requires that the total number of facilities sited be exactly the number specified external to the program. Statement (5) indicates that the decision variables are restricted to the values 0 and 1. The objective function (1) is composed of n^2 terms $a_{ij}X_{ij}$. Since a_{ij} is a population-weighted shortest distance [equations (6)] and since X_{ij} is 0 or 1 depending on the assignment of i to a particular facility j, $n^2 - n$ of these terms will be zero, leaving n values of a_{ij} to be summed.

All problems were solved on the IBM 370/158 under virtual storage in a partition of 200 K by use of the MPSX linear-program solution algorithm (IBM, 1971; 1972). A FORTRAN program [developed by Rosing and ReVelle and used in Rosing and ReVelle (1978)] was run to develop the constraint set, which was then written on disk. This disk was read by the MPSX executor. Constraint (5) had to be relaxed to

$$X_{ij} \geqslant 0 , \qquad \text{for all } i, j, \tag{7}$$

because its integer form cannot be enforced. Since the problem is one of minimization, X_{ij} will never take on a value greater than one and it is unnecessary to introduce an upper bound. Constraint (5) now becomes a test of feasibility of the solution in terms of the integer problem. All six test problems terminated with integer solutions, satisfying this test of feasibility.

As table 1 indicates, the number of iterations required to reach the optimal solution decreases rapidly and consistently with the increase in the number of facilities (the

9-facility problem, whose low number of iterations apparently spoils the sequence, is an exceptional case—see table 1). As a counter-influence, the time per iteration increases as a function of p. The number of nonzero values in the tableau and the dimensions of the tableau remain constant, being a function of n. Therefore the time increase per iteration must be a function of the value of p alone. The cumulative effect is, however, a very significant reduction in total time with an increase in p.

Table 1. The optimal solution for the Talala area (49 demand nodes).

Number of facilities	Optimum facility sites	Functional value	Number of iterations	Time (seconds)	Time per iteration (seconds)
2	3, 44	461245·3	750	699·0	0·932
4	3, 11, 36, 44	338458·9	745	658·2	0·883
5	3, 11, 29, 36, 44	237610·3	643	586·8	0·913
6	3, 11, 28, 29, 34, 44	240767·7	603	556·8	0·923
9	1, 3, 10, 11, 12, 16, 31, 34, 44	174627·3	358 [a]	362·4	1·012
10	1, 3, 10, 11, 12, 16, 31, 34, 44, 46	156182·3	380	417·0	1·097

[a] The low number of iterations in this case is because the 9-facility problem was run utilizing XEPS, one of the control parameters available to the user of MPSX, to increment the right-hand side by 0·001. The program manual (IBM, 1971) suggests this as a device to improve convergence when there is a large number of zeros on the right-hand side of constraints (3). The results tend to confirm the utility of this control.

6 The Teitz and Bart heuristic

The Teitz and Bart heuristic (and the Maranzana heuristic) were run on an IBM 360/65 under operating system in a partition of 70 K by use of the FORTRAN G compiler. The algorithm was run 75 times for each value of p. The initial locations of the p facilities for each run were generated randomly.

The Teitz and Bart heuristic found the optimal solution in all 75 cases for five of the six test problems (see table 2)[(2)]. In the remaining 4-facility problem, this heuristic found the optimal solution 73 times. The two remaining solutions are

Table 2. Solutions from the Teitz and Bart heuristic for 75 different random starts for each number of facilities for the Talala area (49 demand nodes).

Number of facilities	Best solution found	Number of times found	Worst solution found	Ratio of optimal to best solution	Ratio of optimal to worst solution	Robustness [a]	Time per run (seconds)
2	461245·3	75	461245·3	1·0	1·0	1·0	0·66
4	338458·9	73	339573·0	1·0	0·9967	1·0	1·04
5	287610·3	75	287610·3	1·0	1·0	1·0	1·33
6	240767·7	75	240767·7	1·0	1·0	1·0	1·81
9	174627·3	75	174627·3	1·0	1·0	1·0	2·04
10	156182·3	75	156182·3	1·0	1·0	1·0	2·04

[a] See sections 8 and 9 for a discussion of the robustness measure.

(2) The 75 random starts used on the 10-facility problem reported here are the locations generated by Rushton and Kohler (1973) in their research. Our solutions are identical and the comparison of the average execution times reported there, 0·63 seconds and 5·00 seconds per solution for the Maranzana and Teitz and Bart heuristics respectively, indicate the improved efficiency of ALLOC V over ALLOC (see tables 2 and 3).

identical and are very near the optimum. Figure 1 shows the optimal locations of the facilities for the 4-facility problem, as well as the two suboptimal facility sites from this one solution. The village numbers correspond to those given in table 1.

The Teitz and Bart algorithm moves only one facility at a time to a new location to improve the value of the objective function. Teitz and Bart (1968) indicate that their algorithm could encounter situations which would require simultaneous moves of two or more facilities in order to make such an improvement. The fact that the Teitz and Bart algorithm encountered such a situation in only 0·44% of the runs here reported suggests that these suboptimal 'traps' are rare. Further research is needed to determine the frequency with which these situations occur for varying numbers of demand nodes[3]. From our experience the probability of locating any *one* such trap is considerably smaller than the probability of reaching the global optimum. Our preliminary analysis indicates that there is a unique path (pattern of exchanges) leading into each such trap. There are, however, many paths (patterns of exchanges) leading to the optimal solution. The likelihood then of any single run of the Teitz and Bart heuristic terminating prior to achieving optimality would be a function of the density of 'trapped paths'. Knowledge of the exact characteristics of these relationships would allow one to make probabilistic statements about the likelihood that the best solution found is the optimum.

The average time per problem, as shown in table 2, increases directly with the value of p. This is to be expected since each cycle of the algorithm examines all p facilities to determine whether moving one of them to a demand node would improve the value of the objective function. Therefore each cycle consists of $(n-p)p$ evaluations. The time increase from the 9-facility to the 10-facility case is less than for smaller values of p (a matter of thousandths of a second).

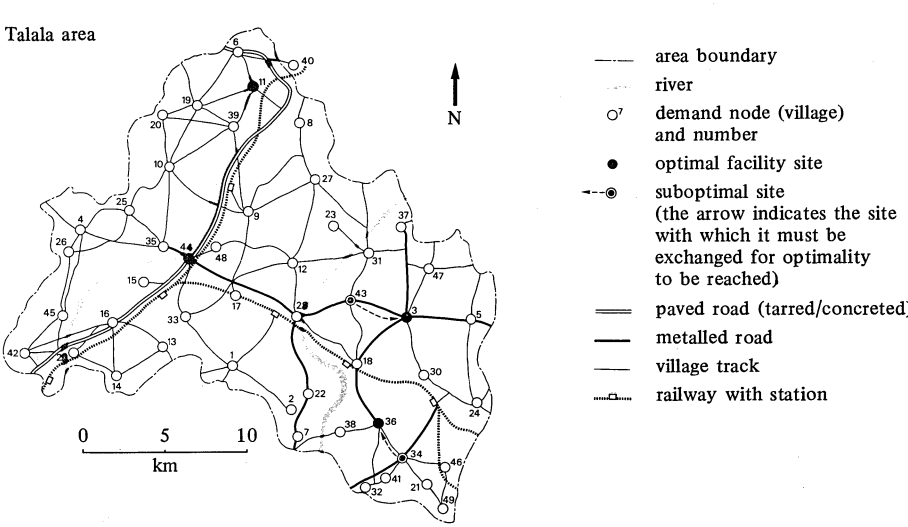

Figure 1. The solution given by the Teitz and Bart heuristic for the 4-facility test problem.

[3] In another, as yet unpublished, study using 25 random starts on a 150-node, 21-facility problem, the heuristic found eight different 'traps' as follows:

aggregate distance	3684480	3695283	3701268	3707246	3712979	3713987	3717124	3717775
number of times found	10	3	1	5	1	1	1	3

7 The Maranzana heuristic

The Maranzana heuristic was run 75 times on each test problem, with the same randomly generated starting locations which were used in the case of the Teitz and Bart heuristic. In the 450 runs the Maranzana heuristic found optimal solutions only 16% of the time, and nearly all of these optimal solutions occurred in the 2-facility problem (see table 3). If the 2-facility problem is excluded, the Maranzana heuristic solved only 0·5% of the remaining cases optimally. On the other hand the Maranzana heuristic did find the optimal solution at least once in three of the six test problems.

It is clear from table 3 that the Maranzana heuristic is most effective in problems involving small numbers of facilities. This appears to be a function of the size of the set of demand nodes which it can search to find a new location for a facility. In searching for a stable partition (Ostresh, 1973), the heuristic assigns each demand node to its nearest facility. Then it tries to move each facility to one of the nodes which it serves in order to reduce the aggregate distance from those nodes. As the number of facilities serving a given number of nodes increases, the number of demand nodes which each facility serves, and thus the number of possible moves considered for each facility, decreases. In turn this apparently reduces the probability of making a move. This results in more stable partitions being found, and hence increases the likelihood that the Maranzana heuristic will terminate before reaching the optimal solution. Although this likelihood is not independent of the size of the set of demand nodes (n), it would seem that the early termination is more directly related to the number of facilities (p) in moderate sized and larger problems, say where $n > 30$ (table 3).

The fact that the Maranzana algorithm found its best solution so rarely for the cases of 4 facilities and above means that multiple runs (multiple starting points) are nearly always needed for an analyst to have any confidence in his solution. This is verified by the ratio of the optimal solution to the mean of the solutions, which moves almost steadily downward, which suggests that some rather poor solutions are among the output. This contrasts with our experience with the Teitz and Bart heuristic, where if a single solution has been found then it is very likely that it is the best solution that will be found.

Table 3. Solutions from the Maranzana heuristic for 75 different random starts for each number of facilities for the Talala area (49 demand nodes).

Number of facilities	Best solution found	Number of times found	Ratio of optimal to best solution	R_b [a]	Mean of solutions	Ratio of optimal to mean of solutions	R_m [a]	Time per run (seconds)
2	461245·3	70	1·0	1·0	473239·0	0·9747	0·9702	0·34
4	338458·9	1	1·0	1·0	354746·1	0·9541	0·9120	0·27
5	288724·4	3	0·9961	0·9950	317697·4	0·9088	0·8658	0·30
6	240767·7	1	1·0	1·0	305491·9	0·7881	0·6167	0·30
9	176707·6	2	0·9882	0·9841	205224·0	0·8509	0·7664	0·34
10	160982·5	1	0·9702	0·9781	216624·9	0·7210	0·6325	0·38

[a] See sections 8 and 9 for a discussion of the robustness measures.

8 Robustness

There is no good index of the efficiency with which a heuristic solves a particular problem. Perhaps this is because few comparisons with optimal solutions have been made. The ratio of the 'best solution found' to the solution from some particular heuristic has been frequently used to compare heuristics in the past (Jarvinen et al, 1972) and, for comparability, values of this ratio are reported in tables 2 and 3, where the optimal solution is used in place of the 'best solution found'. Obviously when the heuristic reaches optimality this ratio is $1 \cdot 0$, and as the heuristic value becomes worse this ratio decreases. We have, however, no knowledge of the lower limit of this ratio scale—it is certainly not zero. Because of this lack of knowledge we cannot interpret the meaning of an incremental change on the scale. Moreover, the location of the lower limit will change with a change in p, which makes comparisons of different problems on one network difficult. Finally, the location of the lower limit will change for each value of n and/or for each network configuration, which makes comparisons between networks impossible.

We define an alternative measure (tables 2 and 3) for comparing heuristic with optimal solutions as follows:

$$R_{\mathrm{b}} = \frac{r - H_{\mathrm{b}}}{r - o} , \tag{8}$$

where

R_{b} is the robustness of the heuristic based on the one best solution,

r is the mean of the objective-function values from randomly generated patterns of facilities,

H_{b} is the best objective-function value from a particular heuristic, and

o is the optimal function value.

In the case of Maranzana heuristic, the robustness ratios were also calculated by use of the mean of the 75 solutions—$R_{\mathrm{m}} = (r - H_{\mathrm{m}})/(r - o)$, where H_{m} is the arithmetic mean of the 75 solutions—instead of the one best solution, and these are reported in table 3. This latter ratio may be more appropriate for measuring the robustness of a heuristic. However, we caution the reader that neither robustness ratio can be used to compare one heuristic solution with another unless optimal solutions are also available.

9 Conclusions

As the robustness measure indicates (table 3), the quality of solutions generated by the Maranzana heuristic gets worse quite rapidly with increases in the value of p. This seems to be a function of the number of stable partitions which it is possible to find and of the proportion of the area to which each facility to be sited has access. It would seem quite dangerous to use the Maranzana heuristic once on a large problem and conclude that its answer is anything more than a stable partition. By making a large number of runs and then selecting the best solution, one can then rely, to some degree, on its being a 'good' stable partition. However, the larger the problem the more questionable it is that the solution is even good.

The Teitz and Bart heuristic is much more robust than the Maranzana. The Teitz and Bart heuristic does not get trapped in the first stable partition, but continues toward the optimal solution until no one-at-a-time change of center location will improve the objective function. This will be a stable partition, but it will generally be better than one found by the Maranzana heuristic. Inspection of the robustness measure indicates that the Teitz and Bart heuristic will find the true optimal solution with a high degree of regularity, at least in problems of the size considered here. We have, however, little knowledge concerning the frequency with which two-at-a-time or higher-order exchanges may be necessary in order to reach optimality in larger problems. The Teitz and Bart heuristic could be used to locate these traps, and their frequency would then give an indication of the likelihood of optimality in large problems. We suspect that occasions requiring two-at-a-time and higher-order changes of center locations are unusual, but we would expect their frequency to increase with n. It would seem reasonable to use the Teitz and Bart heuristic, cautiously, with a large number of starts, and place confidence in the near optimality of the best solution.

It would also seem reasonable to use the number of traps found, and their variety, to temper one's confidence.

Since the Teitz and Bart solution seems quite robust (more work comparing data sets with large values of n is necessary) and the linear-programming approach so expensive, we suggest that the proof of optimality obtained from the latter is worth the expense in relatively few cases. Instances of this might be (1) analyses involving very large investment decisions (Odell and Rosing, 1976, chapter 7 and appendix A; Odell et al, 1976), or (2) detailed sensitivity analyses.

Acknowledgements. This research was conducted while the authors were at the University of Iowa. We wish to acknowledge the support of the Graduate College of the University of Iowa, which met the costs incurred for computer time, Spring 1977. We should like to thank Thomas Wilbanks of the Oak Ridge National Laboratory and an anonymous referee for their critical reading of our manuscript and their constructive comments. Our paper is much improved through their work.

References

El-Shaieb A M, 1973 "A new algorithm for locating sources among destinations" *Management Science* **20** 221-231

Garfinkel R, Neeb A, Rao M, 1974 "An algorithm for the *M*-median plant location problem" *Transportation Science* **8** 217-236

Hillsman E L, 1979 "User's guide to ALLOC IV, ALLOC V, and ALLOC VI: heuristic algorithms for solving *p*-median problems" Institute of Urban and Regional Research, University of Iowa, Iowa City, Iowa (forthcoming)

Hillsman E L, Rushton G, 1975 "The *p*-median problem with maximum distance constraints: a comment" *Geographical Analysis* **7** 85-89

IBM, 1971 *Mathematical Programming System—Extended (MPSX), Control Language Users Manual* Program 5734-XM4, IBM, White Plains, NY

IBM, 1972 *Mathematical Programming System—Extended (MPSX) and Generalized Upper Bounding Programming Description* Program 5734-XM4, IBM, White Plains, NY

Jarvinen P, Rajala J, Sinervo H, 1972 "A branch and bound algorithm for seeking the *p*-median" *Operations Research* **20** 173-178

Maranzana F E, 1964 "On the location of supply points to minimize transport costs" *Operational Research Quarterly* **15** 261-270

Odell P R, Rosing K E, 1976 *Optimal Development of North Sea Oil Fields* (Kogan Page, London)

Odell P R, Rosing K E, Vogelaar H, 1976 "Optimising the oil pipeline system in the UK sector of the North Sea" *Energy Policy* **4** 50-55

Ostresh L, 1973 "An investigation of the multiple location allocation problem" unpublished PhD thesis, Department of Geography, University of Iowa, Iowa City, Iowa

ReVelle C S, Church R, 1980 *Design of Location Systems* (Pergamon Press, New York) forthcoming

ReVelle C S, Rosing K E, 1979 "Planning of oil pipeline systems at sea: an application of a branch and bound *p*-median algorithm" *Naval Research Logistical Quarterly* (forthcoming)

ReVelle C S, Swain R, 1970 "Central facilities location" *Geographical Analysis* **2** 30-42

Rosing K E, ReVelle C S, 1978 "Clustering Dutch water transportation zones: a *p*-median approach" WP A-78-5, Economisch-Geografisch Instituut, Erasmus Universiteit, Rotterdam

Rushton G, Kohler J A, 1973 "ALLOC—heuristic solutions to multi-facility location problems on a graph" in *Monograph 6. Computer Programs for Location Allocation Problems* Eds G Rushton, M F Goodchild, L M Ostresh Jr (Department of Geography, University of Iowa, Iowa City, Iowa)

Swain R, 1974 "A parametric decomposition algorithm for the solution of uncapacitated location problems" *Management Science* **21** 189-198

Teitz M B, Bart P, 1968 "Heuristic methods for estimating the generalized vertex median of a weighted graph" *Operations Research* **16** 955-961