

(wp)
HB
34
-0474
1992

UNIVERSITAT POMPEU FABRA 

*The PQ-Median Problem: Location and Districting of
Hierarchical Facilities. Part II: Heuristic Solution
Methods*

Daniel Serra
and
Charles ReVelle

Economics working paper 13. April 1992

UNIVERSITAT POMPEU FABRA
BIBLIOTECA

The PQ-Median Problem: Location and Districting of Hierarchical Facilities Part II: Heuristic Solution Methods*

Daniel Serra
Universitat Pompeu Fabra
and
Charles ReVelle
The Johns Hopkins University

April 1992

Abstract

The pq-median problem seeks to locate hierarchical facilities so as to obtain a coherent structure. Coherence requires that the entire area assigned to a facility at one level must be assigned to one and the same facility at the next higher level of the hierarchy. Although optimal solutions to the pq-median problem have been obtained by a combination of linear programming and branch-and-bound, large problems are likely to require heuristic approaches for the foreseeable future. This paper proposes several efficient heuristic methods whose solution properties appear to be quite good when compared to those of exact procedures.

*This Research was funded by the Institut Municipal d'Investigacions Mèdiques (IMIM), Barcelona-Spain and the Fondo de Investigación de la Seguridad Social (FISS), grant # 88/1493. Computation was conducted in part using the Cornell National Supercomputer Facility, a resource of the Center for Theory and Simulation in Science and Engineering at Cornell University, which is funded in part by the National Science Foundation, New York State, and the IBM Corporation and Members of the Corporate Research Institute. The authors are grateful to FISS and IMIM for the funds given, and to CNSF for the access to the Cornell facility which has helped to make this work possible.

UNIVERSITAT POMPEU FABRA
BIBLIOTECA



1001961093

1 Introduction: The PQ-Median Problem

It is widely accepted that many facility systems and institutions are hierarchical in nature. Consider, for example, a regionalized health delivery system, where there is an integration and coordination of several levels of health services so that they operate as an articulated, graded single system. The conceptual design consists of a number of hierarchical levels providing differentiated services with each facility of the hierarchical level covering a specific geographical area, and linked to centers in the next level of the hierarchy. As we go "up" in the hierarchy, the geographical areas assigned to the centers in lower levels are completely contained in the area of a center in the next higher level, where there is also a link between these centers (e.g., referral of patients). Coherence, in the context of hierarchical systems, requires that the entire area assigned to a facility at one level must be assigned to one and the same facility at the next higher level.

Another example of a coherent structure is the political division of a country that is divided into regions, that are themselves divided into provinces, which are divided into municipalities. It is hard to think of a municipality belonging to two different provinces.

To achieve an efficient and effective hierarchical system, it is necessary to obtain not only an efficient set of locations but also an effective districting of the catchment areas, since these areas will be the ones to benefit from the services provided by the located facility. While the research on facility location is rich in models aimed at siting hierarchical facilities, there is little research that combines the location and districting of these facilities. Location and districting models have traditionally been studied separately.

The pq-median problem (Serra and ReVelle, 1991) seeks the location of two levels of hierarchical facilities and, at the same time, districts the demand areas utilizing these facilities, achieving in the districting process a "coherent structure". It is a multiobjective model where, at each level, average distance to the facilities is minimized.

Recent computer developments allow solution today of quite large p-median problems using not only variants of the revised simplex algorithm plus branch and bound but also using Lagrangian methods (Rosling et al. 1979, Galvao and Raggi 1989). But when computing resources and machines were scarce, several heuristics and optimal methods were developed to try to find locations that, if not optimal, would provide a good solution to the problem at hand. We face today in the need to solve the pq-median problem similar obstacles to those faced in the solution of the p-median problem two decades ago. Therefore, due to the considerable size of the linear programming model of the pq-median formulation even for relatively small problems, other methods have been sought in order to find good solutions. A brief review of heuristic solution methods for location problems is presented next. Then, four heuristics aimed at solving the pq-median are described and compared.

2 Heuristics for the P-Median Problem and Related Location Problems

The first heuristic for solution of the related uncapacitated plant location problem was developed by Kuehn and Hamburger in 1963. Since then, a myriad of heuristics using different approaches addressed to locational problems have been studied and proposed. Maranzana (1964) developed a heuristic for the warehouse location problem in which the number of warehouses was specified in advance. His heuristic began by initially siting p facilities on the network and then dividing the network into p subsets, each one associated with a facility. Successive facility relocation within the subset, followed by redivision of the points into clusters, produced stable solutions.

One of the classical heuristics used to solve p -median problems is the Tietz and Bart algorithm (1968). In the Tietz and Bart algorithm facilities are systematically interchanged between occupied nodes and unoccupied nodes. At each interchange, the objective is computed, and only trades that improve the objective are considered. One-opt trades continue until no better value of the objective is found. This algorithm has been demonstrated to be both efficient and robust by Diehr (1972), Cornuejols et al. (1977), Rosing et al. (1974), and Hodgson (1986).

Cornuejols et al. (1977) proposed a greedy heuristic consisting of the selection of the first median of the network and then the location, one at a time, of the facility that minimizes the sum of the weighted distances, until p locations are obtained. The authors used this greedy heuristic as the initial solution for the Tietz and Bart's heuristic and obtained good results for the p -median problem with reasonable computational time. They were able to solve quite large problems, with $m=n=100$, in less than ten seconds on an IBM 370/168 (Francis et al. 1983).

Narula et al. (1977) used Lagrangian relaxation of the linear programming formulation and the method of reduced gradients to derive solutions. Since then, there have been a number of attempts to improve the Lagrangean relaxation technique (see for instance, Geoffrion and McBride 1978, Cornuejols et al. 1977, Guignard 1988).

Several heuristics based on the dual of the linear programming version of the p -median model have been used to solve the problem. Diehr (1972) showed similar results to those obtained by the Tietz and Bart heuristic with this approach. Bilde and Krarup (1977), Erlenkotter (1978), and Guignard and Spielberg (1979) used dual ascent procedures to obtain near optimal solutions to the dual of the uncapacitated facility location problem.

The coherent hierarchical pq -median model discussed in the introduction is very computationally intensive to solve using linear programming relaxation and branch and bound or by dual heuristics since it has a very large number of variables and constraints. If the problem were to locate, as an example, 10 A level facilities and 3 B level facilities in a 100-node network, the fully specified problem would have at least 30000 variables and more than a million constraints. It is necessary, as a consequence, to find a heuristic with reasonable computing and storage time that is capable of dealing with the coherence feature of the model. Another problem in solving the pq -median coherent formulation is its multiobjective setting, which increases considerably the computing time in order to obtain

a tradeoff between the objectives.

Observe that one methodological approach to solution of the coherent hierarchical problem is to locate each level independently of the other in a successive manner starting from the top of the hierarchy and proceeding down to the bottom or starting from the bottom of the hierarchy and proceeding up to the top. These types of models have been referred to as top-down or bottom-up. Banergi and Fisher (1977), and Fisher and Rushton (1979) used the p-median model to solve each level of a hierarchy independently. Lea (1978), Narula (1981) and Hodgson (1984) demonstrated that this approach of locating hierarchical facilities would generally produce inferior results to simultaneously locating all levels. The simultaneous approach employed by Hodgson produced better solutions than either the top-down or the bottom-up methods, and the bottom-up approach generally outperformed the top-down. Hodgson observed that this could be due to the much higher weighting applied to the usage of low-level facilities.

Traditional top-down methods optimize the top level location, but the enforced use of such locations as low-level centers produces systems in which the lower level solution is generally inferior to the lower level solution obtained if the location was done without considering the siting of top level facilities. In some cases, the suboptimality of the low-order system outweighs the advantages of the top-level locations resulting in hierarchies that are inferior overall.

Similarly, traditional bottom-up methods generate the best low-order locations because of their unconstrained goal of optimization at that level, but tend to produce very bad results in the location of higher-level facilities. The restriction imposed by choice from only few lower-level potential locations gives worse results at the highest level, and overall, the advantages derived at the lower levels are generally outweighed by the suboptimality of the highest-level location (Hodgson, 1984).

It is necessary then to solve the hierarchical coherent problem using some kind of heuristic that effects a compromise between the top-down and bottom-up heuristics. The pq-median problem formulated here is multiobjective. Each hierarchical level has associated with it an objective of weighted distance to be minimized. This means that in order to find a solution, a compromise between the objectives at both levels has to be found. In the following sections several simple multiobjective heuristics are described. These heuristics are based on the Teitz and Bart algorithm and the top-down and bottom-up methods for the coherent hierarchical model with two levels.

3 Bottom-up Heuristic

The first Solution Algorithm for the P-median **HIER**archical problem (SAPHIER1) uses a modified Teitz and Bart heuristic together with a bottom-up procedure to locate in an n-node network p facilities that offer type A services and q facilities that offer type A and type B services (i.e., successively inclusive services), where $p > q$, and where coherence is observed. The facilities that offer both type A and type B services are referred to as type B facilities. Two objectives are considered, one for each level. The first one

minimizes the average distance (or the total population weighted distances) to type A services. The second one minimizes the average distance to type B services. The modified Tietz and Bart heuristic is used to improve the location of the type B facilities, once the facilities offering type A services have been located optimally. In order to have only one objective so successive solutions from the algorithm can be compared, both objectives are weighted and added. The weights will be such that $w_B = (1 - w_A)$, where w_A and w_B are the weights associated with the population-weighted distance to type A services and the population-weighted distance to type B services. The larger the weight associated with the type B objective (and therefore the lower the weight associated with the type A objective is), the more likely is the final solution to be different from the initial one, since at the beginning the location of type A facilities is optimal since this is a bottom-up methodology.

The SAPHIER1 procedure is iterative, and the first iteration has two phases. In the first phase, $p + q$ facilities offering type A services are located optimally or heuristically, depending on the technique used to do so. The Tietz and Bart heuristic can be used as a method to obtain these locations, using a greedy heuristic for the starting solution, as in Cornuejols (1977). The p -median model using the revised simplex method and branch and bound when necessary can also be used as the first phase solution method for SAPHIER1 as can Lagrangian relaxation. The location of $p + q$ facilities will determine $p + q$ districts² that will be called supernodes. These supernodes will form a second network, where instead of having distances between its supernodes, there will be a coefficient g_{jk} that expresses the relation between supernode j and supernode k . The coefficient g_{jk} is made up of the sum of the products of the population of each demand area in j times the distance from that demand area to the facility that is associated with supernode k . Then, in the second phase of the first iteration, type B services are additively located at the facilities in q supernodes of the $(p + q)$ -supernode network. The location method used can be again the Tietz and Bart heuristic or the revised simplex method with branch and bound when necessary, or a Lagrangian relaxation method. The B objective will be to minimize the sum of the demand weighted distances to the q type B services. That is:

$$\text{Min } Z_B = \sum_{j=1}^{p+q} \sum_{k=1}^{p+q} g_{jk} x_{jk}$$

where:

$$g_{jk} = \sum_{i \in O_j} a_i d_{ik}$$

and O_j is the set of nodes i assigned to facility offering type A services located at node j , and d_{ik} is the distance from node i to the facility that serves supernode k . The variable x_{jk} is one if supernode j is assigned to the facility at supernode k and zero otherwise. The set

2 Each demand area assigns to its closest facility, and the district consists of the set of demand nodes assigned to a particular facility.

of constraints for this problem will be the traditional q-median constraints.

After solving this problem, the facilities associated with the supernodes that did not obtain *type B services, correspond to type A facilities. Facilities with type A and type B services* correspond to type B facilities. Therefore, there will be p type A facilities and q type B facilities located in the initial network. Coherence is observed since all nodes assigned to a type A facility are assigned to one and the same type B facility by virtue of use of the supernodes and the g_{jk} coefficients. The assignment of supernode j to supernode k means that all nodes assigned to the type A facility associated with supernode j will be assigned to the type B facility associated with supernode k.

The solution found so far could correspond to a traditional bottom up heuristic with the additional feature that coherence is observed. Average distance to type A services is optimal. On the other hand, average distance to type B services is most likely not optimal, since some areas may not be assigned to their closest type B facility for type B services (due to coherence). They are also not likely to be optimal because the potential locations for type B services were reduced to a few nodes (the p + q facility sites picked in the first step) instead of using all n nodes of the network.

SAPHIER1 seeks a trade-off between average distance to type A services and average distance to type B services. A set of weights has to be arbitrarily determined so a weighted objective can be used as the decision rule in choosing solutions. If the weights are chosen so that $w_A = 1$ and therefore $w_B = 0$, the solution obtained in the first phase will be the final one, since the only objective is the location of facilities offering type A services, and these are already optimally sited. For any other set of weights, SAPHIER1 will seek an improvement of the weighted objective. It may be possible that, for a given set of weights, by relocating a facility offering type B services the reduction in the B objective will offset the increase in the A objective, since facilities offering type B services also offer type A services. In this case, a better weighted objective is obtained.

The next iterations seek improvement in the solution of the pq-median problem according to the set of weights chosen. The weighted objective for the initial solution has been computed and stored. The procedure used to obtain an improvement of the weighted objective (if any can be obtained) is similar to the Teitz and Bart heuristic. First, one of the p + q facilities offering either type A or A+B services is relocated to a node that does not have a facility. The new A objective (average distance to type A services) is computed. Observe that this objective will most likely be worse than the initial A objective, since the location of type A services has been modified, and prior to this modification it was optimal.

Now the supernodes are computed as in the initial solution. Each facility (both type A and type B) that offers type A services will have a supernode associated with it. This supernode consists of all areas assigned to the facility for type A services. There will be again p + q supernodes. The coefficients g_{jk} are computed for all pairs of supernodes. Then type B services are relocated in q of the p + q supernodes and the B objective (average distance to type B services) computed. The new weighted sum of objectives is computed (w_A x average distance to type A services + w_B x average distance to type B

services). The new B objective may be smaller or larger than the one found in the previous solution. If it is smaller, it may be possible that the weighted sum of objectives will be better than the previous one. If this is the case, a new solution for the pq-median problem is found according to the set of weights specified. If the new weighted objective is not better than the previous one, then this solution is ignored and the previous solution is restored as the current solution.

This procedure is repeated, as in the Teitz and Bart heuristic, for all facilities and nodes. At the end of each iteration the weighted objective is computed and compared to the previous best solution. When all one-opt relocations have been tried, the final solution is compared to the solution found before the relocations started. If the best solution found differs from the initial solution, the one-opt relocation procedure is started again with the best solution as the initial one. On the other hand, if both solutions are equal, SAPHIER1 is completed and the solution is final for the set of weights specified. A new set of weights can be determined and the search for a new best weighted solution can be performed in the same fashion. A step-by-step description of SAPHIER1 follows:

SAPHIER 1

1. Set a weight w_A . Set $Z_w = M$, a very large number.
2. Locate $p + q$ facilities offering type A services using a p-median methodology, exact or heuristic where p is the number of facilities offering only type A services and q is the number of facilities offering type A and type B services.
3. Compute the weighted distance to the $p + q$ facilities offering type A services (Z_A)
4. For each facility offering type A services, create a district consisting of all demand areas assigned to it. The district will be called a "supernode". There will be $p + q$ supernodes.
5. For each pair of supernodes compute the coefficient g_{jk} , consisting of the sum of the demand weighted distances of each demand area in supernode j to the facility offering type A services at supernode k .
6. Locate type B services at q of the $p + q$ supernodes, using an exact p-median method or a p-median heuristic, where the objective (Z_B) is to minimize the average distance from the demand areas in each supernode to type B services.
7. Compute the weighted sum of both objectives ($Z_w = w_A Z_A + (1-w_A) Z_B$) and store the solution.
8. Compare the new Z_w with the old Z_w . If the new solution is better, accept the new solution and go to step 9. If the old solution is better and a full cycle of exchanges has not been completed, store it and go to step 9. If a full cycle of exchanges has been completed, and the old solution is still better, stop.

9. Relocate to a currently empty node one facility out of the $p+q$ facilities offering type A services as in the Teitz and Bart heuristic and re-compute Z_A .

10. Repeat steps 4-7.

Observe again that the use of supernodes to locate type B services is crucial to the SAPHIER1 procedure, since it preserves the characteristic of coherence. The creation of supernodes, formed by the areas assigned to each facility offering type A services, allows type B facilities to be located with coherence, since they will only be allowed to locate at nodes which are already chosen for facilities offering type A services, that is, on top of the $p + q$ already located facilities. In addition, each supernode will be assigned to a facility offering type B services. Therefore all areas included in the supernode (and therefore assigned to the same facility for type A services) will be assigned to one and the same facility offering type B services. Since type B facilities offer type A services, the final result will be the location of p type A and q type B facilities.

4 Top-Down Heuristics

Top-down heuristics generally start locating the top level of the hierarchy (the one with the least facilities in a pyramidal hierarchy) and proceed to successively locate each lower level. In this section three similar top-down heuristics are presented in the context of the coherent location of hierarchical facilities.

Most top-down methods are useful for allocating hierarchical facilities with successively inclusive services, but do not offer trade-offs between the levels of the hierarchy. The top level is first located, and will have an objective value associated with it. Then the next level is located, subject to the location of the top-level facilities. This objective is most likely to be non-optimal from the standpoint of that particular level, since it is constrained by the top level locations. The heuristics presented in this section search trade-offs between the average distances to the facilities at each level of the hierarchy. These heuristics are aimed at locating hierarchical coherent facilities with successively inclusive services.

Again p type A facilities and q type B facilities need to be located, where $p > q$. The starting point of the top-down heuristics described here is the optimal location of the higher-order facilities, i.e., type B facilities, which offer both types of services. Any solution method such as Teitz and Bart or the revised simplex method with branch and bound or Lagrangian relaxation can be used. Once the facilities offering type B services are located, districts are formed. Each district consists of the areas assigned to each type B facility. Consequently, there are q districts, one for each type B facility. Now that q level B facilities are located and the districts formed, it is necessary to know how many of the type A facilities need to be located in each district and where they have to be sited.

The use of districts is necessary to observe coherence. Since all demand areas assigned to a type A facility have to be assigned to one and the same type B facility, by locating the type A facilities within the districts, coherence will be observed. This districting problem

can be solved using several different approaches.

Observe that now there is already one fixed facility offering type A services in each district, since the services are successively inclusive. In this case, the objective in each district may be non-convex, at least when few facilities are located (see ReVelle and Elzinga, 1988). Since type B facilities offer type A services, each district formed will have a fixed facility offering type A services. This facility has to be considered as the first facility offering type A services, and cannot be removed when two or more facilities offering type A services are located, because it is the basis for district formation. Since this facility is fixed (it determines the district) and the facility is not necessarily sited at the best position for a single facility offering type A services, the weighted distance function with respect to the number of facilities offering type A services may be non-convex.

If there are fixed facilities, deterministic dynamic programming can be used to solve the problem, where the stages are the districts and the states are the number of facilities to be located in each district. An alternative is to use lagrangian relaxation of the linear formulation of the problem (Everett 1963). Both procedures can become expensive as the number of districts and locations becomes relatively large. A good discussion of both methods can be found in ReVelle and Elzinga.

A third solution in the case of non-convexities of the objective function is to modify the distance matrix to account for districting and then use any p-median heuristic or optimal procedure to solve the problem. The modification is done in such a way that the distance between areas and potential facility sites that do not belong to the same district is set to a very large number. Therefore, assignments between these areas should not be obtained in the final solution. Once the matrix is modified with such distances, a normal p-median problem can be solved to locate the facilities in the whole region. The solution method can be the revised simplex algorithm with branch and bound, but this might get computationally intensive for relatively large networks. Alternatively, the Tietz and Bart heuristic or lagrangian relaxation can be used to solve the problem.

Both top-down heuristics presented in this section will start with the optimal location of type B facilities using a p-median formulation that can be solved by any conventional solution method. The top-down heuristics will also determine the districts by assigning the population areas to the closest type B facility. On the other hand, each heuristic will use a different method to allocate facilities offering type A services among districts. SAPHIER2 will use the distance matrix modification method. SAPHIER3 will use dynamic programming to find the optimal number of facilities that offer type A services to locate in each district. Then both top-down heuristics will try to improve the location of type A facilities and obtain a trade-off between average distance to type A facilities and average distance to type B facilities. The formal steps of the SAPHIER2 heuristic are as follows:

SAPHIER 2

1. Set a weight w_A reflecting emphasis on the A objective, and a weight w_B , reflecting emphasis on the B objective, such that $w_A + w_B = 1$
2. Locate optimally q type B facilities using a p -median methodology.
3. Compute the objective value Z_B and store it.
4. Form districts by assigning each population area to its closest type B facility.
5. Modify the distance matrix to reflect the districts just created, so that the distance between demand areas which are in different districts is set to a very large number.
6. Locate p type A facilities with the new distance matrix using a p -median formulation and store the A objective (Z_A)
7. Compute the weighted objective (Z_w), where $Z_w = w_A Z_A + (1 - w_A) Z_B$ and store the solution. If the new objective is smaller, store the solution obtained after the trade and its objective. If not, restore the old solution. If all possible one-opt trades for each type B facility have been done, and no improvement has occurred, stop. If not, go to step 8.
8. Trade the location of one of the type B facilities located in step 1 to one of the currently unoccupied positions, as in the Teitz and Bart heuristic.
9. Store final solution for weight w_A and go to step 1.

The heuristic SAPHIER3 uses dynamic programming to allocate p type A facilities among q districts. As in SAPHIER2, the first step is the optimal location of q type B facilities using any of the mentioned solution methods. Once located, each facility supplying type B services will have a corresponding district, composed of all areas assigned to each type B facility. Therefore, there will be q type B districts in the region of study.

The next step is to locate the facilities offering type A services only. Another method than the modification of the distance matrix employed in SAPHIER2 to allocate facilities to districts is the use of a deterministic dynamic program where the number of facilities to locate are states and the stages are given by the districts. This dynamic problem can be cast as an integer program as follows:

$$\text{Min } Z = \sum_{k=1}^q f_k(y_k)$$

subject to:

$$\sum_{k=1}^q y_k = p$$

where p is the number of type A facilities to locate, q is the number of districts, y_k is the number of facilities to locate in each district k ($y_k = 1, 2, 3, \dots, p$), and $f_k(y_k)$ is the total weighted distance in district k from demand areas to their closest facility offering type A services when y_k facilities are located in that district.

Observe that the first facility offering type A services and the associated weighted distance to it from all population areas in each district is already known. These facilities correspond to the type B facilities already located, since these facilities also offer type A services. Therefore, prior to solving the dynamic program, one must find for each district the value of the weighted distance when 2 to p facilities offering type A services are located, that is, $f_k(y_k)$, and such that one facility offering type A services -the type B facility- is fixed. This can be done by solving for each district a modified y -median problem for $y = 2, \dots, p$ to take into account the fixed type B facility offering type A services. Any of the standard solution methods discussed can be used to solve this problem, since the fixed facility means the addition of only one constraint that does not severely affect the solution procedure. These problems are much smaller in size, the size of only the districts.

Once $f_k(y_k)$ is known for $y_k = 1, \dots, p$ in all q districts, the deterministic dynamic program can be solved using any standard recursive method. Once this solution is known the locations of the facilities offering type A services are also known for each district since they were found in order to obtain the weighted distances.

At this stage the value of the objectives and the locations corresponding to both type A and type B levels are known. As in SAPHIER2, average distance to facilities offering type B services is minimum, unlike the average distance to type A services, since the location of facilities offering the type A services was constrained by the existence of districts. The heuristic SAPHIER3 will try to improve the location of facilities offering type A services. This cannot be done without degrading the average distance to type B services. The formal steps of the SAPHIER3 heuristic are as follows:

SAPHIER3

1. Set a weight w_A corresponding to the A objective, and a weight w_B corresponding to the B objective, such that $w_A + w_B = 1$

2. Locate optimally q type B facilities using a p -median formulation
3. Compute the objective value Z_B and store it.
4. Form districts by assigning each population area to its closest type B facility.
5. Locate, for each district, 2 to p facilities offering type A services using a p -median methodology, and such that one facility offering type A services -the type B facility- is fixed.
6. Use dynamic programming to find for each district the optimal number of facilities offering exclusively type A services.
7. Compute the weighted objective (Z_w), where $Z_w = w_A Z_A + (1 - w_A) Z_B$ and record the solution. If the new objective is smaller, store solution obtained after the trade and its objective. If all possible one-opt trades for each type B facility have been done, and no improvement has occurred, stop. If not, go to step 8.
8. Trade the location of one of the type B facilities located in step 1, as in the Teitz and Bart heuristic. Repeat steps 3 to 7.
9. Store final solution for weight w_A and go to step 1.

5 Top-Down Bottom-Up Heuristic

A heuristic that uses features of the top-down and bottom-up heuristics described so far is presented here. This heuristic, SAPHIER4, combines ReVelle and Elzinga's greedy procedure (1988) to find the number of type A facilities to locate in each district with the Teitz and Bart heuristic. As in both SAPHIER2 and SAPHIER3 heuristics, the first step is the median location of q type B facilities using any solution method, such as the linear programming with branch and bound where necessary, or the Teitz and Bart heuristic. Once located, each facility supplying type B services will have a corresponding district, composed of all areas assigned to each type B facility. Therefore, there will be q type B districts in the region of study. The initial locations of the q type B facilities have to be stored because they will be later used in the heuristic. However, for the steps that we describe next, these locations are not taken into account, and only the districts formed by them are utilized.

After the q districts are designed, $p + q$ facilities offering type A services are located among the q districts. The heuristic SAPHIER4 will use the procedure designed by ReVelle and Elzinga to locate facilities offering type A services in the districted region. Basically, the first step of this procedure is to find, using any p -median type formulation, for each of the q districts, the optimal location of 1 to p facilities offering type A services, since there are already q facilities offering type A services that are already located, and the corresponding objectives of weighted distance; then find, again for each district, the reduction in the weighted distance when going from t to $t + 1$ facilities, where $t = 1, \dots,$

$p - 1$, since each district starts with one facility offering type A services already located. The next step is to assign incremental facilities among districts by searching for the largest reduction in the weighted distance. First, one type A facility is conceptually allocated to each of the districts. Its position is not specified unless it is the only facility finally allocated to the district. The next facility to be allocated among the districts will be directed to the district which gives the largest reduction in the weighted distance when going from one to two facilities allocated to it. The next allocation of a facility to any district goes to that district which has the next largest reduction in the costs. The procedure is repeated until $p + q$ facilities offering type A services are located in the q districts. A more complete description of this greedy algorithm is found in ReVelle and Elzinga. Observe that, unlike in SAPHIER2 and SAPHIER3, the $p + q$ facilities offering type A services are located without forcing q of them to be located where the type B facilities were.

Therefore, now it is necessary to find which of these facilities are going to offer type B services, while observing coherence. SAPHIER4 will use the same procedure as in SAPHIER1 to solve this problem. The $p + q$ facilities offering type A services already located determine $p + q$ districts, called supernodes. These supernodes will form a second network, where instead of having distances between its supernodes, there will be a coefficient g_{jk} that consists of the sum of the population at each demand area in supernode j times the distance from this area to the facility associated with supernode k . Then, in the second phase of the first iteration, type B services are located in q supernodes of the $(p + q)$ -supernode network. The location method used can be again the any of the standard p -median methods. The B objective will be such that:

$$\text{Min } Z_B = \sum_{j=1}^{p+q} \sum_{k=1}^{p+q} g_{jk} x_{jk}$$

where:

$$g_{jk} = \sum_{i \in O_j} a_i d_{ik}$$

where variables and notation are described in section 3. The set of constraints for this problem will be the traditional p -median constraints.

After solving this problem, the facilities associated with the supernodes that did not obtain type B services, correspond to type A facilities. Facilities with type A and type B services correspond to type B facilities. Therefore, there will be p type A facilities and q type B facilities located in the initial network. Coherence is observed since all nodes assigned to a type A facility are assigned to one and the same type B facility by virtue of the supernode organization of the network and the g_{jk} coefficients. If supernode j is to receive B type services from supernode k , all nodes assigned to the type A facility in supernode j will be assigned to the type B facility in supernode k .

At this stage the value of the objectives and the locations corresponding to both type A and

type B levels are known. Observe that the B level might not be optimally located, since the location of the facilities offering type B services was subject to the position of the $p + q$ facilities offering type A services. Similarly, the location of the type A facilities might not be optimal, since it was constrained by the districts previously designed. Therefore, the solution so far is a compromise between both levels. The formal steps of the SAPHIER4 heuristic are as follows:

SAPHIER4

1. Set weights w_A and w_B for each objective, such that $w_A + w_B = 1$
2. Locate optimally q type B facilities using a p -median formulation, and form districts. Store Z_B .
3. Locate p type A facilities in the districted region using ReVelle and Elzinga's algorithm. Observe that q facilities offering type A services have already been located. Form the corresponding $p + q$ supernodes.
4. Locate q type B facilities among the $p + q$ supernodes, as in SAPHIER1, using a p -median formulation, compute the new weighted objective (Z_w), where $Z_w = w_A Z_A + (1 - w_A)Z_B$, and store the solution.
5. Form districts corresponding to the new location of the q type B facilities.
6. Locate p type A facilities using ReVelle and Elzinga's algorithm, store new A solution Z_A and form $p + q$ supernodes. Then locate q type B facilities in the $p + q$ supernodes. Compute the new weighted objective Z_w' .
7. Compare the new weighted objective (Z_w') with the previous one (Z_w) obtained in step 4. If Z_w' is smaller ($Z_w' < Z_w$), store solution obtained after the trade, set $Z_w = Z_w'$ and go to step 5. If the new solution is equal or larger, discard it and continue to step 8.
9. Trade the location of one of the type B facilities located in step 2, as in the Teitz and Bart heuristic, and continue to step 3. Repeat until all possible one-opt trades for each type B facility have been done
10. Store final solution for weights w_A and w_B and goto step 1. Generate all necessary solutions using different weights.

6 Results and Computational Experience

The pq -median problem was solved on two test networks using the four different methods described above. The first network had 25 nodes and was used to compare the four SAPHIER heuristics among themselves and to compare them to the optimal solution obtained by using the revised simplex algorithm with branch and bound when necessary.

The second network had 79 nodes and only the heuristics were used to obtain solutions, and the solutions could only be compared to one-another. In both cases several different numbers of type A and type B facilities were used. Results and run times are described for both networks.

The routines and subroutines used for all heuristics have been coded in IBM FORTRAN computer language version 2, which is a vectorized version of FORTRAN that takes advantages of the IBM 3091 E-600 supercomputer facilities used.

Different combinations of type A and type B facilities were used in evaluating the performance of the heuristics. To compare them the problem was first solved using the relaxed linear program plus branch and bound as needed. The programs were solved using a commercial software package, MPSX/MIP³. Therefore, optimal solutions could be obtained for comparison. The full specification of the 25-node problem involved 1,925 integer variables and more than 17,500 constraints. The constraint set on coherence alone had 25^3 elements. In order to reduce the problem size, the number of assignment variables were cut down using the Rosing et al. (1978) approach. The weighting method was used to solve the multiobjective problem. In order to identify the non-inferior set in the objective space, several runs were made with different sets of weights for each combination of type A and B facilities.

The initial step of the SAPHIER1 algorithm was the optimal location of $p + q$ type A facilities using a p -median formulation. These $p + q$ locations were found using the Teitz and Bart algorithm instead of solving a linear relaxation of the p -median model with branch and bound. Even though optimal solutions with this method are not ensured, this was done in order to have a homogeneous approach to the problem. The initial locations for the Teitz and Bart heuristic were computed using the greedy adding heuristic developed by Kuehn and Hamburger.

Once the A locations were found, the initial B locations were computed using again a one-opt approach with a greedy adding heuristic to determine the starting locations. In order to observe coherence only the $p + q$ facilities with type A services already located were candidates to have a type B facility.

The initial locations for the top-down heuristics SAPHIER2 and SAPHIER3, and the top-down bottom-up heuristic SAPHIER4 were found using a similar procedure to the one used for the SAPHIER1 algorithm. The first step in these three heuristics is the location of q type B facilities following a p -median formulation. These facilities were located using the Teitz and Bart heuristic with Kuehn and Hamburger greedy-adding procedure for the starting B locations. Then the initial locations of the $p + q$ facilities offering type A services were found using the same greedy-adding heuristic but with the constraint that

3 MPSX: Mathematical Programming System EXtended/370. MIP: Mixed Integer Programming/370. This software was developed by IBM.

demand areas could assign only to a type A facility that was in the same district, and that q of these facilities had their sites fixed, since q type B facilities were already located, and these facilities also offer type A services. Observe that at the beginning of the greedy heuristic there is a facility offering type A services in each district, since type B facilities also offer type A services.

During the iterative process SAPHIER3 always solves for the location of facilities offering type A services in the districts formed by the B facilities with dynamic programming. To do so, for each district, 2 to p facilities offering type A services are located and their associated weighted distance is computed, with the constraint that one facility offering type A services is fixed, since the associated type B facility in each district also offers type A services. All these p -median problems for each district were solved using the Teitz and Bart heuristic with the additional constraint that one facility was fixed. In each problem, the initial solution was Kuehn and Hamburger's greedy adding heuristic with the additional constraint of a fixed facility.

SAPHIER4 solves the location of type A facilities in a districted region using ReVelle and Elzinga's algorithm. This algorithm requires, for each district, the location of 1 to p facilities offering type A services using a p -median model. All these problems were solved using the Teitz and Bart heuristic with Kuehn and Hamburger's greedy adding heuristic as the initial solution, but any method could be used for all these different p -median problems: the revised simplex algorithm with branch and bound when necessary, one-opt heuristics, or Lagrangian relaxation methods. Then, ReVelle's and Elzinga algorithm is used to find the number of facilities offering type A services that should be located in each district. Once the problem is solved, there will be a new value for the A objective. Therefore, the new weighted objective (Z_w') can be computed using the same weights w_A and w_B , employed when finding the value of the initial weighted objective (Z_w). The new A objective (Z_A') and the new B objective (Z_B'), are combined to yield $Z_w' = w_A Z_A' + w_B Z_B'$.

Observe that the initial location is independent of the weights assigned to each objective. In the case of SAPHIER1, as the B objective is more heavily weighted more iterations of the heuristic are likely to occur, since the starting solution is most favorable to the A objective. Similarly, the number of iterations for top-down heuristics will tend to increase as the A objective is more heavily weighted due to the initial solution.

Table 1 presents the results for the 25-node network obtained using all the solution methods. The table is set up such that all solution methods can be compared for each combination of p and q facilities located. The first column indicates the number of type A and type B facilities located, respectively. Then, for each solution method, the values found for both objectives are presented, corresponding to the different weights applied. The first objective, Z_A , is the average distance per person to facilities offering type A services. The second objective, Z_B , is the average distance from population areas to facilities offering type B services. Both objectives are expressed in kilometers per person. For each set of facilities more than ten weights were used to generate the trade off between objectives.

The solutions obtained with linear programming and branch and bound when needed

(LP+B&B column) for different weights show that there is a clear trade-off between both objectives, as depicted in Figure 1. Both objectives are minimized, so in these figures, the non-inferior solutions are represented by points which are closest to the origin. In other words, the smaller the average distances from demands to closest facilities, the better the solutions are. The trade-off curves are expected to be convex, because they are boundaries of convex sets in objective space. However, some local non convexities such as gap points may appear when the heuristics are used, due to the integer nature of the problem at hand, and, due to the non-optimal characteristics of heuristics.

Since optimal solutions are known (by solving the problem using linear programming and branch and bound when needed) the optimal trade-off curve is also known. It is possible that some of the non-inferior points were not found, even though more than ten runs were done with different weights to try to obtain all of them⁴. Nevertheless, the use of the weighting method for an integer programming problem implies that one cannot be certain that all of the non-inferior solutions are found (Cohon, 1978). Indeed, the use of the weighting method for 0,1 problems is unlikely to generate all non-inferior points because of the occurrence of gap points.

Once an accurate convex hull of the trade off curve is generated, it can be used to compare the effectiveness of the heuristics. The closer the trade-off curves generated with the heuristic are to the optimal trade-off curve, the better the heuristic is. As Figure 1 shows, all the heuristics performed similarly, being close to the optimal trade-off curve except for SAPHIER1. This last heuristic had a worse performance when using a heavy weight in objective B, since the solution associated with this weight showed a relatively larger solution for objective B (the second level of the hierarchy). Nevertheless, the difference is not very significant.

It was expected that the trade-off curves of the heuristics would stay on top or above the optimal trade off curve, but in two graphs in Figure 1 this did not occur, since, as mentioned before, not all the non-inferior solutions were computed.

The models were solved in an IBM 3090-E600 mainframe computer. Average computer times per run were calculated for each problem and solution method. Table 2 shows the results for the heuristics in the 25-node network. The figures correspond to seconds of CPU per run. Approximately twenty runs were done per problem using different weights. As the number of facilities located increases, the heuristics take longer to find a solution. This is because more exchanges are necessary as the number of facilities is increased.

As expected, SAPHIER1 was most efficient in computing time compared to the top-down

4 In the discussion of these models, the term non-inferior points is used for points to be generated by the different solution methods, even though they might not be in reality non-inferior with respect to the optimal solution found with linear programming and branch and bound.

heuristics since the number of one-opt trades is much less. Among the top-down heuristics, the fastest one was SAPHIER2. SAPHIER4 was the slowest in finding solutions. If 6 type A and two type B facilities are located, SAPHIER1 will outperform SAPHIER4 by a factor of ten. Ranking in speed from fastest to slowest was SAPHIER1, SAPHIER2, SAPHIER3 and SAPHIER4.

Table 3 presents the results of the run times for solution of the linear relaxation and the branching and bounding, when this combination was used to solve the pq-median. Average run times vary with the number of type A and type B facilities, but no clear pattern is observed. The pq-median model is quite expensive to solve when 2 type A and 1 type B facilities are located, to find the continuous solution, compared to the other problems. The $p = 2$ $q = 1$ median problem takes twice as much CPU time as the $p = 6$ $q = 2$ median problem to find the continuous optimum, this last one being the second slowest.

Two different combinations of type A and type B facilities were used with the 79-node network. For this size problem, only the four heuristics could be used to solve the two pq-median problems. Linear relaxation and branch and bound was not used because with 79 nodes the number of variables and constraints is very large, and therefore computationally extremely expensive to solve.

The starting solution of SAPHIER1 was computed as in the 25 node network. Table 4 presents the results of the heuristics for the pq-median in the 79-node network. A clear trade-off between average distance to type A facilities and average distance to type B facilities is obtained. The first graph in Figure 2 shows the trade-off curves corresponding to Table 4, when 6 type A and 3 type B facilities are located. SAPHIER1 shows a nice trade off, and finds the best values for both objectives when they are considered separately. SAPHIER2 found only two points in the trade-off. Both SAPHIER3 and SAPHIER4 presented a non-convexity in their trade-off curves, points Y and X respectively. SAPHIER3 found a solution with the same minimum value of B and a smaller minimum value of A than the others.

The second graph in Figure 2 shows the results if 8 type A facilities and 2 type B facilities are located. This time only SAPHIER3 obtained a non-convexity in its trade-off curve (point X). The rest of the heuristics behaved similarly. SAPHIER2 was the one that obtained the best curve, since most of it is below the other trade-off curves.

The heuristics for the 79-node problems were solved again on the IBM 3091-600. Table 5 shows the average run times for each problem and heuristic. Again SAPHIER1 outperformed significantly the top-down heuristics. This time SAPHIER3 was the slowest.

Several conclusions can be drawn from these test problems. SAPHIER1 was much more efficient in computer time than the rest of the heuristics. It also tended to give better values when weights favoring the A objective as opposed to the B objective were used. On the other hand, it was not very efficient in computing the solutions when the B objective was heavily weighted. Both top-down heuristics and SAPHIER4 had the tendency to do the contrary. The solutions found when w_B was set close to 1 were better than when w_B was close to 0. The problems solved in the 25-node network showed that the heuristics

performed relatively well in identifying optimal non-inferior solutions and generating a trade-off. Even though these tradeoffs sometimes present non-convexities, they can be useful for the decision-maker in order to find the best-compromise solution.

Table 1: Results for the pq-median model, 25-node network

#of Fac. Located p-q	LP + B&B		SAPHIER 1		SAPHIER 2		SAPHIER 3		SAPHIER 4	
	Z _A	Z _B	Z _A	Z _B	Z _A	Z _B	Z _A	Z _B	Z _A	Z _B
2-1	15.93	25.04	15.93	25.04	15.93	25.04	15.93	25.04	15.93	25.04
	15.34	25.14	15.34	25.14	14.56	26.85	14.56	26.85	14.77	29.82
	14.53	26.85	14.56	26.85	14.53	30.31				
			14.53	30.20						
3-2	11.18	17.98	12.17	17.98	11.53	17.98	11.53	17.98	11.53	17.98
	10.69	18.60	11.74	18.04	11.08	19.47	11.08	19.47	10.69	18.60
	10.66	19.88	11.01	18.42	10.72	20.17	10.72	20.17		
			10.69	18.60	10.69	20.44	10.69	20.44		
			10.66	20.04						
4-2	10.08	17.98	10.57	17.98	10.08	17.98	10.08	17.98	10.08	17.98
	9.77	18.21	9.65	18.52	9.67	20.24	9.67	20.24	9.65	18.52
	9.73	18.29					9.65	20.45	9.59	19.89
	9.65	18.52								
	9.59	19.84								
5-2	9.04	17.98	9.53	17.98	9.04	17.98	9.04	17.98	9.04	17.98
	9.00	18.02	8.98	18.42	8.98	18.42	8.83	18.52	8.90	19.24
	8.82	18.45	8.82	18.55	8.78	19.48	8.72	19.91	8.78	19.48
	8.75	19.29							8.72	19.91
	8.72	19.91								
6-2	8.24	17.98	8.57	17.98	8.24	17.98	8.24	17.98	8.24	17.98
	8.13	18.21	8.31	18.42	8.07	20.14	8.09	20.17	8.10	18.60
	8.11	18.29	8.16	18.55	8.06	20.24			8.04	19.71
	8.02	19.71	8.06	18.63					8.02	20.07
			8.05	18.97						
			8.02	20.14						
6-3	7.66	14.53	8.18	14.53	7.66	14.53	7.66	14.53	7.66	14.53
	7.39	14.96	8.11	14.64	7.44	15.46	7.58	14.71	7.48	15.18
			7.70	14.71			7.50	15.41	7.42	15.41
			7.50	14.88			7.44	15.46		
			7.39	15.00						

Table 2: Average CPU Time, pq-median Heuristics, 25 nodes

# of facilities located (p-q)	SAPHIER 1 (CPU sec)	SAPHIER 2 (CPU sec)	SAPHIER 3 (CPU sec)	SAPHIER 4 (CPU sec)	LP + B & B (CPU sec)
2-1	0.10	0.41	0.93	0.60	254.3
3-2	0.30	1.35	3.68	5.29	227.7
4-2	0.43	2.01	4.36	6.16	180.6
5-2	0.63	3.11	6.62	8.97	134.7
6-2	0.76	3.56	8.713	10.54	148.7
6-3	0.93	5.83	9.39	12.11	166.1

Table 3: Average CPU Time, pq-Median LP+B&B, 25 nodes

# of facilities located (p-q)	Total number of runs	Optimal int. sol. using linear relax. only	Avg. run time linear solution (CPU sec.)	Avg. additional run time using B&B (CPU sec.)	Total avg. run time (CPU sec.)
2-1	11	3	147.6	106.7	254.3
3-2	10	1	62.6	165.1	227.7
4-2	12	3	79.2	101.4	180.6
5-2	13	2	42.4	92.3	134.7
6-2	11	3	80.2	68.5	148.7
6-3	12	3	27.1	139.0	166.1

Table 4: Results for the pq-median model, 79-node network

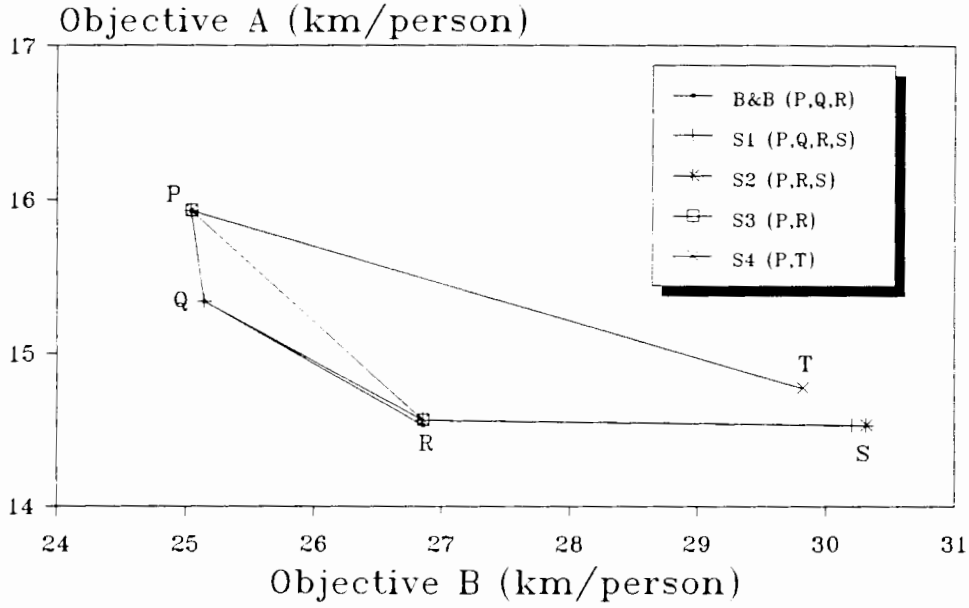
# of Fac. Located p-q	SAPHIER 1		SAPHIER 2		SAPHIER 3		SAPHIER 4																
	Z _A	Z _B	Z _A	Z _B	Z _A	Z _B	Z _A	Z _B															
6-3	226	412	220	412	218	412	224	412															
	219	414							211	422	212	422	215	419									
	215	416													213	429	216	420					
	212	419																	210	441	210	441	
	209	433																					
214	560	205	560	205	560	210	560																
202	562							203	562	203	562	196	580										
201	566													196	569	202	569	195					583
196	196																		195	579	196	588	

Table 5: Average CPU Time per Run, PQ-Median SAPHIER, 79 nodes

# of Facilities Located (p-q)	SAPHIER 1 (CPU sec)	SAPHIER 2 (CPU sec)	SAPHIER 3 (CPU sec)	SAPHIER 4 (CPU sec)
6-3	7.8	448.1	949.2	528.8
8-2	16.0	585.2	1480.5	1225.0

Figure 1: Trade-off Curves, 25-Node Network

2 Type A Facilities
1 Type B Facility



3 Type A Facilities
2 Type B Facilities

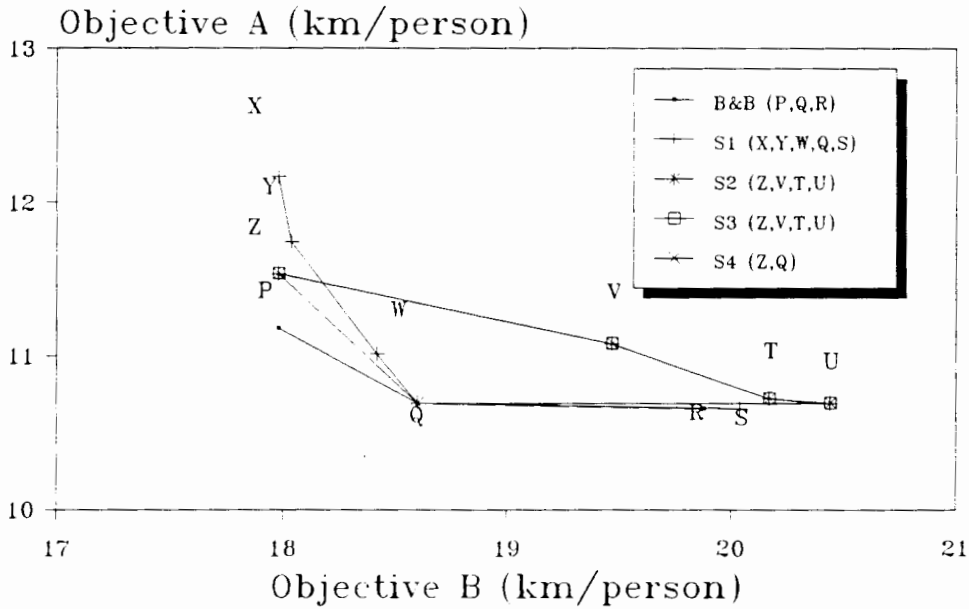
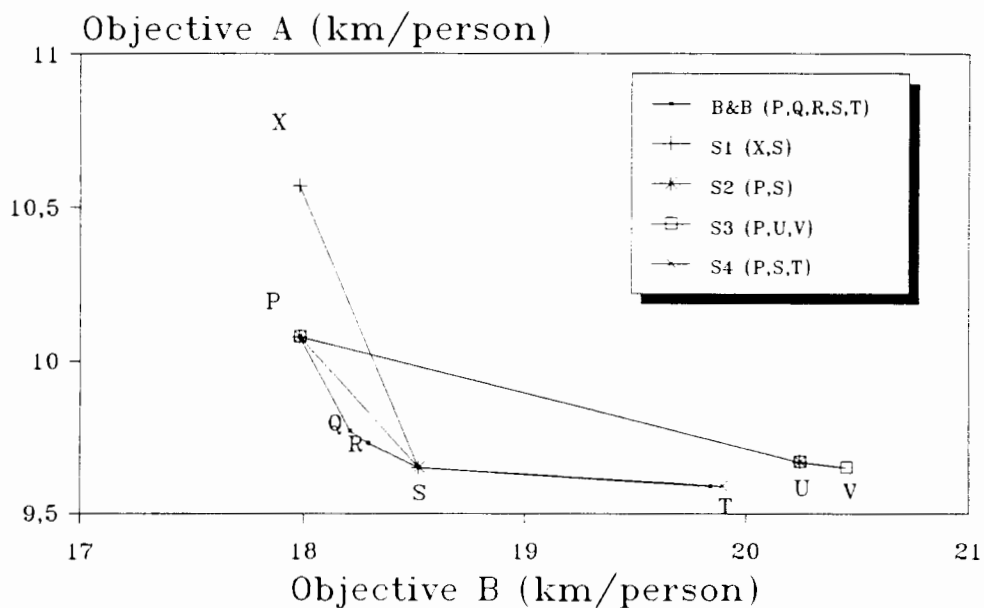


Figure 1 (Cont'd): Trade-off Curves, 25-Node Network

4 Type A Facilities
2 Type B Facilities



5 Type A Facilities
2 Type B Facilities

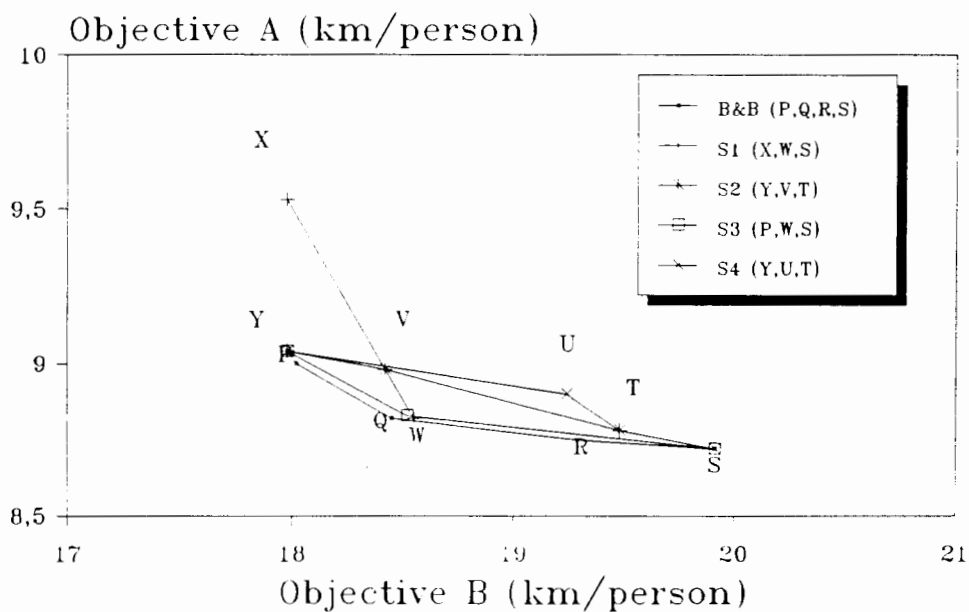
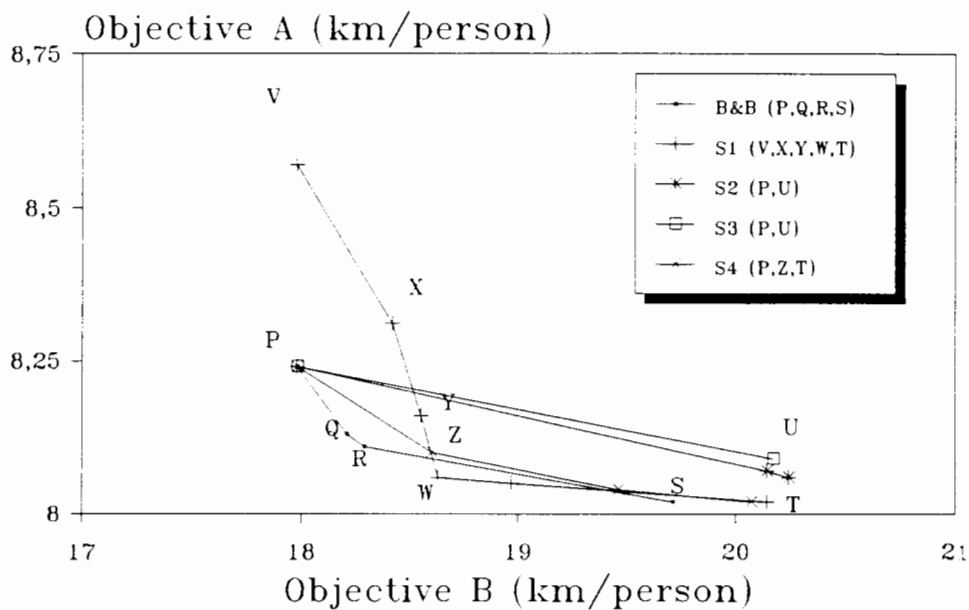


Figure 1 (Cont'd): Trade-off Curves, 25-Node Network

6 Type A Facilities
2 Type B Facilities



6 Type A Facilities
3 Type B Facilities

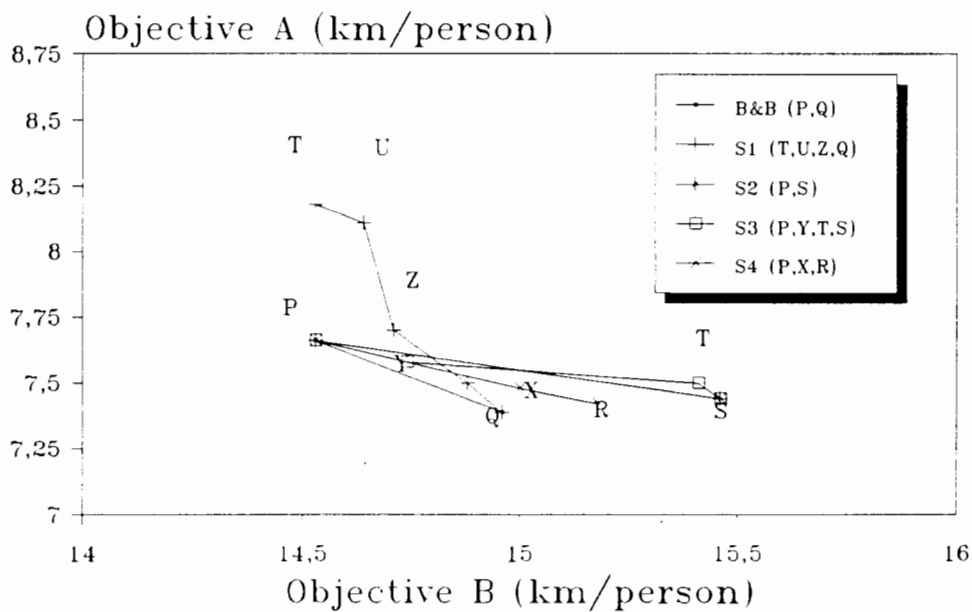
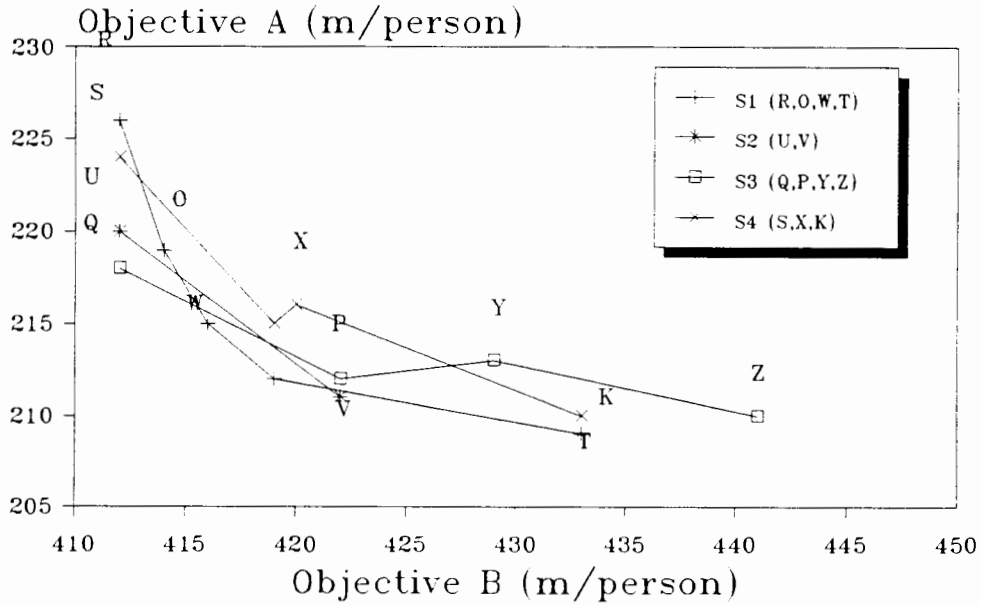
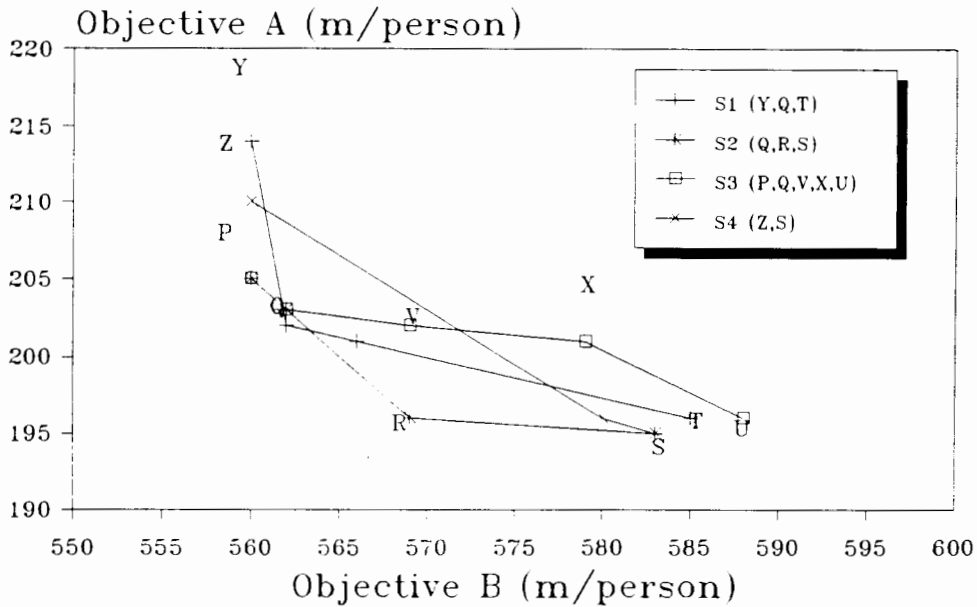


Figure 2: Trade-off Curves, 25-Node Network

6 Type A Facilities
3 Type B Facilities



8 Type A Facilities
2 Type B Facilities



Bibliography

- BANERGI, S. and H. FISHER (1974): "Hierarchical Location Analysis for Integrated Area Planning in Rural India". *Papers of the Regional Science Association* **33**, 177-194.
- COHON, J.L. (1978): *Multiobjective Programming and Planning*. New York, N.Y.: Academic Press, Inc.
- CORNUEJOLS, G, FISHER M.L. and G.L. NEMHAUSER (1977): "Location of Bank Accounts to Optimize Floats: An Analytic Study of Exact and Approximate Algorithms." *Management Science* **23**, 789-799.
- DIERH, G. (1972); "An Algorithm for the P-Median Problem", Working Paper no. 191, Los Angeles, CA: Western Management Institute.
- EVERETT, H. (1963): "Generalized Lagrange Multiplier Method for Solving Problems of Optimum Allocations of Resources". *Operations Research* **12**, 399-417.
- FELDMAN, E., LEHRER F. and T. RAY (1966): "Warehouse Locations under Continuous Economies of Scale". *Management Science* **12**, 670-684.
- FISHER, H. and G. RUSHTON (1979): "Spatial Efficiency of Service Locations and the Regional Development Process". *Papers of the Regional Science Association* **42**, 83-97.
- FRANCIS, R.L., MCGINNIS L.F and A. WHITE (1983): "Location Analysis". *European Journal Of Operations Research* **12**, 220-252.
- GALVAO, R. and L. RAGGI (1989): "A Method for Solving to Optimality Uncapacitated Location Problems". *Annals of Operations Research* **18**, 225-244.
- GARFINKEL, R.S., NEEBE, A.W. and M.R. RAO (1974): "An Algorithm for the M-Median Plant location Problem." *Transportation Science* **8**, 217-236.
- GEOFFRION, A.M. and R. McBRIDE (1978): "Lagrangian Relaxation Applied to Capacitated Facility Location Problems". *AIIE Transactions* **10**, 40-47.
- GUIGNARD, M. and K. SPIELBERG (1978): "A Direct Dual Method for Simple Plant Location Problem with Some Side Constraints". *Mathematical Programming* **13**, 198-228.
- HODGSON, M.J. (1984): "Alternative Approaches to Hierarchical Location-Allocation Systems". *Geographical Analysis* **16**(3), 275-281.
- HODGSON, M. (1986): "A Hierarchical Location-Allocation Model with Allocations based on Facility Size", *Annals of Operational Research* **6**, 273-289.

- KUEHN, A.A. and M.J. HAMBURGER (1963): "A Heuristic Program for Locating Warehouses". *Management Science* 9(4), 609-623.
- LEA, A.C. (1978): "A Model Taxonomy and a View of Research Frontiers in Normative Locational Modeling". Skodsborg, Denmark: Paper presented to International Symposium on Locational Decisions.
- MANNE, A.S. (1964): "Plant Location under Economies of Scale. Decentralization and Computation". *Management Science* 11, 213-235.
- MARANZANA, F.E. (1964): "On the Location of Supply Points to Minimize Transport Costs", *Operations Research Quarterly* 12, 138-139.
- NARULA, S.C., OGBU, U.I, and H.M. SAMULESON (1977): " An Algorithm for the P-Median Problem", *Operations Research* 25, 709-713.
- NARULA, S.C. (1981): "Hierarchical Location-Allocation Problems". Skodsborg, Denmark: paper presented to International Symposium on Location Decision.
- REVELLE, C. and D.J. ELZINGA (1988): "An Algorithm for Facility Location in a Districted Region". *Environment and Planning B: Planning and Design* 16, 41-50.
- ROSING, K.E., REVELLE, C., and H. ROSING-VOGELAAR (1979): "The P-Median and its Linear Programming Relaxation: An Approach to Large Problems", *Journal of the Operational Research Society* 30-9, 815-823.
- SERRA, D. and C. REVELLE (1991): "The PQ-Median Problem: Location and Districting of Hierarchical Facilities". Submitted for publication.
- TIETZ, M.B. and P. BART (1968): "Heuristic Methods for Estimating the Generalized Vertex Median of a Weighted Graph", *Operations Research* 16, 955-965.

