Discussion Paper Series

AN ILLUSTRATION OF THE ALLOC 6B
LOCATION-ALLOCATION ANALYSIS
SYSTEM

by

SOOBYONG PARK
AND
GERARD RUSHTON

DEPARTMENT OF GEOGRAPHY

The University of Iowa

IOWA CITY, IOWA

AN ILLUSTRATION OF THE ALLOC 6B LOCATION-ALLOCATION ANALYSIS SYSTEM

by

SOOBYONG PARK
AND
GERARD RUSHTON

FEBRUARY 1984

DISCUSSION PAPER NO. 37
DEPARTMENT OF GEOGRAPHY
THE UNIVERSITY OF IOWA
IOWA CITY, IOWA 52242

DISCUSSION PAPER EDITOR: GERARD RUSHTON

SOOBYONG PARK IS A PH.D. CANDIDATE IN THE DEPARTMENT OF GEOGRAPHY SPECIALIZING IN LOCATIONAL ANALYSIS.
GERARD RUSHTON IS A PROFESSOR OF GEOGRAPHY.
SUPPORT FROM NSF GRANT #INT 80-17879 IS ACKNOWLEDGED.

Availability of the programs described in this discussion paper

The source programs described in this paper are available on tape or on paper at a cost of $35. The package includes four programs (DISTANCE, UNRAVEL, RETRENCH and ALLOC 6B). They are recorded as separate files at 1600 bpi in 80 character logical records blocked to 1600 characters per block on unlabeled 9-track tape, in EBCDIC code. Orders, specifying whether paper or tape copies are required should be addressed to:

Department Programmer
Department of Geography
The University of Iowa
Iowa City, Iowa 52242
U.S.A.

THE PROBLEM

Many researchers have used the series of programs for location-allocation analyses first published in 1973 by The Department of Geography, The University of Iowa (see Rushton, Goodchild and Ostresh, 1973). A sequel was published in 1980 (see Hillsman 1980), but its use by researchers outside Iowa has apparently been rare. Citations to the 1973 volume continue to appear, even for work initiated since Hillsman's 1980 monograph. In many cases the work could have been done more efficiently and with fewer restrictions using the Hillsman programs, but evidently the diffusion of knowledge that a superior system exists is slow.

Although Hillsman's monograph describes what the programs do and documents how a user controls them, it does not illustrate the outputs of the program nor does it explain the sequence that an analyst would follow to execute a "typical analysis." Based on the experience of Iowa students using the Hillsman system, there is a need for an addendum that illustrates a typical analysis. This paper is an attempt to meet that need.

Like many systems of analysis, analytical methods associated with location-allocation problems are increasingly being adopted by researchers whose principal interest is not the development of location-allocation methods. Instead, they want to solve substantive problems in their field of interest that involve the evaluation of alternative combinations of locations, each of which has a pattern of spatial allocations associated with it. We think it is true to say that Geography, Planning and Regional Science are only in the early stages of identifying the full range of such problems. The textbooks in these fields do not deal adequately with location-allocation problems, even in cases where locational analysis is their theme, (see Rushton, 1981).

In comparison with the study of problems of spatial allocation and spatial choice, the problems involving simultaneous location and allocation were slow to receive the attention of these fields, especially in Europe. Although this is apparently changing, (see Mirchandani and Francis 1984; Hansen and

Thisse 1983; Leonardi 1982, Ghosh and Rushton 1984), there is a need, felt by researchers whose primary interest is not in location-allocation methods, to see the types of data that are typically generated in a location-allocation analysis and to see the sequence of steps that a researcher might follow to produce this data. Too frequently, in the current literature, the results of location-allocation analyses appear as a set of locations that "are optimum" with respect to some objective function. Articles are being written criticizing earlier work as having neglected distributional considerations, (see Adrian 1982). Well-organized location allocation analysis systems, such as Hillsman's, allow authors to assign weights to areas or groups in whatever way they wish and enable them to report the results of sensitivity analyses showing the changes in locational outcomes as distributional issues are considered. In summary, our view is that there is a discrepancy between the sophistication of analysis that is now possible using a contemporary locational-allocation analysis system and the restricted range of issues that are being pursued and reported in the contemporary applied literature where locational variability of services is explored.

AN ILLUSTRATION

In this section an illustration will be described showing a selection of eleven optimal locations in a region of Nigeria. The purpose of the illustration is to show the steps that had to be taken for one typical analysis in a series of analyses conducted by a team of geographers and development specialists at The University of Iowa and The University of Ibadan, Nigeria (see acknowledgements). The illustration was completed within a period of four days in the summer of 1983. For the sake of realism, as well as to communicate the contingency of one stage of analysis with that of the next, the illustration will be described in terms of the stages of the analysis day by day.

Background.

A collaborative agreement between the University of Iowa, U.S.A., The University of Ibadan, Nigeria, and the Indian Institute of Management, Bangalore, India, had been arranged in 1980. The terms were that a series of locational analyses of rural service delivery systems would be completed in Nigeria and India using comparable data and analysis systems. (Support for the work was provided by The National Science Foundation, U.S.A. and by the respective educational institutions of the investigators.) A locational analysis system developed earlier in Iowa was implemented and then substantially modified at the IIMB by the Indian research team under the direction of Professor V.K.Tewari.

A similar attempt in Ibadan was not successful due to several disruptions of computer services there. In order to complete the Nigerian analyses, Dr. Bola Ayeni from The University of Ibadan arranged to visit The University of Iowa in July 1983, to complete the Nigerian analyses there. He brought coding tablets showing the population totals and the locations of the 675 places in his study region, which was a rural region to the southwest of Ibadan. He also brought information on the location of health services such as clinics, maternity units and general hospitals. He had also determined the temporal sequence in which these services had been added to the region during the past three years. Upon arrival in Iowa he was met by Professors Rushton and McNulty from the Department of Geography, who served as technical advisers during the next three days. Mr. Soo Byong Park, a research assistant to Dr. Rushton was technical specialist in charge of the analyses. He is a specialist in locational analysis within the graduate program of The Department of Geography. The computer system used during this period was an IBM 370.

Purpose of This Analysis

This example begins with the analyst having defined a study area, collected basic data on locations of demand for a rural service, estimated the demand at each location, and collected data on the current locations and organization of the service outlets. The example describes the steps that were taken by the technical analyst from the data stage to the results of an analysis to determine the optimal locations of an activity and compare the relative performance of these locations with that of a previously defined set. Often, this set will be the existing locations of a service, so that the effect of the comparison is to assess the current locational efficiency and effectiveness of the existing service delivery system.

Format of the Illustration.

This illustration shows, alternately, the input that Mr. Park submitted, and the output that he received for each step of the analysis. Since the data set is large and the output is also large, selections were made by editing the input submission and the output received. Many of the inputs are explained in lower case lettering in the boxes on the figures. The editing was designed by Dr. Rushton and Mr. Park to reveal all the key functional steps and decisions that would be made by an analyst conducting similar analyses.

Analysis System

The analysis system used is a modified version of the system developed by Hillsman, (see Hillsman, 1980). The modifications were designed by Professors Rushton and McNulty and were programmed and added to Hillsman's original code by S. Park. The principal changes made are those that enable a detailed comparison of the original input locations with those computed by the algorithm for any of the analyses executed by the program. This comparison provides information on changes in geographical access of specific communities to the service between the original and the computed sets of locations. This feature was not available in the program as originally published by Hillsman. Compatibility with the published software documentation was maintained in that the changes affect the output characteristics and are therefore changes that are intrinsic to the code. The changes, with one exception, do not affect the input characteristics and therefore the program can be used with Hillsman's documentation. The program is known as ALLOC 6B and is available from The Department of Geography at The University of Iowa.

Day One: Creating the Distance Matrix.

The first step was to create the distance data between all the places. In the ALLOC system (see Hillsman 1980), this can be achieved by computing shortest paths along route networks between all places (see Ostresh 1973), or by computing distances from the cartesian coordinates of the places. This is the approach followed in this illustration. The program DISTANCE (see Hillsman 1980, p.13) was used (Figure 1). Although 71 lines of code are omitted in this figure, the illustration shows the user loading a short FORTRAN program and reserving memory in the dimension statement for four vectors of values corresponding with the x,y, coordinates, the identification codes and the distance values. Memory needs are, therefore, 4n + 20 where n is the number of places in the study area.

The input data for this analysis, explained in the boxes in Figure 2, show the user reserving a disc drive and naming a disc file to store the distances after they have been created. The control codes show the number of places involved in the analysis and the type of distances to be computed (straight line or city block types of distances). The format of the data and the data itself is added to the input file. Several lines of data are shown in Figure 2.

The output from the program DISTANCE begins by confirming that the input specifications were correctly interpreted. One of the input controls specified that the input data should be shown, so it is here. The listing of the distance matrix, if requested at input time, starts by showing the ID of the place from which distances have been computed (10001 in Figure 3). This is followed by the computed distances to all the other places, including itself, in the order that the places appeared in the input data. Examination of the second set of distances, shown on the bottom of Figure 3, for example, shows that the distance from place 10002 to 10001 is 117, that its distance to itself is 0 and that its distance to 10004 is 68. (We know this because 10004 is the fourth set of coordinates in the input coordinates). For this type of data output we say that the ID connected with any distance is implicit because it is not explicitly shown but, rather, is known because of its relationship to the known structure of the input data; in this case, to the ID's connected to the location coordinates file.

Creating Distance Strings. The second step in the analysis is the creation of distance strings from the distance matrix. The purpose of this step is to reduce the number of distances that will eventually be used in the analyses and to provide a data structure that will allow efficient computation of the steps in the analyses that follow. Both purposes have the effect of allowing large problems to be solved by small computers using small amounts of computing time. A more detailed description of these purposes and how distance strings accomplish the savings in computer memory requirements and in computation time is provided elsewhere (see Hillsman 1980, pp. 81-92). A description of the distance strings is provided below in the description of the output of the program, UNRAVEL, that creates them.

As Figure 4 indicates, UNRAVEL is a small FORTRAN program that takes the output from the previous program, DISTANCE, and rearranges the data into a form more suitable for locational analyses. The first half of Figure 4 describes the dimensioning requirements of UNRAVEL.

Input controls for program UNRAVEL are shown in Figure 5. The first item of information required is the location and description of the distance matrix. Notice that the distances could, at this point, have come from any source, provided they have been organized in a form similar to that produced by program DISTANCE as described earlier. The second item identifies the disc where the results of this analysis are to be stored and assigns a name to this data file. The third item describes key aspects of the distance matrix and, in the second data piece of this line, it defines the largest distance (in this case 750 units), which is to be saved in the results of UNRAVEL. In other words, all distances larger than a given value will be discarded. This option is based on the knowledge that, for most locational analyses, it is never necessary to know the distance from far away places to one another. If the analysis is of hospitals, for example, everyone will have an hospital within some given distance. If this distance can be estimated, then all distances larger than this can usually be discarded without affecting the results of the desired analyses. These distances are simply unnecessary and discarding them saves the amount of memory locations that need to be reserved for the analyses and allows results to be computed in a shorter computation time. The fourth item of information in Figure 5 is the format, describing how the information in the distance matrix is organized.

In Figure 6, the output showing the distance strings is described. The first six lines consist of a confirmation that the input commands were correctly received. The data is organized to be analyzed sequentially in two long data strings with the analysis programs organized so that they can skip over data that are known at any particular stage of an analysis to be redundant. One file is the index file and is essentially a key that is used to interpret the meaning of the distances in the distance file, which is the second of these two files.

The first data line in this example (Figure 6), shows that the first element identifies the index file with a consecutive series starting with one. The index file here shows that it is the first of this index sequence; that it describes distances from place ID 10001 to all other nodes within 750 distance units of itself; that these distances start at the first position in the distance string and end at the 584th position. Before examining the distances themselves below this index file, examine the second record of the index file in the lower half of Figure 6. It shows that there are 558 places within 750 distance units of the ID with which it begins (10002). These distances can be found beginning with the 585th element in the distance string and ending with the one in the 1142nd. position. The last data set on this figure is the description of the third record in the index file.

Returning to the middle of Figure 6, the distance string itself is shown. The three boxes above the distance data relate the distances below to the key in the corresponding part of the index file. Thus, from ID 10001, the closest place is the first place in the distance string (which, in this case is itself) and the distance between these "places" is zero. The second closest place to 10001 is the third place in the distance matrix. The distance from it is 13 units. As indicated in the index file, there are 584 distances within 750 units of place 10001, so there are 584 corresponding pairs of distances and place identifiers all arranged in ascending order of distance from the closest to the farthest from place 10001.

This data structure is discussed in more detail in Hillsman (1980, pp.85-90). It is the key to the solution of large location analysis problems within small computing times. However, note, in this example how program UNRAVEL, because it organizes its own idex file, has been designed to automatically keep track of the data it reorganizes. Consequently, data errors will not occur if the analyses are carefully conducted.

Day Two: Editing the Distance Strings.

The analyses continued on day two with further editing of the distance strings. This phase of the analysis is, in fact, optional in that results of subsequent analyses could have been computed with the use of the output of program UNRAVEL. The decision to perform this phase, which uses program RETRENCH, is a decision which is made in the interest of streamlining the analyses which follow so that they will use less compute time and require less computer memory core.

The philosophy behind the use of RETRENCH is described in Hillsman (1980, pp.137-41). The object of the retrench phase is to eliminate distance data from the UNRAVEL distance strings that can be shown to be unnecessary for any of the locational analyses that will later be required. If, for example, it is known that a certain place will never be a candidate for a service site (although its population will need to receive the service), then the distances from that place to all other places are not needed since they already exist in the distance strings of the other places that might possibly serve them. Such places that require service but which will never, themselves, be service sites, are known as ineligible places. The remaining places are known as candidate places. In Figure 7 the beginning section of the program RETRENCH is shown. Note how, in addition to the distance strings produced by UNRAVEL, this program also uses the "population file." In the RETRENCH "philosophy" it is argued that although candidate places may often be identified arbitrarily by the investigator, at other times the status of being a candidate will be defined in terms of whether the place meets or does not meet a stated level in some variable. Because the analyst can use any variable that can be quantified, this output uses the neutral term "weight" in describing the variable. In this illustration, "population" is used as the "weight."

The first part of Figure 8 shows that four disc areas and related file names must be identified for the purposes indicated in the Figure. The control information (center part of Figure 8) shows that a new and smaller distance limit can be defined in RETRENCH (400 in this example). In this particular sample analysis, the "population" value was used to define "candidacy." All places with more than 300 people were defined as candidate places. In the middle of the data line controlling RETRENCH, the number 127 is interpreted to mean that 127 places here will have their candidacy status defined arbitrarily (see below).

The string of ones toward the bottom of Figure 8 show that the distinction between a place being inside or outside the study region can be recorded so that subsequent analyses can give results describing the geographical accessibility characteristics of people inside or outside the study region.

The output of RETRENCH is compatible with that from UNRAVEL, described earlier. In the case of RETRENCH it is obviously important that a thorough check be made to determine that all edits that were intended by the analyst were correctly executed. If a place was inadvertently declared ineligible (by not declaring it to be a candidate), then all subsequent analyses would show it to be outside the optimal set. It would not be clear to the analyst that the reason might be the misspecification of its eligibility status at this earlier stage of the analysis. Where a place is not a candidate, its distance string is removed and all subsequent index file elements will have their values adjusted to reflect this paring of the distance string length. This happened with the third record in the index file in the example in Figure 9.

Day Three: First Locational Analysis Results.

The first analysis on day three was a test analysis to find the eleven places which together would minimize the average distance of the population to the closest of the eleven places and to compare the results with the present eleven state administrative centers in the region.

The program used was ALLOC 6B, which executes the optimizing phase exactly as designed and programmed by Hillsman (1980), but which performs more computations on the results of the analysis than are done by the ALLOC 6 provided by Hillsman.

The key control information is shown in the middle of Figure 10. This information is telling the program the sources of the data sets and the parameters of the earlier analyses. The ALLOC 6 software is designed to adjust to the different combinations of source data. This particular analysis operated on the distance string data produced by RETRENCH (Figure 6). It is also possible, however, to operate ALLOC6 or ALLOC6B directly on the data produced by UNRAVEL. The input control data, which starts with the number 11 (near bottom of Figure 10), specifies that in this particular analysis, eleven places are to be selected and that the algorithm to be employed is the heuristic location-allocation algorithm developed by Teitz and Bart (1968). Details of other options are described by Hillsman (1980, pp.113-117).

The final set of data on Figure 10 identifies the place ID's of the eleven places that are to be compared with the eleven places selected by the algorithm. These places must, of course, be candidate places. If they are not, the code will identify any places not candidates and will print an error message and will terminate.

A slightly edited (to reduce output size) description of the output is shown on Figures 11 through 16. Much of this output is self explanatory. Figure 11 shows the confirmation of the input data. It is useful for trouble shooting when an analysis is not executing due to an incorrect specification of input data. The item describing the division of weights by 10 (middle of Figure 11) is a feature that allows output to appear in units desired. Populations times distance, for example, when distances are measured in tenths of a kilometer, can lead to large numbers that are cumbersome to manipulate in the output. The program is counting the distances in the distance string and recording its length, (202,828 in this case). At the bottom of Figure 11, a list is provided of the places in the study area by name, ID, and population.

The output shown in Figure 12 is the first part of the analysis. It consists of an analysis of the eleven places (as shown on the bottom of Figure 10), that were to be evaluated before the optimal locations were determined. The first line of the table at the top of this figure is a description of the status of the first of these eleven places. Okenla, ID 40069, has a population (which is the weight in this example) of 130. This is shown as 13 in this output because all weights were divided by 10 in this analysis, (see middle of Figure 11). The next value in this line, reading from left to right, shows that the population of all the places that are closer to Okenla than to any of the other ten places, is 85,040. The total person distance is 1,134,790 kilometers, if every one of these people were to make one visit to Okenla. The final value on this line, 39,286, shows that these people would have to travel an additional 392,860 kilometers if Okenla were to stop offering the service and they then had to travel to the second closest of the eleven places. The phrase "cost if dropped" is used to describe this extra distance cost that would be incurred if the people now receiving service from a place, received it from the next best alternative. It is a measure of the importance of a place in any rural delivery system. The larger the "cost if dropped," the more important is the place in the delivery system. "Drop" algorithms in the location-allocation literature use this value to eliminate, from a set of places, the place with the smallest "cost if dropped." This information is given for each of the eleven places identified as "the starting solution" (see bottom of Figure 10).

The information at the center of Figure 12 is summary information for the eleven places described above. The term "allocated places" refers to the option that places outside the study area can be a part of the data set but ignored in the computation of the summary statistics.

The information in the lower half of Figure 12 identifies, for each of the 675 places in the study, the closest center and its distance from them. Finally, in the bottom section of the Figure, the service areas are described sequentially. For each of the eleven centers, the places that are closer to them than to any alternate center are identified by their ID's. Their populations and their distances from the center are also given. These two tables, which often are quite lengthy, contain the same information. The difference is that in the "list of nodes," the center relationship of any place is easily found because the order of the table is by place ID. In the second table, the service center of any center is easily found because the places have been grouped together by their association with a center.

Figure 13 begins with a re-statement of summary statistics, but then describes some key statistics about the search for a better set of eleven centers by, in this illustration, the Teitz and Bart heuristic algorithm. It shows how center 10032 is replaced by center 10001 and how the total cost (in this case the total weighted distance separation from all places to their closest center), decreases from 809110 (see top line of Figure 13) to 808426. The line notes that this is a net change of 684 and expresses this as a percent of the total distance separation as a measure of its significance.

The bottom half of Figure 13 repeats the format of the table described above and found on Figure 12. The places for which the data are summarized are the new eleven places. Likewise the summary statistics below this table repeat the format described above. Note that the average distance of the places to their closest center is now 5.664 units compared with 7.967 for the original eleven places. This is a 28.9 per cent reduction, (see bottom of Figure 14).

Figure 14 shows the assignments to the new centers of the 675 places in the two ways described earlier, (see Figure 12).

In Figure 15, the effect on both centers and places of adopting the new eleven centers identified by the analysis in comparison with the original eleven centers is shown. The comparison divides the data into three sections. First, (see top of Figure 15), the new centers identified by the algorithm are described. Nine of the eleven original places were replaced in the analysis. Thus, there are nine new centers, nine "old centers" and two centers that were present in the original set of eleven and are called here: "remaining centers." These places, shown in the middle of Figure 15, are described according to their status at the beginning of the evaluation and their status at the end, (see "end set" in Figure 15).

In Figure 16, a comparison is made of the change in status of all 675 places as a result of the analysis. How would people be affected by the adoption of the results of the analysis? In this illustration, (see middle section of Figure 16), 45 percent of people would be unaffected by the change, 35 percent would be closer to a center and 20 percent would be farther than before. The figure shows that 22 per cent of places would be unaffected, 51 percent would be closer and 28 percent would be farther than before. Identification of the specific places and the degree to which they are affected is shown in the section "comparison of node assignments," (see top of Figure 16).

Fourth Day

On the fourth day, Dr. Ayeni defined eighteen analyses that he wished to undertake. Some of them, for example, were analyses to evaluate the locational efficiency of the sequential adding of schools at various locations through time.

These analyses were all completed on this day. Note that the key to the ability of the system to provide solutions so fast is the fact that all the analyses, up to the final ALLOC 6B series, were performed only once. Their purpose was to organize the data for speedy and efficient analysis of any problem subsequently identified. The input requirements to direct the solution of a problem are usually small, usually consisting of the identification of the centers that are to be evaluated; a description of the sources of the data sets developed earlier by the sequence: "DISTANCE, UNRAVEL, RETRENCH"; the identification of place-specific constraints such as discussed earlier; and the identification of the algorithm and the objective function that the user wishes Program ALLOC 6B to use.

Resources Used in the Case Study

We estimate that the resources required to produce the eighteen analyses requested by Dr. Ayeni were:

Professional time: seven person days.
Clerical time (data encoding): two person days.
Computer time: approximately $20. per analysis and approximately $250 for the development of the geocoded data files.

These resource estimates presume that the software system is operational on the computing installation (in this case an IBM 370), and that a person is available who is trained in the use of the system and knowledgeable about the theory and methods of location-allocation analysis and of the specific computational techniques that are used in the Hillsman ALLOC system.

Conclusion

This illustration has shown how the steps to be taken for an efficient location-allocation analysis of a realistically sized study area involve effort and care in the pre-processing phase of the actual location-allocation analysis itself. By developing an appropriate data structure for an area, the analyst is in the position of one who has invested a great deal of effort in the organization of the data system with comparably less effort in the design and execution of the particular location-allocation analysis. In the several dozen such analyses which we have seen done at The University of Iowa in the past three years, we have been impressed to notice the large proportion of the ultimate effort that is required to accomplish the first analysis with very little effort being required to do other analyses after the first has been successfully realized.

Acknowledgements.

The subject matter of this illustration was provided to us, fortuitously, by Dr. Bola Ayeni, Department of Geography, The University of Ibadan, Nigeria. The design of the changes made to Hillsman's ALLOC 6 was a team effort directed by G. Rushton and M.L. McNulty. Participating in addition to S. Park were Dan Berglove, A. Krishnamurthi, Don McKeagney and N. Sivagnanum. At The Indian Institute of Management, Bangalore, V.K.Tewari and S. Dwarakinath made some useful suggestions to us.

References.

Adrian, C. 1983. Analysing service equity: fire services in Sydney. Environment and Planning A 15: 1083-1100.

Ghosh, A. and Rushton, G. eds. 1984. Spatial Analysis and Location-Allocation Models, in preparation.

Hansen, P., Peeters, D., and Thisse, J.-F. 1983. Public facility location models: A selective survey. In Locational Analysis of Public Facilities, eds. J.F. Thisse and H.G. Zoller, pp. 223-62. New York: North-Holland Pub. Co.

Hillsman, E.L. 1980. Heuristic Solutions to Location-Allocation Problems: A User's Guide to ALLOC IV, V, and VI. Iowa City, Iowa: Geography Department, University of Iowa, Monograph 7.

Leonardi, G. 1982. Public facility location, in G. Leonardi (ed) Public Facility Location: Issues and Approaches. Laxenburg, Austria: International Institute for Applied Systems Analysis, (Reprinted from Sistemi Urbani, Vol. 3, 1981). pp. 293-97.

Mirchandani, P. and Francis, R. (eds.) 1984. Discrete Location Theory. forthcoming, J. Wiley & Sons, New York.

Ostresh, Jr., L.M. 1973. SPA: a shortest path algorithm. in G. Rushton, M.F. Goodchild and L.M. Ostresh Jr. eds. Computer Programs for Location-Allocation Problems. Iowa City, Iowa: Department of Geography, The University of Iowa, monograph 6, pp. 141-62.

Rushton, G. 1980. Review of Locational Analysis in Human Geography by Haggett, Cliff and Frey. Geographical Review 70: 112-14.

Rushton, G., Goodchild, M.F. and Ostresh, Jr., L.M. eds. 1973. Computer Programs for Location-Allocation Problems. Iowa City, Iowa: Department of Geography, The University of Iowa, monograph 6.

Teitz, M.B. and Bart, P. 1968. Heuristic Methods for estimating the generalized vertex median of a weighted graph. Operations Research 16: 955-61.

Figure 1. Input File of the Program DISTANCE for Creating
Inter-place Distance Matrix from Coordinates

```
//BLAQQQDS JOB (15001808,40),'I2PARK',TIME=3
//   EXEC FORTGCLG,REGION=350K
//FORT.SYSIN DD *
C                         D I S T A N C E
C
C     PROGRAM TO COMPUTE A DISTANCE MATRIX FROM CARTESIAN COORDINATES.
C     WRITTEN BY EDWARD L. HILLSMAN
C     DEPARTMENT OF GEOGRAPHY, THE UNIVERSITY OF IOWA, 1975.
C     DIMENSIONING INFORMATION
C     THE FOLLOWING ARRAYS ARE TO HAVE A LENGTH EQUAL TO THE NUMBER OF
C     NODES:
C     ID        ID NUMBER OF EACH NODE.
C     IX        X-COORDINATE OF EACH NODE.
C     IY        Y-COORDINATE OF EACH NODE.
C     IDIST     DISTANCES IN ONE ROW OF THE DISTANCE MATRIX.
C     IFMT      DOES NOT AFFECT THE NUMBER OF NODES THAT CAN BE HANDLED
C               BY THE PROGRAM.  IFMT IS USED TO STORE VARIABLE FORMATS.
C               IT IS MACHINE-DEPENDENT AND MAY HAVE TO BE CHANGED FOR
C               MACHINES OTHER THAN THE IBM 360 AND 370.
C     THE FOLLOWING FEATURES OF THE CODE ARE KNOWN TO BE MACHINE- OR
C     INSTALLATION-DEPENDENT:
C     (1)  IFMT AND FORMAT STATEMENTS 2000 AND 3300 (SEE NOTE ABOVE).
C     (2)  NREAD, NPRINT, AND KPUNCH ARE THE UNIT NUMBERS FOR THE CARD
C     READER, LINE PRINTER, AND CARD PUNCH, RESPECTIVELY.  THEY ARE SET
C     IN THE FIRST THREE EXECUTABLE STATEMENTS IN THE PROGRAM.
C     (3)  USE OF END= IN THE STATEMENT DIST
      DIMENSION ID(677),IX(677),IY(677),IDIST(677)
      DIMENSION IFMT(20)
      NREAD=5
      NPRINT=6
      KPUNCH=7
      WRITE(NPRINT,999)
C     READ CONTROL CARD AND FORMAT.
      READ (NREAD,1000) NODES,METRIC,NPUNCH,NWRIT
      IF (NPUNCH.EQ.0) NPUNCH=KPUNCH
      READ (NREAD,2000) IFMT
      WRITE(NPRINT,3000) NODES,NPUNCH
      IF (NWRIT.GT.0) WRITE(NPRINT,3050)
      IF (NWRIT.EQ.0) WRITE(NPRINT,3060)
      IF (METRIC.EQ.1) WRITE(NPRINT,3100)
      IF (METRIC.EQ.2) WRITE(NPRINT,3200)
      WRITE(NPRINT,3300) IFMT
C     READ ID NUMBERS AND COORDINATES.  CHECK FOR DUPLICATE ID NUMBERS.
      DO 100 I=1,NODES
      READ (NREAD,IFMT,END=900) ID(I),IX(I),IY(I)
      WRITE(NPRINT,3400) ID(I),IX(I),IY(I)
      .    .    .    .
      .    .    .    .     71 lines of program are not shown here.
      .    .    .    .
 6000 FORMAT(28HODISTANCE MATRIX IS COMPLETE)
      END
```

Figure 2. Input Controls for Creating Matrix of Distance from Coordinates
Using the Program "DISTANCE"

```
//GO.FT10F001 DD UNIT=DISK,DISP=(NEW,CATLG,DELETE),     opens disc file from
                                                        unit 10 to store
                                                        distance matrix

//  DSN=USER.A5001808.PARK.DSNIG.EUC675,                name of distance
                                                        matrix for catalo-
                                                        guing on disc for
                                                        input to UNRAVEL or
                                                        ALLOC III, IV or V.
//  DCB=(RECFM=FB,BLKSIZE=7000,LRECL=70),
//  SPACE=(TRK,(500,20),RLSE)
//GO.SYSIN DD *
  675   2   10    1          control codes for this analysis:
                             675 places
                             2 means Euclidean distances to be computed
                             10 means the disc unit where the distance
                                matrix will be written.
                             1 means - also make a paper copy of the
                                distance matrix.

(3I5)                        (3I5) describes the organization of the data:
10001    73  820             node ID's and X,Y coordinates
10002   172  883
10003    78  808
10004   135  793
10005   150  770
10006   195  755             one of the 675 lines of input data: this one
10007   190  742             is for place ID 10006 which is located at
10008   182  743             X coordinate 195 and Y coordinate 755.
   .     .    .
   .     .    .
   .     .    .
40262   971  351              last data item- 675th place ID and coordinates
```

Figure 3. Sample Output of the Program "DISTANCE"

```
COMPUTE A SQUARE DISTANCE MATRIX FOR  675 NODES

WRITE THIS MATRIX ON OUTPUT UNIT 10            This confirms that the distance
                                               matrix was written on unit 10

LIST THE MATRIX                                These lines confirm that the
                                               distances will be printed, will
                                               be straight line distances and
COMPUTE DISTANCES USING EUCLIDEAN OR           that the data format was 3I5.
 STRAIGHT-LINE METHOD
READ NODE DATA USING FORMAT OF:  (3I5)

ECHO CHECK OF INPUT DATA FOLLOWS

      ID          X          Y

    10001        73        820           Echo check means a copy of the input
    10002       172        883           data as the program found it.
    10003        78        808
    10004       135        793
      .           .          .
      .           .          .           remainder of input data here
      .           .          .
    40262       971        351

LISTING OF DISTANCE MATRIX
                                      distance
                         distance     from 10001
            distance     from 10001   to 10009
place ID    from 10001   to 10002
            to itself

10001    0  117   13   68   92  138  141  133  142  126  111  141  124  141
  169  179  187  191  205  223  249  278  284  301  319  298  325  332  328
   .    .    .
   .    .    . remainder of distances from ID 10001 to the rest of the 675
   .    .    . other places in the output data
   .    .    .
                                      distance
                         distance     from 10002
            distance     from 10002   to 10009
place ID    from 10002   to itself
            to 10001

10002  117    0   13   68   92  138  141  133  142  126  111  141  124  141
   .    .    .remainder of distance from ID 10002 to the rest of the 675
   .    .    .other places in the output data
   .    .    .
```

Figure 4. Input File of the Program "UNRAVEL" for Creating Inter-place Distance "strings" from the Distance Matrix Created by the Program "DISTANCE"

```
//BLAQQQUR JOB (15001808,50,,,,,0),'I2PARK',TIME=3,MSGLEVEL=(1,1)
//  EXEC FORTGCLG,REGION=250K
//FORT.SYSIN DD *
C                            U N R A V E L
C
C
C     PROGRAM TO CONVERT A COMPLETE DISTANCE MATRIX INTO A FORM WHICH
C     MAY BE USED BY ALLOC VI.(i.e., distance strings)
C     WRITTEN BY EDWARD L. HILLSMAN, DEPARTMENT OF GEOGRAPHY, THE
C     UNIVERSITY OF IOWA, NOVEMBER, 1974.
C     DIMENSIONING INFORMATION.
C     THE FOLLOWING ARRAY MUST BE DIMENSIONED IN THE MAIN PROGRAM AND IN
C     THE SUBROUTINE UNRAVEL.
C     LABEL(N)      STORES ONE COLUMN OF A COMPLETE DISTANCE MATRIX.
C     THE FOLLOWING ARRAYS MUST BE DIMENSIONED IN THE SUBROUTINE ONLY.
C     IPOOL(N)      STORES SUBSCRIPTS OF DISTANCES TO BE SAVED AND SORTED
C     NEXT(N)       SCRATCH VECTOR FOR OUTPUT OF DISTANCES AND SUBSCRIPTS
C     IRB(N),ILB(N) INDEX VECTORS FOR THE SOTRING PROCEUDRE.
C     N IS THE NUMBER OF ROWS (COLUMNS) IN THE COMPLETE DISTANCE MATRIX.
C     IFMT(20) STORES THE FORMAT TO BE USED IN READING ONE COLUMN OF THE
C     DISTANCE MATRIX.  ITS SIZE DOES NOT AFFECT THE SIZE OF THE MATRIX
C     THAT CAN BE PROCESSED.  HOWEVER, IT IS MACHINE-DEPENDENT AND MAY
C     HAVE TO BE CHANGED FOR MACHINES OTHER THAN THE IBM 360 AND 370.
C------------------------------------------------------------------------
      COMMON NPUNCH,NPRINT,LABEL
      DIMENSION IFMT(20)
      DIMENSION LABEL(680)
      INTEGER*2 LABEL
      NREAD0=5
      NPRINT=6
      KPUNCH=7
      WRITE(NPRINT,4000)
C     READ CONTROL CARD AND FORMAT.
      READ (NREAD0,5000) KNODES,LAMBDA,NREAD1,NPUNCH,IFORM
       . . .
       . . . 8 lines of programs are not shown here.
       . . .
      READ (NREAD0,7000) IFMT
      WRITE(NPRINT,7100) IFMT
C     MASTER LOOP TO READ EACH COLUMN OF THE MATRIX AND CALL SUBROUTINE
C     UNRAVL.  UNRAVL WILL REFORMAT THE COLUMN AND WRITE IT OUT.
C      READ (NREAD1,7000) DUMMY
      DO 500 JDEX=1,KNODES
      READ (NREAD1,IFMT) IDZOUT,(LABEL(J), J=1,KNODES)
      CALL UNRAVL (JDEX,IDZOUT,KNODES,LAMBDA,IFORM)
  500 CONTINUE
      .    .    .    .
      .    .    .    .               approximately 122 lines of the
      .    .    .    .               program are not shown here.
      END
```

Figure 5. Input Controls for the Program "UNRAVEL"

```
//GO.FT10F001 DD UNIT=DISK,DISP=(OLD,KEEP),
// DSN=USER.A5001808.PARK.DSNIG.EUC675,
//   DCB=(RECFM=FB,BLKSIZE=7000,LRECL=70)
```

shows that the distance matrix data will be read from disc unit 10 and that its file name name is "DSNIG.EUC675"

```
//GO.FT11F001   DD   UNIT=DISK,DISP=(NEW,CATLG,DELETE),
//    DSN=USER.A5001808.PARK.UNRAV.UNFMT.NIGERIA,
//    DCB=(RECFM=VBS,LRECL=1000,BLKSIZE=1004),
//    SPACE=(TRK,(500,100),RLSE)
```

reserves space for new distance data on disc as unformatted file and names it as: "UNRAV.UNFMT.NIGERIA"

```
//GO.SYSIN   DD  *
  675  750   10   11    1
```

control information: 675 places; 750 is the largest distance to be saved; 10 is the unit number containing the distance matrix; 11 is the unit number for the new distance data(distance string and index string) created by this program. 1 means that the new distance data will be created as unformatted data. (see Hillsman, 1980, p.91)

```
(48(14I5/),4I5)
//
```

describes the organization of the distance matrix data created by the program DISTANCE. In this case: 48 records each containing 14 data items followed by one with six(i.e., making total of 676 data items--1 ID and 675 distances)

Figure 6. Sample Output of the Program "UNRAVEL"

```
CREATE INDEX AND DISTANCE FILES FOR  675 NODES
ORDER DISTANCES IN INCREASING VALUES UP TO AND
INCLUDING VALUES OF 750
READ DISTANCE MATRIX FROM UNIT 10
WRITE INDEX AND DISTANCE FILES ON UNIT 11
THESE FILES WILL BE UNFORMATTED
READ DISTANCE MATRIX WITH FORMAT OF (48(14I5/),4I5)
A LISTING OF THE INDEX AND DISTANCE FILES FOLLOWS
```

This information confirms that the input instructions were correctly interpreted.

This is the first record of the index file. **It describes distances from node 10001 to all other nodes within 750 distance units.** **These distances start at the first position and go to the 584th position.** **There are 584 distances in this string.**

```
      1   10001        1      584        1      584
```

From 10001, the closest place is place 1 (itself) at 0 distance. **From 10001, the second closest place is the 3rd node at 13 units distance.** **From 10001, the fifth closest place is the 49th node at 59 distance units.**

```
  1    0    3   13   56   24   48   34   49   59    4   68
 51  103   50  104   57  110   11  111    2  117  113  118
  .    .    .    .
  .    .    .    .
  .    .    .    .
425  745  545  747  543  747  430  749
```

54 lines of output are not shown here.

From 10001, the 584th place is the 430th node at 749 distance units.

This is the second record of the index file.

From ID 10002, there are 558 places within 750 distance units and these begin in the 585th position in the distance string and end in the 1142nd position.

```
      2   10002      585     1142        1      558
  2    0  113   25  .  .  .
 56  128   44  130  .  .  .
  .    .    .    .  .  .  .
  .    .    .    .  .  .  .
  .    .    .    .  .  .  .
121  292   54  298  .  .  .
 36  310  126  314  .  .  .
```

The second closest place is 113rd place at 25 units distance. It is the 586th data pair in the "distance string". 50 lines of output are not shown here.

This is the 3rd record of the index file.

```
      3   10003     1143     1736        1      594
  3    0    1   13   48   25   56   35   47   56    4   59
 11   98   50  110   51  111   13  111   10  114  113  119
 14  128    6  128   12  129    9  130    7  130   26  156
```

Figure 7. Input File of the Program "RETRENCH"

```
//BLATRDRT JOB (15001622,50,10),'I2PARK'
/*ROUTE  PRINT REMOTE2
// EXEC FORTGCLG,REGION.GO=350K
//FORT.SYSIN DD *
      DIMENSION IWT(677), MIN(677), IDN(677), NODE(677), IDIST(677),
     * LTRY(677), NEAR(677), ELIGBL(677), KFIXED(677), DISCRD(677),
     * KANDD(677)
      DIMENSION IFMT(20), ICARD(3), INDEX(11), KTOTD(6000)
      INTEGER*2 NODE, LTRY, NEAR, IDIST, INDEX, MIN, KANDD
      INTEGER*2 INOUT, IHZERO
      LOGICAL*1 KIN, ELIGBL, KFIXED, DISCRD
      DATA ICARD /4HDECL,4HFIXE,4HDELE/
      NREAD1 = 5
      NPRINT = 6
      IZERO = 0
      IHZERO = 0
      LARGE = 32000
      WRITE (NPRINT,99999)
      READ (NREAD1,99997) N, NOCRIT, KMAX, LAMBDA, LOWLIM, INSIDE,
     * KARB, MFIX, JUNK, KPASS, NREAD2, NREAD3, NREAD4, NREAD5, NREAD9,
     * NWRIT0, NWRIT1, NWRIT2, NWRIT3
      WRITE (NPRINT,99996) N, NOCRIT, KMAX, LAMBDA
      II = 0
C
C-----------------------------------------------------------------------
C     READ THE POPULATION FILE FORMAT AND POPULATION FILE.  USE THE
C     POPULATION FILE TO DECLARE NODES TO BE CANDIDATES.
C-----------------------------------------------------------------------
      READ (NREAD2,99995) IFMT
      WRITE (NPRINT,99994) NREAD3
      READ (NREAD3,IFMT) (IDN(I),IWT(I),I=1,N)
      WRITE (NPRINT,99993)
      DO 10 I=1,N
         MIN(I) = LARGE
         KANDD(I) = LARGE
         ELIGBL(I) = .FALSE.
         IF (IWT(I).LT.LOWLIM) GO TO 10
         .      .      .
         .      .      .               About 420 lines of the program
         .      .      .               are not shown here.

      END
/*
```

Figure 8. Input Controls for the Program "RETRENCH"

```
//GO.FT09F001 DD UNIT=DISK,DISP=OLD,
// DSN=USER.A5001808.PARK.UNRAV.UNFMT.NIGERIA,
// DCB=(RECFM=VBS,LRECL=1000,BLKSIZE=1004)
```

This describes the location of the distance data created by UNRAVEL.

```
//GO.FT10F001 DD UNIT=DISK,DISP=(NEW,CATLG,DELETE),
// DSN=USER.A5001808.FSTPSS.PARK.NIGER.DS675,
// DCB=(RECFM=FB,LRECL=80,BLKSIZE=8000),
// SPACE=(TRK,(400,10),RLSE)
//GO.FT20F001 DD UNIT=DISK,DISP=(NEW,CATLG,DELETE),
// DSN=USER.A5001808.RETR.INDEX.PARK.KM40.UFMT,
// DCB=(RECFM=VBS,LRECL=1000,BLKSIZE=1004),
// SPACE=(TRK,(450,10),RLSE)
//GO.FT30F001 DD UNIT=DISK,DISP=(NEW,CATLG,DELETE),
// DSN=USER.A5001808.RETR.DSTFILE.PARK.KM40.UFMT,
// DCB=(RECFM=VBS,LRECL=1000,BLKSIZE=1004),
// SPACE=(TRK,(450,10),RLSE)
//GO.SYSIN DD *
  675    1  750  400          300    1  127              1       5 5 9 9  102030 6
```

These are the unit numbers for writing three new data files created by RETRENCH onto the disc.
Unit 10 stores "FSTPSS"
Unit 20 stores "INDEX"
Unit 30 stores "DISTANCE"

control information: 675 places in the region, one class of distances will be created, largest distance in the file(created by UNRAVEL) was 750; new largest distance to be saved by this program is 400; minimum populaion size for a node to be a candidate is 300; non-candidacy is allowed for nodes outside the region; 127 arbitrary declarations for candidacy that overrule the candidacy by population above; both the read format and the population file are input from unit 5; both the index and the distance files(unformatte d) are input from unit 9; 10, 20 and 30 are unit numbers for writing new data files created by this program(see above); 6 is the unit number for writing the information of fixed centers and candidates.

population file for the 675 nodes:
(I5,I10) is the read format for the pop. file.
The weight of node 10001 is 2734.

```
(I5,I10)
10001        2734
10002        7400
10003         225
  .      .      .
  .      .      .
  .      .      .
```

remainder of the 675 places ID's and populations

region declaration:"1" in col.5 means that the 5th node in the list of nodes above is within the study region.

```
11111111111111111111111111111111111111111111111111111111111111111111111111111111
  .   .   .
  .   .   .
```

remainder of study region declarations

The following 127 arbitrary declarations for candidacy overrule any of the previous candidacy declarations:

```
DECL
1     10003     F
1     10006     F
1     10011     T
  .      .      .
  .      .      .
  .      .      .
```

DECL is the header card of the folowing declaration file.
"F" means that node 10003 is no longer a candidate.
"T" means that node 10011 is now forced to be a candidate.
remainder of candidacy declaration.

Figure 9. Sample Output of the Program "RETRENCH"

```
EDIT INDEX AND DISTANCE FILES FOR   675 NODES
INPUT INDEX FILE CONTAINS  1 CRITICAL DISTANCE CLASS(ES)
  DISTANCE CLASS(ES)--OUTPUT FILE WILL CONTAIN 1
MAXIMUM DISTANCE IN INPUT DISTANCE FILE IS       750--
  MAXIMUM IN OUTFILE WILL BE       400
READ POPULATION FILE FROM I/O UNIT  5
POPULATION FILE HAS BEEN READ
CANDIDACY
USING A POPULATION OF       300 AS A LOWER LIMIT FOR CANDIDACY,
  548 NODES ARE CANDIDATES
      0 OF THESE ARE OUTSIDE THE MAIN STUDY REGION AND
        HAVE BEEN DROPPED FROM THE LIST OF CANDIDATES
MAKE  127 ARBITRARY DECLARATIONS OF NODE CANDIDACY
THESE MAY DUPLICATE THE PROGRAMMED DECLARATIONS ABOVE

    The above information confirms that the input instructions
    were correctly interpreted.

THE FOLLOWING   585 NODES ARE CANDIDATES FOR CENTERS
     10001     10002     10004     10005     10007     10008
       .         .         .   the remainder of list of candidate nodes
       .         .         .   59 lines are not shown here.
       .         .
ALL OTHER NODES ARE INELIGIBLE
MAIN EDIT TO USE GENERAL MAXIMUM DISTANCE OF      400

THE FOLLOWING NODE IS A CANDIDATE
    This is the first item of the index file.
    It describes distances from node 10001 to all other nodes within
    400 units of distance.
    These distances start at the first position and continues to the
    93rd position in the file of distance strings.
        1    10001       1      93     1    93
   1    0    3   13   56   24   48   34   49   59    4   68   47   69
   .    .  7 lines are not shown here. From node 10001, the closest place
   .    .  is place 1(itself) at 0 distance.
   .    .  The second closest place is the 3rd place at 13 units distance.
   .    .  The 5th colsest place is the 49th node at 59 units distance.
   .    .  The 93rd closest place is 238th node at 397 units distance.
 128  384  668  388  127  388  126  388  236  389  366  390  360  390
 375  396  122  396  238  397
CUMULATIVE NUMBER OF DISTANCES=          93
   .  .  18 lines for 2nd node is not shown here.
   .  .
CUMULATIVE NUMBER OF DISTANCES=         224

    This is the 3rd record of the index file.
    This shows that the 3rd place(ID 10003) is NOT eligible as a candidate.

        3    10003       0       0     1     0
THE FOLLOWING NODE IS A CANDIDATE
        4    10004     225     373     1   149
   4    0    5   27   48   34   45   46   46   55    3   59   47   65
   .    .    .
   .    .    .
```

Figure 10. Input for one Analysis to Compute the p-median Solution
for 11 Centers using the Teitz and Bart Heuristic Algorithm

```
//BLAQQQA6 JOB (15001808,20,,,,,0),'I4PARK',TIME=10,MSGLEVEL=(1,1)
//STEP01  EXEC PGM=NIGERIA,REGION=2000K
//STEPLIB  DD  DSN=USER.A5001622.NIGERIA,DISP=SHR
//LKED.SYSPRINT DD SYSOUT=A
//GO.FT09F001 DD UNIT=DISK,DISP=OLD,
     DSN=WYL.BLA.QQQ.NIGWTS
//GO.FT10F001 DD UNIT=DISK,DISP=OLD,
// DSN=USER.A5001808.RETR.INDEX.PARK.KM40.UFMT,
//    DCB=(RECFM=VBS,LRECL=1000,BLKSIZE=1004)
//GO.FT11F001 DD UNIT=DISK,DISP=OLD,
// DSN=USER.A5001808.RETR.DSTFILE.PARK.KM40.UFMT,
//    DCB=(RECFM=VBS,LRECL=1000,BLKSIZE=1004)
//GO.SYSIN DD *
P-MEDIAN FOR NIGERIA,11 CENTERS
```

These are unit numbers for three sets of input data files. unit 9 for weight file, unit 10 for index file, and unit 11 for distance file.

This is the title information- any title.

number of places in the distance matrix	index file of distances came from unravel or retrench	longest distance to be stored in this analysis	divide all populations by 10	these show the storage units for the input data files.

```
  675    1  400     10      10    11    9    9
```

For further information (Hillsman, 1980, P.108)

```
     0
```

This means that this problem involves no editing of distance strings (see Park and Rushton 1983, ALLOC 6B). This control item does not appear in ALLOC 6.

number of center locations to be evaluated	select the algorithm in this case the Teitz and Bart alg.	this information and the blanks in certain cases control the analysis by specifying the objective function and the material to be printed or stored (see Hillsman, 1980, pp.113-117)

```
   11    1                 1    1
```

These are the eleven place Id's that will be evaluated at the beginning then, in this case the Teitz and Bart heuristic algorithm will compute the best 11 locations to minimize average distance from the demand points to their closest center

```
   10002   10005   10032   10037   10052   20035   20054   20094
   30072   40069   40095
99999
/*
//
```

This completes the problem definition of this job.

Figure 11. Sample Output for the 11 Center p-median Problem

```
I  PROGRAM ALLOC VI                                  I
I                                                    I
I  WRITTEN BY  EDWARD L. HILLSMAN                    I
I              DEPARTMENT OF GEOGRAPHY               I
I              THE UNIVERSITY OF IOWA                I
I              IOWA CITY, IOWA  52242  USA           I
I                                                    I
I              COPYRIGHT C 1977 EDWARD L. HILLSMAN   I
------------------------------------------------------

RUN TITLE/ P-MEDIAN FOR NIGERIA,11 CENTERS          /
NUMBER OF NODES IN FILES                          675
PLUS ONE DUMMY NODE WITH ID OF 0
MAXIMUM DISTANCE TO BE SAVED                      400
NUMBER OF DISTANCE CLASSES IN INDEX FILE            1
READ INDEX FILE FROM UNIT                          10
READ DISTANCE FILE FROM UNIT                       11
READ PROBLEM DEFINITIONS FROM UNIT                  5
INDEX AND DISTANCE FILES ARE UNFORMATTED
READ WEIGHTS FORMAT FROM UNIT                       9
READ FORMATTED WEIGHTS FROM UNIT                    9
DIVIDE ALL WEIGHTS BY                              10
NO FIXED CENTERS
SET SECOND FACTOR EQUAL TO ZERO
------------------------------------------------------
NUMBER OF DISTANCES STORED IS                  202828
MAXIMUM PERMITTED IS                           210000
LENGTH OF LONGEST STRING IS                       518
NUMBER OF NODES INSIDE STUDY REGION IS            675
TOTAL WEIGHT AFTER SCALING IS                  101559
TOTAL INSIDE IS                                101559
PROVIDE INFEASIBLE SERVICE AT A COST OF       1164083
------------------------------------------------------
LIST OF NODE ID NUMBERS AND POPULATIONS


 10001 IDIYA      273   10002 IMALA     740   10003 KETU       23
 10005            142   10006 GBOGILAW   25   10007 IKEREKUO   59
 10009 OKEODO     139   10010 OLORUNDA  133   10011 AKINIYI     7
 10013 OCOWOYIN    93   10014 KESAM      98   10015 ILUGUN     47
   .    .    .
   .    .    .
   .    .    .
```

This information confirms that the input instructions were correctly interpreted.

The remainder of list of node ID's, names and populations

Figure 12.
========================

```
P-MEDIAN FOR NIGERIA,11 CENTERS
                    PROBLEM NUMBER      1

LOCATE    11 CENTERS                     MALG  ICON  MALG2  KRIT  NMAP  MMAP  MACH
                                            1     0     0     0     1     1     0
LIST OF CENTERS
 CENTER(ID & NAME)       WT(CENTER  OUTSIDE   TOTAL)  WT*DISTANCE  COST IF
                                                                   DROPPED
---------------------------------------------------------------------------------
    40069 OKENLA            13        8504       8517      113479     39286
      .     .     .
      .     .     .  remaining 9 centers
      .     .     .
    40095 IFO             1830       15513      17343      197350    112045

FOR THE LIST OF CENTERS ABOVE:
TOTAL ALLOCATED WEIGHTED DISTANCE IS                               809110
AVERAGE DISTANCE OF ALLOCATED PLACES TO NEAREST CENTERS IS          7.967
TOTAL UNALLOCATED POPULATION IS                                         0
  WEIGHT:       TOTAL=      101559
              CENTERS=       33976
      OUTSIDE CENTERS=       67583
            % OUTSIDE=       66.55
AV DISTANCE OF OUTSIDE=      11.97

LIST OF NODES

      NODE       CENTER     WEIGHT  DIST          NODE       CENTER     WEIGHT  DIST

II 10025   10052 ABEOKUTA     47     4  II 10026   10032 ISAGA        32      8
II 10028   10032 ISAGA        22     6  II 10029   10032 ISAGA        32      5
II 10031   10032 ISAGA       470     5  II 10032   10032 ISAGA       667      0
II 10034   10037 KUTA         14     2  II 10035   10037 KUTA         11      4
II 10037   10037 KUTA         23     0  II 10038   10037 KUTA          8      3
II 10040   10037 KUTA         35     3  II 10041   10037 KUTA        157      5
      .     .     .
      .     .     .   the remainder of list of nodes in the starting solution
      .     .     .

LIST OF TRADE AREAS (NODE ID, WEIGHT, AND DISTANCE TO CENTER)

  10002  740    0       20055   57    2          10050    9    3
    .     .     .
    .     .     . the remainder of nodes assigned to the center 10002
  10058  443   16       30131   47   19

  10005  142    0       10004   43    2          10045   47    3
    .     .     .
    .     .     . the remainder of nodes assigned to center 10005
  10010  133    4       10012    8    5          10003   23    8
    .     .     .
    .     .     . the remaining 9 groups of trade areas
    .     .     .143 lines are not shown here.
```

Figure 13.

SUMMARY STATISTICS
TOTAL ALLOCATED WEIGHTED DISTANCE IS 809110
AVERAGE DISTANCE OF ALLOCATED PLACES TO NEAREST CENTERS IS 7.967
TOTAL UNALLOCATED POPULATION IS 0
OVER ENTIRE PROBLEM, MAXIMUM DISTANCE TRAVELED IS 39
FROM NODE 20001 TO CENTER 20035
INSIDE STUDY REGION, MAXIMUM DISTANCE TRAVELED IS 39
FROM NODE 20001 TO CENTER 20035
MOST EXPENDABLE CENTER IS 10032
WHICH WOULD INCREASE THE OBJECTIVE FUNCTION BY 5816
IF DROPPED WITHOUT REPLACEMENT

START TEITZ AND BART ALGORITHM

OLD CENTER	COST IF DROPPED	NEW CENTER	TOTAL COST	NET CHANGE	PERCENT CHANGE
10032	5816	10001	808426	684	0.0845
10005	7612	10007	805956	2470	0.3055
10007	10082	10009	805540	416	0.0516
10009	10498	10016	804592	948	0.1177
40098	82221	40237	601064	8829	1.4476
---END CYCLE 1					
CHANGES = 60					
10002	202253	10001	599804	1260	0.2096
. . . 17 lines are not shown here.					
20125	72432	20132	575211	1485	0.2575
---END CYCLE 2					
CHANGES = 19					
---END CYCLE 3					
CHANGES = 0					

END TEITZ AND BART ALGORITHM

LIST OF CENTERS

CENTER(ID & NAME)	WT(CENTER	OUTSIDE	TOTAL)	WEIGHT * DISTANCE	AVERAGE DISTANCE (ALL	& OUTSIDE)	COST IF DROPPED
10010 OLORUNDA	133	4264	4397	51888	11.80	12.17	1208048
20132 TOSUN	375	6173	6548	54183	8.27	8.78	73917
. . . 9 lines are not shown here.							
40237 AIYEDE	47	9181	9228	64782	7.02	7.06	90755

FOR THE LIST OF CENTERS ABOVE:
TOTAL ALLOCATED WEIGHTED DISTANCE IS 575211
AVERAGE DISTANCE OF ALLOCATED PLACES TO NEAREST CENTERS IS 5.664
TOTAL UNALLOCATED POPULATION IS 0

WEIGHT:	TOTAL=	101559
	CENTERS=	33303
	OUTSIDE CENTERS=	68256
	% OUTSIDE=	67.21
AV DISTANCE OF OUTSIDE=		8.43

Figure 14.

```
=======================

LIST OF NODES

     NODE       CENTER    WEIGHT  DIST          NODE        CENTER   WEIGHT  DIST
DIST

II 10001  10010 OLORUNDA    273    12  II 10002  10010 OLORUNDA      740     15
II 10004  10010 OLORUNDA     43     6  II 10005  10010 OLORUNDA      142      4
 .     .  .
 .     .  .  the remainder of list of nodes in the optimal solution
 .     .  .
II 40257  10052 ABEOKUTA     23    19  II 40258  10052 ABEOKUTA       46     19
II 40260  10052 ABEOKUTA     47    19  II 40261  10010 OLORUNDA       14     20


LIST OF TRADE AREAS (NODE ID, WEIGHT, AND DISTANCE TO CENTER,
  WHICH IS FIRST ID IN EACH AREA)
  10010  133    0        10012    8    1        10009  139    2    2
  .      .      .  the remainder of nodes assigned to center 10010
  .      .      .  41 nodes were assigned to this center including itself.
  .      .      .
  10002  740    15       10050    9   16        10051    5   17   18
  10058  443    23
  41

  .      .      .  remaining 10 groups of trade areas
  .      .      .  About 150 lines were not shown here
  .      .      .


SUMMARY STATISTICS
TOTAL ALLOCATED WEIGHTED DISTANCE IS                              575211
AVERAGE DISTANCE OF ALLOCATED PLACES TO NEAREST CENTERS IS         5.664
TOTAL UNALLOCATED POPULATION IS                                        0
OVER ENTIRE PROBLEM, MAXIMUM DISTANCE TRAVELED IS          23
  FROM NODE    10058 TO CENTER    10010
INSIDE STUDY REGION, MAXIMUM DISTANCE TRAVELED IS          23
  FROM NODE    10058 TO CENTER    10010
AVERAGE VALUE OF SECOND FACTOR IS    0.0
MOST EXPENDABLE CENTER IS    20009
WHICH WOULD INCREASE THE OBJECTIVE FUNCTION BY       42663
  IF DROPPED WITHOUT REPLACEMENT
PERCENT CHANGE IN OBJECTIVE FUNCTION FROM INITIAL LIST  OF CENTERS IS   28.9082
                                      FROM LAST PRINTING IS             28.9082
------------------------------------------------------------------------------
------------------------------------------------------------------------------
```

Figure 15.
========================

<COMPARISON OF CENTERS BETWEEN BEGINNING AND END SOLUTIONS>>

NEW CENTERS(CENTERS NOT IN THE BEGINNING SET)

ID	NAME	WT(CENTER	OUTSIDE	TOTAL)	WT*DIST	NODES	AV DIST(ALL	& OUTSIDE)
10010	OLORUNDA	133	4264	4397	51888	41	11.80	12.17
. . .	. . .	**7 lines are not shown here.**						
40237	AIYEDE	47	9181	9228	64782	89	7.02	7.06
TOTAL	9 CENTERS	3805	50561	54366	403373	515		
AVERAGE		422.8	5617.9	6040.7	44819.2	57.2	7.42	7.98
AV WT/NODE OUTSIDE			99.9					

REMAINING CENTERS(CENTERS BOTH IN BEGINNING AND END SETS)

ID	NAME	WT(CENTER	OUTSIDE	TOTAL)	WT*DIST	NODES	AV DIST(A	& O)
10052	ABEOKUTA	29175	6128	35303	59115	60	1.67	9.65
20054	AROSA	323	11567	11890	112723	100	9.48	9.75
TOTAL	2 CENTERS	29498	17695	47193	171838	160		
AVERAGE		14749.0	8847.5	23596.5	85919.0	80.0	3.64	9.71
AV WT/NODE OUTSIDE			112.0					

**** BEGINNING SET ****

	WT(OUTSIDE	TOTAL)	WT*DIST	NODES	AV DIST(A	& O)
	2910	32085	18214	23	0.57	6.26
	7927	8250	76578	76	9.28	9.66
TOTAL	10837	40335	94792	99		
AVERAGE	5418.5	20167.5	47396.0	49.5	2.35	5.36

OLD CENTERS(CENTERS NOT IN THE END SET)

ID	NAME	WT(CENTER	OUTSIDE	TOTAL)	WT*DIST	NODES	AV DIST(ALL	& OUTSIDE)
10002	IMALA	740	718	1458	10066	7	6.90	14.02
. . .	. . .	**7 lines are not shown here.**						
40095	IFO	1830	15513	17343	197350	154	11.38	12.72
TOTAL	9 CENTERS	4478	56746	61224	714318	576		
AVERAGE		497.6	6305.1	6802.7	79368.6	64.0	11.67	12.59
AV WT/NODE OUTSIDE			100.1					

Figure 16.

<<COMPARISON OF NODE ASSIGNMENTS>>

NODE		WEIGHT	CENTER		DISTANCE GAIN/LOSS				WT*DIST GAIN/LOSS		
ID	NAME		BEGIN	END		BEGIN	END		BEGIN	END	
10001	IDIYA	273	10005	10010	1	9	12	-3	2457	3276	-819
10002	IMALA	740	10002	10010	1	0	15	-15	0	11100	-11100
10003	KETU	23	10005	10010	1	8	11	-3	184	253	-69
10004	IMALAW	43	10005	10010	1	2	6	-4	86	258	-172
10005		142	10005	10010	1	0	4	-4	0	568	-568
10006	GBOGILAW	25	10005	10010	1	4	4	0	100	100	0
NODE 10006 IS ASSIGNED TO DIFFERENT CENTER BUT HAS SAME DISTANCE.											
10007	IKEREKUO	59	10005	10010	1	4	3	1	236	177	59

. . .
. . . **the remainder of node assignment comparison**
. . . **About 710 lines were not shown here.**

<<THE SPATIAL ACCESSIBILITY OF PEOPLE AND PLACES FOR THE BEGINNING AND END SET OF CENTERS>>

	CHANGES IN SPATIAL ACCESSIBILITY		
	SAME	CLOSER	FARTHER
NUMBER OF PEOPLE	45406	35397	20756
PERCENT OF PEOPLE	44.71	34.85	20.44
NUMBER OF PLACES	147	342	186
PERCENT OF PLACES	21.78	50.67	27.56

<<COMPARISON OF THE SPATIAL ACCESSIBILITY BETWEEN THE BEGINNING AND ENDING SET OF CENTERS>>

	SAME	CLOSER	FARTHER
AVERAGE DISTANCE IN THE BEGINNING	3.00	15.12	5.11
AVERAGE DISTANCE AT THE END	3.00	5.86	9.49
% RATIO OF DISTANCES(END/BEGIN)	100.00	38.77	185.56
AVERAGE NO. OF PERSONS IN PLACES	308.88	103.50	111.59

CENTER LOCATIONS AT BEGINNING OF PROBLEM

10002 10005 10032 10037 10052 20035 20054 20094
30072 40069 40095

LOCATIONS OF CENTERS AT END OF ALGORITHM

10010 20132 20009 40143 10052 20041 20054 40228
20100 40033 40237

END OF PROBLEM
OBJECTIVE FUNCTION AT START AND END
1 809110 575211