CHAPELLE, René Adrien, 1933-
LOCATION OF CENTRAL FACILITIES--
HEURISTIC ALGORITHMS FOR LARGE
SYSTEMS.

The University of Oklahoma, Ph.D., 1969
Engineering, industrial

THE UNIVERSITY OF OKLAHOMA

GRADUATE COLLEGE

LOCATION OF CENTRAL FACILITIES
HEURISTIC ALGORITHMS FOR LARGE SYSTEMS

A DISSERTATION

SUBMITTED TO THE GRADUATE FACULTY

in partial fulfillment of the requirements for the

degree of

DOCTOR OF PHILOSOPHY

BY

RENÉ ADRIEN CHAPELLE

Norman, Oklahoma

1969

LOCATION OF CENTRAL FACILITIES

HEURISTIC ALGORITHMS FOR LARGE SYSTEMS

APPROVED BY

DISSERTATION COMMITTEE

ACKNOWLEDGMENT

I wish to express grateful appreciation to all who have made this undertaking possible.

In particular I extend thanks to my committee chairman Dr. Robert A. Shapiro for his guidance and suggestions. Special gratitude is due to all other members of my committee consisting of Professors Robert J. Block, Rodney L. Boyes, Bobbie L. Foote, Raymond P. Lutz.

I first became interested in location of central facilities during a summer employment with the Post Office Department in Washington, D.C. The motivation provided by the director of the Industrial Engineering Bureau, Mr. Alvin P. Hanes, has been instrumental in the development of this study. I am also grateful for the communication with Professor Leon Cooper of Washington University at St. Louis.

ABSTRACT

This study deals with the optimization of location of multiple central facilities in a large system of dependent elements connected by communication links.

This problem is of major importance in the location of service facilities. The function to optimize is a minimization of link costs due to time delay or connection expenses.

The case of small systems with one central location is a time-honored problem solved in the case of Euclidean distances by analog, geometric and numerical methods. The exact solution of the problem in a closed and explicit form has not yet been found. It often deals with the problems of industries or warehouses or communication center locations connected by straight links, and it has been extensively studied by Launhardt, Weber, Isard, and Cooper to mention a few. Other investigators have considered the case of a Manhattan metric in which the connecting links are perpendicular segments, and the results are tentatively applied to plant or city layouts.

The case of large systems and multiple central locations leads to excessive computing efforts often impossible even on the largest computers. Some algorithms are available but are somewhat inflexible. It is for this reason that new algorithms were developed; the variable grid algorithms and the variable discrimination algorithm. Whatever the size of the system, the combination of these algorithms allows a possible and rapid location of central facilities within the constraints of computer time and memory expenditure. A complete study and programming of these algorithms are presented to allow direct application by the engineer or the economist. Some possible applications to the Post Office Department are shown for the location of the sectional centers serving the numerous post offices of any given state.

It is shown that those algorithms may be extended to n dimensional space and can be used to large cluster analysis.

TABLE OF CONTENTS

Chapter | Page

Chapter Page

LIST OF ILLUSTRATIONS

Figure | Page

Figure | Page

LIST OF TABLES

LIST OF LOGIC DIAGRAMS

LIST OF COMPUTER PROGRAMS

I.1. LOCATION OF CENTRAL FACILITIES

CHAPTER I

INTRODUCTION

Generalities

Our physical, social and economical behavior depends on a set of networks: biological networks to control our thoughts and motions, social networks allowing interchange of ideas and generation of actions, and economical networks to sustain our wants and distribute our goods.

These networks are limited in connection richness and flow capacity. To connect various elements of the system it may take time and it may generate cost. We are familiar with the old adage that "time is money", a more appropriate statement should be distance is money.

For the housewife it takes time to go shopping, time to take the children to school, time to prepare the dinner. For the engineer it takes time to materialize an idea into a design, time to do some market research, time to analyze a feasible production method, time to receive the material, time to route it through the plant, time to evaluate, organize, implement. We behave to create a comfortable balance with our environment, but it takes time to respond to environmental changes because of spatial limitation in our information network and a delayed reaction might be uncomfortable or even fatal.

Distance is the constraining element in most of our actions. The daily commuter complains about the long ride to work, the housewife considers that the shopping center is too far from home, that the kitchen is poorly laid out; the child does not appreciate the long walk to school on a chilly morning. The postman would prefer a shorter route, the fireman less hurry on a long stretch, the salesman less territory to cover and the wounded less distance to the hospital. The industrialist would like the markets to be situated close to the resources, the store manager would offer a better service if the warehouse were a block from his retail store, the weatherman could be more accurate if the data gathering station were in his backyard, the engineer more inventive if a well furnished library were in his office and more effective if he could communicate readily without ambiguity with his staff of foremen in the shop or salesmen in the field.

It takes time and effort because of distances, messages are distorted and goods are scarce because of long network branches with constrained capacity. The problem of time is often a problem of space and the problem of space a problem of location.

I.1. LOCATION OF CENTRAL FACILITIES

CHAPTER I

INTRODUCTION

Generalities

Our physical, social and economical behavior depends on a set of networks: biological networks to control our thoughts and motions, social networks allowing interchange of ideas and generation of actions, and economical networks to sustain our wants and distribute our goods.

These networks are limited in connection richness and flow capacity. To connect various elements of the system it may take time and it may generate cost. We are familiar with the old adage that "time is money", a more appropriate statement should be distance is money.

For the housewife it takes time to go shopping, time to take the children to school, time to prepare the dinner. For the engineer it takes time to materialize an idea into a design, time to do some market research, time to analyze a feasible production method, time to receive the material, time to route it through the plant, time to evaluate, organize, implement. We behave to create a comfortable balance with our environment, but it takes time to respond to environmental changes because of spatial limitation in our information network and a delayed reaction might be uncomfortable or even fatal.

Distance [illegible]training element in most of our actions. The [illegible]lains about the long ride to work, the h[illegible] the shopping center is too far from h[illegible]poorly laid out; the child does not a[illegible] school on a chilly morning. Th[illegible] a shorter route, the fireman less hurry[illegible], the salesman less territory to cover and the wo[illegible] distance to the hospital. The industrialist would like the markets to be situated close to the resources, the store manager would offer a better service if the warehouse were a block from his retail store, the weatherman could be more accurate if the data gathering station were in his backyard, the engineer more inventive if a well furnished library were in his office and more effective if he could communicate readily without ambiguity with his staff of foremen in the shop or salesmen in the field.

It takes time and effort because of distances, messages are distorted and goods are scarce because of long network branches with constrained capacity. The problem of time is often a problem of space and the problem of space a problem of location.

In all our previous examples we had these constraints because of space requirements. It takes time to think and act because our nervous system is formed of multiple paths of finite lengths, traversed by pulses of finite velocity, we have a finite number of neurones with finite number of interconnections. The computer manufacturer has quickly realized the importance of space on the computation time of their machines, and made their circuits with microelements. With our brain we must be able to control the writing on this page or the movement of our foot; the computer may have to drive a printer as well as disc packs or tape reels. To minimize message transmission to tape outputs it is wise to think that a central location of the main processor would be advisable. Nature has furnished us with a central processor shrinking nervous paths into some of our mental processes but also peripheral processes for our reflexes. In the operation of our processing system we must ingest some input data or raw materials, for example the design problem we expect to solve requires the use of a book in fluid mechanic at the University library, a similar design has been used on a machine in Japan, a technical study of this machine has been published in Russia. Similarly the computer may have to use a magnetic tape in the computing room, punch card readers from the manufacturing plant, light pen oscilloscopes from the various engineering departments, punched tape transmitted by teletype from remote terminals across the country. It is a problem of space allocation to define the location of our laboratory in function of the available literature and expertise on the subject. It is a problem of space allocation to install our computer at the right place so as to minimize message delays and cost of wiring and transmission.

In a manufacturing environment for example the economic factor of time is often a disguised problem of space. Labor and overhead costs accumulate with time and it takes time for procurement of raw material, time to transform the input into finished products, time to move parts from one work station to another, time to move carriages, slides, tables, spindles, tools into working position, time to gather components to assemble a final product. It takes time because new materials must be produced in distant markets, the tools must be moved along a cutting path, the assembly components must be handled from the storage bins. To minimize time is to minimize the space transport of materials or messages. This is the problem of plant location with respect to raw material input and market output for the finished product. Plant layout deals with the problem of geometrical arrangements of machines and men to decrease distances, and motion study tries to simplify work station to reduce body displacements and fatigue.

In a community environment we rely on a multitude of county, state, or national service agencies. Their location should allow a rapid contact in case of need.

Who cares about an excellent fire department if our house has time to burn to the ground while the fire truck is on the road; who likes a modern hospital so far out that the patient may die en route? To minimize time we must minimize space. This is clearly evident in our modern societies in the tremendous rate of growth of urban communities and high-rise buildings where space is hopefully condensed so as to simplify our constant communications. The city service agencies cater to the people and they must be situated so as to please the largest number of citizens by a correct location of their premices. However, no city is independent, no country is autonomous and the optimum location of a facility will be different if we are only concerned with a restricted environment or of the whole possible interconnections. Service agencies respondent to many cities or many countries are part of a very large system and the optimum location of center minimizing transmission time of message or goods deals with a multitude of elements.

Therefore the problems of location that we consider to tackle in this dissertation are closely related to the theory of graphs and circuits, to the theory of flow and transportation and to the theory of information. We will limit this study to the location of central facilities and the corresponding rational clustering of satellites which will be serviced in the optimum manner within the physical or economical constraints imposed on the system.

I.2. Plan of Study

In the general problem we define the location of n elements with their respective requirements or supplies. These n elements are involved in transaction with a set of m elements.

The set of n elements may be restricted in size and the elements sufficiently apart to consider space as discrete; on the other hand, the set may be extremely large and the n elements so closely located that a measure of density in this continuous space will have to be introduced.

The set of m elements may be a subset of the n elements or may be a completely different set.

To the transaction is affixed a figure of merit: time, distance, cost, etc., which must be optimized by the proper choice of the set of m elements with respect to their location, constraints of capacity, characteristics of the connecting channels and economic constraints.

When a central facility must be reached by depending satellites, the means of connection may be direct along geometric straight lines. This is the case, for example, of airfreight with straight paths between the central suppliers and the various cities, the case also of telecommunication between central emitters and related receptors, etc.

Quite often in practice the transportation network between central facilities and related elements must be done through a maze of streets, roads and sinuous paths. At the scale of a city the connecting links may be a succession of perpendicular streets and avenues that a citizen, for example, must follow to reach the closest post office. At the scale of the United States it is a set of road elements often oriented North-South and East-West, that a delivery truck must drive to reach the city of the retail store. In a plant it is the nicely aligned aisles along which are lined the machines and which are crisscrossed by forklifts or other material handling equipment.

Our n elements not only have spatial characteristics but they may have a complete vector of characteristics and we might be interested not in the location of various central facilities but in the grouping of elements with nearly similar entities. We are no longer dealing in space allocation but in cluster analysis or entity allocation.

Our investigation could therefore consider various levels of complexity. We could look first at a discrete space with an interconnecting network of straight lines between satellite elements and central facilities. In this Euclidean space we would study the location of one central facility deserving few elements then we would extend our study to the problem of multiple central locations. We could note the possible expansion to an N dimensional space by a brief mention of clusters. We could then tackle the problem of a Manhattan space of intersecting streets and avenues or a network of roads properly aligned along longitude and latitude. We could then consider the case of a dense set of elements to be served in a continuous space of given density. Table 1 represents most of the possible combinations which may occur in locational theory.

Space (elements to serve)	Discrete	,	Continuous (large number)
Central Facilities (or clusters)	One	,	Multiple
Distances	Euclidean 1 dimensional N dimensional	, Manhattan,	Sinuous Paths
Transactions	Equal	,	Unequal
Constraints	Constrained	,	Unconstrained

Table 1. Location-Allocation Problems

This dissertation cannot deal with all the possible combinations of problems presented in the figure above; therefore, some areas will be emphasized at the expense of others and the concluding chapter will show possible areas for future research.

The optimal location of central facilities is an age-old problem which attracted many mathematicians of reknown since the 17th century, however, the simplified assumptions lead to mathematical models which do not really apply to some common complex situations. In our large communities and at the scale of our country it is rare to find only one centralized facility serving the individuals. Cities have many fire stations, states have many central post offices, countries have multiple centralized data gathering and distributing locations. With the advent of the computer, heuristic solution of the locational problem becomes possible but we are still limited in our theoretical study by the immensity of the problem. Flexible heuristic algorithms for very large systems will be emphasized to allow direct application by the engineer or economist.

I.3. Applications

A tentative tabulation of possible applications follows. It is quite incomplete but will give an idea of the problems which may be tackled with the present algorithms.

Table 2. Applications of Locational Problems

Elements Requiring Service	Centralized Facilities or Clusters
Individual as a member of a community	For Transportation taxi stations bus stations railway stations air terminals airports For Communication post offices public telephones telegraph offices For Education schools libraries churches For Security police stations fire stations hospitals, doctors For Supplies stores For Entertainment theaters stadiums TV & radio stations For Administration city govt. agencies county govt. agencies state govt. agencies federal govt. agencies

Resources and Markets	Industrial Plants Warehouses Distribution Centers Communication Centers Data Gathering Centers Production Centers Regulating Centers Group of Markets with similar characteristics
Entities Machines, controls, organizations, elements etc.	Central Processors and Regulators Management Centers Taxonomy, etc.

CHAPTER II

LOCATION OF CENTRAL FACILITIES

DISCRETE, TWO-DIMENSIONAL SPACE, EUCLIDEAN DISTANCES

ONE CENTRAL LOCATION

We are considering in this case a finite set of n discrete elements associated with a set of two characteristics which may be their cartesian coordinates in a plane, or a set of two entities X_1 and X_2.

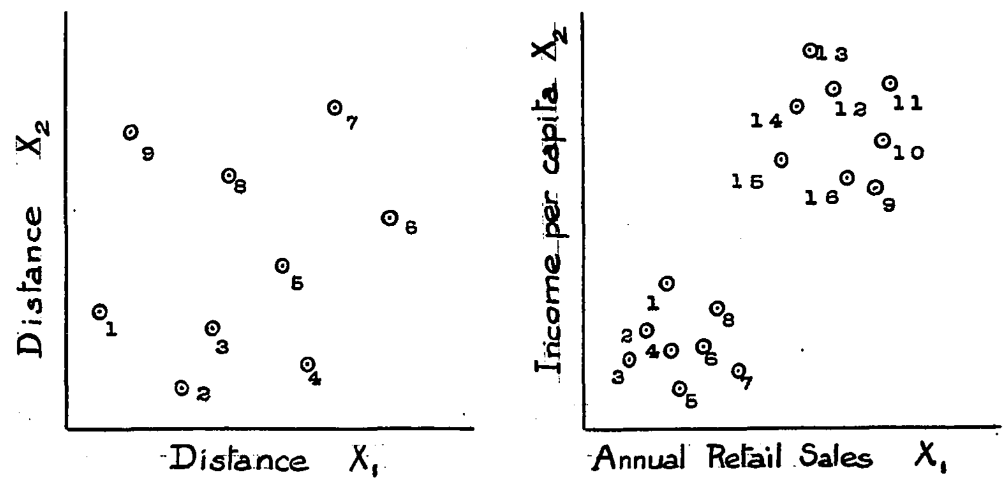


Fig. 1. Discrete Two-dimensional Space

In considering a network of straight lines connecting these points, we try to optimize the layout of this network of lines with respect to a figure of merit.

In the case of one central location the network of straight lines must be connected to one central node through which all transfers will be made.

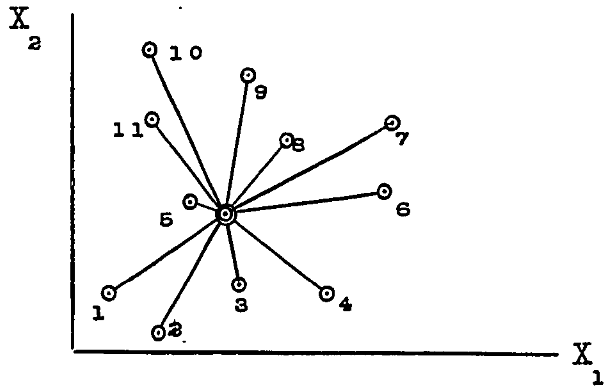


Fig. 2. Discrete Two-dimensional Space
Euclidean Distances
1 Central Location

There is no lateral transfer between 2 elements except through the common central facility.

The problem is often considered as the minimization of the sum of Euclidean distances joining each facility to the central location. However, the amount of transfer along each line may be variable in size, the branches and facilities may have limit in capacities and what may appear to be the shortest route may not be the optimum one when considering time or cost of communication. In this case, weighted distances will have to be introduced and sets of constraints will be added to the problem.

When constraints are ignored, the problem in this simplest form may be solved by various methods which will be described in the sections which follow.

II.1. ANALOG SOLUTION

II.1.1. Lights and Mirrors

When light is emitted from point A and impinges on a flat mirror to reach point B it follows a minimum distance path. The problem is to find the point of incidence. A similar problem was solved by the ancient Greeks to find the shortest path between 2 locations close to a stream if on the way a pail of water had to be fetched.

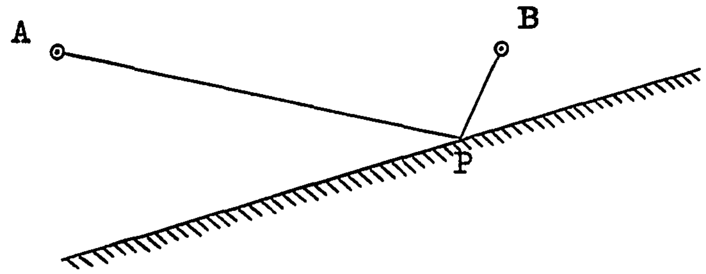


Fig. 3. Light and Mirror

When the distance AP+PB is given, P is located on an ellipse of loci A and B. The set of ellipses with loci A and B represent the possible location of point P. As the distance AP+PB decreases, it reaches a minimum value for which the ellipse is tangent at point P which is the solution to our problem

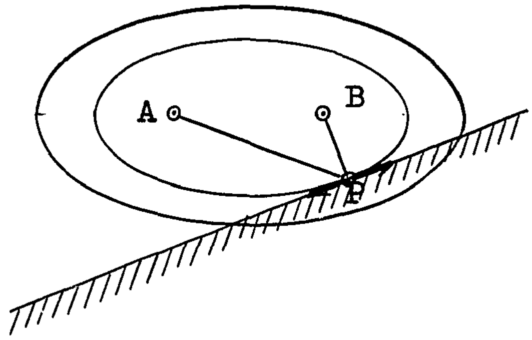


Fig. 4. Light and Mirror

Minimum Distance Path

In the case of 3 points A,B,C which must be connected by a network to a central facility so as to minimize distances, Polya [48] shows that a similar physical approach involving light can be made. In that case if we assume that the central facility P is located at a distance r from C, then it must be on a circle centered on C and of radius r. The minimization of distance AP+PB corresponds to a path of light impinging on a circular mirror. According to the laws of reflection $\alpha_1 = \alpha_2$ and due to the symmetry of the situation a similar reasoning undertaken from C can be done from A, then B. Therefore, all angles α, β and γ are equal to 60° and the central facility is at the intersection of the network of links oriented at 120°.

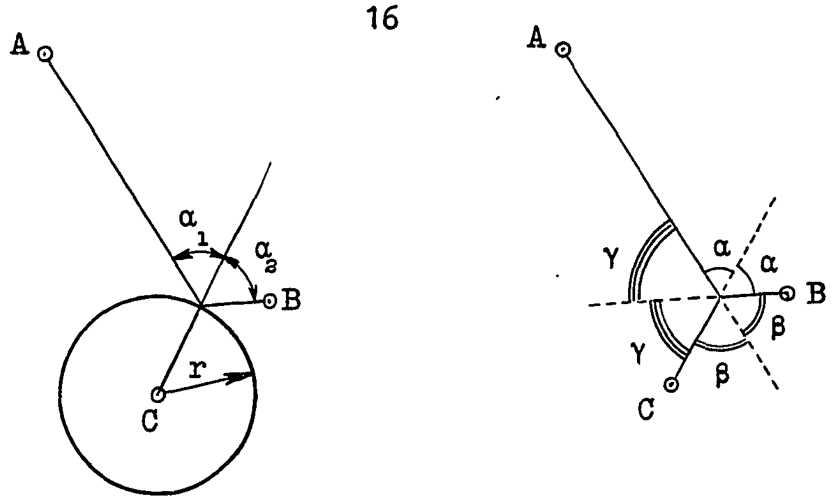


Fig. 5. Light and Mirrors

Central Location of 3 Facilities

II.1.2. Weights and Pulleys

This solution is not historically the first one, but it allows a visualization which aids greatly in comprehension as the problem gains in complexity.

Let us first consider 3 locations A, B, C engaged in transaction with a central facility P. If the volume of transaction to and from these three locations are identical and if the links are not constrained in capacity then the optimum location of the central facility will minimize the sum of distances PA+PB+PC. If the 3 points are plotted on a vertical plane and equipped with small pulleys, the optimum location of the central facility is given by the position of equilibrium of a knot connecting 3 strings passing over the pulleys and carrying 3 equal weights at their extremities.

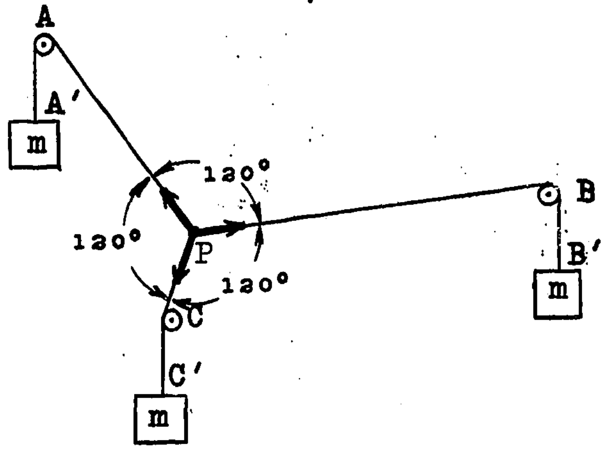


Fig. 6. Weight and Pulleys
Central Location of 3 Facilities

We will prove that equilibrium is reached when the potential energy of the system is at a minimum, that is when all the weights are in the lowest position possible. This corresponds to the maximum of AA'+BB'+CC' or the minimum of PA+PB+PC. Through static consideration the equilibrium is obtained when the system of forces are equally inclined at 120° to each other.

If the transactions from point A, B and C to point P are not identical, proportional weights could be used. For example, if costs are to be minimized in the transport of a, b, c tons of materials respectively from A,B,C to P, then we should minimize $a.|AP|+b.|BP|+c.|CP|$ with weights m_1, m_2, m_3 at A,B,C proportional to the transport weights a, b, c. In this general case, the system of mass in equilibrium gives the minimization of link distances [47].

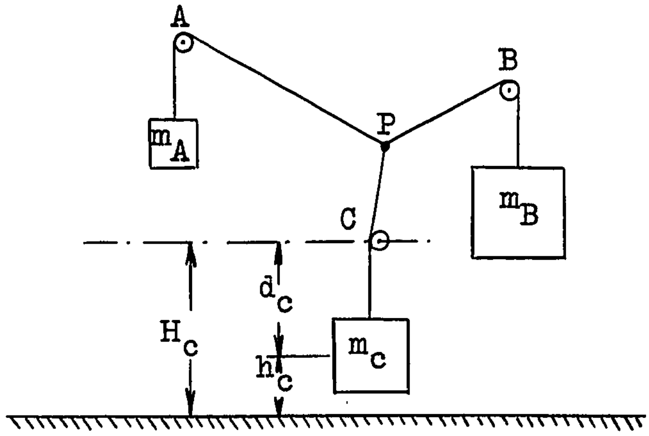


Fig. 7. Weight and Pulley

Minimization of Potential energy

With L_i : length of the string at each respective point

d_i : length of the string from the pulley to the weight

l_i : length of the string from the pulley to the knot

H_i : distance of the pulley to the reference horizontal plane

m_i : weight at end of string

$$\sum_i m_i H_i = \sum_i m_i h_i + \sum_i m_i d_i$$

$$\sum_i m_i L_i = \sum_i m_i d_i + \sum_i m_i l_i$$

$$\sum_i m_i L_i - \sum_i m_i H_i = \sum_i m_i l_i - \sum_i m_i h_i$$

$$\sum_i m_i L_i - \sum_i m_i H_i + \sum_i m_i h_i = \sum_i m_i l_i$$

as $\sum_i m_i L_i = \text{Constant}$

and $\sum_i m_i H_i = \text{Constant}$

the minimization of $\sum_i m_i l_i$ corresponds to the minimization of $\sum_i m_i h_i$ which is at the minimum of potential energy.

In 1775 Fagnano studied the location of a point minimizing distances in a system of 4 elements. In the case of equal transaction, the central facility was found at the intersection of the segments connecting opposite points. The result can be seen immediately by using our analog weight system.

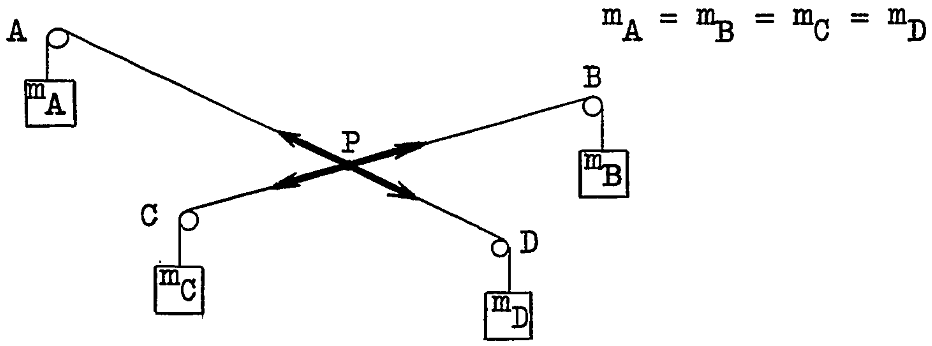


Fig. 8. Weight and Pulleys
Central Location of 4 Facilities

A similar central facility would be found if we had identical transaction from opposite points $m_A = m_D$, $m_B = m_C$ with $m_A \neq m_B$.

This analog approach can be used, theoretically, for any set of n facilities; however, it is, in practice, restricted by the following limitations: the strings are not perfectly flexible and the friction is not negligible. It is interesting to note that the location of the central facility as found by this model is not the center of masses as one might subjectively assume. A clever use of this analog procedure may be extremely useful in complex problems, it has already been used in a number of practical problems [16] [33] [8].

II.1.3. The Link-Length Minimizer

In 1957 Miehle [43] under contract with the Department of the Army designed a mechanical device able to define the central location of a relatively large system. On a horizontal surface are installed some fixed pegs with pulleys at the location of elements to be serviced, a movable peg represents the central facility. The interconnections are made with a loop of string. By pulling the end of the string, because of potential energy consideration the movable peg will be positioned to minimize the total string length.

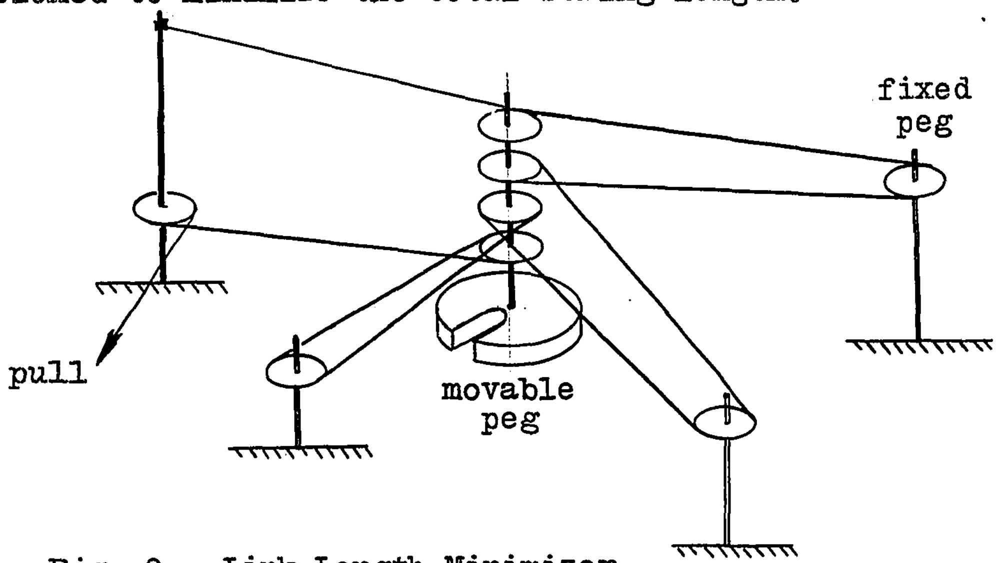


Fig. 9. Link-Length Minimizer
Location of One Central Facility

This method is relatively simple; however, it requires a physical model and the friction at the pulleys may be large and the movable peg must be manually positioned in order to retrieve the slack on the string. A relatively large system can be treated and we will see in Chapter III that the device may be applied to the location of multiple central facilities. Transaction weight on a given link may be added to the model by multiple increments created by multiple looping.

II.1.4. Electrical Field

Electrodes connected to a DC power source through a resistive network are located on a plane map of facility locations. At a given point of this plane, the total field is the sum of the elementary fields weighted by the proper resistances. This total field may be measured by an omni-directional detector and, with a scale factor, is analogous to the total transport cost. Equi-field lines can then be constructed representing iso-cost lines, these are concentric around the optimum location and converge toward that optimum when the detector sensitivity is increased. The method is relatively simple but lacks accuracy as we shall see later (II.2.3). An ingenious electrical analogue machine has been developed by Mr. William Bernard under Air Force contract AF18(600)-125 [7].

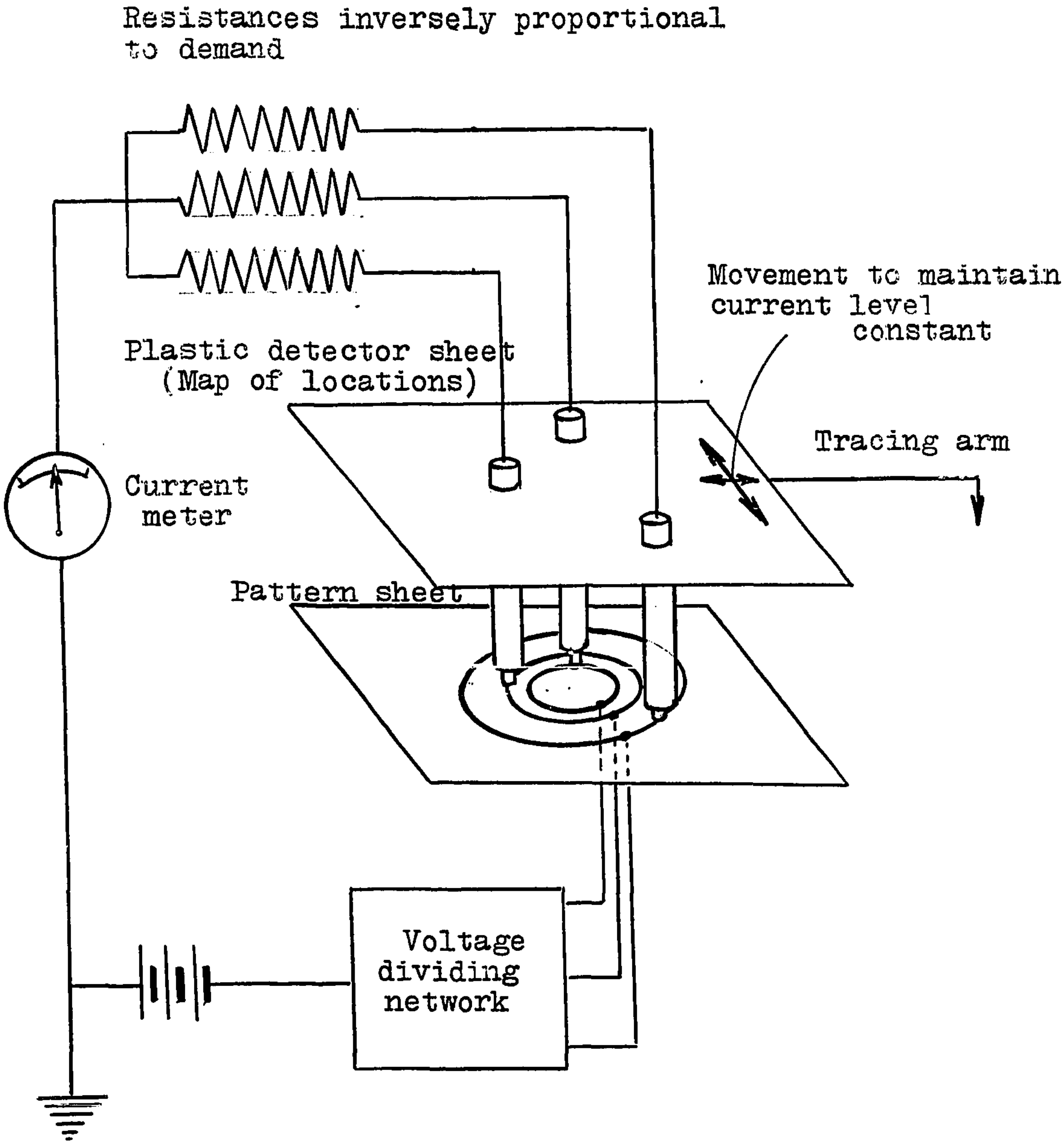


Fig. Electric Field Analogue Computer

II.2. GEOMETRIC SOLUTION

II.2.1. The Force Polygon

In the analog model of weight and pulleys the forces acting on the common string knot are in equilibrium and form a force parallelogram. The study of which is put to use in the following geometric construction defining the position of the central facility P.

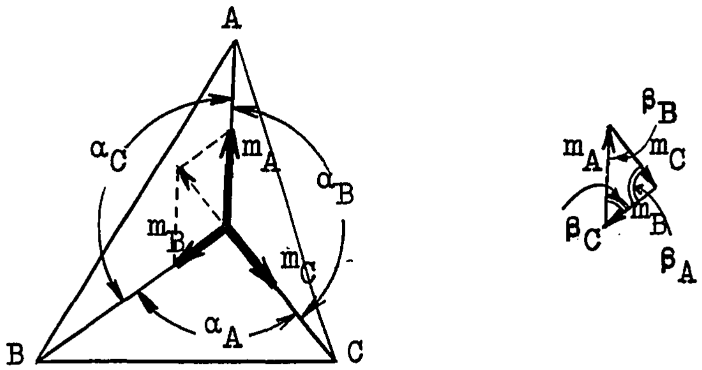


Fig. 11. The Force Polygon

The angles β_A, β_B, β_C are the supplements of the respective angles α_A, α_B, α_C. The following construction is suggested by George Pick in the mathematical appendix of Weber's theory on location of industries [54] [26]. In a circle sustaining the arc $\widehat{AB}$, the angle β is the supplement of α, and the point C can be anywhere on the arc $\widehat{ACB}$.

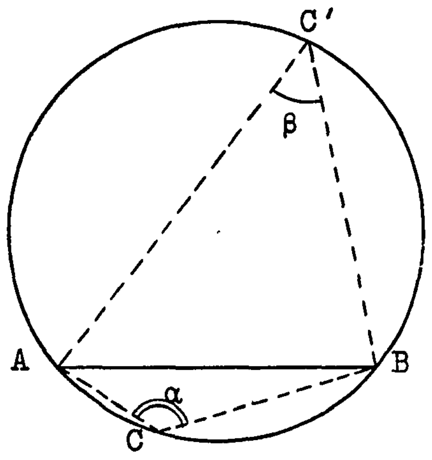


Fig. 12. Construction of Supplemental Angles

Knowing the weights acting at each location, a polygon of force may be built, thus defining the angles β_i. The supplemental angles α_i are found on circles sustending β_i on one side.

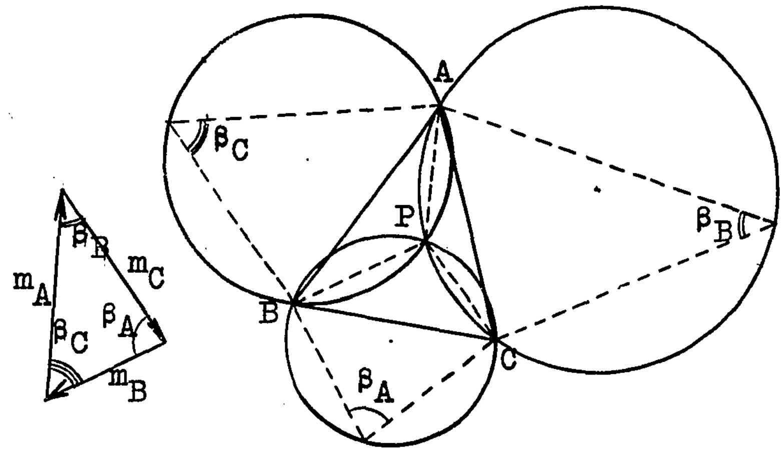


Fig. 13. Geometric Location of a Central Facility

For computational ease we will mention the trigonometric formulation of angles β_i.

With $2p = m_A + m_B + m_C$

$$\sin \frac{\beta_A}{2} = \left(\frac{(p - m_B)(p - m_C)}{m_B\ m_C}\right)^{\frac{1}{2}}$$

$$\sin \frac{\beta_B}{2} = \left(\frac{(p - m_C)(p - m_A)}{m_C\ m_A}\right)^{\frac{1}{2}}$$

$$\sin \frac{\beta_C}{2} = \left(\frac{(p - m_A)(p - m_B)}{m_A\ m_B}\right)^{\frac{1}{2}}$$

In the case of a negligible weight m_i at one location i, the corresponding angle β_i is negligible and α_i is practically 180°, the location of the central facility is then on the opposite line.

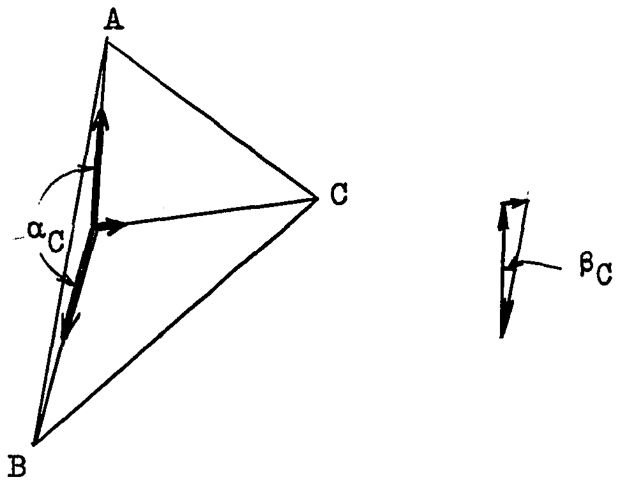


Fig. 14. Case of a Negligible Weight

The location of P on that particular line, according to Pick, is then at the center of mass so that:

$$m_A\ x_A = m_B\ x_B$$

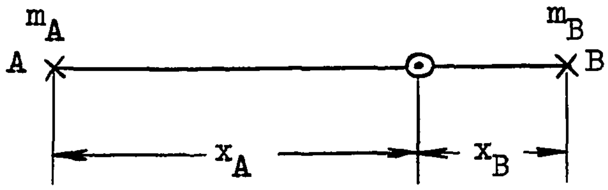


Fig. 15. The Central Facility as a Center of Mass

Even Yassen in 1956 [58] uses this principle of center of moments to optimize the facility location, but a simple numerical example shows that this reasoning is false.

If for example

$$m_A = 8$$
$$m_B = 12$$
$$AB = 10$$

the equality of moments would give

$$m_A\ x_A = m_B\ x_B = m_B\ (AB - x_A)$$
$$8\ x_A = 12(10 - x_A)$$
$$x_A = 6$$
$$x_B = 4$$

and the total weight distance is $(8\times 6) + (12\times 4) = 96$.
However, if the central facility is located at B the total of weight distance is only $8\times 10 = 80$, which is an improvement. The central location should therefore be located at the point of maximum weight in the case of 2 facilities.

In the case of a very large weight, for example if

$$m_A > m_B + m_C$$

it is then impossible to build the weight triangle and the geometric construction is impossible. In our case, point A has so much weight that it is necessary to locate the central facility at that point.

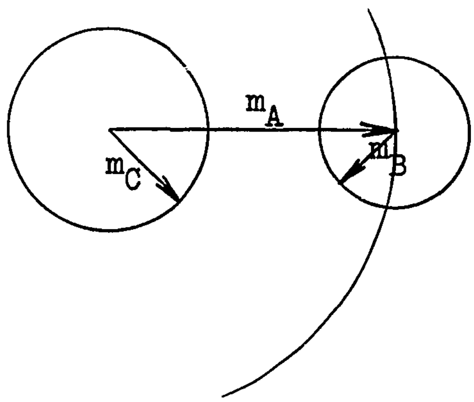


Fig. 16. Case of a very large weight in A
The polygon of forces cannot be constructed

If one angle of the triangle is greater than 120° there is no point at which each side subtends 120°, hence the minimum point P coincides with the vertex [A].

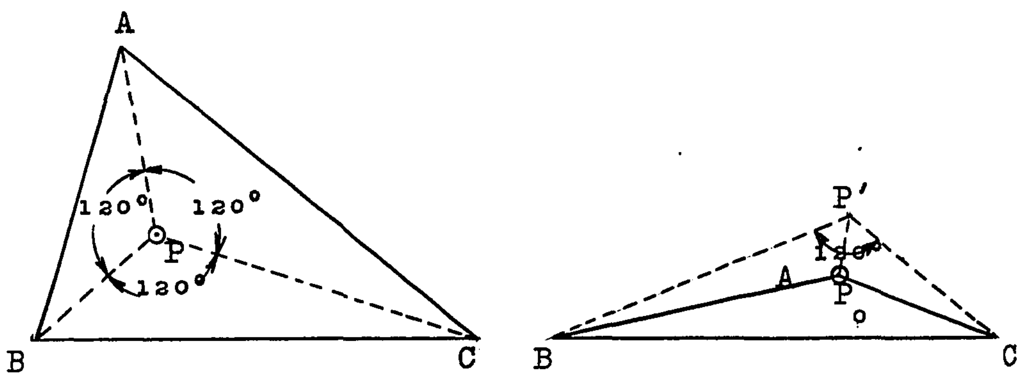


Fig. 17. Locational Triangle
One angle is greater than 120°

II.2.2. The Launhardt-Palander Construction

In 1882 Launhardt [37] developed a graphical solution to define the central facility for a set of 3 locations, which is making use also of the force polygon.

On one of the triangle sides is built a force polygon, the intersection P of the circumscribed circle with a segment joining the third triangle corner to the so-called "pole" F of the circle defines the location of the central facility.

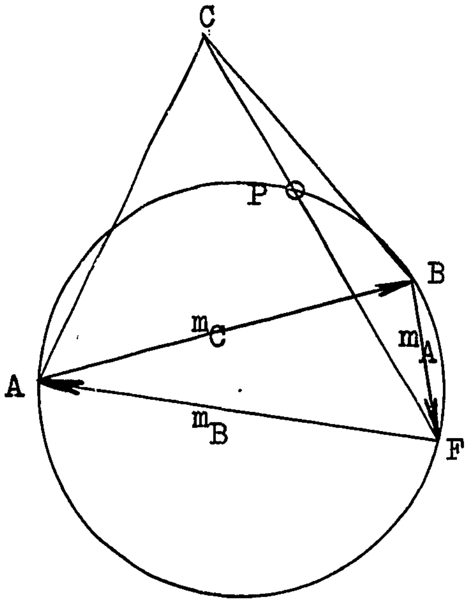


Fig. 18. The Launhardt-Palander Construction

It is to be noted that no geometrical reasoning was given for this particular construction.

Using this simple construction Tord Palander [46] developed a diagram which depicts the influence of the location of a consumer C_i using the products manufactured at a central facility P from raw materials coming from sources A and B.

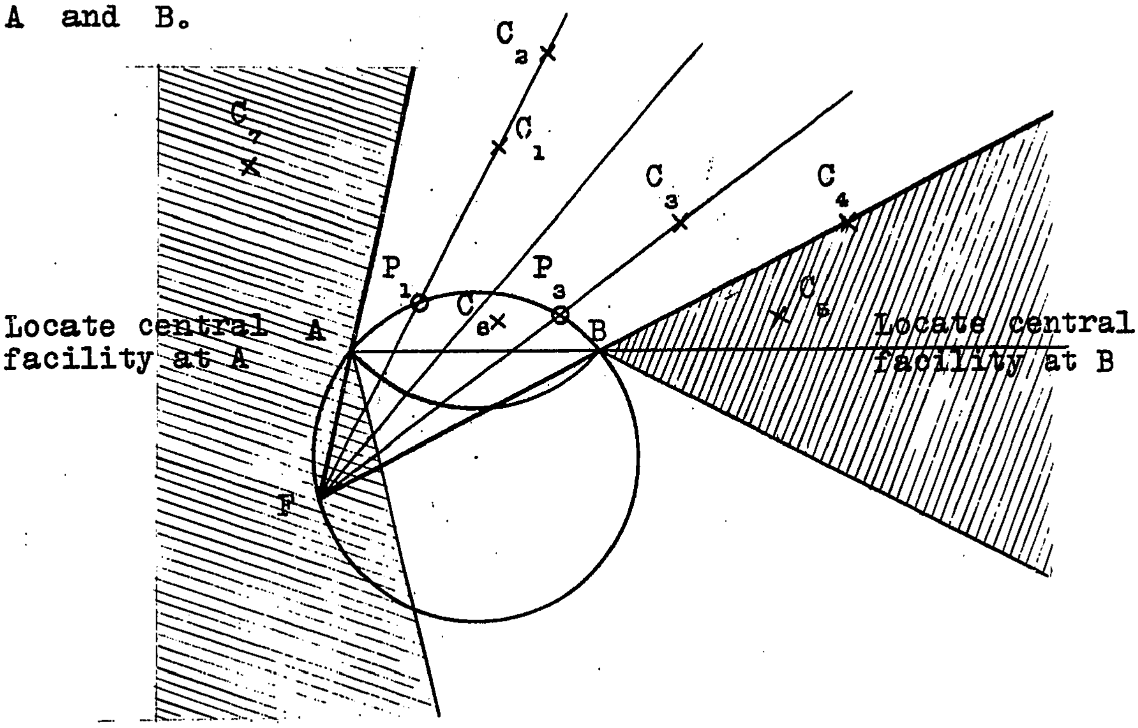


Fig. 19. Influence of Markets and Raw Material Sources on Industry Location

The production facility deserving C_1 or C_2 should be at P_1. If the consumer were at C_3, he could be best served by a production center at P_3. Similarly B is the best location to serve C_4 or C_5. If the customer were at C_6, he would be served best by a production facility at C_6 itself. On the other hand, a customer at C_7 would be served best by a facility located at A.

This construction gives a good insight into locational shifts in case of weight changes in A and B, however if we look at Figure 20, we will see that the point M_A found by using the Launhardt-Palander construction is far from being the optimum location. In fact, we may obtain 3 locations M_A, M_B, M_C, if we use the different sides of the triangle as a base for our force polygon, all of them sharing an increased sum of weighted distances compared to the optimum facility P.

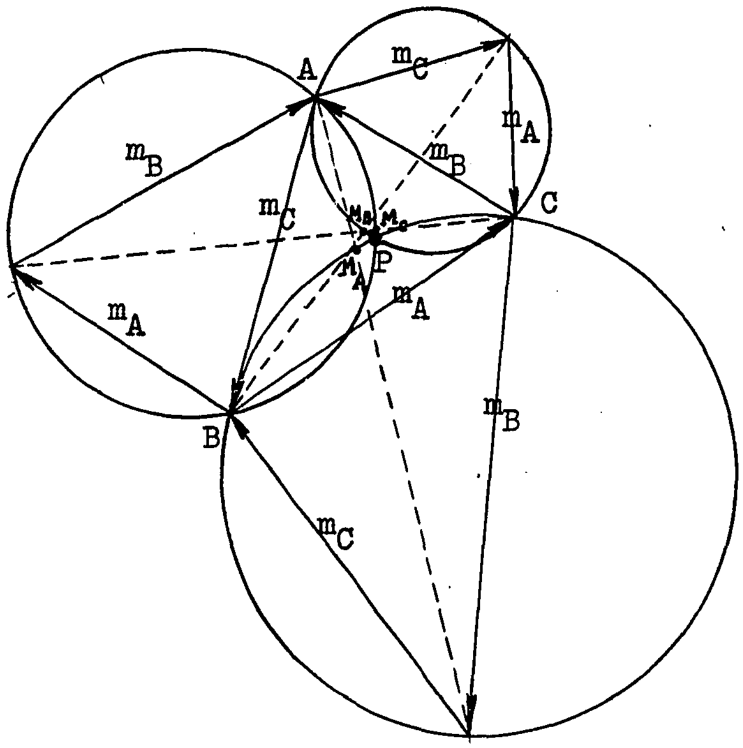


Fig. 20. Inconsistency in the Launhardt-Palander Construction

II.2.3. Isovectures and Isodapanes

All points situated on a circle centered on a facility are equidistant from that facility in Euclidean space. It is a line of equipotential for distances, but also for costs if transportation rates are identical in all azimuths, and for time if the straight distance is covered at equal speed in all directions from the facility. If a particular azimuth is advantaged because of the location of a cheap line of transport by railroad or canal for example, the line of equal cost will be a distorted circle in that particular direction. Similarly the isochrone circles may be distorted by the presence of a faster transportation system in a given direction. The set of lines representing equal distances, costs or transportation time will be representing a family of isovectures. If we consider 2 facilities A and B and their respective families of distance isovectures we obtain a set of intersecting circles. If a facility is set at a point M on one of these circle interceptions the sum of distances $PA+PB=D$ can be read directly by summing the corresponding radii. For example in Figure 21, $R_{A3} + R_{B7} = D = 10$. Another facility M′ set at the intersection of $R_{A3} + \triangle R = R_{A4}$ and $R_{B7} - \triangle R = R_{B6}$ will be connected to A and B by an identical sum of distances

$$(R_{A3} + \triangle R) + (R_{B7} - \triangle R) = R_{A3} + R_{B7} = D = 10$$
$$R_{A4} + R_{B6} = D = 10$$

The location of the point P_1, P_2, P_3, etc., are on a curve of equal sum of distances and called isodapane.

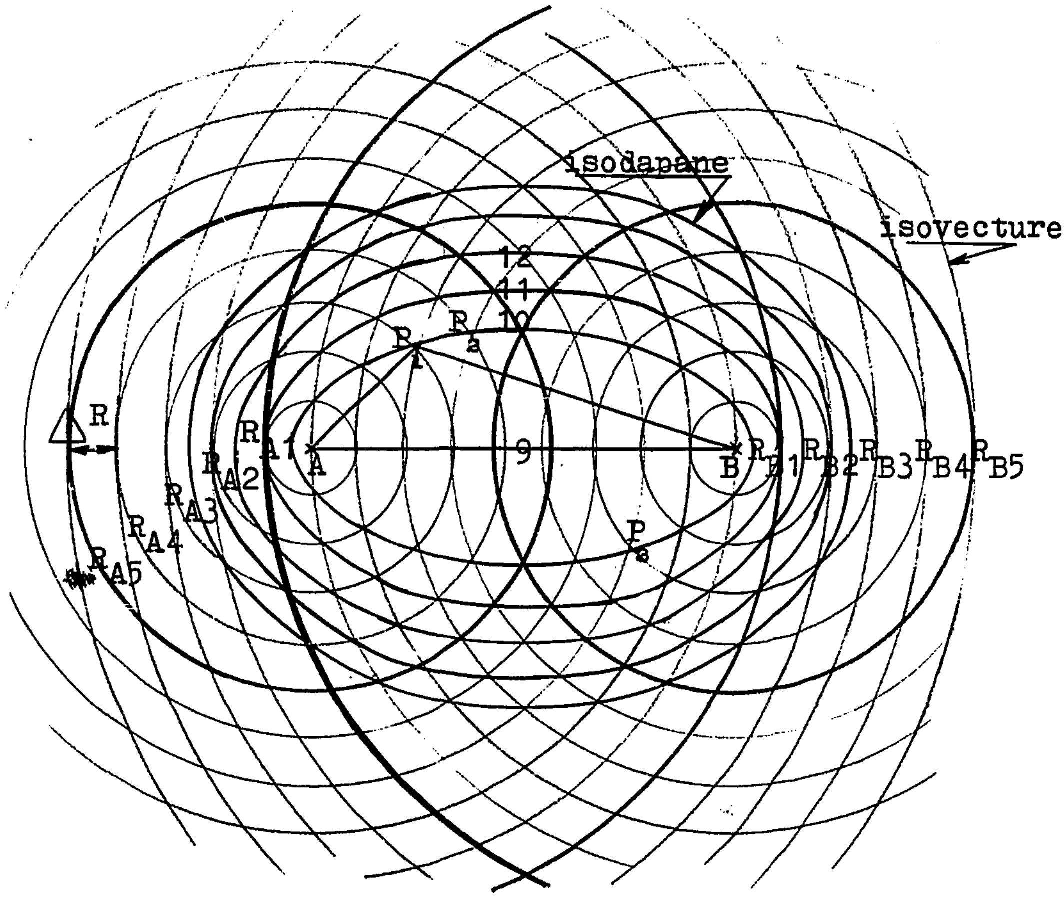


Fig. 21. Isovectures and Isodapanes of Distances

Similarly if we consider costs, these are proportional not only to distances but to the volume of freight and the transportation rate, the isovectures will be modified accordingly. Also, isovectures corresponding to time may have different spacing when corresponding to one facility or another because of limitations in channel capacity or slower transmission means. However, isodapanes of total equal costs or equal time can be readily constructed.

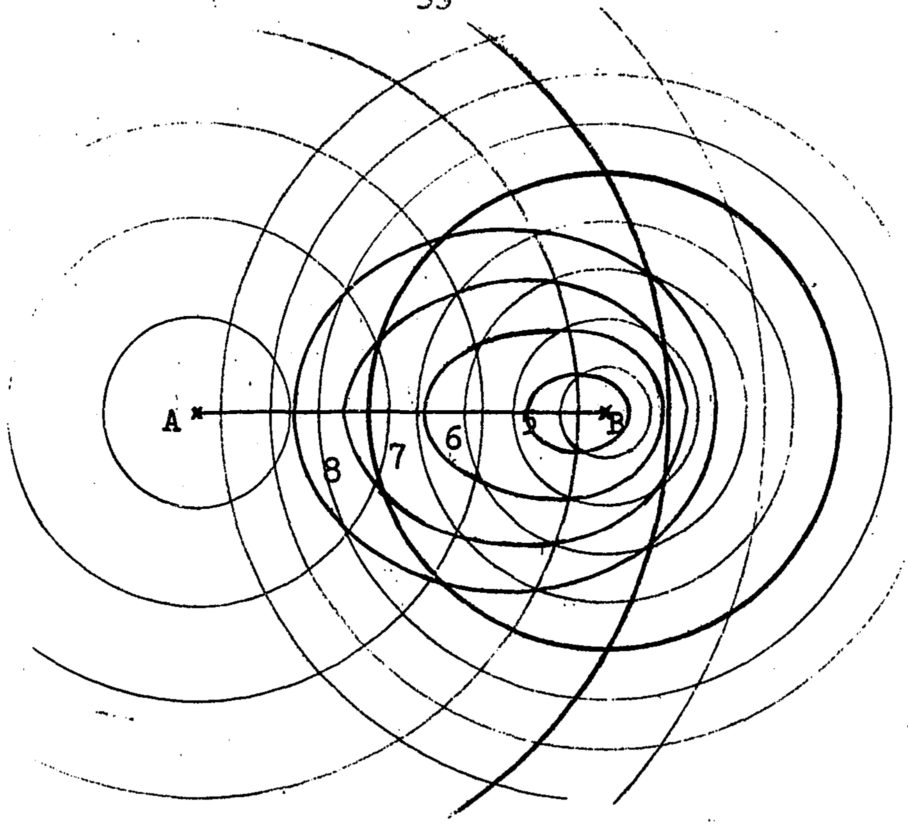


Fig. 22. Isovectures and Isodapanes of Costs or Times

When looking at the isodapanes we see that when the costs (or distances, or times) decrease, they converge toward the optimum location of the central facility. This construction could then be readily used in the case of multiple facilities to locate an optimum central location [31] [32].

A construction of the isovectures and isodapanes is given in Figure 23 in the case of 3 facilities. It may be readily seen that this construction is very long and grossly inaccurate as we get closer to the optimum central location. It is of very little use to locate the optimum point except if the plot is accelerated by means of a Computer. This construction is, however, self-explanatory with respect to

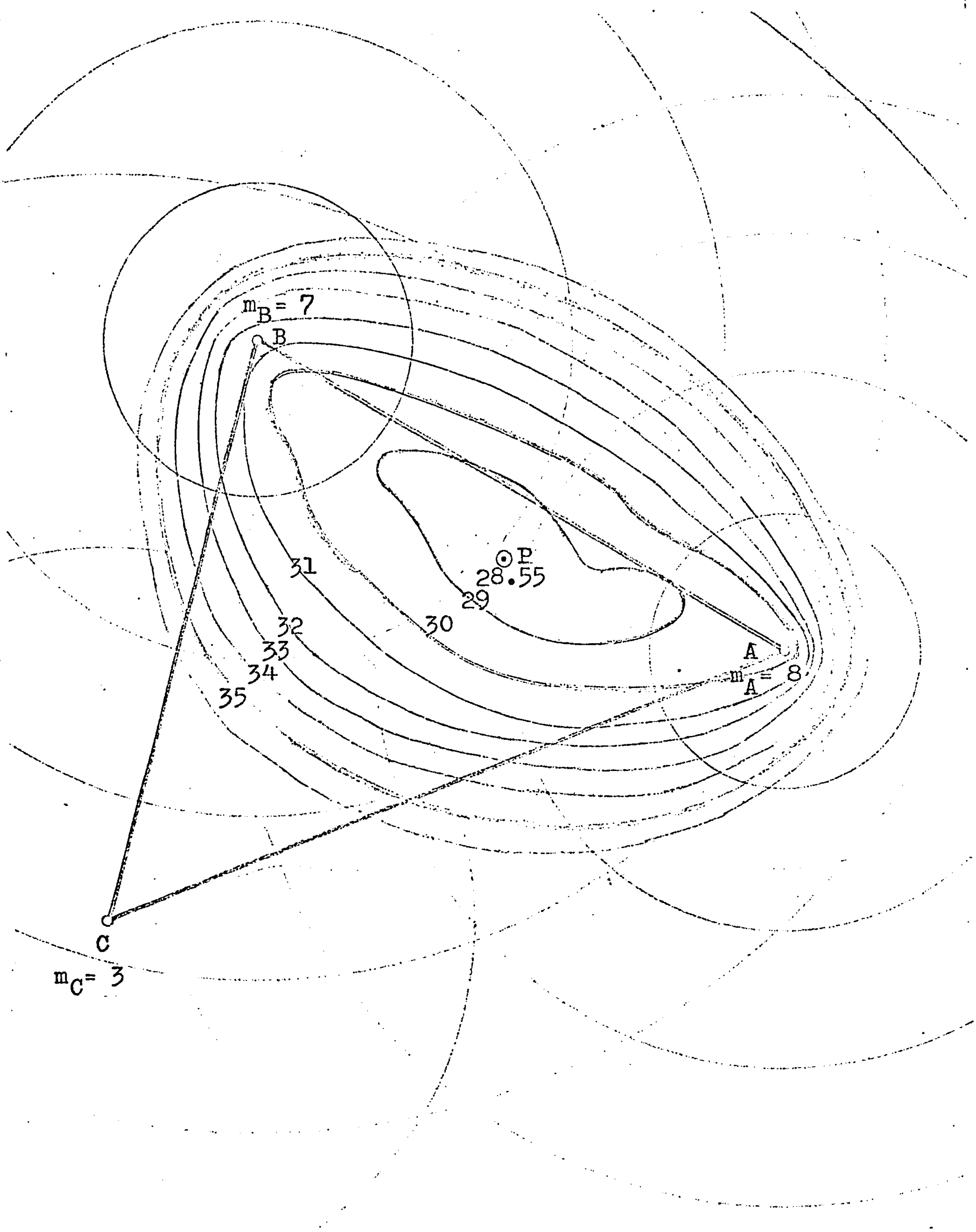


Fig. 23 Isovectures and Isodapanes of Costs

Case of 3 facilities

sensitivity analysis, if the center must be located in another area for economic reasons or other types of constraints, the construction can readily inform us of the penalty we would have to pay in this suboptimum location.

II.2.4. Topographic Mapping of Costs

The locations being defined by x and y in a system of rectangular coordinates, we may consider a third axis of costs perpendicular to the plane of facilities. For each point P taken as a candidate for the central facility we can compute the corresponding total transportation cost which then can be plotted vertically above the point P. If the computation of cost is repeated for a certain number of points, we obtain a three dimensional convex surface, whose minimum distance to the plane of facilities gives the optimum location of the central facility. If this topographical map of cost is cut by parallel planes corresponding to given differences in costs, we obtain the set of isodapanes which can be projected on the plane x,y.

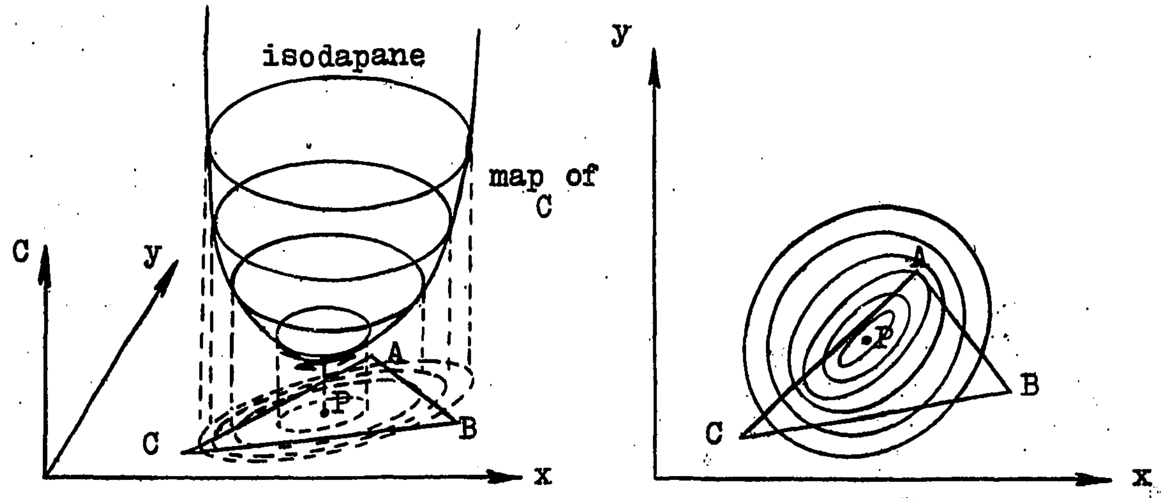


Fig. 24. Cost Function and Topographic Mapping

If the plane x,y is divided into a checker-work and if we consider the central facility to be successively located at each line intersection we could obtain a matrix of total transportation costs which could be used to visually delineate the isodapanes and therefore define the optimum central location [45].

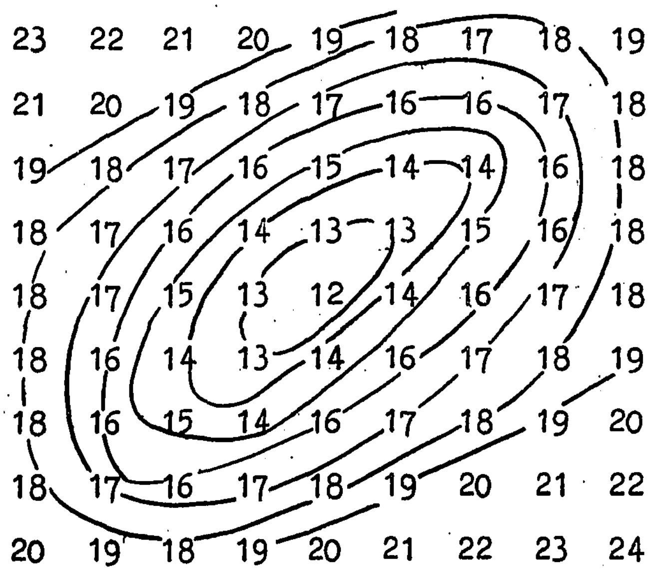


Fig. 25. Digital Mapping of Cost Function

II.2.5. Extension of the Geometrical Construction

The extension of Pick's graphical method to a set of multiple facilities is impossible because the various orientations of the force polygon segments are unknown.

Launhardt has extended his construction to a larger set of facilities however its inadequacy has been shown in Figure 20.

The limitation is created by the indetermination in the orientation of the force polygon. If a central point is chosen intuitively from which the polygon is built, it will close through an error vector. We should investigate the possibility of reducing this closure vector by a rational rule to relocate our first estimated central location. No sure convergent method has been found to date by the author.

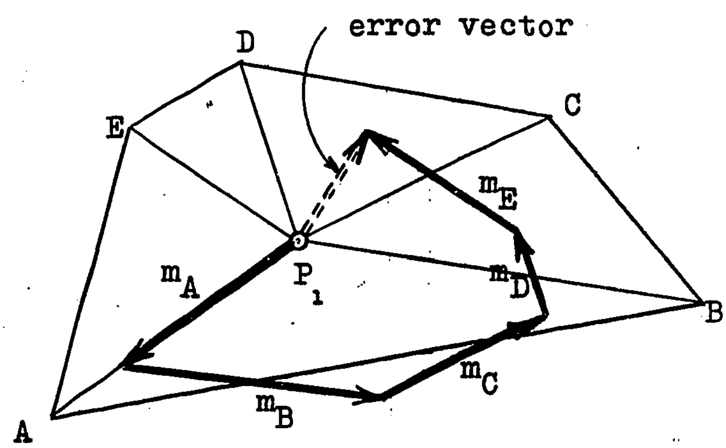


Fig. 26. Error Closure Vector

In 1810 Tedenat found a trigonometric relationship between the angles formed by an arbitrary line intersecting all segments connecting n points to a central facility. In 1837 Steiner gave the formal demonstration of this relation. However, this relation is interesting as long as the central facility is located but it is of no help in locating it.

II.3. ALGEBRAIC SOLUTION

II.3.1. Determination of Minimum Point

In the locational problem, we know the location of each facility and their corresponding requirements as well as the set of shipping rates (or speed of transport). In the most general problem we must define the number of central facilities as well as their location so as to minimize transportation costs (or communication time).

A comprehensive study in the case of a single central facility is presented by Walter Isard [33].

For example in the case of n facilities: $i = 1,2,3,\ldots,n$; the transport costs to the central facility P is given by

$$C = \sum_{i=1}^{n} r_i \, m_i \, D_i$$

where r_i : represents the transport rate on the route from P to i

m_i : represents the quantity to transport from P to i

D_i : represents the distance connecting P to i.

The distance D_i is function of the location of the central facility of coordinates (X,Y). The cost function C will also vary according to the location of the central facility.

$$C(X,Y) = \sum_{i=1}^{n} r_i \circ m_i \circ D_i (X,Y)$$

We are looking for the minimum of the function C as it varies with the locational vector (X,Y) of the central facility. A stationary point of this function C is obtained by equating to zero the first partial derivatives of C with respect to X and Y. For example a stationary point for a set of 3 facilities is given by

$$\begin{aligned} dC = 0 &= d(r_1 m_1 D_1 + r_2 m_2 D_2 + r_3 m_3 D_3) \\ &= r_1 d(m_1 D_1) + r_2 d(m_2 D_2) + r_3 d(m_3 D_3) \end{aligned}$$

since r_i are fixed, we have

$$\frac{r_1}{r_2} = - \left.\frac{d(m_2 D_2)}{d(m_1 D_1)}\right|_{(m_3 D_3 = C^{st})}$$

$$\frac{r_1}{r_3} = - \left.\frac{d(m_3 D_3)}{d(m_1 D_1)}\right|_{(m_2 D_2 = C^{st})}$$

$$\frac{r_2}{r_3} = - \left.\frac{d(m_3 D_3)}{d(m_2 D_2)}\right|_{(m_1 D_1 = C^{st})}$$

This represents a set of 3 equations with 3 unknowns D_1, D_2, D_3. For this stationary point to be a minimum of cost it is sufficient that the second derivative of C with respect to an arbitrary line passing through P of arc length u be positive to prove that the transport cost surface is convex downward

$$\frac{d^2C}{du^2} = \sum r_i m_i \frac{d^2D_i}{du^2}$$

that is $$\frac{d^2D_i}{du_2} \geq 0 \quad \text{so} \quad \frac{d^2C}{du_2} > 0$$

In the general case of n points it is enough to define D from 2 facilities to find the location of the central one.

As previously, P will give a stationary point to the function C if

$$\frac{r_i}{r_j} = -\frac{d(m_j D_j)}{d(m_i D_i)} \Bigg| \qquad \sum r_k m_k D_k = C^{st}$$

for $i \neq j \neq k$

In a system of cartesian coordinates with Euclidean distances connecting the facilities to the central location we may write the following equations

$$D_i = [(X - x_i)^2 + (Y - y_i)^2]^{\frac{1}{2}}$$

$$C(X,Y) = \sum_{i=1}^{n} r_i m_i \; D_i(X,Y) = \sum_{i=1}^{n} r_i m_i [(X-x_i)^2 + (Y-y_i)^2]^{\frac{1}{2}}$$

X, Y : represent the cartesian coordinates of the central facility

x_i, y_i : represent the cartesian coordinates of the facility i.

To prove that C has a minimum we must consider the Hessian matrix

$$H = \begin{bmatrix} \dfrac{\partial^2 C}{\partial X^2} & \dfrac{\partial^2 C}{\partial X \, \partial Y} \\ \dfrac{\partial^2 C}{\partial X \, \partial Y} & \dfrac{\partial^2 C}{\partial Y^2} \end{bmatrix}$$

and prove that C_{XX} as well as $C_{XX}C_{YY} - C^2_{XY}$ are always positive, then it will be necessary to check that C can have at most one minimum and at least one minimum.

$$C(X,Y) = \sum_{i=1}^{n} r_i m_i \left[(X-x_i)^2 + (Y-y_i)^2\right]^{\frac{1}{2}}$$

$$C_X(X,Y) = \sum_{i=1}^{n} r_i m_i \frac{1}{2} 2(X-x_i) \left[(X-x_i)^2 + (Y-y_i)^2\right]^{-\frac{1}{2}}$$

$$C_{XX}(X,Y) = \sum_{i=1}^{n} r_i m_i \left\{(1) \left[(X-x_i)^2 + (Y-y_i)^2\right]^{-\frac{1}{2}} + (X-x_i)(-\frac{1}{2})\, 2\,(X-x_i) \left[(X-x_i)^2 + (Y-y_i)^2\right]^{-\frac{3}{2}}\right\}$$

$$C_{XX}(X,Y) = \sum_{i=1}^{n} r_i m_i \left[\frac{1}{D_i} - \frac{(X-x_i)^2}{D_i^3}\right]$$

$$C_{XX}(X,Y) = \sum_{i=1}^{n} r_i m_i \left[\frac{D_i^2 - (X-x_i)^2}{D_i^3} \right]$$

$$C_{XX}(X,Y) = \sum_{i=1}^{n} r_i m_i \frac{(X-x_i)^2 + (Y-y_i)^2 - (X-x_i)^2}{D_i^3}$$

$$C_{XX}(X,Y) = \sum_{i=1}^{n} r_i m_i \frac{(Y-y_i)^2}{D_i^3}$$

$C_{XX}(X,Y)$ is therefore always positive because r_i, m_i, D_i are always positive and the numerator is a square.

Similarly

$$C_{YY}(X,Y) = \sum_{i=1}^{n} r_i m_i \frac{(X-x_i)^2}{D_i^3}$$

and

$$C_{XY}(X,Y) = \sum_{i=1}^{n} r_i m_i \frac{(X-x_i)(Y-y_i)}{D_i^3}$$

So

$$C_{XX}C_{YY} - C_{XY}^2 = \left[\sum_{i=1}^{n} \frac{r_i m_i (Y-y_i)^2}{D_i^3} \right] \left[\sum_{j=1}^{n} \frac{r_j m_j (X-x_j)^2}{D_j^3} \right] - \left[\sum_{i=1}^{n} \frac{r_i m_i (X-x_i)(Y-y_i)}{D_i^3} \right] \left[\sum_{j=1}^{n} \frac{r_j m_j (X-x_j)(Y-y_j)}{D_j^3} \right]$$

$$C_{XX}C_{YY} - C_{XY}^2 = \sum_{i,j=1}^{n} \frac{r_i r_j m_i m_j}{D_i^3 D_j^3} \Big[(Y-y_i)^2(X-x_j)^2 - (X-x_i)(X-x_j)(Y-y_i)(Y-y_j)\Big]$$

if $i = j$ the term in the bracket is zero

if $i \neq j$

$$C_{XX}C_{YY} - C_{XY}^2 = \sum_{i \neq j}^{n} \frac{r_i r_j m_i m_j}{D_i^3 D_j^3} \Big[(Y-y_i)^2(X-x_j)^2 - (X-x_i)(X-x_j)(Y-y_i)(Y-y_j) + (Y-y_j)^2(X-x_i)^2 - (X-x_j)(X-x_i)(Y-y_j)(Y-y_i)\Big]$$

$$C_{XX}C_{YY} - C_{XY}^2 = \sum_{i \neq j}^{n} \frac{r_i r_j m_i m_j}{D_i^3 D_j^3} \Big[(Y-y_i)(X-x_j) - (Y-y_j)(X-x_i)\Big]^2$$

$C_{XX}C_{YY} - C_{XY}^2$ is therefore always positive because r_i, m_i, D_i are always positive and the numerator is a square.

The stationary point obtained by differentiation is therefore a minimum point.

Palermo [47] has proven that C can have at most one minimum on the plane and that the function C has at least a minimum.

This minimum occurs at

$$C_X(X,Y) = 0$$

$$C_Y(X,Y) = 0$$

That is

$$\sum_{i=1}^{n} \frac{r_i m_i (X-x_i)}{\left[(X-x_i)^2 + (Y-y_i)^2\right]^{\frac{1}{2}}} = 0$$

$$\sum_{i=1}^{n} \frac{r_i m_i (Y-y_i)}{\left[(X-x_i)^2 + (Y-y_i)^2\right]^{\frac{1}{2}}} = 0$$

This is a set of two non-linear equations which cannot be explicitly solved. An iterative method is necessary such as the Newton-Raphson procedure.

If the 2 equations are independent, the coordinates of the central facility are given by

$$X = \frac{\displaystyle\sum_{i=1}^{n} \frac{r_i m_i x_i}{\left[(X-x_i)^2 + (Y-y_i)^2\right]^{\frac{1}{2}}}}{\displaystyle\sum_{i=1}^{n} \frac{r_i m_i}{\left[(X-x_i)^2 + (Y-y_i)^2\right]^{\frac{1}{2}}}} = \frac{\displaystyle\sum_{i=1}^{n} \frac{r_i m_i x_i}{D_i}}{\displaystyle\sum_{i=1}^{n} \frac{r_i m_i}{D_i}}$$

$$Y = \frac{\sum_{i=1}^{n} \frac{r_i m_i y_i}{\left[(X-x_i)^2 + (Y-y_i)^2\right]^{\frac{1}{2}}}}{\sum_{i=1}^{n} \frac{r_i m_i}{\left[(X-x_i)^2 + (Y-y_i)^2\right]^{\frac{1}{2}}}} = \frac{\sum_{i=1}^{n} \frac{r_i m_i y_i}{D_i}}{\sum_{i=1}^{n} \frac{r_i m_i}{D_i}}$$

As D_i is a function of X and Y these equations cannot be solved directly, they must be solved by iteration. A set of starting values $X^{(0)}$ and $Y^{(0)}$ must be assumed a priori and they are used to compute $X^{(1)}$ $Y^{(1)}$ using the equations above, and so on $X^{(1)}$ $Y^{(1)}$ is used to compute $X^{(2)}$ $Y^{(2)}$ etc. The process hopefully converges if the starting values are chosen adequately.

We are now going to consider a method to derive a plausible iterative starting value $X^{(0)}$ and $Y^{(0)}$.

It happens quite frequently that transportation costs may be proportional to distances raised to some power k

$$C = \sum_{i=1}^{n} w_i \, D_i^k \qquad k > 1$$

w_i being the weighted index bearing on the location of the center. It can be proven [9] that for $k \geq 1$ the function C is convex. Being a convex function, every local minimum is a global minimum.

$$C(X,Y) = \sum_{i=1}^{n} w_i \left[(X-x_i)^2 + (Y-y_i)^2\right]^{\frac{k}{2}}$$

$$C_X(X,Y) = \sum_{i=1}^{n} k\, w_i\, (X-x_i) \left[(X-x_i)^2 + (Y-y_i)^2\right]^{\frac{k}{2}-1} = 0$$

$$C_Y(X,Y) = \sum_{i=1}^{n} k\, w_i\, (Y-y_i) \left[(X-x_i)^2 + (Y-y_i)^2\right]^{\frac{k}{2}-1} = 0$$

Only an implicit solution for X and Y coordinates of the central facility is obtained and an iteration procedure must also be used to solve the non-linear expressions:

if $\quad L_i = \left[(X-x_i)^2 + (Y-y_i)^2\right]^{\frac{k}{2}-1}$

$$C_X(X,Y) = k\,X \sum_{i=1}^{n} w_i\, L_i - k \sum_{i=1}^{n} w_i\, x_i\, L_i = 0$$

$$C_Y(X,Y) = k\,Y \sum_{i=1}^{n} w_i\, L_i - k \sum_{i=1}^{n} w_i\, y_i\, L_i = 0$$

and

$$X = \frac{\sum_{i=1}^{n} w_i\, x_i\, L_i}{\sum_{i=1}^{n} w_i\, L_i}$$

$$Y = \frac{\sum_{i=1}^{n} w_i\ y_i\ L_i}{\sum_{i=1}^{n} w_i\ L_i}$$

As previously, if we had a set $X^{(o)}$, $Y^{(o)}$ of good starting values giving us convergence, we could solve by iteration

$$X^{(k+1)} = \frac{\sum_{i=1}^{n} w_i\ x_i\ L_i^{(k)}}{\sum_{i=1}^{n} w_i\ L_i^{(k)}}$$

$$Y^{(k+1)} = \frac{\sum_{i=1}^{n} w_i\ y_i\ L_i^{(k)}}{\sum_{i=1}^{n} w_i\ L_i^{(k)}}$$

the superscript (k) meaning the k^{th} iteration.

In this general case also we are faced with the problem of finding the starting values.

Let us consider the particular case where $k = 2$, then

$$C = \sum_{i=1}^{n} w_i \left[(X-x_i)^2 + (Y-y_i)^2\right]$$

$$C_X = \sum_{i=1}^{n} 2w_i \ (X-x_i) = 0$$

$$C_Y = \sum_{i=1}^{n} 2w_i \ (Y-y_i) = 0$$

This set of 2 linear equations can be explicitly solved:

$$X = \frac{\sum_{i=1}^{n} w_i \ x_i}{\sum_{i=1}^{n} w_i}$$

$$Y = \frac{\sum_{i=1}^{n} w_i \ y_i}{\sum_{i=1}^{n} w_i}$$

It has been proven by McHose [42] that this solution of the second degree equation is a good first approximation for the location of the central facility when $k \geq 1$. Therefore we can take these values as starting solution of our iterative process and experience proves that the procedure rapidly converges.

In our original problem of Euclidean distance our iterative equation will then be:

$$X^{(k+1)} = \frac{\displaystyle\sum_{i=1}^{n} \frac{r_i m_i x_i}{\left[(X^{(k)}-x_i)^2 + (Y^{(k)}-y_i)^2\right]^{\frac{1}{2}}}}{\displaystyle\sum_{i=1}^{n} \frac{r_i m_i}{\left[(X^{(k)}-x_i)^2 + (Y^{(k)}-y_i)^2\right]^{\frac{1}{2}}}}$$

$$Y^{(k+1)} = \frac{\displaystyle\sum_{i=1}^{n} \frac{r_i m_i y_i}{\left[(X^{(k)}-x_i)^2 + (Y^{(k)}-y_i)^2\right]^{\frac{1}{2}}}}{\displaystyle\sum_{i=1}^{n} \frac{r_i m_i}{\left[(X^{(k)}-x_i)^2 + (Y^{(k)}-y_i)^2\right]^{\frac{1}{2}}}}$$

with

$$X^{(o)} = \frac{\displaystyle\sum_{i=1}^{n} r_i m_i x_i}{\displaystyle\sum_{i=1}^{n} r_i m_i}$$

$$Y^{(o)} = \frac{\displaystyle\sum_{i=1}^{n} r_i m_i y_i}{\displaystyle\sum_{i=1}^{n} r_i m_i}$$

This starting value is the weighted mean coordinate. An example of computer program to solve that algorithm is given in Chapter IV.

It must be kept in mind that before beginning the algorithm solution each facility should be checked for weight dominance, that is the central location may be located at point $k = 1, 2, \ldots, n$ only if

$$w_k \geq \left[\left(\sum_{j \neq k} w_j \cos \theta_j \right)^2 + \left(\sum_{j \neq k} w_j \sin \theta_j \right)^2 \right]^{\frac{1}{2}}$$

with
$$\cos \theta_j = \frac{x_j - x_k}{D_{kj}}$$

$$\sin \theta_j = \frac{y_j - y_k}{D_{kj}}$$

This check may become quite cumbersome computationally when dealing with a very large number n of facilities. We should ignore this test unless we become suspicious of a weight dominance creating a convergence of the algorithm to a close proximity of a facility k, then the check is only necessary on that particular facility k.

II.3.2. Geometric Programming

Minimization of the function

$$D(X,Y) = \sum_{i=1}^{n} \left[(x_i-X)^2 + (y_i-Y)^2\right]^{\frac{1}{2}}$$

may be done by minimizing the related function

$$G = \sum_{i=1}^{n} t_{oi}^{\frac{1}{2}}$$

subject to the constraints

$$t_{oi} \geq (x_i-X)^2 + (y_i-Y)^2 \qquad i=1,\ 2,\ \cdots,\ n$$

where t_{oi} are additional independent variables.

There are $6n$ terms in the constraint inequalities put in canonical form :

$X^2 t_{oi}^{-1}$

$$1 \geq x_i^2 t_{oi}^{-1} - 2x_i X t_{oi}^{-1} + y_i^2 t_{oi}^{-1} - 2y_i Y t_{oi}^{-1} + Y^2 t_{oi}^{-1} \qquad i=1,\ 2,\ \cdots,\ n$$

and n terms in the function G. There are n variables t_{oi} and 2 variables: X,Y. Even if the function G were a posynomial, which it is not, we would expect $6n-3$ degrees of freedom.

When considering a relatively small system of 50 facilities, the problem involves a minimum of 297 degrees of difficulty which is equivalent to the optimization of a function of 297 variables, a problem which is theoretically possible but economically infeasible.

II.3.3. Expenditures at Vertice Points

In the above algorithm we have assumed that money or time costs were proportional to distances. This is not always the case and some fixed charges of loading and unloading must be added. Under such a case the function C becomes discontinuous.

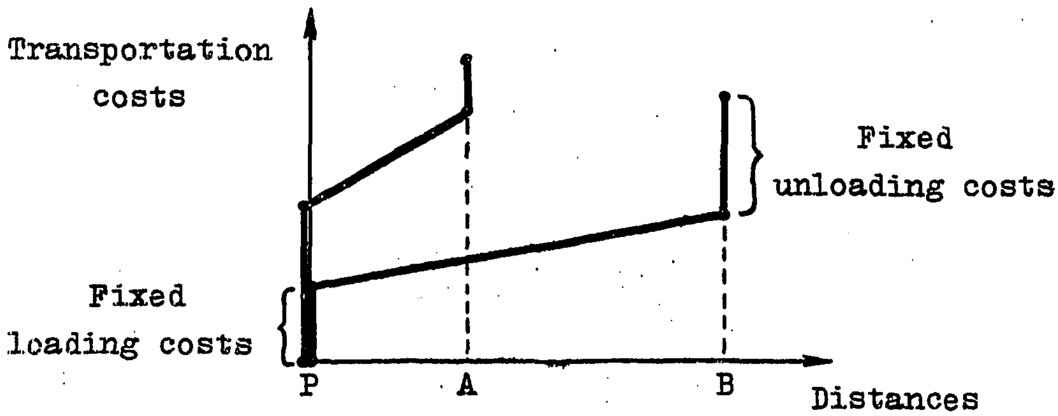


Fig. 27. : Expenditures at Vertice Points .

if l_i : is the total loading cost for goods transferred from facility P to facility i

u_i : is the total unloading cost for goods transferred from facility P to facility i

then the function C is given by

$$C = \sum_{i=1}^{n} (l_i + r_i m_i D_i + u_i)$$

It may be assumed that the loading and unloading costs are directly proportional to the amount transited : m_i

$$l_i = a_i m_i$$

$$u_i = b_i m_i$$

a_i and b_i being factors of proportionality then the function C is given by

$$C(X,Y) = \sum_{i=1}^{n} \left(a_i m_i + r_i m_i D_i + b_i m_i \right)$$

$$C(X,Y) = \sum_{i=1}^{n} a_i m_i + \sum_{i=1}^{n} r_i m_i D_i (X,Y) + \sum_{i=1}^{n} b_i m_i$$

$$C(X,Y) = C_I + C_{II}(X,Y) + C_{III}$$

Mathematically speaking, as the cost C_I and C_{III} are not functions of the location of the central facility, a differentation of the function C will give an identical set of equations as derived above and their solution will give the same central location coordinates. However, some fallacy appears in this reasoning; for example: if facility j has an extremely large cost of loading and unloading its own goods, shown by the importance of the coefficients a_j and b_j, this cost may outweigh all transportation costs and therefore it might be more economical to locate the central facility at point j in order to eliminate the transfer cost of its own goods, and also drastically lower the loading and unloading unit cost by the corresponding modernization and mechanization imparted to this center.

A procedure to avoid this pitfall will be to derive the coordinates X and Y obtained by the solution of the set of equations of the partial derivatives of C_{II}, compute the corresponding cost C then check the various transportation costs C_k using one of the locations as central facility.

$$C_k = \sum_{i=1}^{n} m_i \; r_i \; D_i + \sum_{i \neq k}^{n} (a_i + b_i) \, m_i \qquad k = 1, 2, \dots, n$$

if one of the C_k is less than C then the central facility must be located at that particular k place.

II.3.4. Case of Variable Transport Rate

It is common in practice to deal with transportation media where the rate of transport decreases with distance. Accordingly this complicates the cost function. Timewise also it is quite frequent to use for example, slower aircrafts on shorter routes than on longer ones.

When the transportation rate is function of distance, then

$$r_i = f_i(D_i)$$

$$C = \sum_{i=1}^{n} f_i \, (D_i) \cdot m_i \cdot D_i$$

Stationary points may be derived by differentiation:

$$dC = \sum_{i=1}^{n} m_i \left[f_i (D_i) + D_i . f_i' (D_i) \right] dD_i$$

and

$$\left. \frac{f_i (D_i) + D_i \cdot f_i' (D_i)}{f_j (D_j) + D_j . f_j' (D_j)} = - \frac{d (m_j D_j)}{d (m_i D_i)} \right| \sum_k f_k (D_k). m_k . D_k = C^{st}$$

$$i \neq j \neq k$$

The development of these equations may lead to multiple optima. The function C should be evaluated at all of these optima to find the best one.

If the route passes through a congested area the expenditure in time and money per unit carried versus distance may not follow a linear function. Extra costs due to obstacles like towns and rivers in the case of ground transportations for example, must be taken into consideration [8].

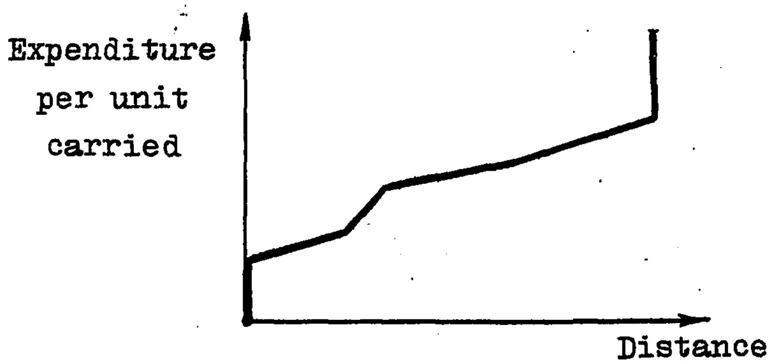


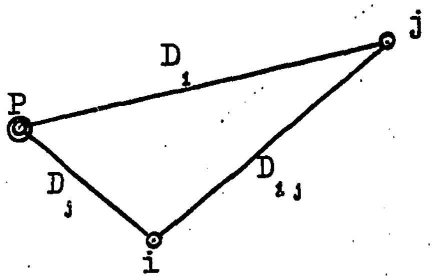
Fig. 28. Variable Transport Rate in Congested Area

II.3.5. Bounds on Sum of Distances

When considering a central location P minimizing the sum of distances, we consider a network with branches of length D_i, from P to the facility i. The total length of the network is

$$D = \sum_{i=1}^{n} D_i$$

Considering a pair of vertices : i, j



$$D_i + D_j > D_{ij}$$

$$D_i + D_j > D_{ij}$$

$$D_i + D_j = D_{ij}$$

For each pair of vertices the following inequality holds

$$D_i + D_j \geq D_{ij}$$

In a locational problem of n vertices, there are

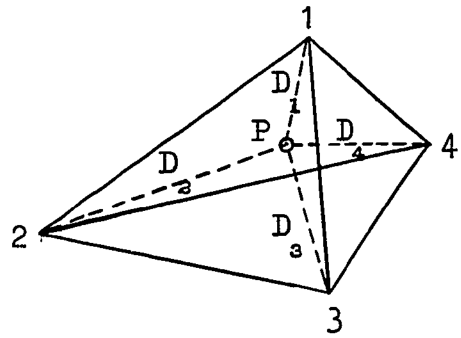
$$\sum_{i=1}^{n-1} n - i = \frac{n\,(n-1)}{2}$$

distances D_{ij}.

The distance D_i connecting P to the facility i is a part of (n-1) triangles on which can be applied the triangle inequality.

In the case of 4 facilities, for example, we can write the following set of inequalities

$$
\begin{aligned}
D_1 + D_2 &\geq D_{12}\\
D_1 + D_3 &\geq D_{13}\\
D_1 + D_4 &\geq D_{14}\\
D_2 + D_3 &\geq D_{23}\\
D_2 + D_4 &\geq D_{24}\\
D_3 + D_4 &\geq D_{34}
\end{aligned}
$$



$$3D_1 + 3D_2 + 3D_3 + 3D_4 \geq D_{12} + D_{13} + D_{14} + D_{23} + D_{24} + D_{34}$$

On the right-hand side are the possible combinations of distances connecting n facilities taken 2 at a time.

The left-hand side

$$3(D_1 + D_2 + D_3 + D_4) = 3 \sum_{i=1}^{4} D_i = 3D$$

corresponds to (n-1)D.

Therefore

$$D \geq \frac{1}{n-1} \sum_{i=2}^{n} \sum_{j=1}^{i-1} D_{ij}$$

The right-hand side is then a lower bound of the sum of distances [10].

In our iterative process we took the weighted average as a starting value, the

$$D \leq \sum_{i=1}^{n} \left[(X-\bar{x})^2 + (Y-\bar{y})^2\right]^{\frac{1}{2}}$$

in which

$$\bar{x} = \frac{1}{n} \sum_{i=1}^{n} x_i$$

$$\bar{y} = \frac{1}{n} \sum_{i=1}^{n} y_i$$

This is therefore a possible upper bound for the sum of distances.

II.3.6. The Constrained Problem

The n distinct locations form a convex hull H limited by linear boundaries. Either P is located at a facility, in case of weight dominance, and is trivially an element of H or it is first obtained by a weighted sum of facility coordinates and therefore is an interior point. We will use this convex hull in the heuristic algorithm with variable grid and linear constraints (Chapter IV) to limit our investigation.

Sometimes the location of a facility is restrained within a given space limited by physical boundaries. Sometimes the facilities generally considered as punctual may have a spatial area or a zone of influence and cannot be located too close because of possible interference.

We could be faced by the following constraints

$$\left[(X-x_i)^2 + (Y-y_i)^2\right]^{\frac{1}{2}} \geq \Phi_i$$

$$x_{min} \leq X \leq x_{max}$$

$$y_{min} \leq Y \leq y_{max}$$

This is an example of optimization theory in which the objective function and some of the constraints are non-linear. The following possible solutions should be considered: the method of Lagrangian multipliers in which the inequalities would be investigated in turn in their equality sense, or using the Khun-Tucker conditions.

It must be noted that according to the demonstration of Kuhn and Kuenne [36], the coordinates X and Y of P are necessarily elements of the convex hull and the set of 2 inequalities $x_{min} \leq X \leq x_{max}$

$$y_{min} \leq Y \leq y_{max}$$

are automatically met and are therefore redundant constraints.

The minimization problem is then limited to

minimize $\quad C(X,Y) = \sum_{i=1}^{n} r_i \; m_i \left[(X-x_i)^2 + (Y-y_i)^2\right]^{\frac{1}{2}}$

subject to $\quad \left[(X-x_i)^2 + (Y-y_i)^2\right]^{\frac{1}{2}} \geq \Phi_i \qquad i = 1, 2, \cdots, n$

then we have a set of n inequalities.

In the Lagrangian method the inequality constraints are considered by including one active constraint at a time. Thus for

$$\left[(X-x_1)^2 + (Y-y_1)^2\right]^{\frac{1}{2}} \geq \Phi_1$$

to be active the corresponding Lagrangian function becomes

$$L(X,Y,\lambda_1) = C(X,Y) + \lambda_1 \left(\left[(X-x_1)^2 + (Y-y_1)^2\right]^{\frac{1}{2}} - \Phi_1 \right)$$

the necessary conditions are given by

$$\frac{\partial L}{\partial X} = 0 \qquad \frac{\partial L}{\partial Y} = 0 \qquad \frac{\partial L}{\partial \lambda_1} = 0$$

$$\frac{\partial L}{\partial X} = \sum_{i=1}^{n} \frac{r_i\, m_i\, (X-x_i)}{\left[(X-x_i)^2 + (Y-y_i)^2\right]^{\frac{1}{2}}} + \frac{\lambda_1\, (X-x_1)}{\left[(X-x_1)^2 + (Y-y_1)^2\right]^{\frac{1}{2}}} = 0$$

$$\frac{\partial L}{\partial Y} = \sum_{i=1}^{n} \frac{r_i\, m_i (Y-y_i)}{\left[(X-x_i)^2 + (Y-y_i)^2\right]^{\frac{1}{2}}} + \frac{\lambda_1\, (Y-y_1)}{\left[(X-x_1)^2 + (Y-y_1)^2\right]^{\frac{1}{2}}} = 0$$

$$\frac{\partial L}{\partial \lambda_1} = \left[(X-x_1)^2 + (Y-y_1)^2\right]^{\frac{1}{2}} - \Phi_1 = 0$$

The solution of these 3 equations should give X, Y and λ_1. Then it should be necessary to check if the solution violates the other inequality constraints. A similar computation should be undertaken for the n inequalities. If the three above equations were linear, still the computational effort would be large and the procedure computationally unattractive. But, moreover, the implicit nature of these equations cannot give a direct solution, except through an iterative process. In this rigorous form the method should be abandoned. However, we have seen that in the unconstrained problem a good approximation of X and Y is given by considering

$$C(X,Y) = \sum_{i=1}^{n} r_i \; m_i \; D_i^2 \; (X,Y)$$

In this particular case the Lagrangian equations become

$$L(X,Y,\lambda_1) = \sum_{i=1}^{n} r_i \; m_i \left[(X-x_i)^2 + (Y-y_i)^2\right] + \lambda_1 \left[(X-x_1)^2 + (Y-y_1)^2 - \Phi_1^2\right]$$

$$\frac{\partial L}{\partial X} = \sum_{i=1}^{n} 2r_i \; m_i \; (X-x_i) + 2\lambda_1 \; (X-x_1) = 0$$

$$\frac{\partial L}{\partial Y} = \sum_{i=1}^{n} 2r_i \; m_i \; (Y-y_i) + 2\lambda_1 \; (Y-y_1) = 0$$

$$\frac{\partial L}{\partial \lambda_1} = (X-x_1)^2 + (Y-y_1)^2 - \Phi_1^2 = 0$$

An explicit solution is possible in that case, but the final solution is not exact and must be improved by iterative process to correspond to the effective Euclidean distances. If the initial approximation is known to be $X^{(o)}$ and $Y^{(o)}$ then a better solution is determined by the equations

$$X^{(n+1)} = X^{(n)} - u \left| \frac{\partial C(X,Y)}{\partial X} \right|^{(n)}$$

$$Y^{(n+1)} = Y^{(n)} - u \left| \frac{\partial C(X,Y)}{\partial Y} \right|^{(n)}$$

the u value : distance to move in the good direction to improve the value of the variable, if taken too small may produce slow convergence or if taken too large may miss the optimum solution altogether. Moreover, u may be positive or negative and a trial and error procedure will be necessary.

A procedure used by Stewart [45] considers all the inequalities in their equality sense and he uses a plotting technique to check that the results given by iteration do not violate the constraints.

We could also apply the Kuhn-Tucker conditions for P to be a stationary point of the minimization problem

$$C(\underline{X}) = \sum_{i=1}^{n} r_i \; m_i \; \left[(X_1 - x_i)^2 + (X_2 - y_i)^2\right]^{\frac{1}{2}}$$

subject to

$$\underline{g}(\underline{X}) = - \left[(X_1 - x_i)^2 + (X_2 - y_i)^2\right]^{\frac{1}{2}} + \Phi_i \le 0$$

These conditions are summarized below

1) $\underline{\lambda} \le \underline{0}$
2) $\underline{\nabla} C(\underline{X}) - \underline{\lambda}\,\underline{\nabla}\,\underline{g}(\underline{X}) = \underline{0}$
3) $\underline{\lambda}\,\underline{g}(\underline{X}) = \underline{0}$
4) $\underline{g}(\underline{X}) \le \underline{0}$

Developing these conditions will lead to

1) $(\lambda_1, \lambda_2, \lambda_3, \cdots, \lambda_n) \le \underline{0}$

2) $$\left(\frac{\partial C}{\partial X_1}, \frac{\partial C}{\partial X_2}\right) - (\lambda_1, \lambda_2, \cdots, \lambda_n) \begin{bmatrix} \dfrac{\partial g_1(\underline{X})}{\partial X_1} & \dfrac{\partial g_1(\underline{X})}{\partial X_2} \\ \dfrac{\partial g_2(\underline{X})}{\partial X_1} & \dfrac{\partial g_2(\underline{X})}{\partial X_2} \\ \vdots & \vdots \\ \dfrac{\partial g_n(\underline{X})}{\partial X_1} & \dfrac{\partial g_n(\underline{X})}{\partial X_2} \end{bmatrix} = \underline{0}$$

3) $(\lambda_1, \lambda_2, \ldots, \lambda_n) \begin{bmatrix} g_1(\underline{X}) \\ g_2(\underline{X}) \\ \vdots \\ g_n(\underline{X}) \end{bmatrix} = \underline{0}$

4) $\underline{g}(\underline{X}) \leq \underline{0}$

When developed we obtain

1) $(\lambda_1, \lambda_2, \lambda_3, \ldots, \lambda_n) \leq \underline{0}$

2) $$\sum_{i=1}^{n} \frac{r_i\ m_i\ (X_1 - x_i)}{\left[(X_1 - x_i)^2 + (X_2 - y_i)^2\right]^{\frac{1}{2}}} + \sum_{i=1}^{n} \lambda_i \frac{X_1 - x_i}{\left[(X_1 - x_i)^2 + (X_2 - y_i)^2\right]^{\frac{1}{2}}} = 0$$

$$\sum_{i=1}^{n} \frac{r_i\ m_i\ (X_2 - y_i)}{\left[(X_1 - x_i)^2 + (X_2 - y_i)^2\right]^{\frac{1}{2}}} + \sum_{i=1}^{n} \lambda_i \frac{X_2 - x_i}{\left[(X_1 - x_i)^2 + (X_2 - y_i)^2\right]^{\frac{1}{2}}} = 0$$

$$3) \quad \lambda_i \left(\Phi_i - \left[(X_1 - x_i)^2 + (X_2 - y_i)^2 \right]^{\frac{1}{2}} \right) = 0$$

for $i = 1, 2, 3, \cdots, n$

$$4) \quad - \left[(X_1 - x_i)^2 + (X_2 - y_i)^2 \right]^{\frac{1}{2}} \leq - \Phi_i$$

for $i = 1, 2, 3, \cdots, n$

which we must solve for X_1, X_2 and λ_i $(i = 1, 2, \cdots, n)$. This method is also not very attractive computationally and the quadratic formulation of C should be used to obtain approximative but useful results.

CHAPTER III

LOCATION OF CENTRAL FACILITIES

DISCRETE, TWO-DIMENSIONAL SPACE, EUCLIDEAN DISTANCES

MULTIPLE CENTRAL LOCATIONS

The location of the n facilities to service, the requirement at each destination as well as the rate of shipping in a particular region are known. The problem is to determine the number m and location of central facilities supplying the service and their corresponding set of satellites.

It is assumed that the number m of central locations is less than the number n of facility locations. If not, it would be possible to have a zero total transport cost by putting a center at each facility.

III.1. ANALOG SOLUTION

III.1.1. The Soap Film Method

Plane films of soap bubbles formed between 2 close planes and a set of posts connecting them gives surfaces of minimum potential energy. The lines connecting the posts are then of minimum total length through a network of 120° angle boundaries.

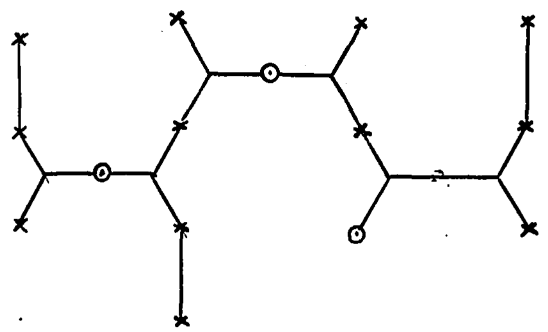


Fig. 29. Shortest Network Joining More Than 3 Points

The necessary junction points cannot be specified but are automatically created. The method gives unreliable results when the number of points is above 15 to 20 because of variable drainage as the model is pulled out of the soap-forming solution. Moreover, the solution of the problem is not uniquely defined.

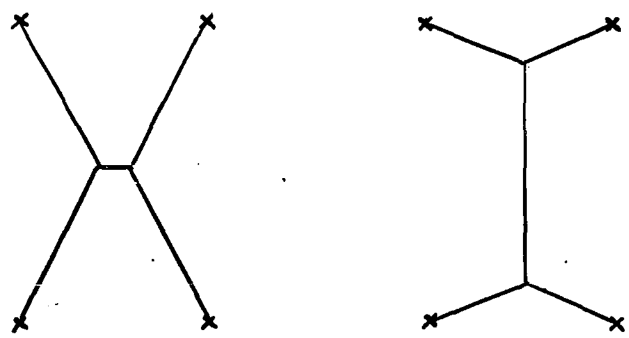


Fig. 30. Network Not Uniquely Defined

III.1.2. The Link-Length Minimizer

The mechanical system developed by Miehle [43] can be applied to multiple central facilities but the number of these central facilities and their dependent satellites must be known. This choice has to be made subjectively by looking at the concentration of points in a given area and it is never certain that the choice will bring the minimum circuit length. However, during the minimization process if one central facility is brought closer to a point than the central facility on which it depends, then a change in connections may improve the minimization.

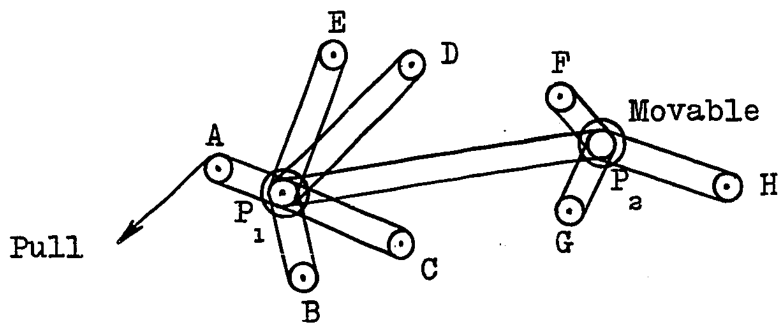


Fig. 31. Link-Length Minimizer
Multiple Central Facilities

Constraints of distance between central facilities may be obtained by connecting them with a rigid spacing bar. Constraints of minimum distance between central location and satellite facility may be obtained by installing a corresponding round base of given radius at the base of the facility constrained.

Transaction weights on a given link may be added to the model by multiple increments created by multiple looping.

It may take about an hour to find the optimum location of 17 central facilities in a system of 62 fixed points.

III.2. ALGEBRAIC SOLUTION

Even if we assume no restriction on the capacity of the central facility and if the shipping costs are supposed to be independent of the total central facility supply, we still are faced with a very large problem. If we arbitrarily decide on the number m of central facilities, there are $S(n,m)$ possible assignments of n destinations to m sources [11], where S is the Stirling number of the second kind:

$$S(n,m) = \frac{1}{m!} \sum_{k=0}^{m} \binom{m}{k} (-1)^k (m-k)^n$$

These possible assignments are enormously large for large n. Moreover, we might find that another value of m may lead to smaller total transportation costs. Each value of m brings a new arrangement of satellite locations and we cannot tell a priori without exhaustive study what will be the optimum value m giving a global minimum of transaction costs. Moreover, as the number of central facilities increases, the cost of invested capital and operating costs increase at the same time.

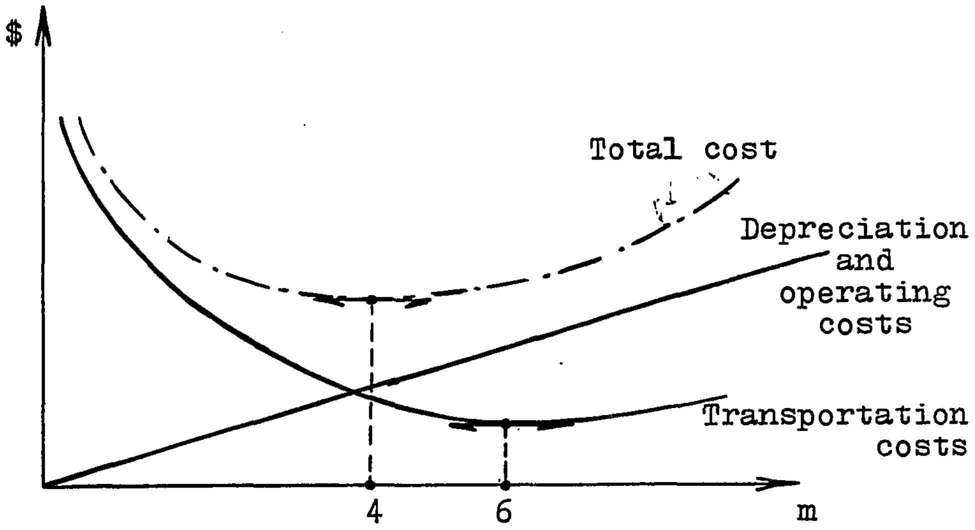


Fig. 32. Total Cost Global Minimum

If we refer to the above figure we see that a minimum transportation cost is reached with 6 central facilities, but the optimum number minimizing the total cost is 4.

Our cost function $C(m)$ is the sum of transportation costs $C_1(m)$ and the depreciation and operating costs $C_2(m)$

$$C(m) = C_1(m) + C_2(m)$$

The shipping costs are proportional to distances as well as to quantity shipped, this cost may be discontinuous in the case, for example of quantity discount.

The cost of invested capital and operating costs could readily be estimated by standard economic analysis if we knew the corresponding satellites and their respective demand or supply. The location of the central facilities and their respective assignments must first be solved.

II.2.2.1. <u>Central Facility Location and Assignment</u>

The transportation cost is function of the location of the n facilities (x_i, y_i) $i = 1, 2, 3, \ldots, n$ as well as the number m and location of the central facilities (X_j, Y_j) $j = 1, 2, \ldots, m$. We will assume that each facility is connected only to a unique central point, therefore, only the distances connecting a central point to its respective satellites should be considered. A facility i may or may not be connected to a central location j and we will use the Kronecker delta δ_{ij} of value 1 if i is connected to j or value 0 if it is not.

Therefore, the transportation cost in Euclidean space can be written as

$$C = \sum_{j=1}^{m} \sum_{i=1}^{n} \delta_{ij} \; w_{ij} \left[(X_j - x_i)^2 + (Y_j - y_i)^2 \right]^{\frac{1}{2}}$$

A set of m stationary points is found by solving the m equations in X_j and m equations in Y_j

$$C_{X_j} = 0$$

$$C_{Y_j} = 0$$

or for $j = 1, 2, 3, \cdots, m$

$$\sum_{i=1}^{n} \frac{\delta_{ij}\ w_{ij}\ (X_j - x_i)}{\left[(X_j - x_i)^2 + (Y_j - y_i)^2\right]^{\frac{1}{2}}} = \sum_{i=1}^{n} \frac{\delta_{ij}\ w_{ij}\ (X_j - x_i)}{D_{ij}} = 0$$

$$\sum_{i=1}^{n} \frac{\delta_{ij}\ w_{ij}\ (Y_j - y_i)}{\left[(X_j - x_i)^2 + (Y_j - y_i)^2\right]^{\frac{1}{2}}} = \sum_{i=1}^{n} \frac{\delta_{ij}\ w_{ij}\ (Y_j - y_i)}{D_{ij}} = 0$$

Following the same development as for one central facility we must prove that the principal minor determinants of the Hessian matrix are all positive for the stationary point to be a minimum.

$$\left.\begin{array}{l} C_{X_jX_j} > 0 \\ C_{X_jX_j}\ C_{Y_jY_j} - C^2_{X_jY_j} > 0 \end{array}\right. \quad \text{for } j = 1, 2, 3, \cdots, m$$

This minimum is then found by solving the extremal equations

$$\sum_{i=1}^{n} \frac{\delta_{ij}\ w_{ij}\ X_j}{D_{ij}} - \sum_{i=1}^{n} \frac{\delta_{ij}\ w_{ij}\ x_i}{D_{ij}} = 0$$

$$\sum_{i=1}^{n} \frac{\delta_{ij} \ w_{ij} \ Y_j}{D_{ij}} - \sum_{i=1}^{n} \frac{\delta_{ij} \ w_{ij} \ y_i}{D_{ij}} = 0$$

which lead to

$$\left.\begin{aligned} X_j &= \frac{\displaystyle\sum_{i=1}^{n} \frac{\delta_{ij} \ w_{ij} \ x_i}{D_{ij}}}{\displaystyle\sum_{i=1}^{n} \frac{\delta_{ij} \ w_{ij}}{D_{ij}}} \\ Y_j &= \frac{\displaystyle\sum_{i=1}^{n} \frac{\delta_{ij} \ w_{ij} \ y_i}{D_{ij}}}{\displaystyle\sum_{i=1}^{n} \frac{\delta_{ij} \ w_{ij}}{D_{ij}}} \end{aligned}\right\} j = 1, 2, 3, \cdots, m$$

As D_{ij} is a function of X_j and Y_j, these equations cannot be solved directly, they must be solved by iteration. We will assume as starting values of the iterative process the values $X_j^{(o)}$ and $Y_j^{(o)}$ obtained by using the quadratic formulation of distances

$$C = \sum_{j=1}^{m} \sum_{i=1}^{n} \delta_{ij} \ w_{ij} \ \left[(X_j - x_i)^2 + (Y_j - y_i)^2\right]$$

$$\left.\begin{aligned} C_{X_j} &= \sum_{i=1}^{n} 2\, \delta_{ij}\, w_{ij}\, (X_j - x_i) = 0 \\ C_{Y_j} &= \sum_{i=1}^{n} 2\, \delta_{ij}\, w_{ij}\, (Y_j - y_i) = 0 \end{aligned}\right\} j = 1,\ 2,\ \cdots,\ m$$

$$\left.\begin{aligned} X_j^{(0)} &= \frac{\sum_{i=1}^{n} \delta_{ij}\, w_{ij}\, x_i}{\sum_{i=1}^{n} w_{ij}} \\ Y_j^{(0)} &= \frac{\sum_{i=1}^{n} \delta_{ij}\, w_{ij}\, y_i}{\sum_{i=1}^{n} w_{ij}} \end{aligned}\right\} j = 1,\ 2,\ \cdots,\ m$$

then the solution of the exact Euclidean distances is given by the iterative process

$$X_j^{(k+1)} = \frac{\displaystyle\sum_{i=1}^{n} \frac{\delta_{ij}\, w_{ij}\, x_i}{\left[(X_j^{(k)} - x_i)^2 + (Y_j^{(k)} - y_i)^2\right]^{\frac{1}{2}}}}{\displaystyle\sum_{i=1}^{n} \frac{\delta_{ij}\, w_{ij}}{\left[(X_j^{(k)} - x_i)^2 + (Y_j^{(k)} - y_i)^2\right]^{\frac{1}{2}}}}$$

$$Y_j^{(k+1)} = \frac{\displaystyle\sum_{i=1}^{n} \frac{\delta_{ij}\ w_{ij}\ y_i}{\left[(X_j^{(k)}-x_i)^2 + (Y_j^{(k)}-y_i)^2\right]^{\frac{1}{2}}}}{\displaystyle\sum_{i=1}^{n} \frac{\delta_{ij}\ w_{ij}}{\left[(X_j^{(k)}-x_i)^2 + (Y_j^{(k)}-y_i)^2\right]^{\frac{1}{2}}}}$$

for $j = 1, 2, \cdots, m$ and all possible combinations of Kronecker δ_{ij}.

We are quickly limited by the size of the problem and it may become uneconomical to use this exhaustive technique for more than 10 facilities. We must in fact, not only compute all the possible assignments for m central locations which may run easily into many million combinations but also we must investigate the variation of total cost as m is varied, the development of other techniques is necessary when we must deal with some common problems involving many hundred elements.

III.2.2. Bound on Sum of Distances

When considering a set of m central locations P_j minimizing the sum of distances, we consider a network with branches of length D_{ij} from P_j to the facility i. The total length of the network is

$$D = \sum_{j=1}^{m} \sum_{i=1}^{n} \delta_{ij} \; D_{ij}$$

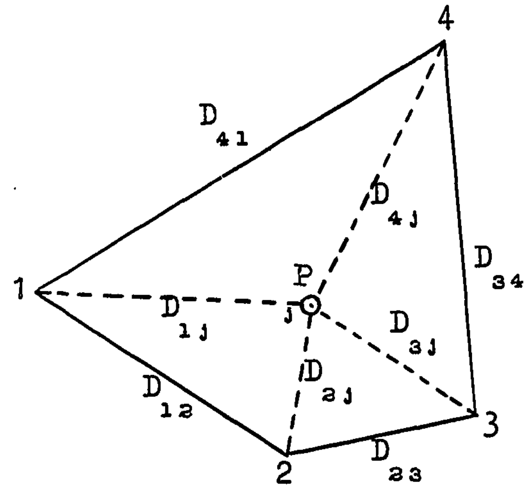
We have evaluated the bounds for the sum of distances in the case of one central facility. In the case of m central facilities we do not have a priori the value of the Kronecker delta δ_{ij} nor the satellites assigned to a given central facility. However, if we knew this allocation of satellites we could apply to that particular set the triangle inequality relating the distance between 2 facilities i and k (D_{ik}) and the distances of these facilities i and k to the central one j (D_{ij} and D_{kj}). We have

$$D_{ij} + D_{kj} \geq D_{ik}$$

If h facilities are satellites of the central location j then for this set of satellites the total optimum distance is D_j

$$D_j = \sum_{i=1}^{h} D_{ij}$$

In the case of 4 satellite facilities, for example



$$
\begin{aligned}
D_{1j} + D_{2j} &\geq D_{12} \\
D_{2j} + D_{3j} &\geq D_{23} \\
D_{3j} + D_{4j} &\geq D_{34} \\
D_{4j} + D_{1j} &\geq D_{41}
\end{aligned}
$$

$$2D_{1j} + 2D_{2j} + 2D_{3j} + 2D_{4j} \geq D_{12} + D_{23} + D_{34} + D_{41}$$

$$2 \sum_{i=1}^{4} D_{ij} \geq D_{12} + D_{23} + D_{34} + D_{41}$$

$$D_j = \sum_{i=1}^{4} D_{ij} \geq \frac{1}{2}(D_{12} + D_{23} + D_{34} + D_{41})$$

Then in a generalized problem

$$D = \sum_{j=1}^{m} D_j$$

will be greater or equal to one half of n distances of the D_{ik} matrix and certainly greater than one half of the n least distances.

CHAPTER IV

HEURISTIC ALGORITHMS

DISCRETE - TWO DIMENSIONAL SPACE - EUCLIDEAN DISTANCES

When considering the unconstrained or constrained locational problem with single or multiple central facilities we always reach a point at which an iteration technique is required because of the implicit nature of the equations. Manual computation in such a case is quite tedious; computer programming is necessary when dealing with a large set of facilities. In the previous chapter we dealt at length on the mathematical reasoning supporting our method. It is often found in practice that working tools are also necessary and the development of these tools is rarely presented in the literature. This explains why a large amount of valuable research is frequently wasted or ignored because of a missing link between the scientist and the potential user. The following programs have been developed to be applied as an easy tool by any prospective user. Some of these algorithms have been studied extensively by Kuehn and Hamburger [35], Cooper [10], Feldman, Lehrer and Ray [24], Vergin and Rogers [53], but if their range of accuracy is discussed at length, their innerworkings are not directly available to the user. Some other programs are new approaches giving more flexibility for the particular case of very large systems often present in federal government locational problems.

For each heuristic algorithm we shall study the general principle, the detailed logic diagram, some characteristics of programming, an application to an actual problem and a discussion on the results obtained and the corresponding expenses in computer time and memory.

The programs are written in FORTRAN which is a very known language but expensive in memory and computational requirements, a more careful programming or the use of AUTOCODER might be necessary in some cases.

Terminology of variables in the following algorithms and computer programs. Some of these variables will be more fully explained in the corresponding algorithms using them.

x_i, y_i or X(I), Y(I): Cartesian coordinates of facilities i

r_i or XR(I): Rate of transport from facility i

If we try to minimize a cost function it might be in dollar per pound per mile for example. If we try to minimize a time function it might be in nanosecond per bit per meter for example.

m_i or XM(I): Amount to transport from facility i to the central location P.

It might represent for example the poundage of goods to carry or the number of digitized bits of a message to transmit.

N : Number of facilities

M : Number of central facilities

ITERA : Number of iterations in random search of facilities

IGRID : Initial number of grid divisions on each X and Y axis

ITGRD : Number of grid size changes

INC : Incremental number of divisions on each X and Y axis when passing from one grid size to the next

DOLD(I), DNEW(I): Old and new Euclidean distances from facility I to optimum central location

SDOLD, SDNEW: Old and new sum of distances to the central facilities

COLD(I), CNEW(I): Old and new transportation costs from facility I to optimal central location

SCOLD, SCNEW: Old and new sum of transport cost to the central facility

JOSAV(I), JNSAV(I): Old and new code number allocation of the facility I

IOSAV(J), INSAV(I): Old and new code number of the randomly selected central location

RAINC : Class width on cumulative distribution of locations

RAD(I) : Class boundaries on cumulative distribution of locations

KITER : Iteration counter

YFL : Random number between 0 and 1.000

XC(J), YC(J): Cartesian coordinates of randomly selected central facilities

D(I,J) : Euclidean distances from facility I to central location J

L : Grid spacing counter

XMIN, XMAX: Minimum and maximum values of X(I)

YMIN, YMAX: Minimum and maximum values of Y(I)

IV.1. ONE CENTRAL LOCATION

IV.1.1. One Central Facility Heuristic Algorithm

The program is based on the iterative algorithm presented on page 50.

$$X^{(k+1)} = \frac{\displaystyle\sum_{i=1}^{n} \frac{r_i\, m_i\, x_i}{\left[(X^{(k)}-x_i)^2 + (Y^{(k)}-y_i)^2\right]^{\frac{1}{2}}}}{\displaystyle\sum_{i=1}^{n} \frac{r_i\, m_i}{\left[(X^{(k)}-x_i)^2 + (Y^{(k)}-y_i)^2\right]^{\frac{1}{2}}}}$$

$$Y^{(k+1)} = \frac{\displaystyle\sum_{i=1}^{n} \frac{r_i \; m_i \; y_i}{\left[(X^{(k)}-x_i)^2 + (Y^{(k)}-y_i)^2\right]^{\frac{1}{2}}}}{\displaystyle\sum_{i=1}^{n} \frac{r_i \; m_i}{\left[(X^{(k)}-x_i)^2 + (Y^{(k)}-y_i)^2\right]^{\frac{1}{2}}}}$$

with

$$X^{(o)} = \frac{\displaystyle\sum_{i=1}^{n} r_i \; m_i \; x_i}{\displaystyle\sum_{i=1}^{n} r_i \; m_i}$$

$$Y^{(o)} \quad \frac{\displaystyle\sum_{i=1}^{n} r_i \; m_i \; x_i}{\displaystyle\sum_{i=1}^{n} r_i \; m_i}$$

The logic diagram of the program is given on pages 86-87. The computer print-out and a set of results corresponding to 50 hypothetical facilities are given from page 182 to page 187.

It is to be noted that there is no built-in check for weight dominance and the user should scrutinize more thoroughly a solution which would be very close to an existing facility. The case of weight dominance of one facility is however, quite improbable when studying a large system.

The iterative process was stopped when

$$X^{(k+1)} - X^{(k)} \leq \text{ERR}$$

$$Y^{(k+1)} - Y^{(k)} \leq \text{ERR}$$

the value of ERR being read in as a problem variable.

The algorithm was run with problems of various sizes from $n = 3$ to $n = 500$. It is to be noted that with $\text{ERR} = 0.01$, in the case of 3 facilities the central location was found in 23 iterations while for a problem of 500 facilities it required 28 iterations. The number of iterations depends largely on the extreme locations of some facility as the weighted distances will give a poor first estimate [36] and the number of steps of the iterative search are not necessarily more numerous with a large system than with a small one.

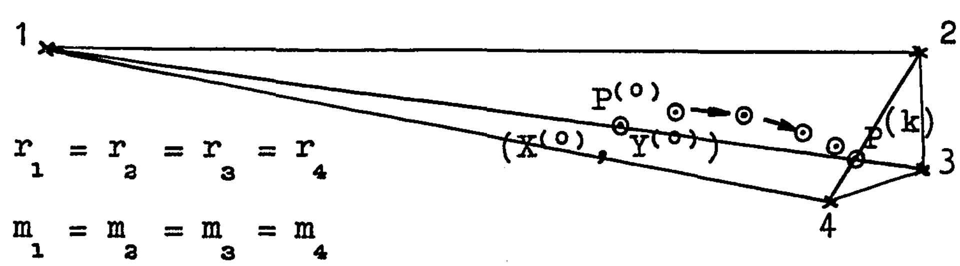


Fig. 33. Case of Extreme Locations in the Determination of One Central Facility

When looking at the variation of total transportation costs during the iterative process on the computer print-out, we can see that there is a lack of sharp minimum, which means that a rough location of the central facility, as given for example by the starting value in the case of non-extreme locations may be sufficient in practical use.

The program was run in the case of weight dominance and the algorithm rapidly converges toward the dominant facility, however, never exactly reaching it. (see page 189)

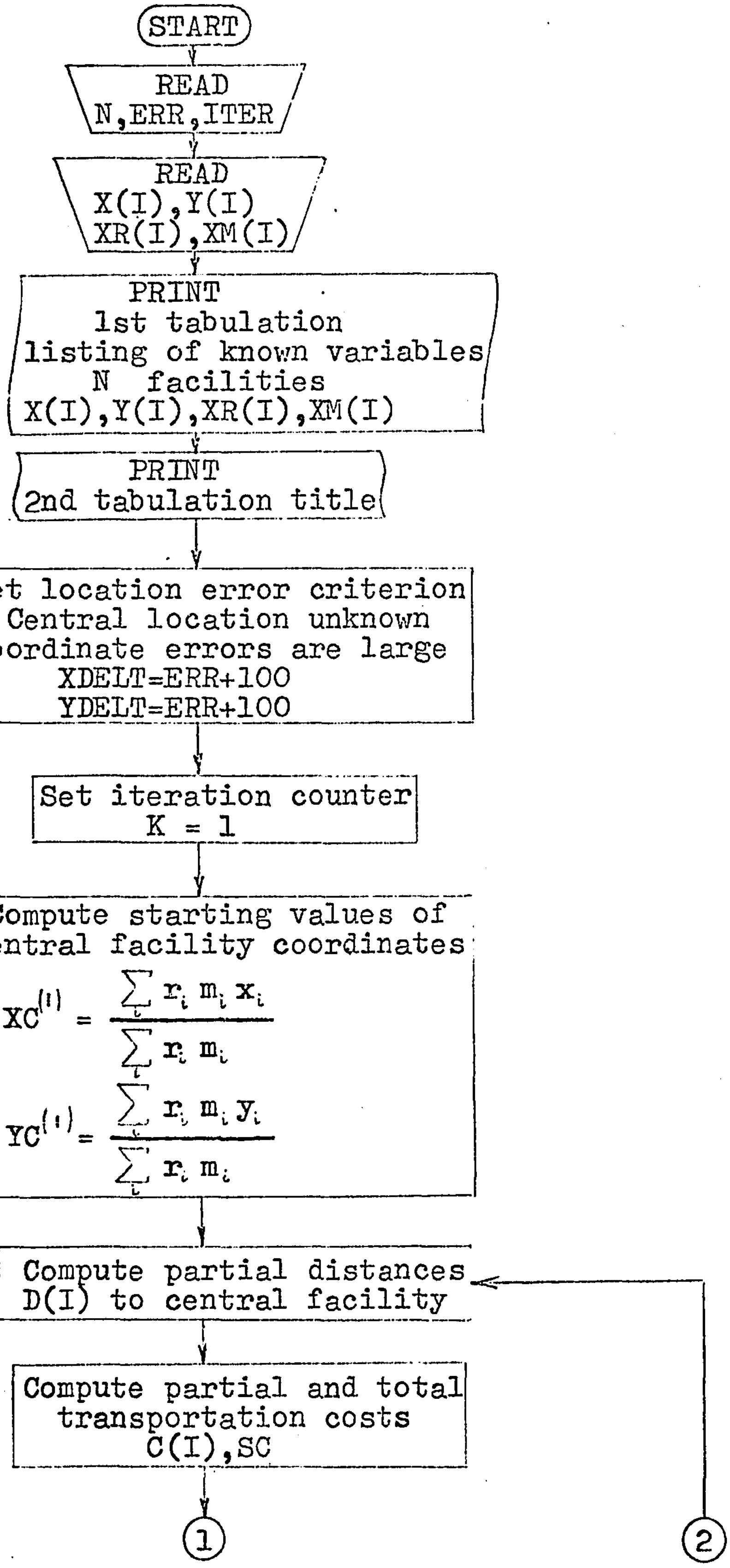
Computational time on an electronic digital computer may be quite expensive and a tally of time was kept to try to derive a relation between the number of facilities investigated and the corresponding computational time on an IBM 360/40. In the particular system used the computer operates on two problems at a time (mode MFP2) and the variable demands on its elements and stored sub-routines do not allow a time print-out reliable for each of the problems. This inconsistency can be readily seen on the following Table 3, where compilation time varies drastically from one run to the next. There is no sure trend, but it can be considered that computational time is of insignificant importance in the total economics of most of the problems.

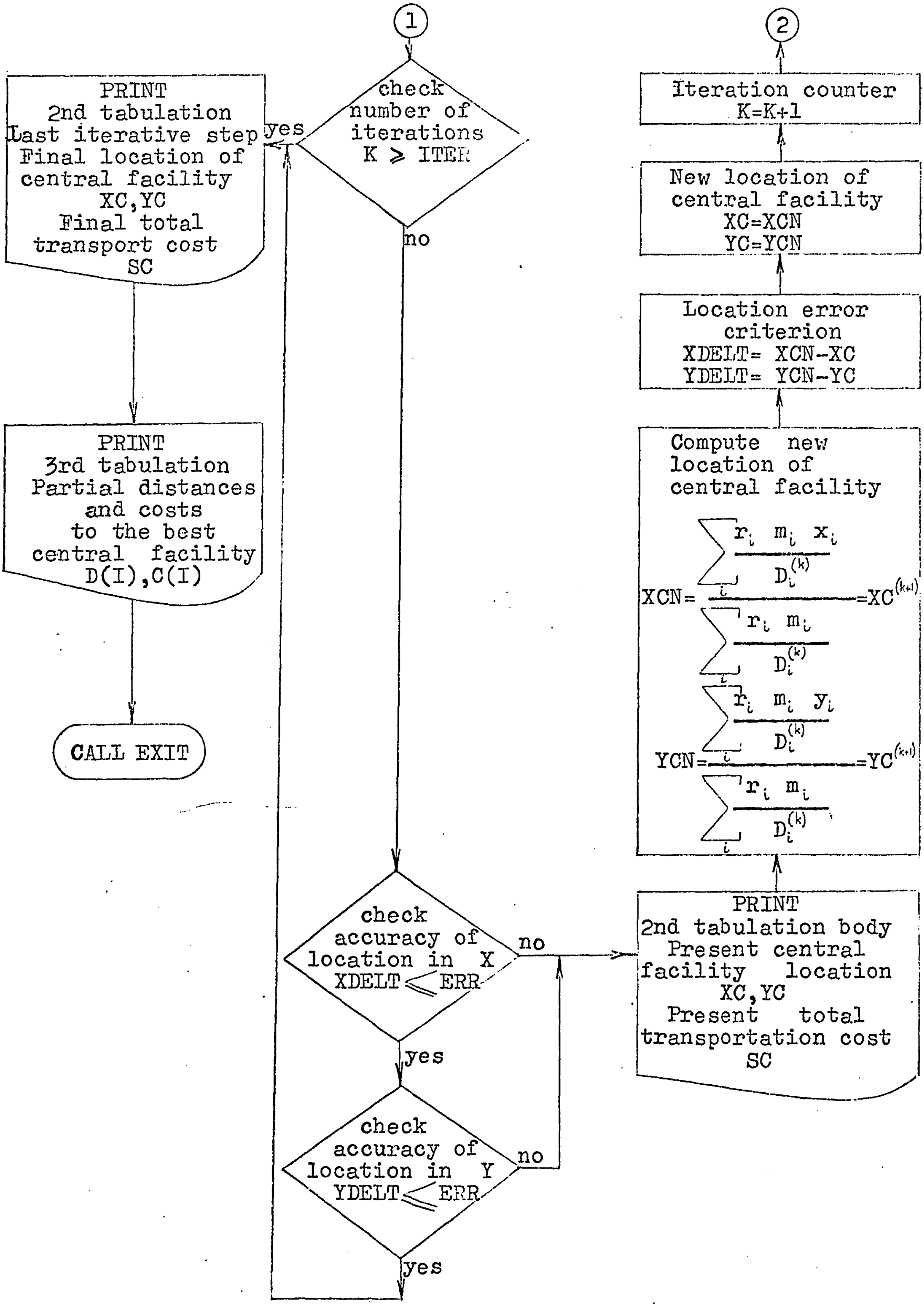
The total memory requirement in the case of 1000 facilities is 66F2 or 26,354 bytes of which 742 are used for the program.

Real life data were used in the computation of one optimum central facility for the postal system, considering the continental 50 states of the union and their corresponding output in first class letter mail in the year 1965 [52]. This output volume was assumed to be originating from the capital city of the state the longitudes and latitudes of which were given [23]. With this particular limitation in the type of mail and assuming plane geometry, an optimum location for a postal institute for example, should be at 39°39'N and 83°27'W around Columbus, Ohio.

The program was also run for the sets of 20 facilities and 125 facilities which are used later on, in the analysis of the multiple central facilities algorithms. The results are shown in Figure 34 and Figure 35.

Diagram 1. One Central Facility Heuristic Algorithm




1
check number of iterations K ≥ ITER
yes
no
PRINT 2nd tabulation Last iterative step Final location of central facility XC,YC Final total transport cost SC
PRINT 3rd tabulation Partial distances and costs to the best central facility D(I),C(I)
CALL EXIT
check accuracy of location in X XDELT ≤ ERR
no
yes
check accuracy of location in Y YDELT ≤ ERR
no
yes
PRINT 2nd tabulation body Present central facility location XC,YC Present total transportation cost SC
Compute new location of central facility
$XCN = \frac{\sum_i \frac{r_i m_i x_i}{D_i^{(k)}}}{\sum_i \frac{r_i m_i}{D_i^{(k)}}} = XC^{(k+1)}$
$YCN = \frac{\sum_i \frac{r_i m_i y_i}{D_i^{(k)}}}{\sum_i \frac{r_i m_i}{D_i^{(k)}}} = YC^{(k+1)}$
Location error criterion XDELT= XCN-XC YDELT= YCN-YC
New location of central facility XC=XCN YC=YCN
Iteration counter K=K+1
2

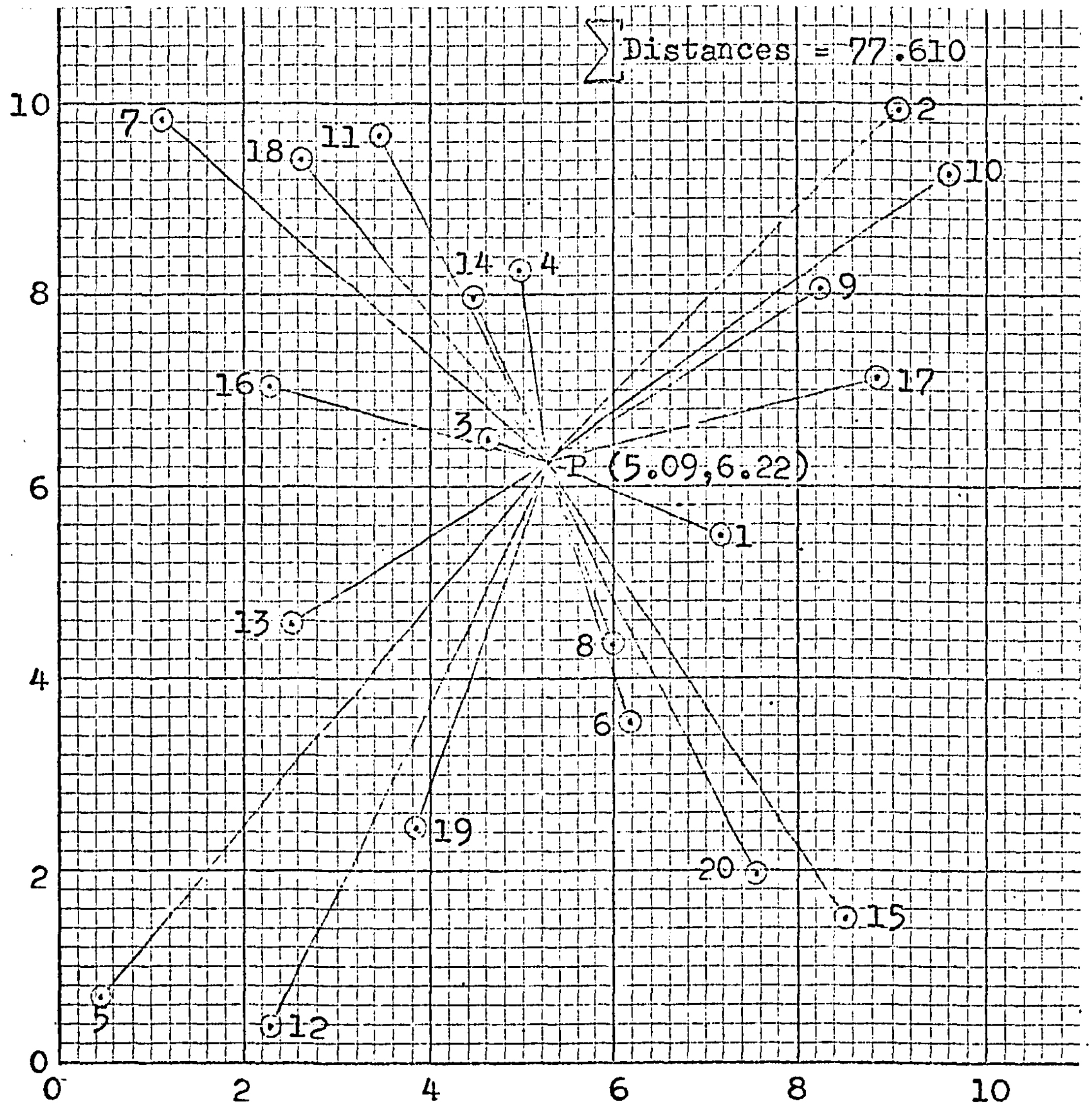


Fig. 34 Location of one central facility
Heuristic algorithm , N = 20

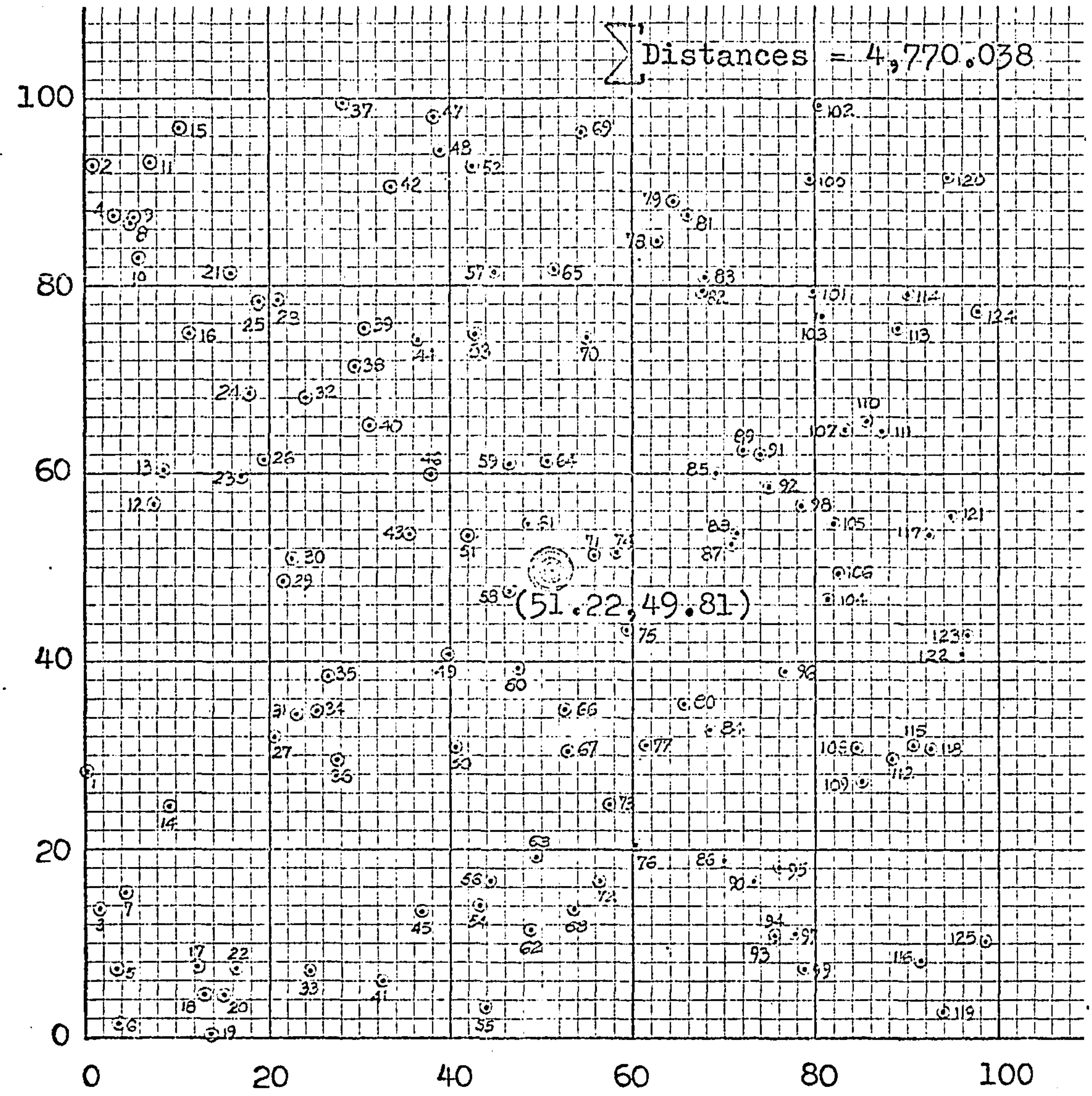


Fig. 35 Location of one central facility
Heuristic algorithm, N = 125

Table 3 – Computation Time Requirement –

Discrete – Two Dimensional Space – Euclidean Distances

Algorithm to Locate One Central Facility

Computer: IBM 360/40, Printer IBM 1403 N1,

Computer Operating, Under MFP2

Time in Hours.Minutes.Seconds (Sexagesimal)

No. of Facil-ities	No. of Itera-tions	Time JOB-LKED Compilation	Time LKED-GO Subroutine Ass.	Time GO-END Oper.Time	Total Time
5	14	00.01.10	00.01.42	00.00.50	00.03.42
15	10	00.00.47	00.01.58	00.00.21	00.01.56
15	10	00.00.50	00.01.03	00.01.33	00.03.26
25	10	00.01.35	00.01.41	00.00.52	00.04.08
50	24	00.01.47	00.01.33	00.00.29	00.03.49
75	13	00.01.00	00.00.47	00.01.03	00.02.50
125	19	00.01.01	00.00.49	00.00.39	00.02.29
500	28	00.01.21	00.02.04	00.02.35	00.05.60

Table 3a - Computation Time Requirement -
Discrete - Two Dimensional Space - Euclidean Distances
Algorithm to Locate One Central Facility

Computer: IBM 360/40, Printer IBM 1403 N1,
Computer Operating, Under MFP2

Time in Hours.Minutes.Seconds (Sexagesimal)

Number of Facilities	Begin Time JOB	Begin Step LKED	Begin Step GO	End JOB
5	11.06.20	11.07.30	11.09.12	11.10.02
15	04.11.05	04.11.52	04.12.40	04.13.01
15	20.11.10	20.12.00	20.13.03	20.14.36
25	11.10.15	11.11.50	11.13.31	11.14.23
50	04.13.09	04.14.56	04.16.29	04.16.58
75	14.14.50	14.15.50	14.16.37	14.17.40
125	12.49.26	12.50.27	12.51.16	12.51.55
500	20.05.04	20.06.25	20.08.29	20.11.04

Table 4 Postal System

Optimal Location of One Central Facility

Processing All States Daily Output of

First Class Mail

#	State	Capital	Lat. Deg-Min	Long. Deg-Min	Mail Output Pounds
1	Alabama	Montgomery	32° 23 'N	86° 17 'W	62,650
2	Alaska	Juneau	58° 25 'N	134° 30 'W	1,163
3	Arizona	Phoenix	33° 30 'N	112° 00 'W	22,288
4	Arkansas	Little Rock	34° 42 'N	92° 16 'W	20,916
5	California	Sacramento	38° 35 'N	121° 30 'W	394,139
6	Colorado	Denver	39° 44 'N	104° 59 'W	47,453
7	Connecticut	Hartford	41° 45 'N	72° 40 'W	74,813
8	Delaware	Dover	39° 10 'N	75° 30 'W	15,863
9	Washington, D.C.	Washington	38° 50 'N	77° 00 'W	238,476
10	Florida	Tallahassee	30° 25 'N	84° 17 'W	80,791
11	Georgia	Atlanta	33° 45 'N	84° 23 'W	79,198
12	Idaho	Boise	43° 38 'N	116° 12 'W	9,502
13	Illinois	Springfield	39° 46 'N	89° 37 'W	387,961
14	Indiana	Indianapolis	39° 45 'N	86° 08 'W	91,294
15	Iowa	Des Moines	41° 35 'N	93° 37 'W	60,204
16	Kansas	Topeka	39° 02 'N	95° 41 'W	47,086

Optimal Central Location: 39° 39 'N, 83° 27 'W

#	State	Capital	Lat. Deg-Min	Long. Deg-Min	Mail Output Pounds
17	Kentucky	Frankfort	38° 10 'N	84° 55 'W	41,344
18	Louisiana	Batan Rouge	30° 28 'N	91° 10 'W	52,978
19	Maine	Augusta	44° 19 'N	69° 42 'W	17,631
20	Maryland	Annapolis	39° 00 'N	76° 25 'W	73,834
21	Massachusetts	Boston	42° 15 'N	71° 07 'W	153,409
22	Michigan	Lansing	42° 45 'N	84° 35 'W	127,187
23	Minnesota	St. Paul	44° 57 'N	93° 05 'W	90,323
24	Mississippi	Jackson	32° 17 'N	90° 10 'W	24,448
25	Missouri	Jefferson City	38° 34 'N	92° 10 'W	139,140
26	Montana	Helena	46° 35 'N	112° 01 'W	17,322
27	Nebraska	Lincoln	40° 49 'N	96° 43 'W	35,658
28	Nevada	Carson City	39° 10 'N	119° 45 'W	9,207
29	New Hampshire	Concord	43° 10 'N	71° 30 'W	11,631
30	New Jersey	Trenton	40° 13 'N	74° 46 'W	184,397
31	New Mexico	Sante Fe	35° 10 'N	106° 00 'W	17,645
32	New York	Albany	42° 40 'N	73° 50 'W	662,584
33	North Carolina	Raleigh	35° 45 'N	78° 39 'W	73,749
34	North Dakota	Bismark	46° 48 'N	100° 46 'W	11,646
35	Ohio	Columbus	40° 00 'N	83° 00 'W	219,330
36	Oklahoma	Oklahoma City	35° 27 'N	97° 32 'W	59,159
37	Oregon	Salem	44° 55 'N	123° 03 'W	45,733

Optimal Central Location: 39° 39 'N, 83° 27 'W

#	State	Capital	Lat. Deg-Min	Long. Deg-Min	Mail Output Pounds
38	Pennsylvania	Harrisburg	40° 15′N	76°50′W	302,933
39	Rhode Island	Providence	41°50′N	71°23′W	22,769
40	South Carolina	Columbia	34°00′N	81°00′W	28,434
41	South Dakota	Pierre	44°22′N	100°20′W	10,292
42	Tennessee	Nashville	36°10′N	86°48′W	68,770
43	Texas	Austin	30°15′N	97°42′W	233,041
44	Utah	Salt Lake City	40°45′N	111°52′W	23,110
45	Vermont	Montpelier	44°20′N	72°35′W	14,082
46	Virginia	Richmond	37°35′N	77°30′W	74,408
47	Washington	Olympia	47°02′N	122°52′W	53,472
48	West Virginia	Charleston	38°20′N	81°35′W	23,240
49	Wisconsin	Madison	43°05′N	89°23′W	86,544
50	Wyoming	Cheyenne	41°10′N	104°49′W	7,489

Optimal Central Location: 39°39′N, 83°24′W

IV.2. MULTIPLE CENTRAL LOCATIONS

IV.2.1. Multiple Central Facilities Heuristic Algorithm

The program should be based on the iterative algorithm presented on page 74.

$$X^{(k+1)} = \frac{\displaystyle\sum_{i=1}^{n} \frac{\delta_{ij}\; r_i\; m_i\; x_i}{\left[(X_j^{(k)}-x_i)^2 + (Y_j^{(k)}-y_i)^2\right]^{\frac{1}{2}}}}{\displaystyle\sum_{i=1}^{n} \frac{\delta_{ij}\; r_i\; m_i}{\left[(X_j^{(k)}-x_i)^2 + (Y_j^{(k)}-y_i)^2\right]^{\frac{1}{2}}}}$$

$$Y^{(k+1)} = \frac{\displaystyle\sum_{i=1}^{n} \frac{\delta_{ij}\; r_i\; m_i\; y_i}{\left[(X^{(k)}-x_i)^2 + (Y^{(k)}-y_i)^2\right]^{\frac{1}{2}}}}{\displaystyle\sum_{i=1}^{n} \frac{\delta_{ij}\; r_i\; m_i}{\left[(X_j^{(k)}-x_i)^2 + (Y_j^{(k)}-y_i)^2\right]^{\frac{1}{2}}}}$$

for $j = 1, 2, 3, \cdots, m$ and $\delta_{ij} = 0, 1$

with

$$X_j^{(o)} = \frac{\displaystyle\sum_{i=1}^{n} \delta_{ij}\; r_i\; m_i\; x_i}{\displaystyle\sum_{i=1}^{n} r_i\; m_i}$$

$$Y_j^{(0)} = \frac{\sum_{i=1}^{n} \delta_{ij} \; r_i \; m_i \; y_i}{\sum_{i=1}^{n} r_i \; m_i}$$

The amount of computation becomes rapidly prohibitive even for less than 10 facilities and it is of little interest to develop such an algorithm. Other procedures must be sought to reduce the computational effort. In some practical problems we might know the set of satellite facilities depending on one central location, we are then brought back to the previous case of locating only one central facility. Such a simplification is often realistic, for example several market areas may have their own individual source. In some other cases we might know the position of all the facilities as well as the central ones but we do not know the value of the Kronecker delta, that is, we must find the set of appropriate satellites connected with each central facility.

IV.2.2. <u>Destination Subset Algorithm</u> [10]

In some practical problems one often considers enlarging or modifying an existing facility so as to use it as a central location for a set of satellites. We must then consider a subset m of the n facilities such that it minimizes transportation costs when connected to the proper set of satellites.

There are

$$\frac{n!}{m!\ (n - m)!}$$

possible choices of n facilities taken m at a time. For each of these combinations of m central facilities we must consider all the distances to all the other points and allocate the satellites which give the minimum distances.

It is to be noted that for $m = 1$ the method is trivial. Every location is tried as a candidate for a central facility and the sum of transportation costs computed for each of them, the location giving the least transportation cost is chosen as central location. A program was written for this particular case of one facility used as central location. The logic diagram is shown on pages 101-102. The program was applied to a set of 20 facilities with equal weight as per Figure 36. The computer print-out and results are given from page 190 to page 193 and the problem was processed in 3 minutes 49 seconds on the IBM 360/40 computer. The program uses 542 bytes, the total memory requirement varies with the size of the problem but in the particular case of 50 facilities it takes 418 bytes more.

In the case where m is relatively large, the amount of computation may be cumbersome because of the large number of combinations: $\binom{n}{m}$. It is also complicated to make a general computer program which is able to consider all the possible combinations for all m. In the case where m is known a program can be readily developed using reference [38].

In practice, this assumption of using an existing facility as a central location is quite realisitc and it is not uncommon, for example, in the postal system to enlarge and mechanize an existing post office to use it as a "sectional center" handling mail of satellite post offices.

If because of the structure of the problem or its economic constraints the central facilities must be separate entities, then the use of the subset algorithm would not be correct. However, this algorithm allows the definition of the optimum δ_{ij}, that is the optimum set of satellites best served by one central location. For this particular set taken alone we may then compute the exact location of the central facility by using the program developed in paragraph IV.1.1. using the extremal equations found for the location of one and only center.

We are not sure however, that the method gives us the absolute minimum, first of all because we assume a priori a value of m, and secondly, because we intuitively decide that the set of satellites found by the subset algorithm are the best ones, even after correction is made to locate exactly the central facility. We also assume that the corrected algorithm will bring some improvement which is not necessarily the case. In fact, in case of partial weight dominance from some facilities then chosen as central one by the subset algorithm, the exact solution may increase substantially the total cost of transport. A high loading and unloading cost at the chosen center may also create higher total cost using the algorithm IV.1.1.

In the destination subset algorithm we do not have any choice in the length of the investigation procedure. It is only through exhaustive enumeration of all the combinations that a correct allocation may be found. We should keep in mind for example, that for $n = 100$, $m = 10$ there are

$$\binom{n}{m} = \binom{100}{10} = \frac{100!}{10!\ 90!} = 17{,}310 \text{ billion combinations of}$$

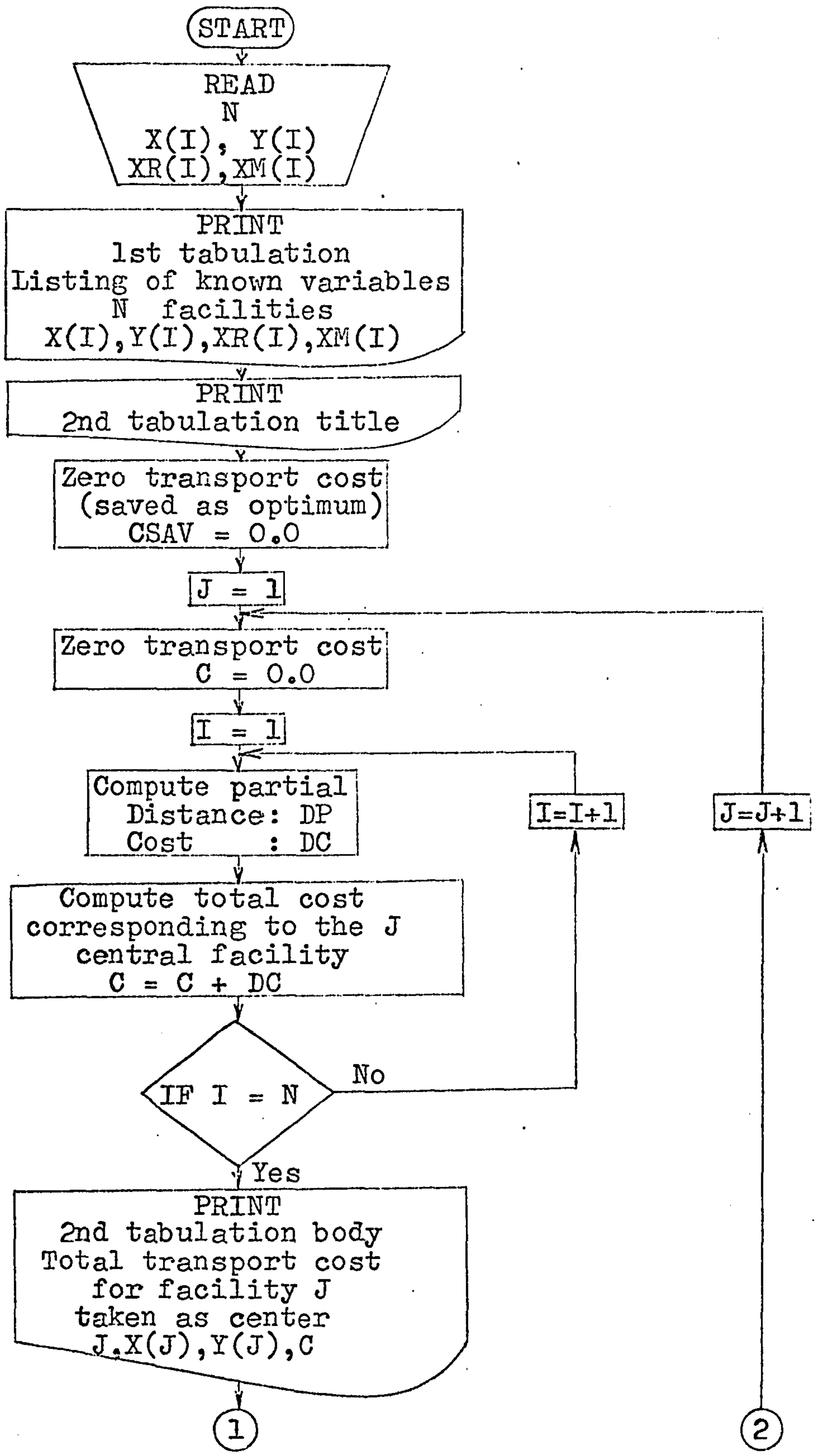
10 central facilities. Then for each of these centers

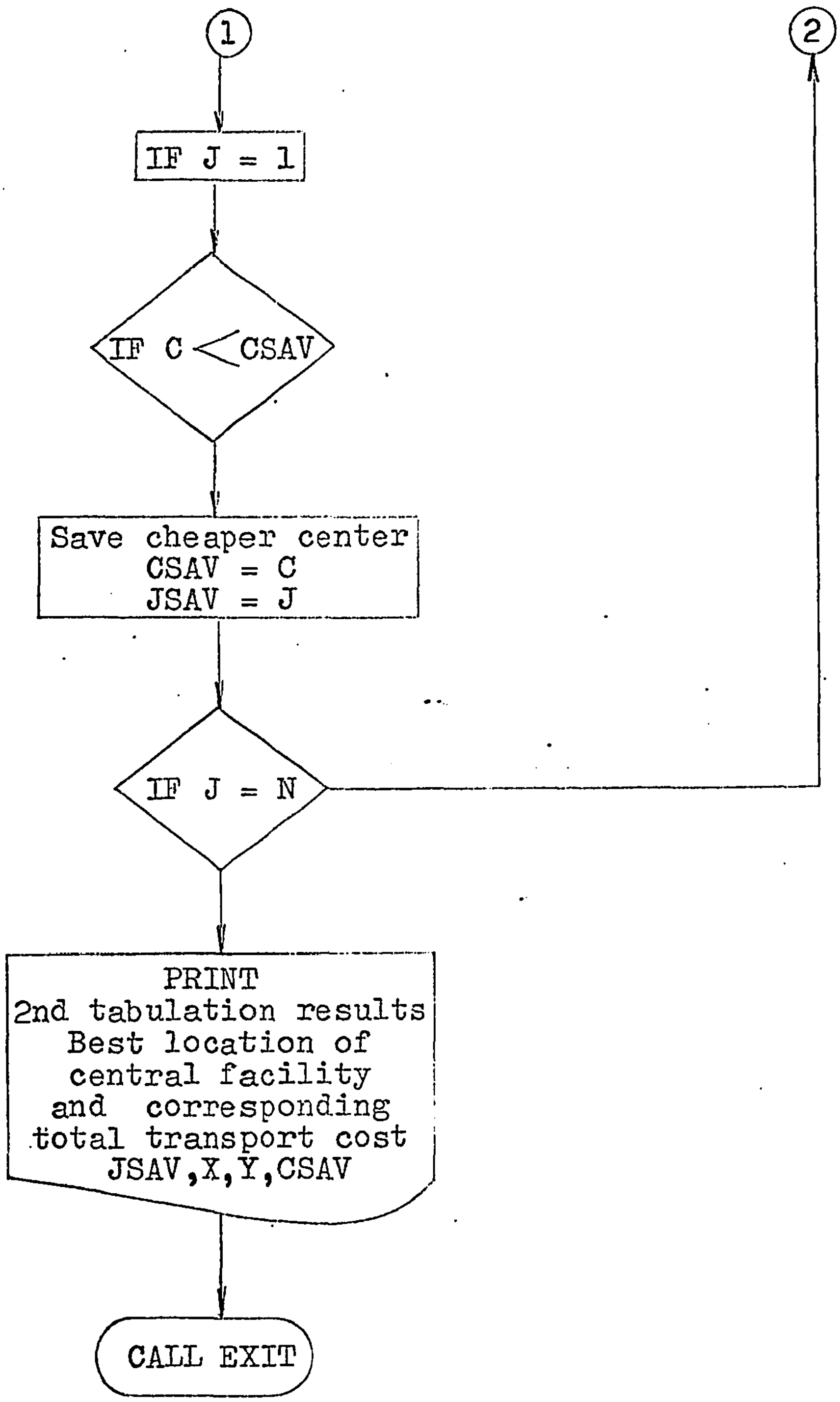
$(n - m) = 100 - 10 = 90$ distances must be computed, that is $m\ (n - m) = 900$ distances are computed then compared for each allocation. This represents for example, 15,579 trillion computations involving square and square roots to find the right set of central facilities and satellites in the case of 100 facilities and 10 centers.

We have seen in section II.3.5. that the central facilities will be definitively situated within the convex hull defined by straight lines connecting extreme points.

Except in very rare cases of weight dominance or large obtuse angles of the convex hull, the central facilities will be found at the extreme points, we could then make abstraction of these k extreme points, thus reducing our possible combinations from $\binom{n}{m}$ to $\binom{n-k}{m}$.

Diagram 2. Destination Subset Algorithm
One facility is used as central location




1
2
IF J = 1
IF C < CSAV
Save cheaper center
CSAV = C
JSAV = J
IF J = N
PRINT
2nd tabulation results
Best location of
central facility
and corresponding
total transport cost
JSAV,X,Y,CSAV
CALL EXIT

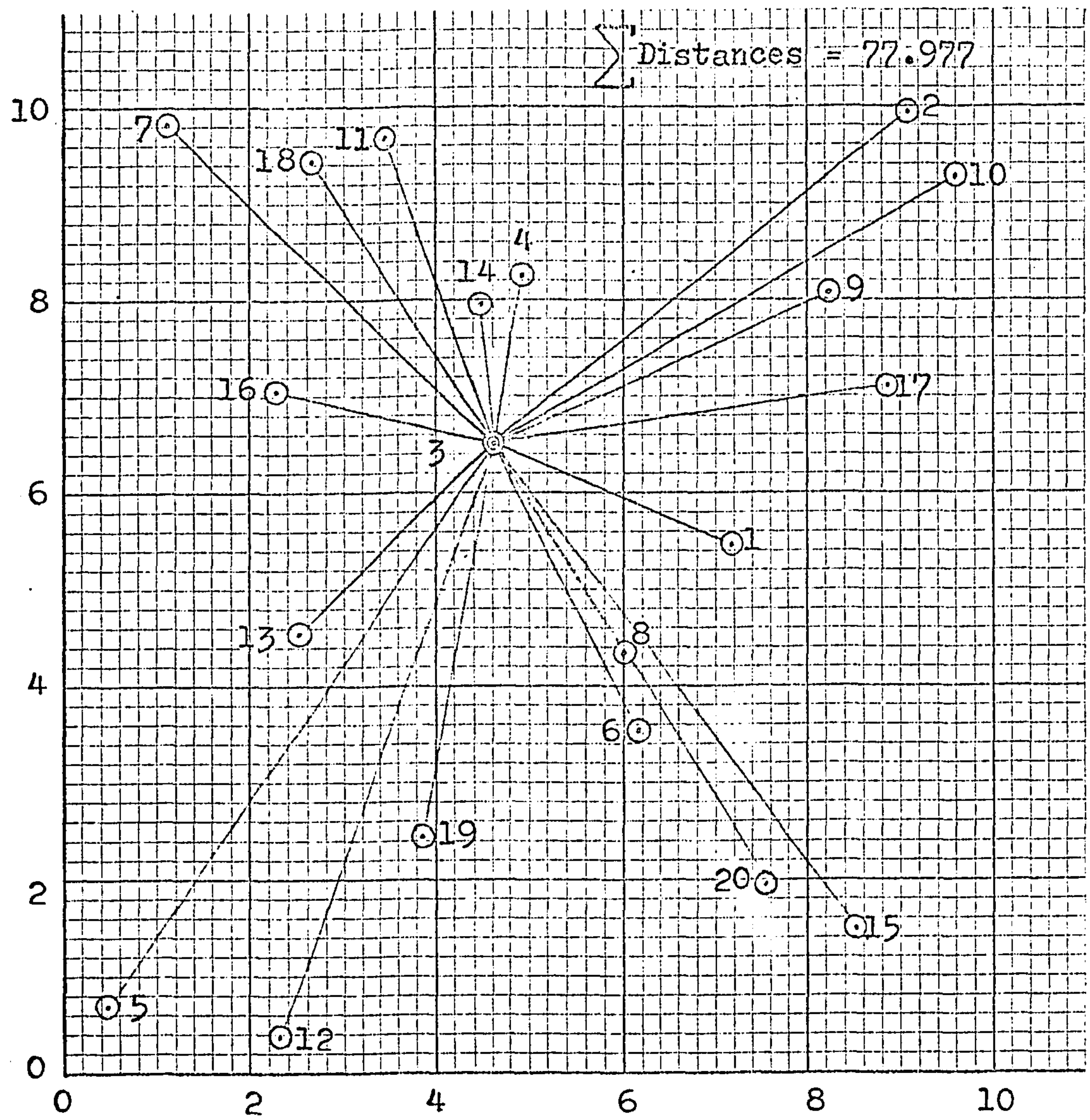


Fig. 36 Destination Subset Algorithm
One facility taken as a center
N = 20

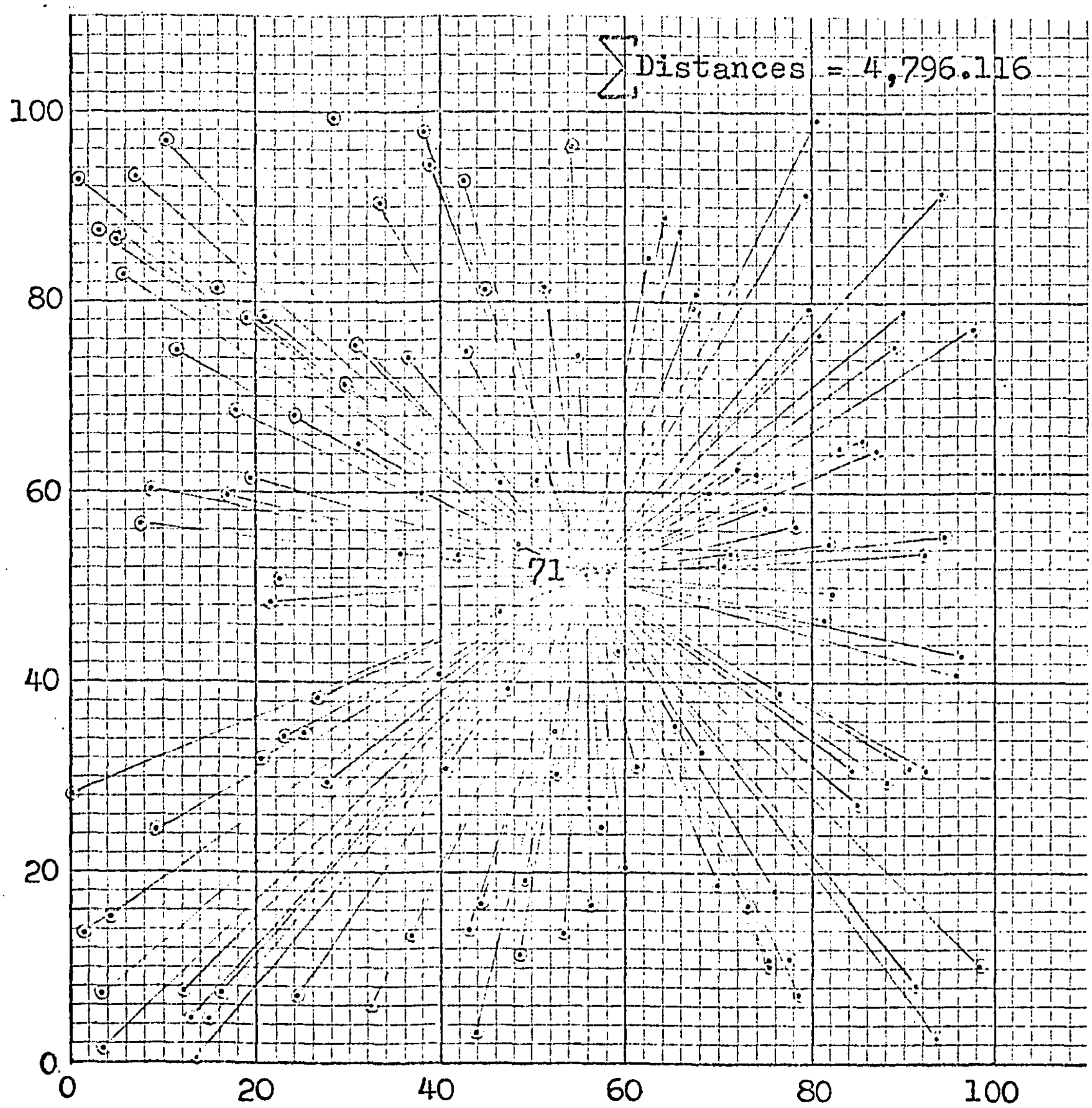


Fig. 37 Destination Subset Algorithm
One facility taken as a center
N = 125

IV.2.3. Variable Grid Algorithm

When considering many hundred facilities and numerous central ones, the destination subset algorithm, although practical in assumption, may be too cumbersome to use because of the length of exhaustive computations.

We know that the optimum location of all the j centers (X_j, Y_j), must satisfy the system of inequalities

$$x_{minimum} \leq X_j \leq x_{maximum}$$

$$y_{minimum} \leq Y_j \leq y_{maximum}$$

The area delimited by $x_{minimum}$, $x_{maximum}$ and $y_{minimum}$, $y_{maximum}$, can be divided into a mesh of large or fine spacing. The N intersection points of our mesh can then be treated as possible central facilities, m subsets of these N points can be found using a similar algorithm as in IV.2.2., so as to define the proper satellites minimizing transportation costs. In substance, this grid algorithm is similar to the destination subset algorithm but it is much more flexible. The choice of spacing will definitively influence our computing time (and possibly our locational accuracy), but at least we have a means of control on our computer time expenditure.

This algorithm is attractive as long as $N < n$, because it reduces the number of possible combinations to $\binom{N}{m}$. With this algorithm we are not too much interested in finding the correct location of the central facility but the correct subset of allocations, once these subsets are known they can be studied independently and the algorithm IV.1.1. can give us for each one the correct location of the center. We can start with a very loose mesh containing at least m intersection points, define from it the corresponding δ_{ij}. A new mesh is then re-defined with a tighter spacing; if the new subsets of allocations remain the same, then it is probable that these allocations are correct. Similarly as with the destination subset algorithm we are never sure that these allocations are the best ones.

IV.2.4. Variable Grid Algorithm with Linear Constraints

The central facilities being located within the convex hull, a more efficient variable grid algorithm should discard the mesh points outside this convex hull. The linear constraints defining this convex hull can be found by a relatively involved separate sub-routine program, but it must be kept in mind that these facilities will have to be plotted sooner or later in order to present the results to management, then it is quite easy on that plot to locate the extreme points.

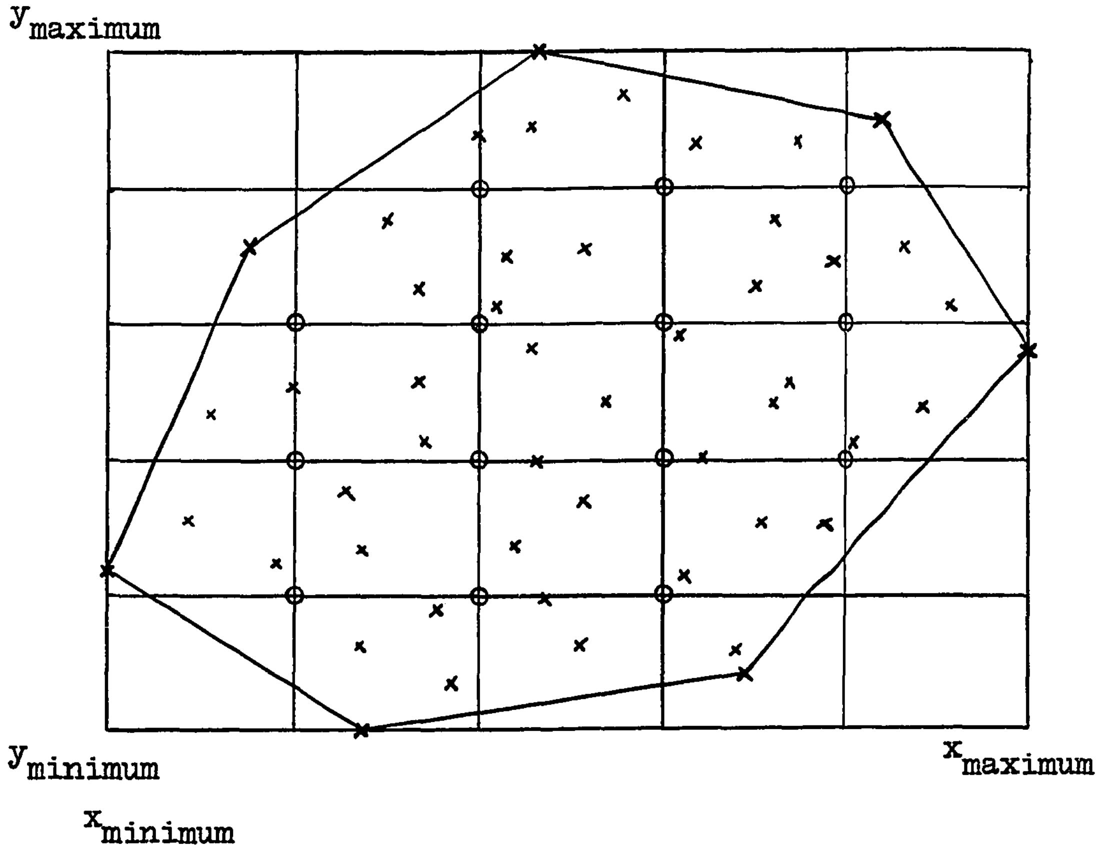


Fig. 38. Variable Grid Algorithm
with Linear Constraints

For this particular figure with $n = 50$ facilities, if we try to find the location of $m = 3$ central locations, we must consider $\binom{50}{3} = 19{,}600$ combinations or 2,763,600 computations. In the case of the variable grid algorithm with linear constraints only $\binom{14}{3} = 364$ combinations or 51,324 computations would be necessary. It is possible that the set of assignments in both cases may be identical and consequently the location of the central facility for each subset would be the same using the correcting algorithm IV,1.1.

Inversely, in the case of very few facilities, the variable grid algorithm can present many more intersecting points and consequently the use of algorithm IV.1.1. might not be necessary in the final analysis and the optimum set of grid points might be accurate enough. This case is somewhat realistic as centers are sometimes located at the intersection of ranges and townships.

Some extra effort is necessary to define the convex hull. In the case of a large system the plot can be done quite rapidly for example, with a Calcomp plotter at the output of an IBM 1130 computer using the plotting sub-routines or the powerful "data presentation system". Another method would not use any plot but would define the hull by the following procedure. The points are ordered by increasing values of x_i. The point corresponding to $x_{minimum}$ is definitively an extreme point. The set of lines connecting this extreme point to the other points of the set will have facilities located above and below them, except for two lines which delineate a part of the convex boundary. These extreme lines are connected to new extreme points; from these, new sets of lines are defined, new boundary lines are found, some of these boundary lines will give new extreme points not previously defined. Furthermore, the method allows the definition of all the bounding lines and extreme points.

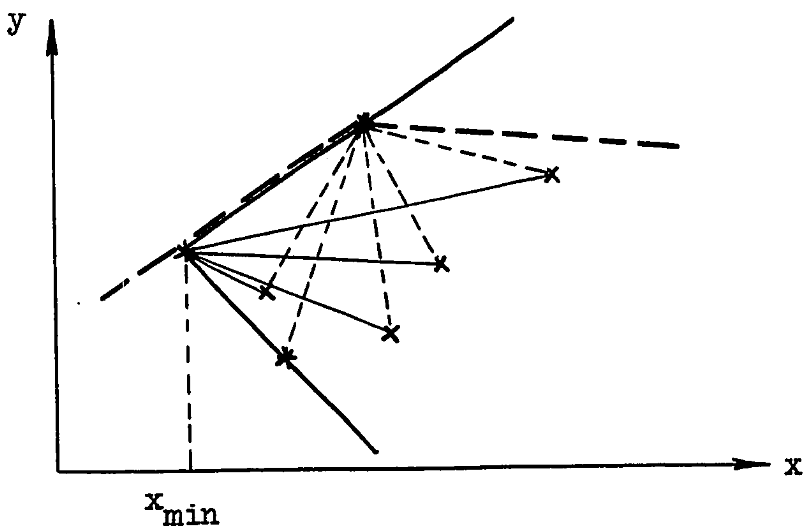


Fig. 39. Investigation Procedure to Define the Convex Hull

This particular grid algorithm with linear constraints may appear somewhat cumbersome to use. It may be pointed out that the destination subset algorithm by its very nature, automatically defines points within the convex hull. However, by using this variable grid, we have a direct handle on our computational effort and we may stop at any level of accuracy without having to go through the exhaustive set of $\binom{n}{m}$ combinations which might become enormous in the case of a very large system.

IV.2.5. Random Destination Algorithm [10]

The problem with the destination algorithm and even the variable grid algorithm is that the amount of computational effort may be very large. Also quite a number of combinations of grid points or destinations "rationally" chosen by a replacement process can easily be recognized as poorly chosen when considering the layout of facilities.

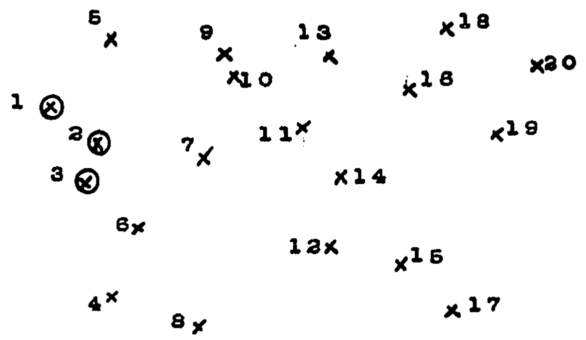
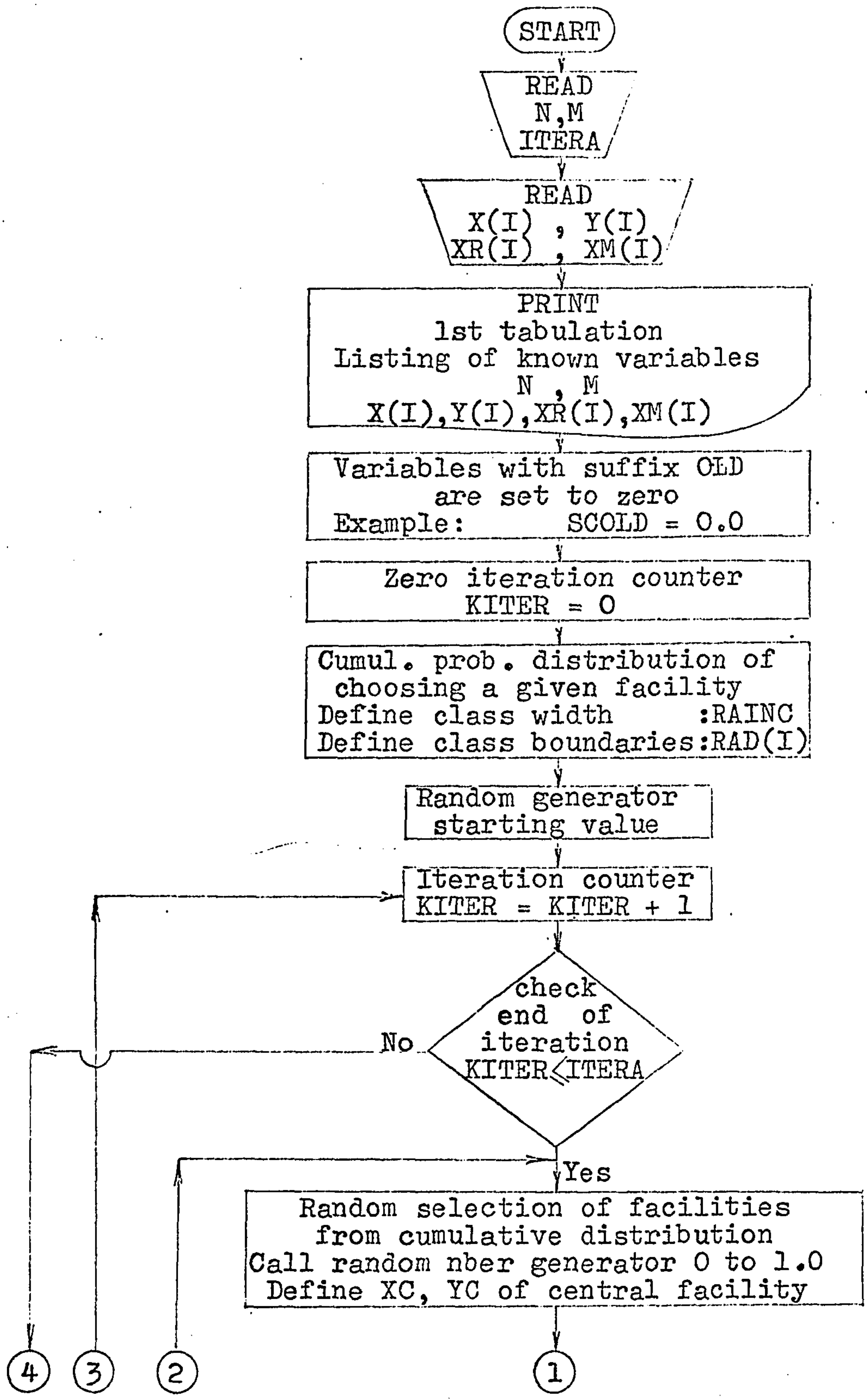


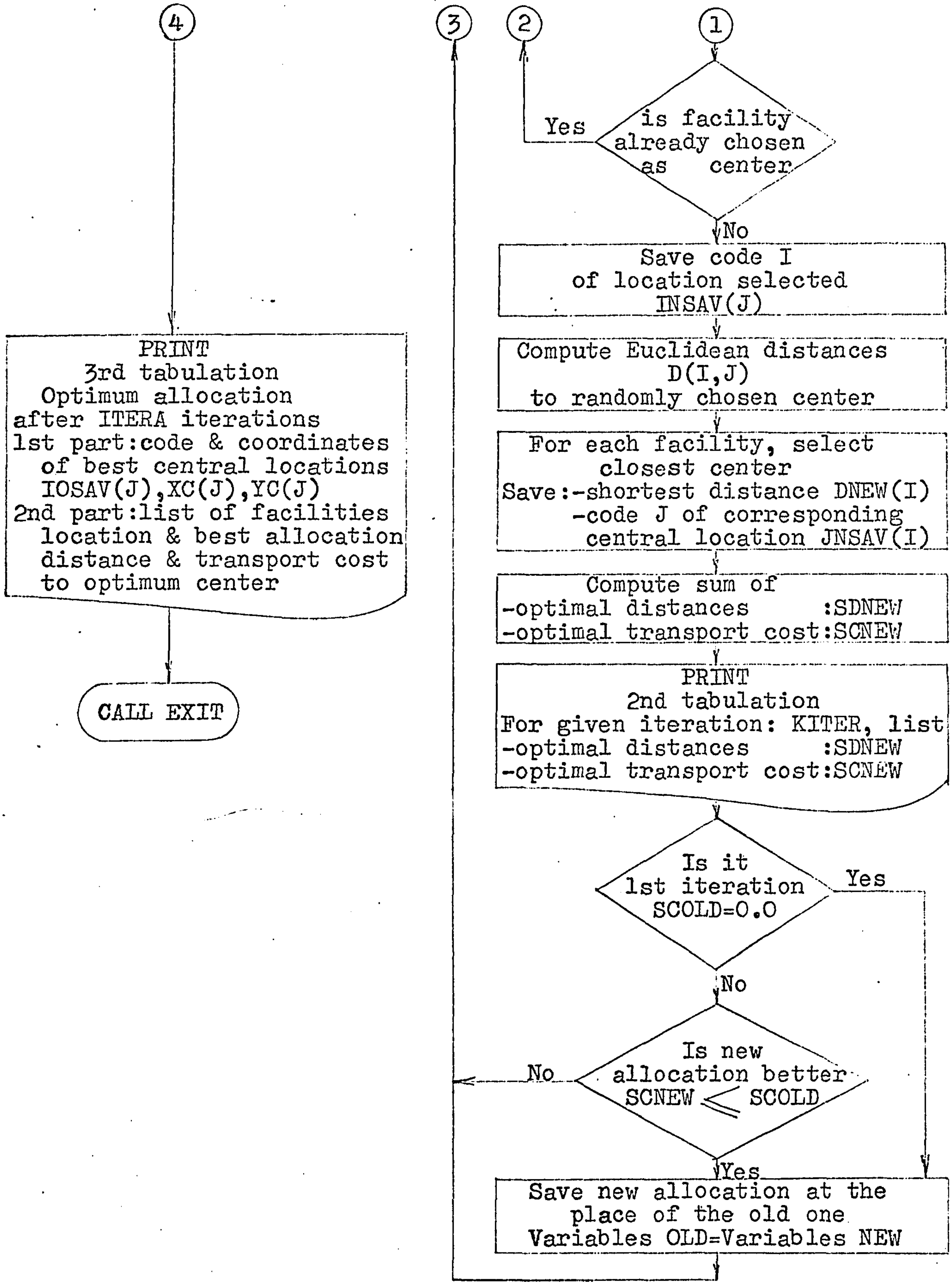
Fig. 40. Irrational Choice of Central Facilities

If for example, we consider the "rational" set of combination $\binom{20}{3}$ 1, 2, 3; 1,2,4; 1,2,5; etc. we can readily see from Fig. 40 that they are poor contenders for the title of optimum central locations. A possible random choice of these combinations could lead more rapidly to a better solution. To avoid duplication of computational effort we could input all possible combinations from a pack of nicely shuffled cards. However, if we can write all the combinations it means that the problem is small enough to easily allow an exhaustive computation of all the combinations in any order presented. When the system becomes too large, then a sampling procedure might be advantageous, the subset of m facilities being chosen at random through a Monte Carlo technique from the set of n facilities. For this particular subset, allocations can be determined and the corresponding sum of weighted distances evaluated. During the sampling procedure it is possible to keep in the computer memory the set of m facilities with the best characteristics. As with every Monte Carlo procedure it is of utmost importance to know when to stop the procedure so as to obtain an acceptable level of error. A simple criterion would be to stop after a given number of samplings. A more sophisticated method would be to look at the distribution of weighted distances by maintaining a running talley of mean, μ, and standard deviation, σ, and stop the sampling if the allocation falls below $\mu - x\sigma$, x being determined through experience.

The logic diagram of a possible computer program is given on page 112 . The program was applied to a set of 20 facilities as per Fig. 48, to be served by 3 central locations. The computer print-out and results are given from page 194 to page 200 and the 500 iterations were processed in 6 minutes 27 seconds on an IBM 360/40 computer. The program uses 1264 bytes. The total memory requirement varies with the size of tne problem; for example, the variables occupy 1286 bytes more when sampling 20 facilities with 5 central locations. The program was also applied to 4 large systems of 125 facilities each; the results for the first system are shown on figure 50, and the quantitative results for all are given on page 160.a.

Diagram 3. Random Destination Subset Algorithm




4
3
2
1
Yes
is facility already chosen as center
No
Save code I of location selected INSAV(J)
Compute Euclidean distances D(I,J) to randomly chosen center
For each facility, select closest center
Save:-shortest distance DNEW(I)
-code J of corresponding central location JNSAV(I)
Compute sum of
-optimal distances :SDNEW
-optimal transport cost:SCNEW
PRINT
2nd tabulation
For given iteration: KITER, list
-optimal distances :SDNEW
-optimal transport cost:SCNEW
Is it 1st iteration SCOLD=0.0
Yes
No
Is new allocation better SCNEW ≤ SCOLD
No
Yes
Save new allocation at the place of the old one
Variables OLD=Variables NEW
PRINT
3rd tabulation
Optimum allocation after ITERA iterations
1st part:code & coordinates of best central locations IOSAV(J),XC(J),YC(J)
2nd part:list of facilities location & best allocation distance & transport cost to optimum center
CALL EXIT

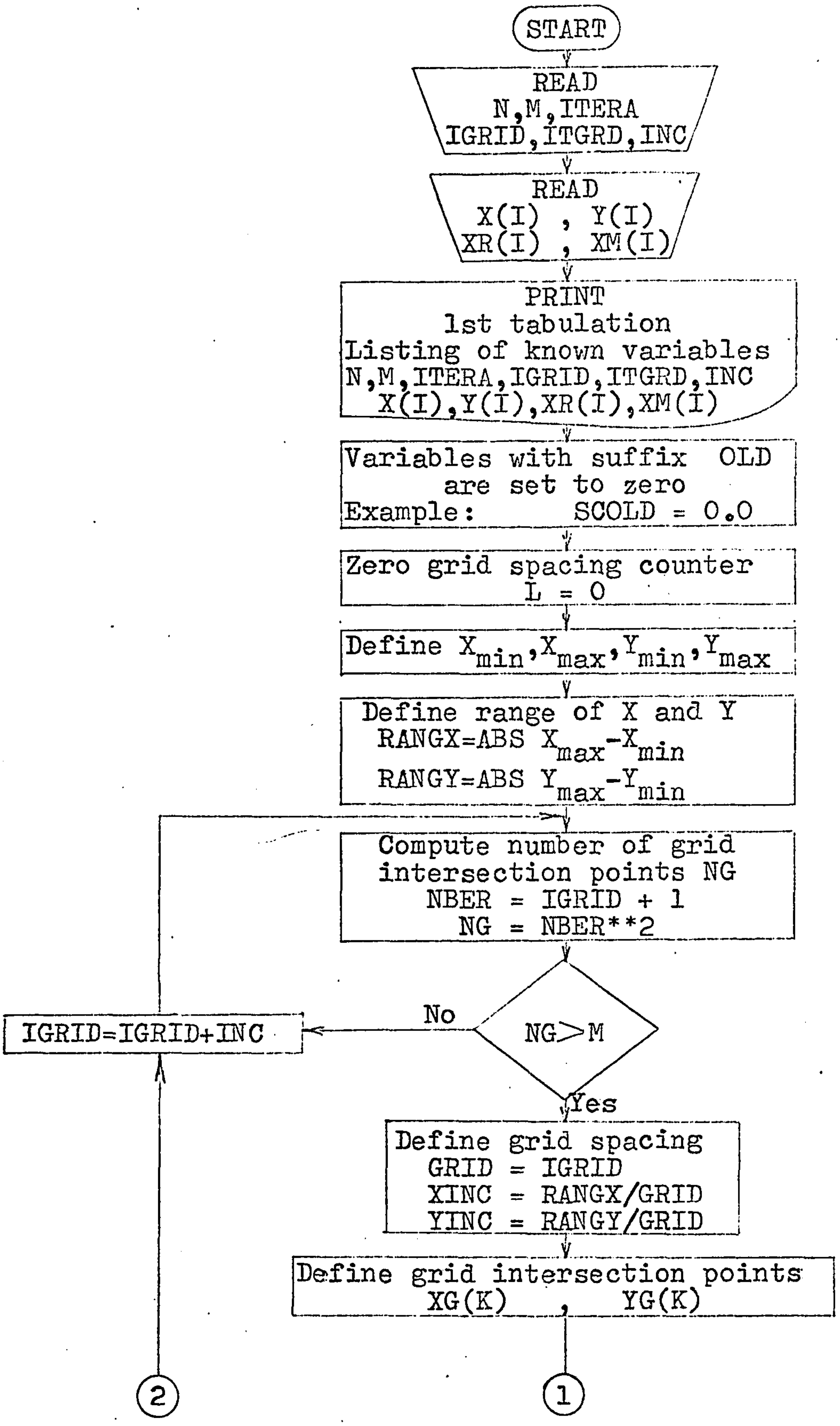
IV.2.6. Random Grid Location Algorithm

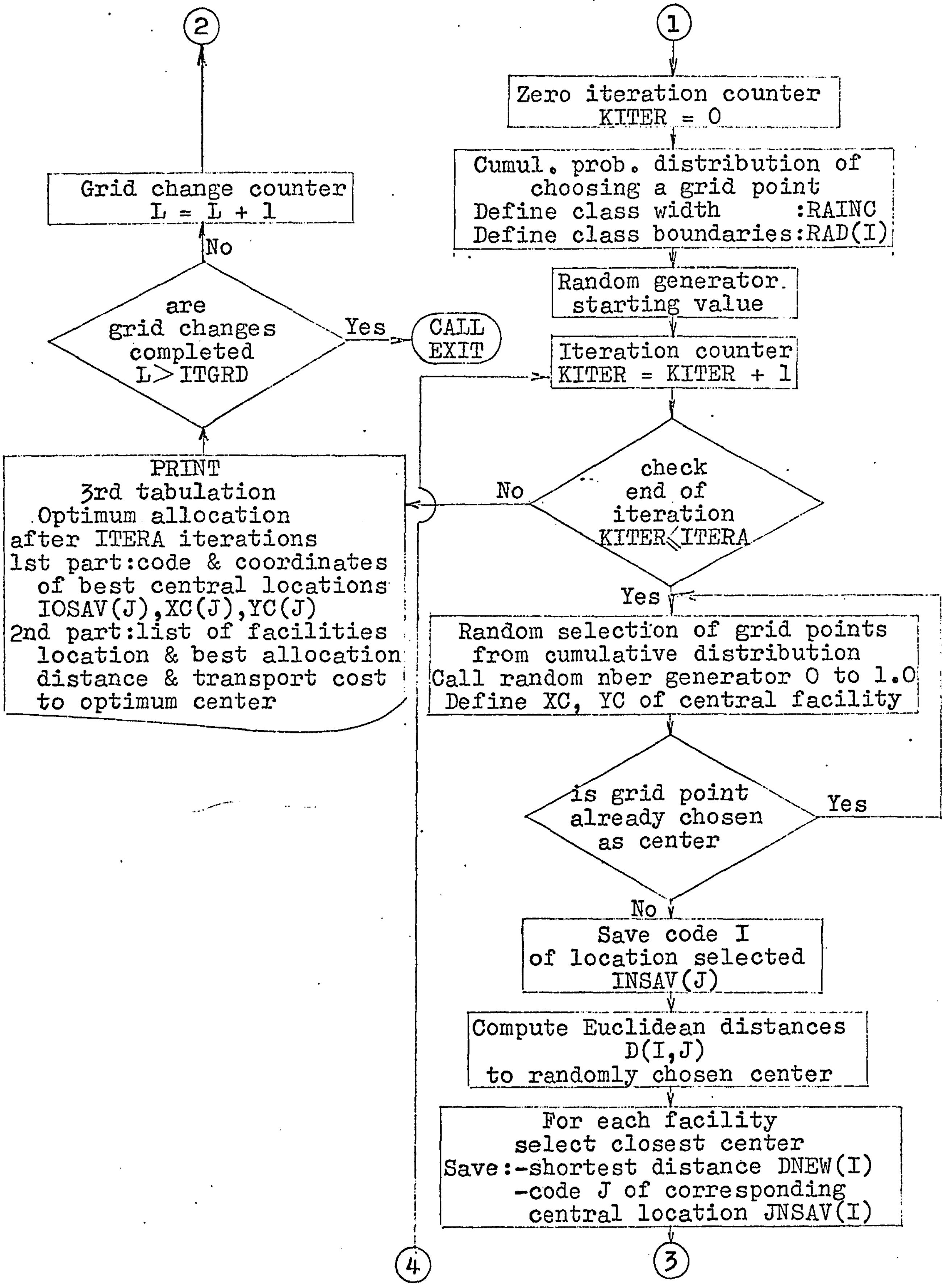
When we are using the random destination algorithm, we are sampling from a population of $\binom{n}{m}$ combinations. We have seen that this set might be enormous if n and m are ralatively large (17 trillions in the case of $\binom{100}{10}$). Consequently the results given by a relatively small amount of sampling may be very much suboptimal. To avoid this pitfall we could take a much larger sample but this goes against our intent to reduce our computational effort. We could also divide the facility space into a mesh with a spacing such that the number of mesh intersection points is much smaller than n. In this last case, we are sampling from a reduced population of N mesh intersecting points still covering the whole area to be investigated. We are assuming in these grid algorithms that the distribution of facilities in space is uniformly spread, which is frequently the case for large systems. This method however, would not be very efficient in the case of remote clusters.

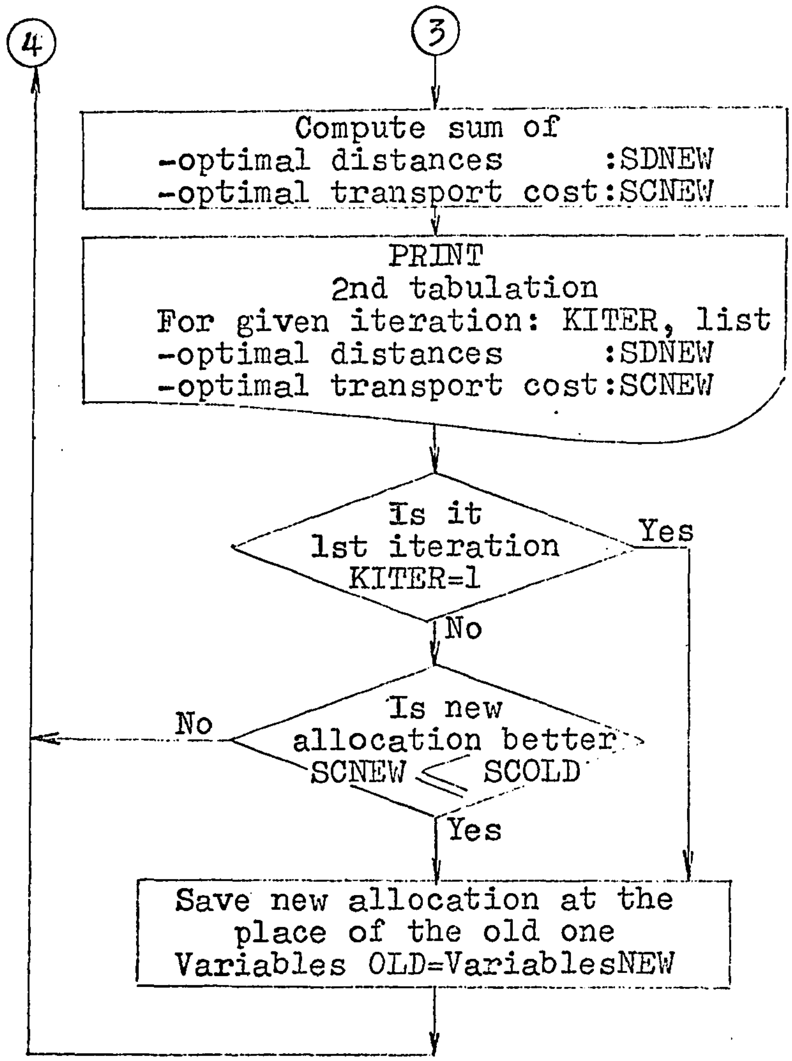
The logic diagram of a possible computer program is given on page 116. The program was applied to the same set of 20 facilities served by 3 central locations of algorithm IV.2.5. The computer print-out and results are given from page 201 to page 217 . Five possible grid spacings were investigated by dividing the circumscribed rectangle in 3, 4, 5, 6, then 7 divisions on each axis.

For each grid spacing 100 samplings were made. The corre- sponging 500 samplings were done in 6 minutes 33 seconds on an IBM 360/40 computer. The program uses 1694 bytes. The total memory requirement varies with the size of the problem, for example, the variables occupy 1494 bytes more when sampling 20 facilities with 5 central locations and 50 possible grid intersection points.

Diagram 4. Random Grid Location Algorithm




2
1
Zero iteration counter
KITER = 0
Grid change counter
L = L + 1
Cumul. prob. distribution of
choosing a grid point
Define class width :RAINC
Define class boundaries:RAD(I)
No
Random generator
starting value
are
grid changes
completed
L>ITGRD
Yes
CALL
EXIT
Iteration counter
KITER = KITER + 1
PRINT
3rd tabulation
Optimum allocation
after ITERA iterations
1st part:code & coordinates
of best central locations
IOSAV(J),XC(J),YC(J)
2nd part:list of facilities
location & best allocation
distance & transport cost
to optimum center
No
check
end of
iteration
KITER≤ITERA
Yes
Random selection of grid points
from cumulative distribution
Call random nber generator 0 to 1.0
Define XC, YC of central facility
is grid point
already chosen
as center
Yes
No
Save code I
of location selected
INSAV(J)
Compute Euclidean distances
D(I,J)
to randomly chosen center
For each facility
select closest center
Save:-shortest distance DNEW(I)
-code J of corresponding
central location JNSAV(I)
4
3


4
3
Compute sum of
-optimal distances :SDNEW
-optimal transport cost:SCNEW
PRINT
2nd tabulation
For given iteration: KITER, list
-optimal distances :SDNEW
-optimal transport cost:SCNEW
Is it
1st iteration
KITER=1
Yes
No
Is new
allocation better
SCNEW ≤ SCOLD
No
Yes
Save new allocation at the
place of the old one
Variables OLD=VariablesNEW

IV.2.7. Random Grid Algorithm with Linear Constraints

This algorithm is very similar to the random grid location algorithm. A mesh is defined to cover the facility space and the mesh intersection points selected at random as possible central facilities. To decrease further the number of combinations we exclude from our sampling process the mesh intersection points which fall outside the convex hull enclosing the facilities. Definition of the hull can be done by the methods described under paragraph IV.2.4.

The logic diagram of a possible computer program is given on page 121. The program was applied to the same set of 20 facilities served by 3 central locations of algorithm IV.2.5. The computer print-out and results are given from page 218 to page 234. Five possible grid spacings were investigated by dividing the circumscribed rectangle in 3, 4, 5, 6 then 7 divisions on each axis. The linear constraints are defined and during sampling each grid intersection is checked against these constraints, this intersection point is rejected if it violates any one of the constraints. For each grid spacing 100 samplings were made. The corresponding 500 samplings were done in 5 minutes 59 seconds on an IBM 360/40 computer. The program uses 2052 bytes. The total memory requirement varies with the size of the problem; for example, the variables occupy 1742 bytes more when sampling 20 facilities with 5 central locations and 50 possible grid intersection points.

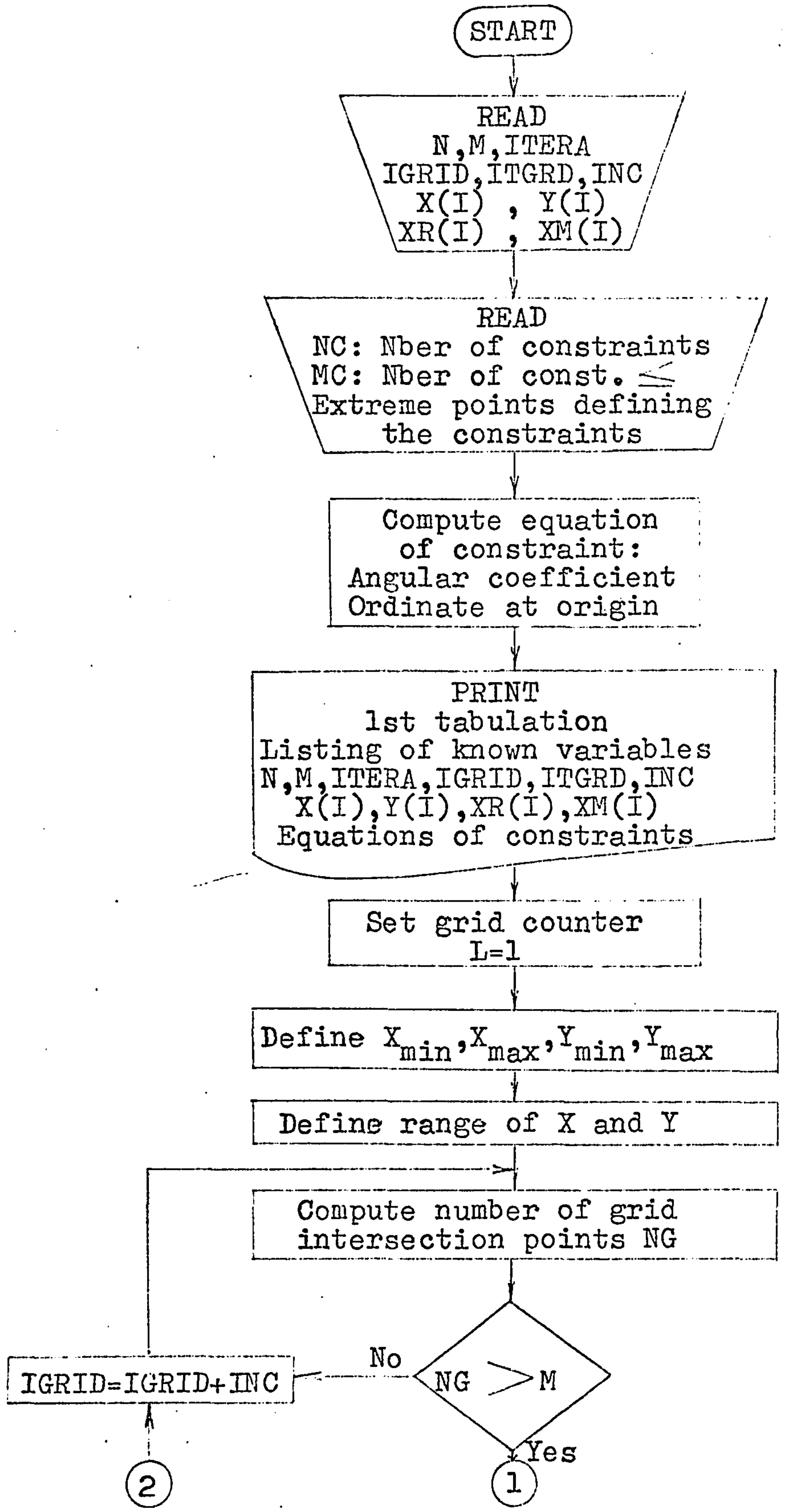
Every point (x,y) of the linear constraint passing through the extreme points (x_i, y_i) and (x_j, y_j), is subjected to the equation:

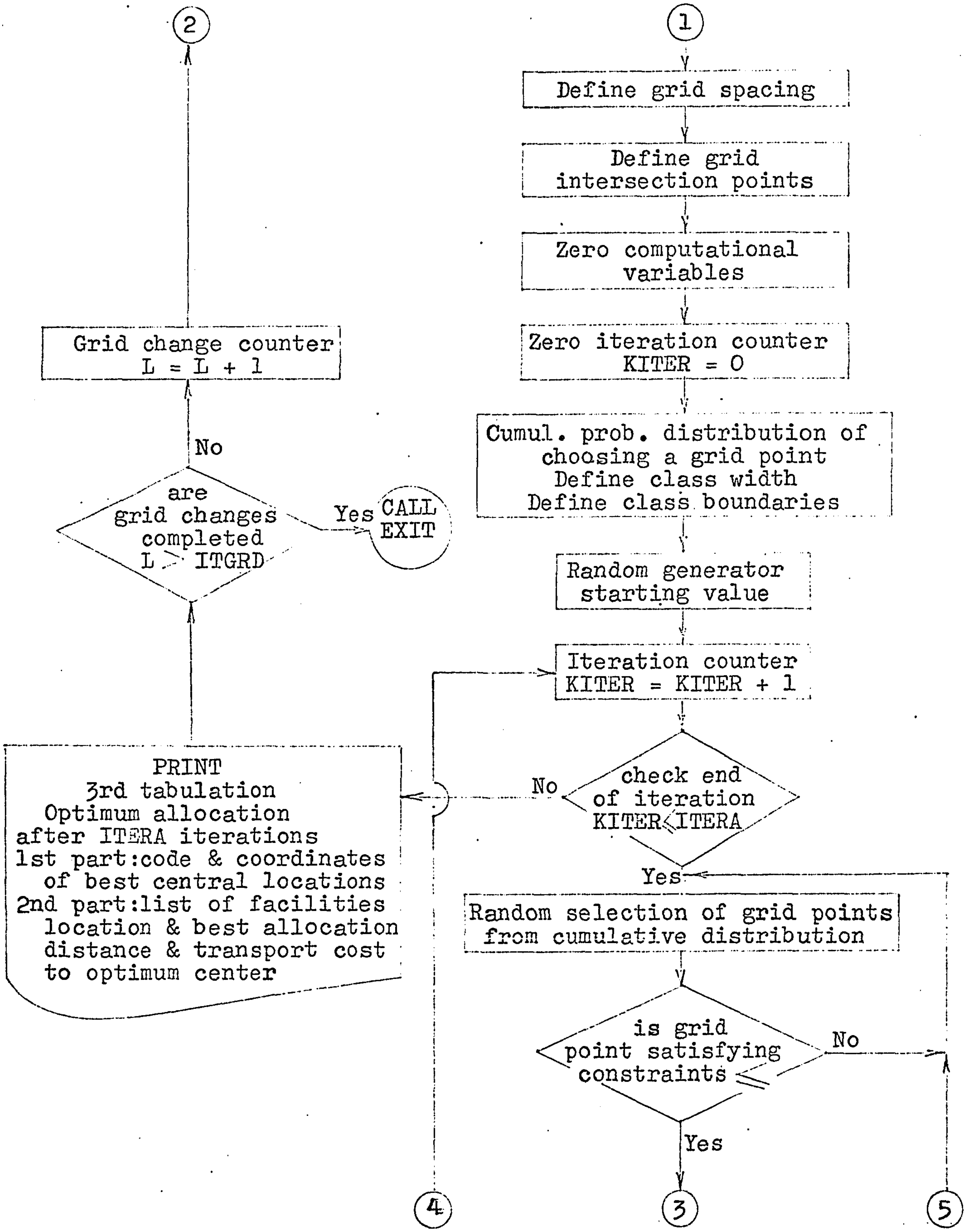
$$\frac{y - y_i}{y_i - y_j} = \frac{x - x_i}{x_i - x_j}$$

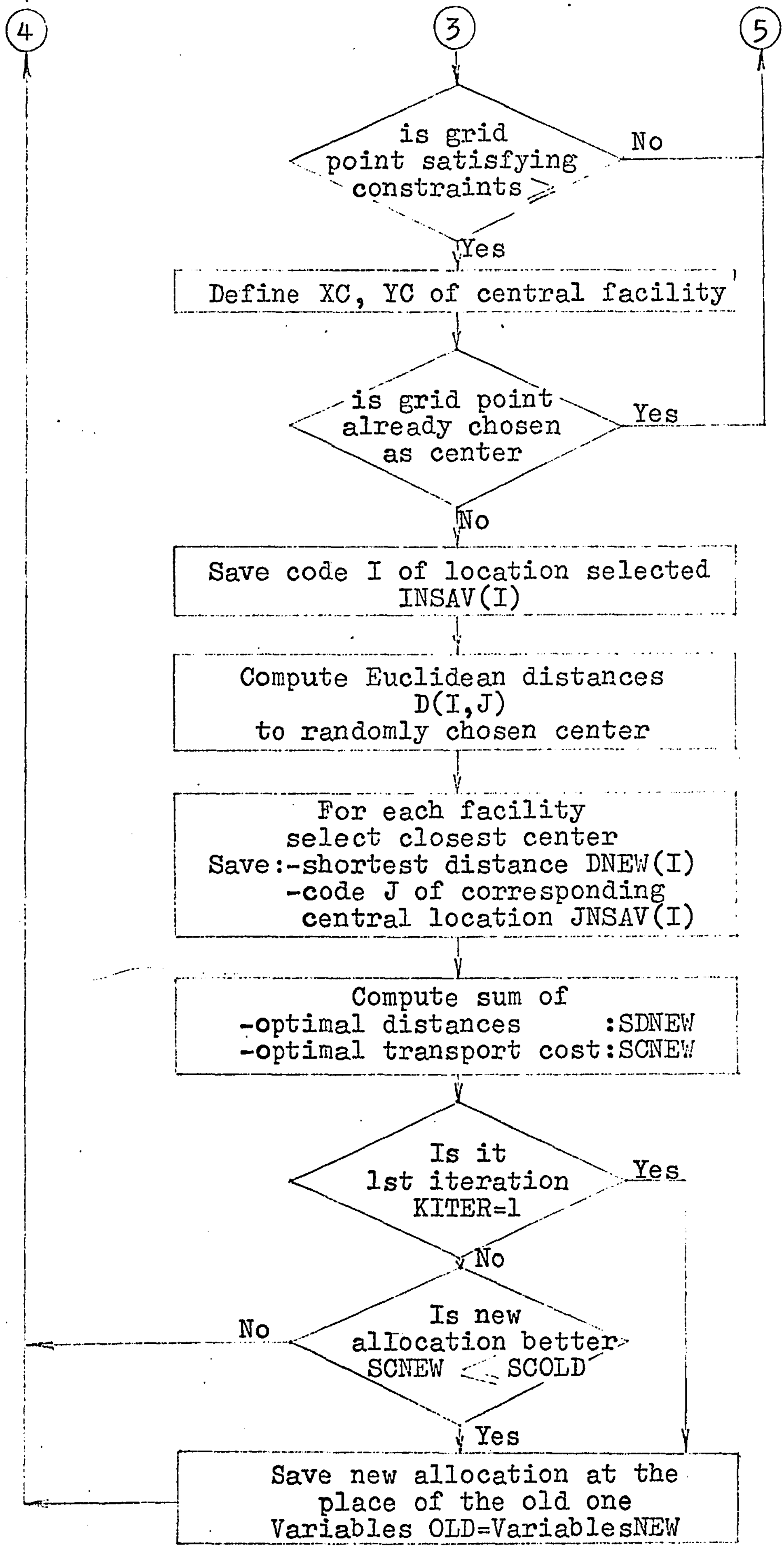
A part of the data input includes the number of linear constraints (NC), number relatively small even for large sets, and the extreme points defining each constraint. The constraints are grouped in two classes: first the one "smaller than or equal to" (MC of them), then the one "larger than or equal to". The program automatically defines the elements of the linear equation: angular coefficient and ordinate at origin, and rejects the random grid points violating any of the constraints.

The program was also applied to 4 large systems of 125 facilities each; the results for the first system are shown on figure 58 and the quantitative results for all are given on page 160.a.

Diagram 5. Random Grid Algorithm with Linear Constraints




2
1
Define grid spacing
Define grid intersection points
Zero computational variables
Grid change counter L = L + 1
Zero iteration counter KITER = 0
No
Cumul. prob. distribution of choosing a grid point
Define class width
Define class boundaries
are grid changes completed L > ITGRD
Yes
CALL EXIT
Random generator starting value
Iteration counter KITER = KITER + 1
PRINT
3rd tabulation
Optimum allocation after ITERA iterations
1st part:code & coordinates of best central locations
2nd part:list of facilities location & best allocation distance & transport cost to optimum center
No
check end of iteration KITER ≤ ITERA
Yes
Random selection of grid points from cumulative distribution
is grid point satisfying constraints
No
Yes
4
3
5


4
3
5
is grid point satisfying constraints ≥
No
Yes
Define XC, YC of central facility
is grid point already chosen as center
Yes
No
Save code I of location selected INSAV(I)
Compute Euclidean distances D(I,J) to randomly chosen center
For each facility select closest center
Save:-shortest distance DNEW(I)
-code J of corresponding central location JNSAV(I)
Compute sum of
-optimal distances :SDNEW
-optimal transport cost:SCNEW
Is it 1st iteration KITER=1
Yes
No
Is new allocation better SCNEW ≤ SCOLD
No
Yes
Save new allocation at the place of the old one
Variables OLD=VariablesNEW

IV.2.8. Successive Approximation Algorithm [10]

The complexity of the locational problem increases with n but even more drastically with m. It is a relatively small problem to find the optimum allocation when $m = 2$ because of the limited number of combinations $\binom{n}{2}$. If the allocation were found for such a set we could then try to introduce a third center by placing it for example, at one of the facilities. Then we would have to test each facility to see if it could not be better served from the new central one, and re-allocate accordingly. The process can then be carried up to m centers.

In this method the problem is to choose adequately each new center. In some practical problems the subjective choice may be sufficient to lead to the optimum allocation, however, in most cases it is difficult to pick good contenders and the resulting allocation may be suboptimal. After the 1st approximation we still must consider $(n - 2)$ facilities as possible centers, and this number may be quite large.

IV.2.9. Grid Successive Approximation Algorithm

In a like manner to the successive approximation algorithm of paragraph IV.2.8, we choose to select two grid intersection points as central facilities, and once the best allocations are defined a third grid point is introduced for an improved re-allocation. The process is carried until m

sets of allocations are defined, then each subset may be processed with the algorithm IV.1.1. to find the exact center locations.

In this case we are limiting our investigation to a set of grid points N which are directly under our control. Once 2 optimum grid points are defined, we can more easily investigate all the remaining grid points as possible third center. The program can also be devised as to reject grid points violating the linear constraints of the convex hull.

IV.2.10. Alternate Location - Allocation Algorithm

The set of n locations is divided into m sets of approximately identical number of points, and for each of these subsets the best central location is defined. Each point is then tested to see if it is closer to its central facility than the neighboring one. If the neighboring one is closer, new subsets are defined and new central locations computed. The process is combined until further improvement is not possible.

The method is sometimes known as the ALA Algorithm and was suggested by both Cooper [10], Vergin and Rogers [52]. This algorithm has been applied to some practical problems, for example, it was used by Devine and Lesso [17] in the optimization of offshore oil fields.

IV.2.11. Variable Discrimination Algorithm

When we look at a very large system of facilities we might consider reducing our power of discrimination. In so doing, we are assuming that in a set of facilities the locations are so closely located that they may be considered as a single entity.

This set of n_k facilities will have a corresponding mean weight

$$w_k = \frac{1}{n_k} \sum_{i=1}^{n_k} r_i \, m_i$$

acting at a point

$$x_{(n_k)} = \frac{\sum_{i=1}^{n_k} r_i \, m_i \, x_i}{\sum_{i=1}^{n_k} r_i \, m_i}$$

$$y_{(n_k)} = \frac{\sum_{i=1}^{n_k} r_i \, m_i \, y_i}{\sum_{i=1}^{n_k} r_i \, m_i}$$

If the system is very large we might use a very small power of discrimination and therefore include large clusters of facilities into the same set. If on the other hand the problem is relatively small we might increase our power of discrimination and include only very few facilities in each set.

Once all the sets $n_1, n_2, \cdots, n_k$ are defined, we may apply any one of the previous algorithms to solve the locational problem. The only purpose of our method is to condense the total number of facilities into a smaller set.

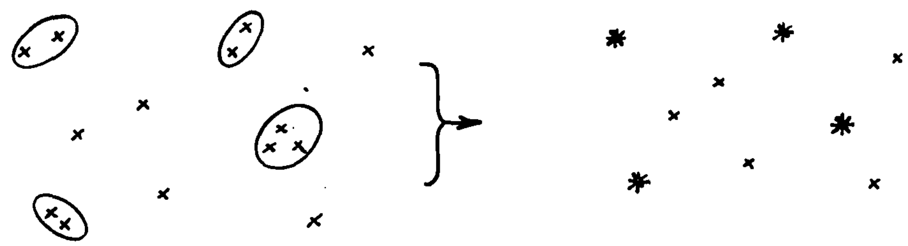


Fig. 42. Variable Discrimination Algorithm

We will assume that each facility is surrounded by a blurry area, if another facility is present in this area it is not recognized as separate. If two or more facilities have intersecting blurred areas they will be considered as a single one.

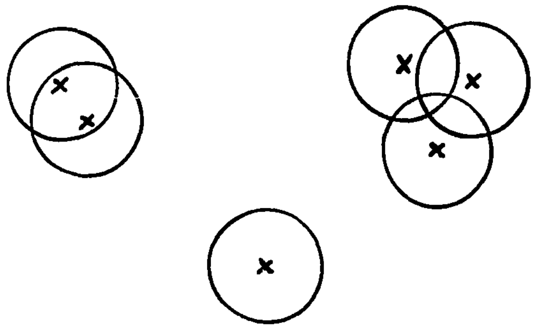


Fig. 43. Variable Discrimination Algorithm
Blurred Areas

We have direct control over the size of the blurr and accordingly over the size of the problem.

In this process we make the assumption that closely located facilities will finally depend on the same center. In theory it might be wrong (see Fig. 44.), but in practice it is highly improbable that closely located facilities will depend on different centers, if only for avoiding disruption created by discontinuity of management or operation.

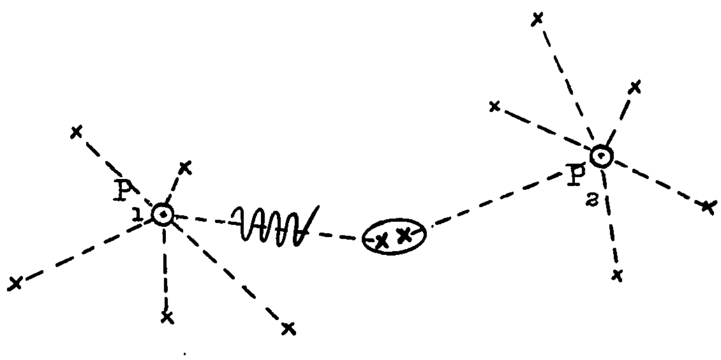


Fig. 44. Variable Discrimination Algorithm
Underlying Assumption

It is to be noted that we would like to discriminate more sharply the facilities with heavy weight and more loosely the facilities with little weight; the blurred area should then be inversely proportional to the facility weight.

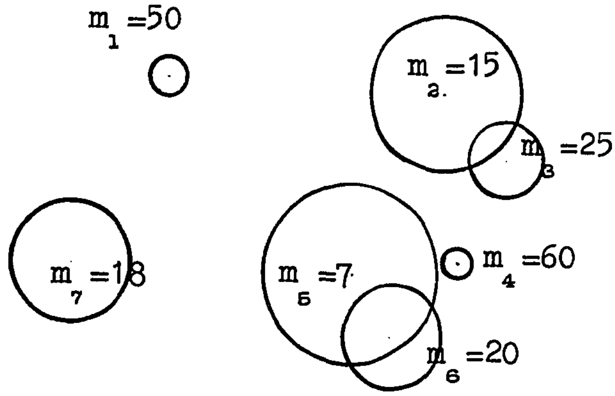


Fig. 45. The Discriminating Power is Proportional to the Respective Weights

In practice the problem is to define the subsets $n_1, n_2, \ldots, n_k$.

In a large system it is highly improbable that the plot will be found to be of much help as every facility must be scrutinized independently, we must then be able to automatically define these subsets. Two approaches are possible. For each facility we may compute all the corresponding Euclidean distances to all other facilities and select the shortest one within a given value 2ε, this would amount to computing $\frac{n(n-1)}{2}$ distances. We may also order the abcissae of our facilities and check the one with corresponding x and y falling less than 2ε apart; in this particular case the blurred areas would be assimilated to Squares.

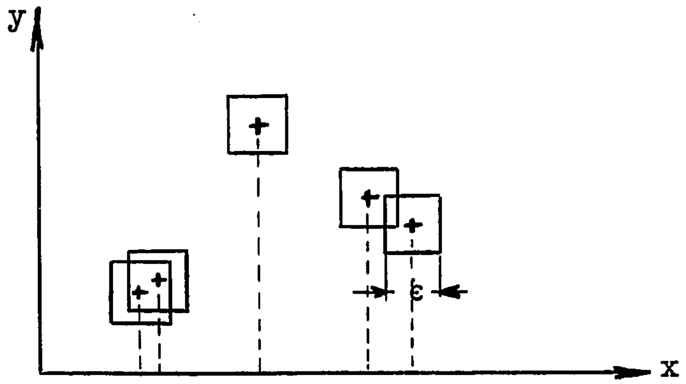
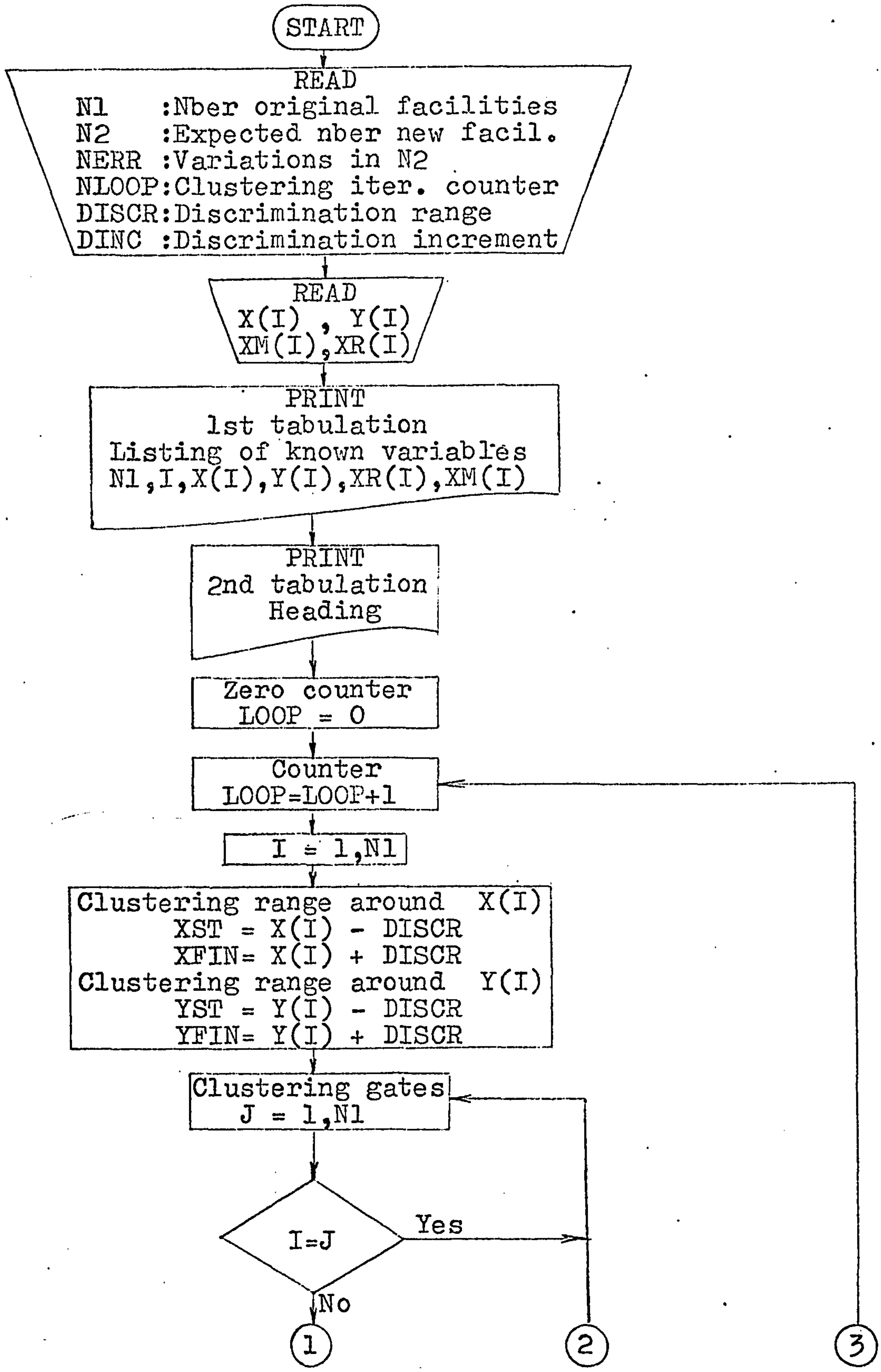


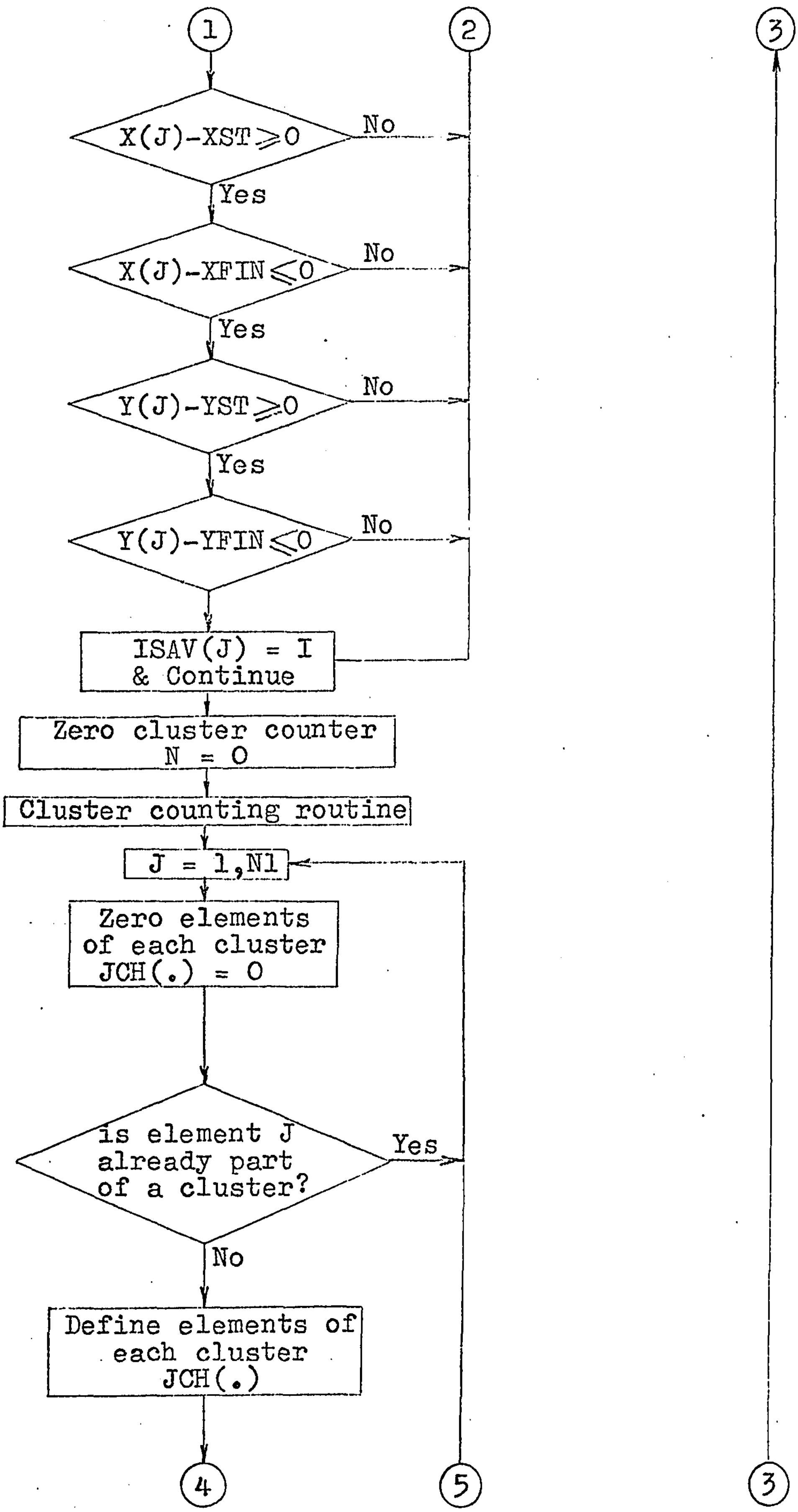
Fig. 46. Variable Discrimination Algorithm
Agglomeration by Closeness of Coordinates

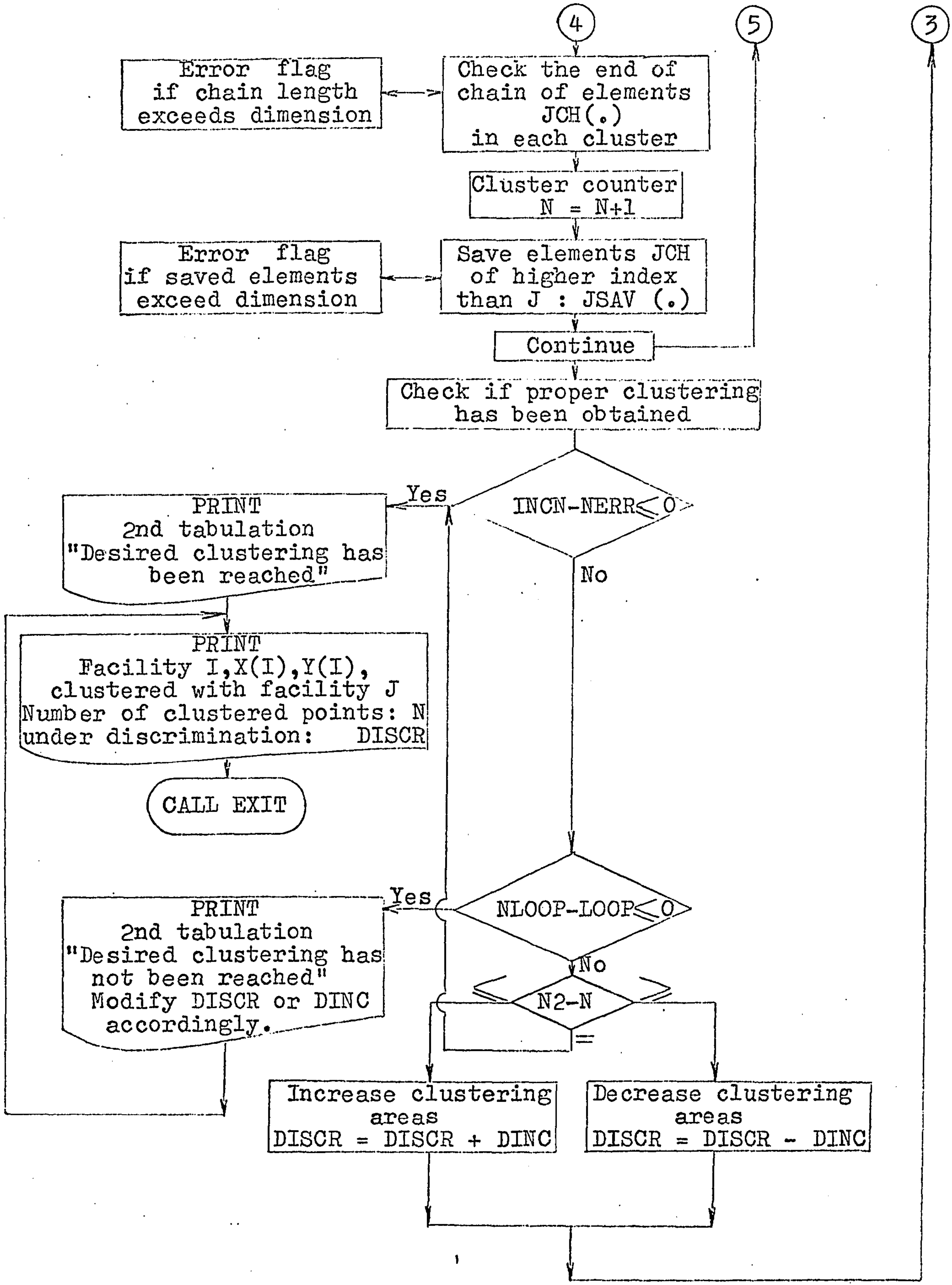
The value of ε depends on the size of the set we want finally to handle with the previous algorithms. The computer program could be built in such a manner that we would define the size of the desirable set, a given ε would be tested and increased if the agglomerated set was found to be too large, or decreased in the other case.

With this algorithm we can reduce a very large set to a smaller one easier to handle and although the results of the locational problem may be approximate, we can deal with any size problem and we will not be stopped by the computer memory or computational time. In some cases the problems may be so large that it is not even possible to store all the facilities coordinates and weights in the computer memory space, it is then necessary to partition the problem and agglomerate the sets successively in each partition.

Diagram 6. Variable Discrimination Algorithm




1
2
3
X(J)-XST⩾0
No
Yes
X(J)-XFIN⩽0
No
Yes
Y(J)-YST⩾0
No
Yes
Y(J)-YFIN⩽0
No
ISAV(J) = I
& Continue
Zero cluster counter
N = 0
Cluster counting routine
J = 1,N1
Zero elements
of each cluster
JCH(.) = 0
is element J
already part
of a cluster?
Yes
No
Define elements of
each cluster
JCH(.)
4
5
3


4
5
3
Error flag if chain length exceeds dimension
Check the end of chain of elements JCH(.) in each cluster
Cluster counter N = N+1
Error flag if saved elements exceed dimension
Save elements JCH of higher index than J : JSAV (.)
Continue
Check if proper clustering has been obtained
Yes
INCN-NERR≤0
No
PRINT 2nd tabulation "Desired clustering has been reached"
PRINT Facility I,X(I),Y(I), clustered with facility J Number of clustered points: N under discrimination: DISCR
CALL EXIT
Yes
NLOOP-LOOP≤0
No
PRINT 2nd tabulation "Desired clustering has not been reached" Modify DISCR or DINC accordingly.
<
N2-N
>
=
Increase clustering areas DISCR = DISCR + DINC
Decrease clustering areas DISCR = DISCR - DINC

CHAPTER V

ANALYSIS OF RESULTS

Some of the previous algorithms have been extensively studied by Leon Cooper and applied to 100 locational problems with 60 facilities and 4 central locations. In one of his research papers [12] he gives the following percent deviation from the lower bound (ref: II.3.5, III.2.2).

Table 5. Errors in Algorithms
(Applied to 100 Problems, $n = 60$, $m = 4$)

	Mean Percent Error	Mean Percent Deviation from Lower Bound
Destination Subset	0.948	360.1
Random-Destination	2.518	367.2
Successive-Approximation	7.086	387.9
Alternate Location-Allocation	2.582	367.7

In judging these methods we should not be comparing only the minimum sum of distances that they give, but mainly how fast they offer the correct set of satellites, as each of these sets gives the exact solution very rapidly and with little memory requirement by applying algorithm IV.1.1. With these criteria, the destination subset algorithm with the lowest mean percentage error is one of the poorest, while the variable grid algorithm is much better.

We should also look at the memory requirement and computing time, above 50 facilities the destination subset algorithm is not practical, above a few hundred facilities the random destination and random grid may reach the limit of memory space (around 1,000 bytes of memory is required for each facility with the random grid algorithm and linear constraints in the search for 5 central locations). For larger systems of many thousand facilities the use of the discrimination algorithm is a necessity, it may rank poorly in mean percent error but it is the only method now available which can solve such a system.

The valuable grid algorithms and discrimination algorithms have been mainly devised for large systems. However, to compare with existing algorithms they were tested for a set of 20 facilities and 3 central locations (Fig. 48). Then to compare the results in the case of a larger system it was applied to 125 facilities served by 3 centers. Because of computer time limitations it was not possible to make a more exhaustive study.

V.1. RANDOM DESTINATION ALGORITHM

Distribution of Distances

V.1.1. Small System: $n = 20$, $m = 3$

500 Samples

Range

$98.009 - 43.927 = 54.082$

Number of classes

Sturges rule: $k = 1 + 3.3 \log_{10} N$

k: number of classes to use

N: total number of data

$k = 1 + 3.3 \times 2.69897 = 1 + 8.9 \cong 10$ classes

Class interval

$54.082/10 = 5.4082$

Table 6. Random Destination Algorithm

Distribution of Distances

Small System

	Class Boundaries	Frequency
1	43.927 - 49.335	53
2	49.336 - 54.743	154
3	54.744 - 60.151	97
4	60.152 - 65.559	80
5	65.560 - 70.968	56
6	70.969 - 76.376	26
7	76.377 - 81.784	17
8	81.785 - 87.192	6
9	87.193 - 92.600	5
10	92.601 - 98.009	6
		500

Mean: 59.629

Standard Deviation: 9.928

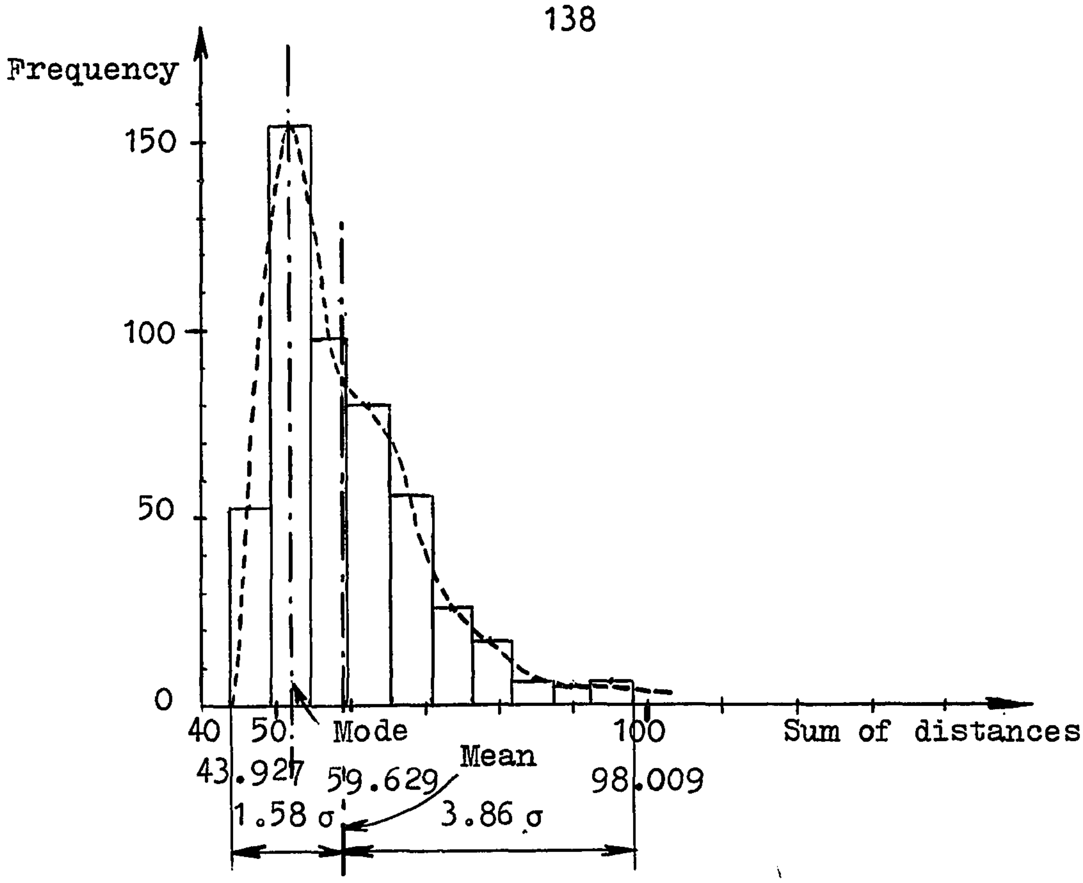


Fig. 47. Random Destination Algorithm

Distribution of Distances

Small System

The distribution of distances has a positive skewness, the optimum assignment lies at 1.58 σ from the mean, while poor assignments extend up to 3.86 σ from the mean.

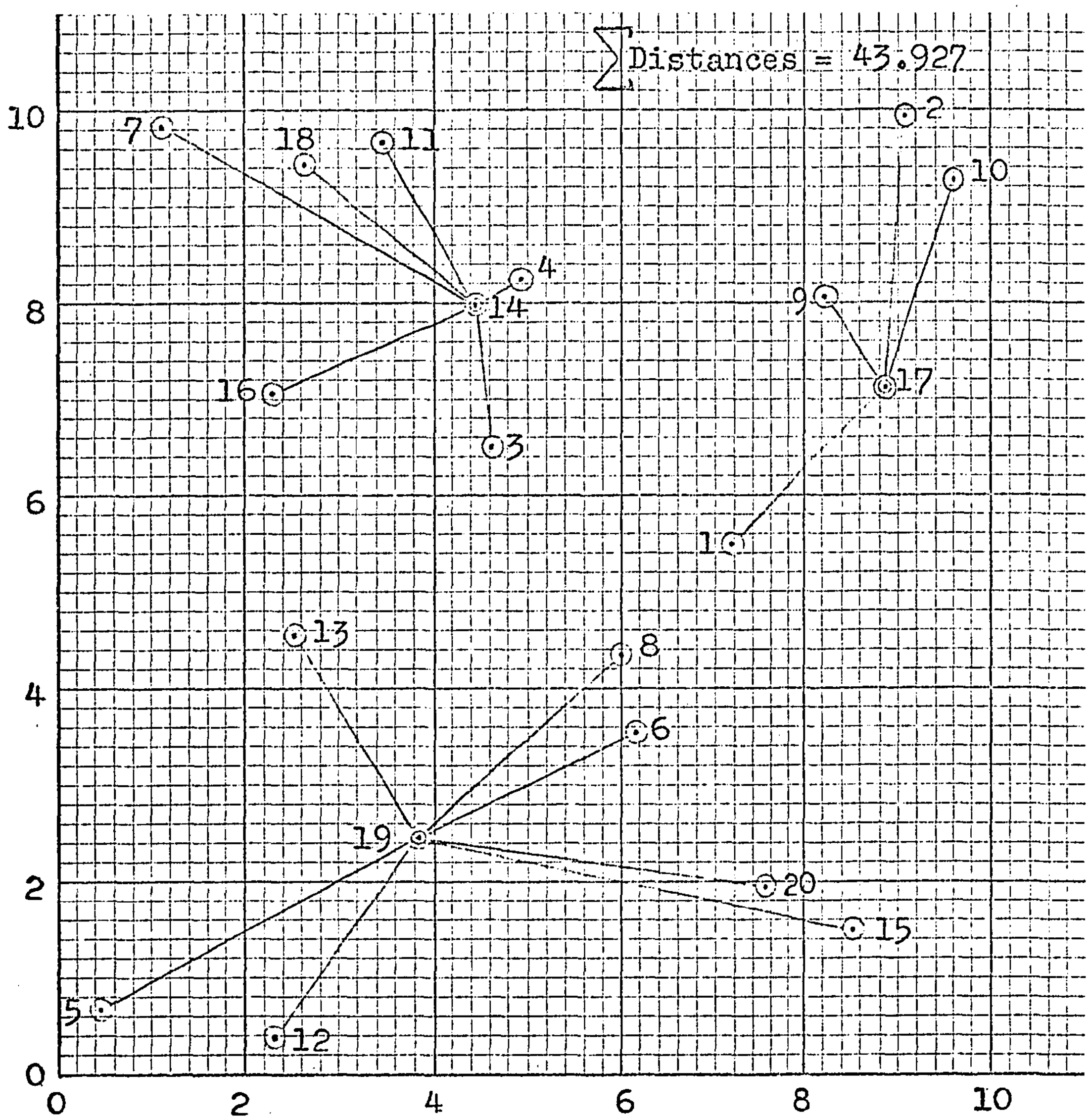


Fig. 48 Random destination algorithm
N = 20 , M = 3

V.1.2. Large System: $n = 125$, $m = 3$

25 Samples

Range

$$6186.930 - 3323.594 = 2863.336$$

Number of classes

$$k = 1 + 3.3 \log_{10} 25 = 1 + 3.3 \times 1.39794 = 5.6 \cong 6$$

Class Interval

$$2863.336/6 = 477.2226$$

Table 7. Random Destination Algorithm
Distribution of Distances
Large System

	Class Boundaries	Frequency
1	3323.594 - 3800.816	6
2	3800.817 - 4278.039	7
3	4278.040 - 4755.261	7
4	4755.262 - 5232.484	3
5	5232.485 - 5709.707	0
6	5709.708 - 6186.930	2
		25

Mean: 4331.586

Standard Deviation: 750.026

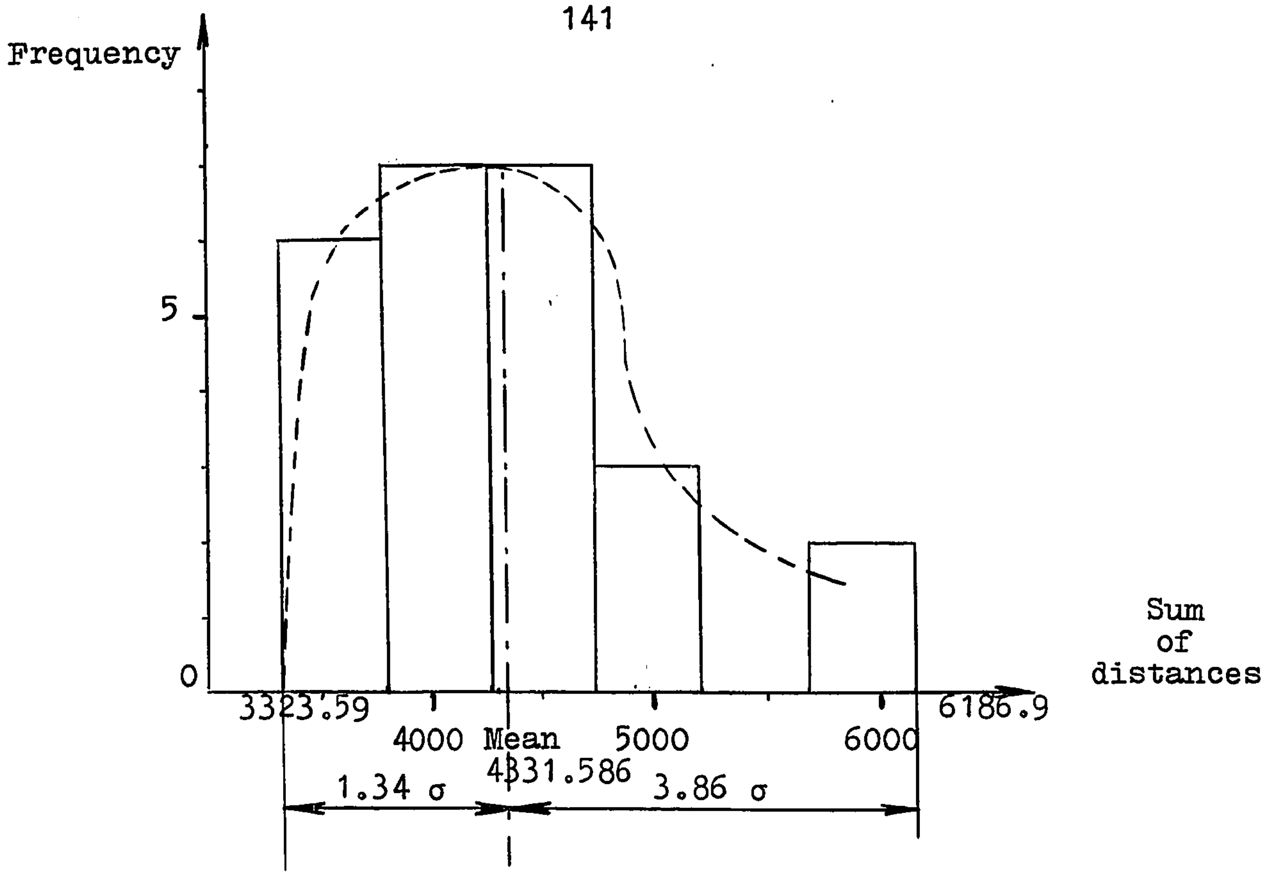


Fig. 49. Random Destination Algorithm
Distribution of Distances
Large System

The distribution of distances has a positive skewness, the optimum assignment lies at 1.34 σ from the mean, while poor assignments extend up to 3.86 σ from the mean.

The 25 iterations were made in 4 minutes 31 seconds on an IBM 360/40 computer.

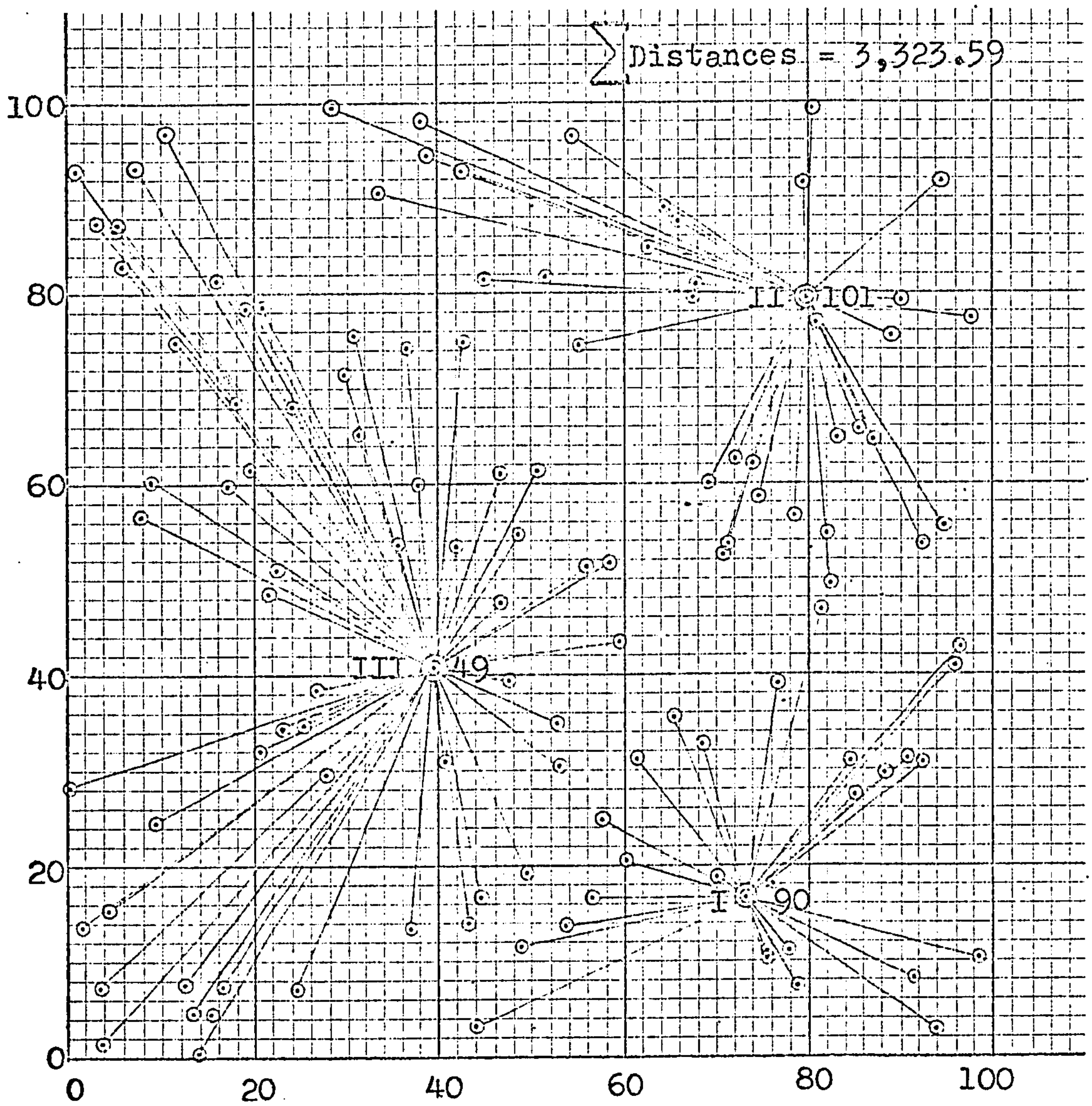


Fig. 50 Random destination algorithm
N = 125 , M = 3

V.2. RANDOM GRID LOCATION ALGORITHM DISTRIBUTION OF DISTANCES

V.2.1. Small System: $n = 20$, $m = 3$

100 Samples per Grid Division

5 Grid Divisions from 3 to 7

Number of classes

$$k = 1 + 3.3 \log_{10} 100 = 1 + 3.3 \times 2 \cong 8 \text{ classes}$$

Table 8. Random Grid Location Algorithm

Ranges and Class Intervals

Grid Divisions	Range	Class Interval
3	116.045 - 57.769 = 63.276	63.276/8 = 7.9095
4	104.580 - 53.128 = 51.452	51.452/8 = 6.4315
5	108.658 - 50.847 = 57.811	57.811/8 = 7.2263
6	102.296 - 52.866 = 49.430	49.430/8 = 6.1787
7	119.858 - 61.436 = 58.422	58.422/8 = 7.3027

Total distribution

Number of classes: $k = 1 + 3.3 \log_{10} 500 \cong 10$

Range: 119.858 = 50.847 = 69.011

Class interval: 69.011/10 = 6.9011

Table 9. Random Grid Location Algorithm
Distribution of Distances
Small System

Classes	Grid Division 3: Class Boundaries	Freq.	Grid Division 4: Class Boundaries	Freq.	Grid Division 5: Class Boundaries	Freq.
1	52.769- 60.678	22	53.128- 59.559	14	50.847- 58.073	5
2	60.679- 68.588	32	59.560- 65.991	21	58.074- 65.299	18
3	68.589- 76.497	16	65.992- 72.422	22	65.300- 72.526	28
4	76.498- 84.407	8	72.423- 78.854	19	72.527- 79.752	20
5	84.408- 92.316	7	78.855- 85.285	11	79.753- 86.978	19
6	92.317-100.226	1	85.286-91.717	8	86.979- 94.205	5
7	100.227-108.135	2	91.718- 98.148	3	94.206-101.431	4
8	108.136-116.045	2	98.149-104.580	2	101.432-108.658	1
	Mean: 70.727		Mean: 71.946		Mean: 73.414	
	Std. Dev.: 12.781		Std. Dev.: 10.920		Std. Dev.: 11.347	

Table 9. Random Grid Location Algorithm
Distribution of Distances
Small System

Classes	Grid Division 6 Class Boundaries	Freq.	Grid Division 7 Class Boundaries	Freq.	Total Distritution Class Boundaries	Freq.
1	52.866- 59.044	13	61.436- 68.738	21	50.847- 57.748	36
2	59.045- 65.223	24	68.739- 76.041	37	57.759- 64.649	87
3	65.224- 71.402	31	76.042- 83.344	27	64.650- 71.550	136
4	71.403- 77.581	27	83.345- 90.647	5	71.551- 78.451	122
5	77.582- 83.759	1	90.648- 97.949	1	78.452- 85.352	62
6	83.760- 89.938	0	97.950-105.252	3	85.353- 92.253	27
7	89.939- 96-117	3	105.253.112.555	2	92.254- 99.154	12
8	96.118-102.296	1	112.556-119.858	4	99.155-106.055	10
9					106.056.112.956	3
10					112.957-119.858	5
	Mean: 67.792 Std. Dev.: 8.233		Mean: 78.153 Std. Dev.:12.108		Mean: 72.405 Std. Dev.:11.655	

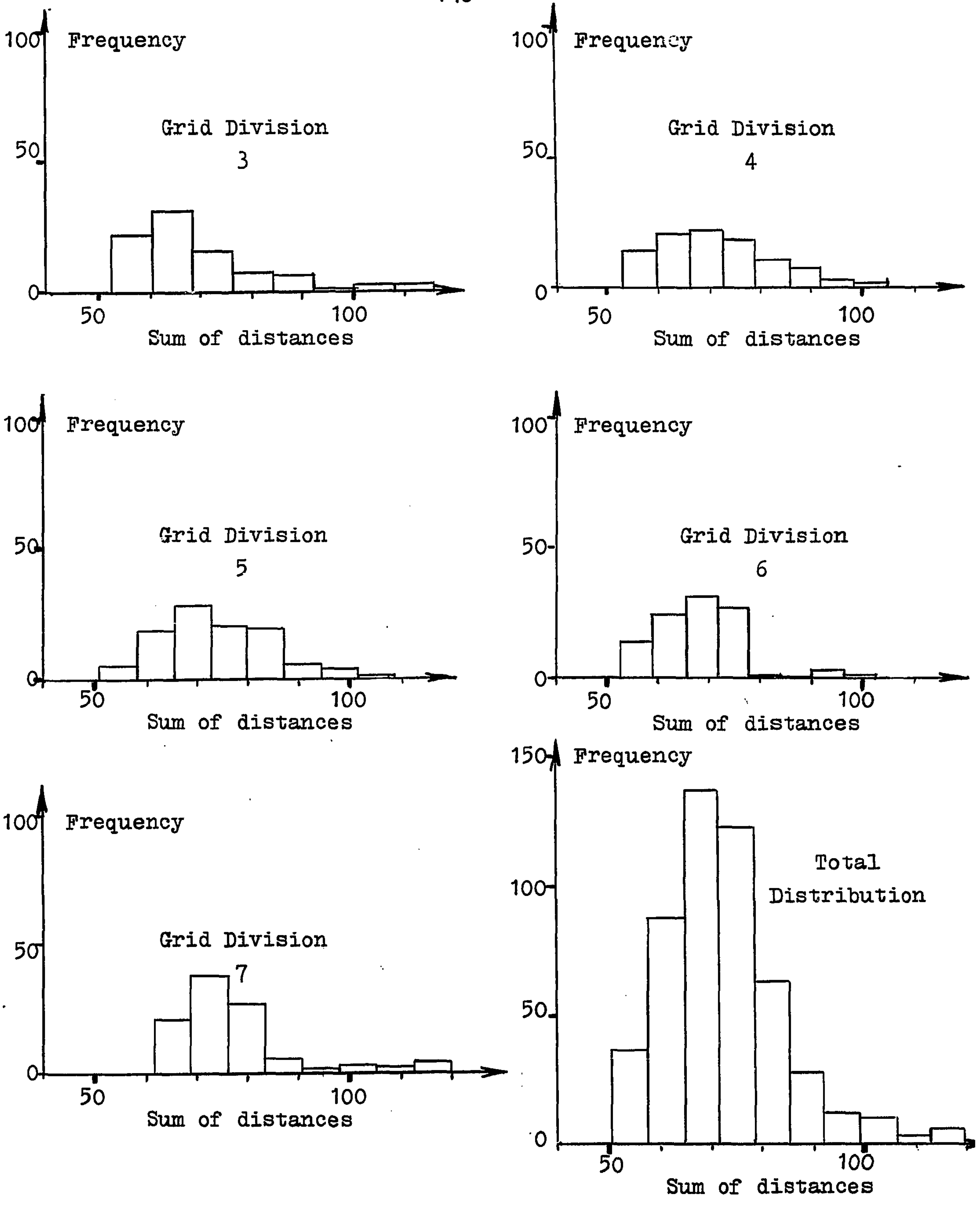


Fig. 51. Random Grid Location Algorithm
Distribution of Distances - Small System

In this particular layout of facilities, whatever the grid division may be, the mean sum of distances, the standard deviation and the minimum sum are all larger than the ones found with the random destination algorithm. It took 43 samplings to reach the minimum with the random destination method. With the random grid method only the division in 3 offered a total number of grid points less than 20 and, in this particular case, the minimum distance was reached in 2 samplings. The other grid division required an average of 47 samplings before reaching optimum. This single problem is not enough to bear any definite judgment on the speed of the method except that it consistently gives larger error than the random destination algorithm; we must then consider that the random grid algorithm is inadequate for a small number of facilities. We will show that for a larger set this method becomes more and more efficient.

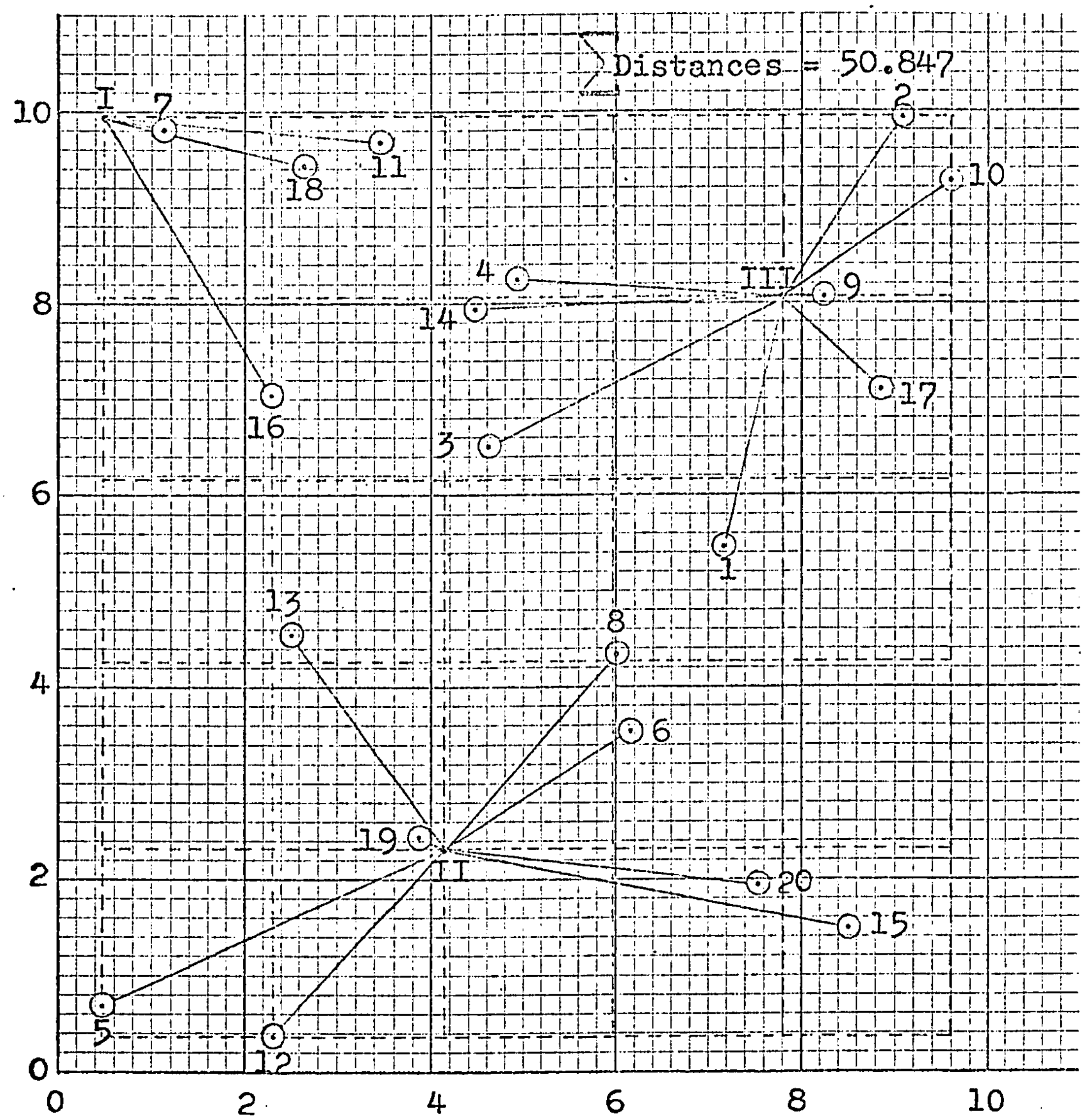


Fig. 52 Random grid location algorithm
N = 20 , M = 3

V.2.2. Large System: $n = 125$, $m = 3$

25 Samples per Grid Divisions

One Grid Division in 5

Range

$$6064.238 - 3387.996 = 2676.242$$

Number of classes

$$k = 1 + 3.3 \log_{10} 25 \cong 6$$

Class interval

$$2676.242/6 = 446.0403$$

Table 10. Random Grid Location Algorithm
Distribution of Distances
Large System

	Class Boundaries	Frequency
1	3387.996 - 3834.036	6
2	3834.037 - 4280.076	6
3	4280.077 - 4726.116	4
4	4726.117 - 5172.157	3
5	5172.158 - 5618.197	5
6	5618.198 - 6064.238	1
		25

Mean: 4459.856

Standard Deviation: 770.290

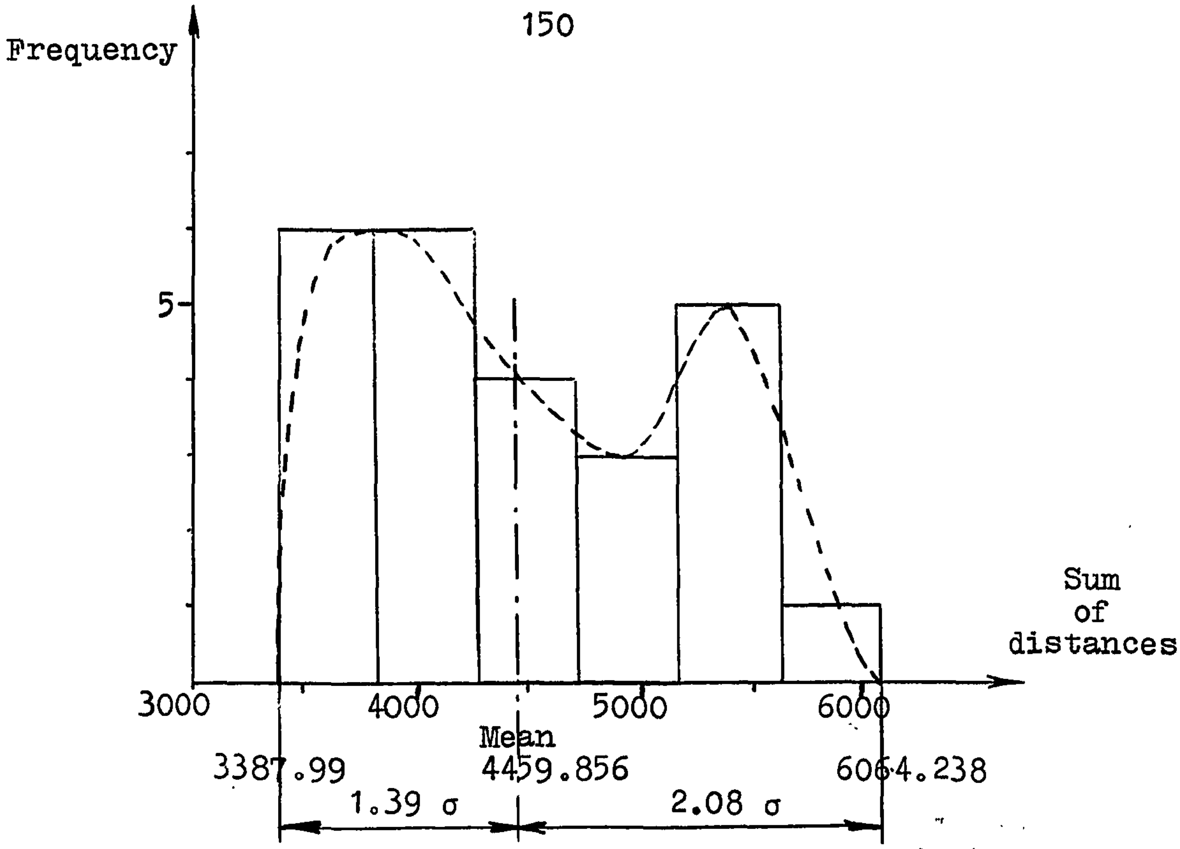


Fig. 53. Random Grid Algorithm
Distribution of Distances
Large System

The optimum assignment lies at 1.39 σ from the mean, while poor assignments extend up to 2.08 σ from the mean. It is to be noted that although the minimum sum of distances is nearly identical to the random destination algorithm, the assignment is quite different. We must apply the exact re-location (IV.1.1.) to find out what is the optimum method of the two.

The 25 iterations were made in 5 minutes 42 seconds on an IBM 360/40 computer.

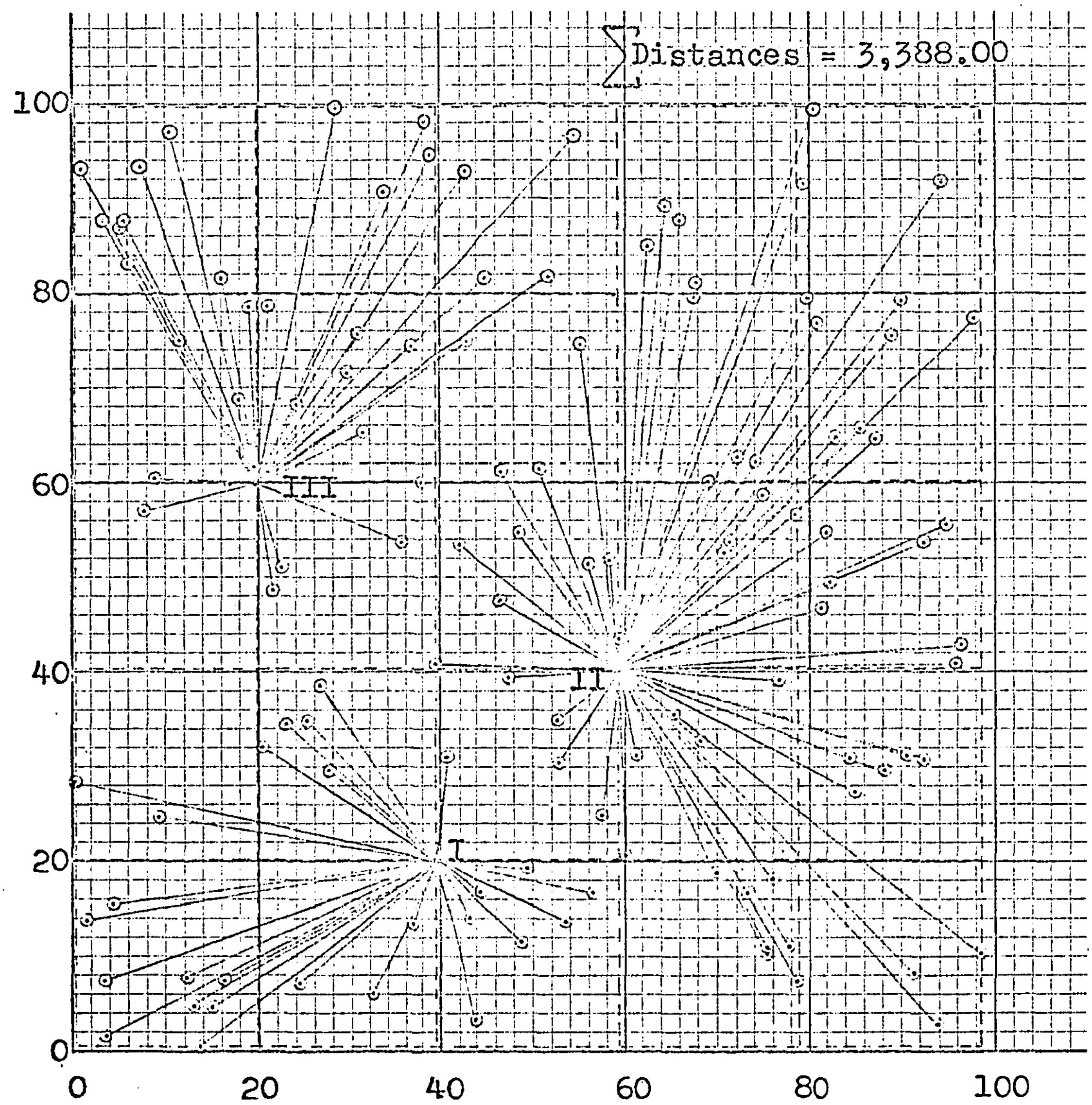


Fig. 54 Random grid location algorithm
N = 125 , M = 3

V.3. RANDOM GRID WITH LINEAR CONSTRAINTS

V.3.1. Small System, n = 20, m = 3

100 Samples per Grid Division

5 Grid Divisions from 3 to 7

Table 11. Random Grid with Linear Constraints

Ranges and Class Intervals

Grid Divisions	Range	Class Interval
3	76.490 - 52.941 = 22.549	22.549/8 = 2.8186
4	73.176 - 46.781 = 26.395	26.395/8 = 3.2992
5	88.660 - 50.051 = 38.609	38.609/8 = 4.8261
6	81.441 - 54.283 = 27.158	27.158/8 = 3.3947
7	121.155 - 64.102 = 57.053	57.053/8 = 7.1316

Total Distribution

Number of classes: 10

Range : 121.155 - 46.781 = 74.374

Class interval : 74.374/10 = 7.4374

Table 12. Random Grid with Linear Constraints

Distribution of Distances

Small System

Classes	Grid Division 3: Class Boundaries	Freq.	Grid Division 4: Class Boundaries	Freq.	Grid Division 5: Class Boundaries	Freq.
1	52.941- 55.759	53	46.781- 50.080	7	50.051- 54.877	17
2	55.760- 58.578	15	50.081- 53.379	9	54.878- 59.703	24
3	58.579- 61.396	0	53.380- 56.679	21	59.704- 64.529	20
4	61.397- 64.215	5	56.680- 59.978	27	64.530- 69.355	15
5	64.216- 67.034	24	59.979- 63.277	11	69.356- 74.181	11
6	67.035- 69.852	2	63.278- 66.577	15	74.182- 79.007	7
7	69.853- 72.671	0	66.578- 69.876	5	79.008- 83.833	4
8	72.672- 75.490	1	69.877- 73.176	5	83.834- 88.660	2
	Mean: 57.901 Std. Dev.: 5.368		Mean: 58.767 Std. Dev.: 5.852		Mean: 63.731 Std. Dev.: 8.708	

Table 12. Random Grid with Linear Constraints

Distribution of Distances

Small System

Classes	Grid Division 6: Class Boundaries	Freq.	Grid Division 7: Class Boundaries	Freq.	Total Distribution: Class Boundaries	Freq.
1	54.283- 57.677	9	64.102- 71.233	30	46.781- 54.218	64
2	57.678- 61.072	28	71.234- 78.365	52	54.219- 61.655	154
3	61.073- 64.467	19	78.366- 85.496	17	61.656- 69.093	154
4	64.468- 67.862	16	85.497- 92.628	0	69.094- 76.530	81
5	67.863- 71.256	12	92.629- 99.760	0	76.531- 83.968	42
6	71.257- 74.651	10	99.761-106.891	0	83.969- 91.405	4
7	74.652- 78.046	2	106.892-114.023	0	91.406- 98.842	0
8	78.047- 81.441	4	114.024-121.155	1	98.843-106.280	0
9					106.281-113.717	0
10					113.718-121.155	1
	Mean: 64.680 Std. Dev.: 6.230		Mean: 74.576 Std. Dev.: 6.937		Mean: 63.930 Std. Dev.: 8.959	

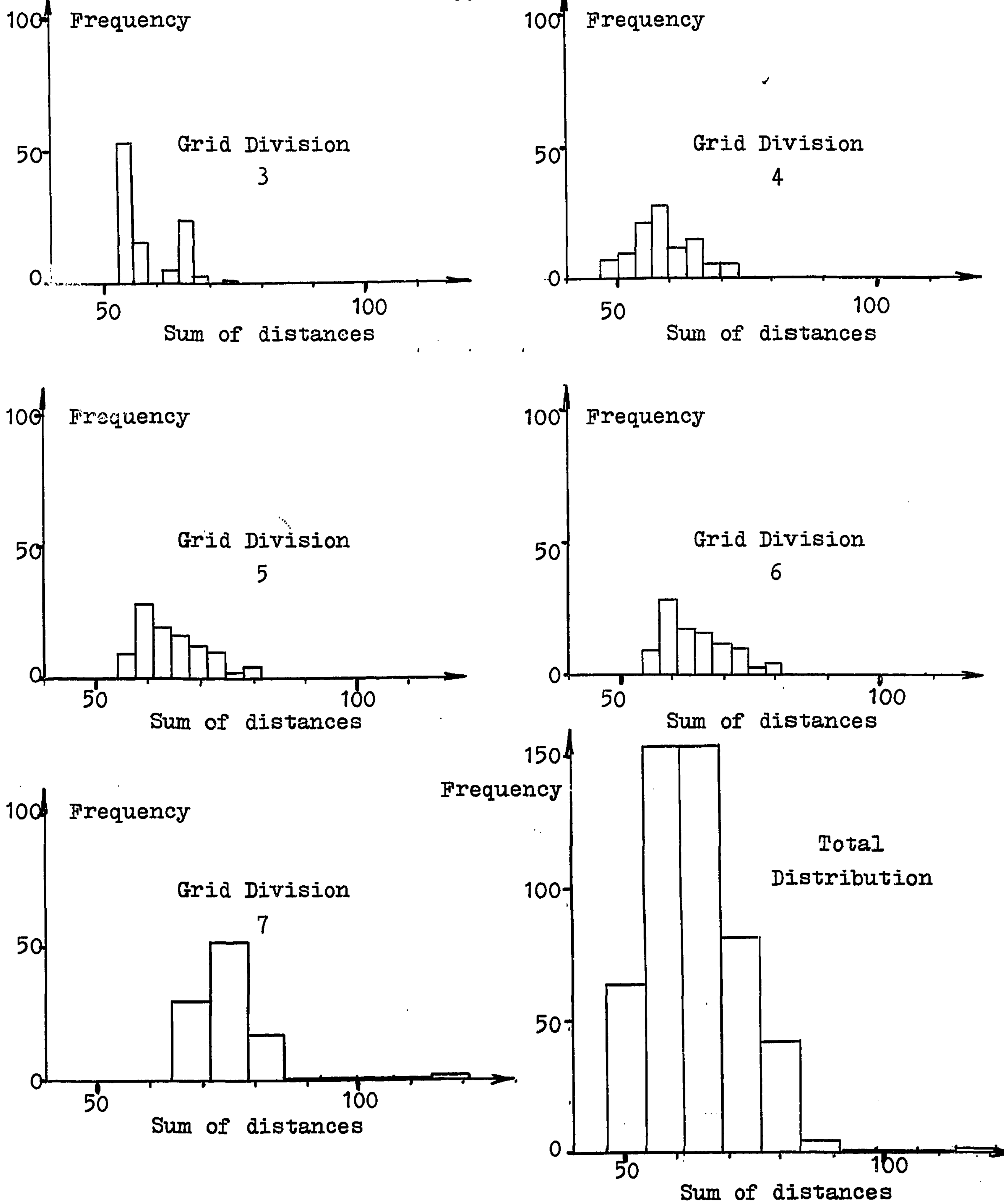


Fig. 55. Random Grid with Linear Constraints
Distribution of Distances - Small System

In this particular layout of facilities, the variability in the sum of distances is less with the random grid with linear constraints than with the random destination method. When the number of grid points is smaller than the number of facilities, the mean value of the sum of distances is definitively improved (57.9 and 58.7 compared to 59.6). The optimum value of 46.781 obtained with a grid division of 4 is still larger than 43.927 given by the random destination method, however, the assignment in satellites is the same and consequently after final location with algorithm IV.1.1. the sum of distances will be identical.

It is interesting to note that when the number of grid points is too small the distribution becomes multimodal, for example, 2 modes appear at 54 and 65 for a grid division of 3. As the number of grid points increases the distribution gets skewed more and more to the right; this is due to the fact that we take percentagewise fewer and fewer samples as the population increases. This fact explains why the random destination algorithm will not be efficient when used with a very large system, because it will then be necessary to take also very large samples from the population to finally get a point near optimum.

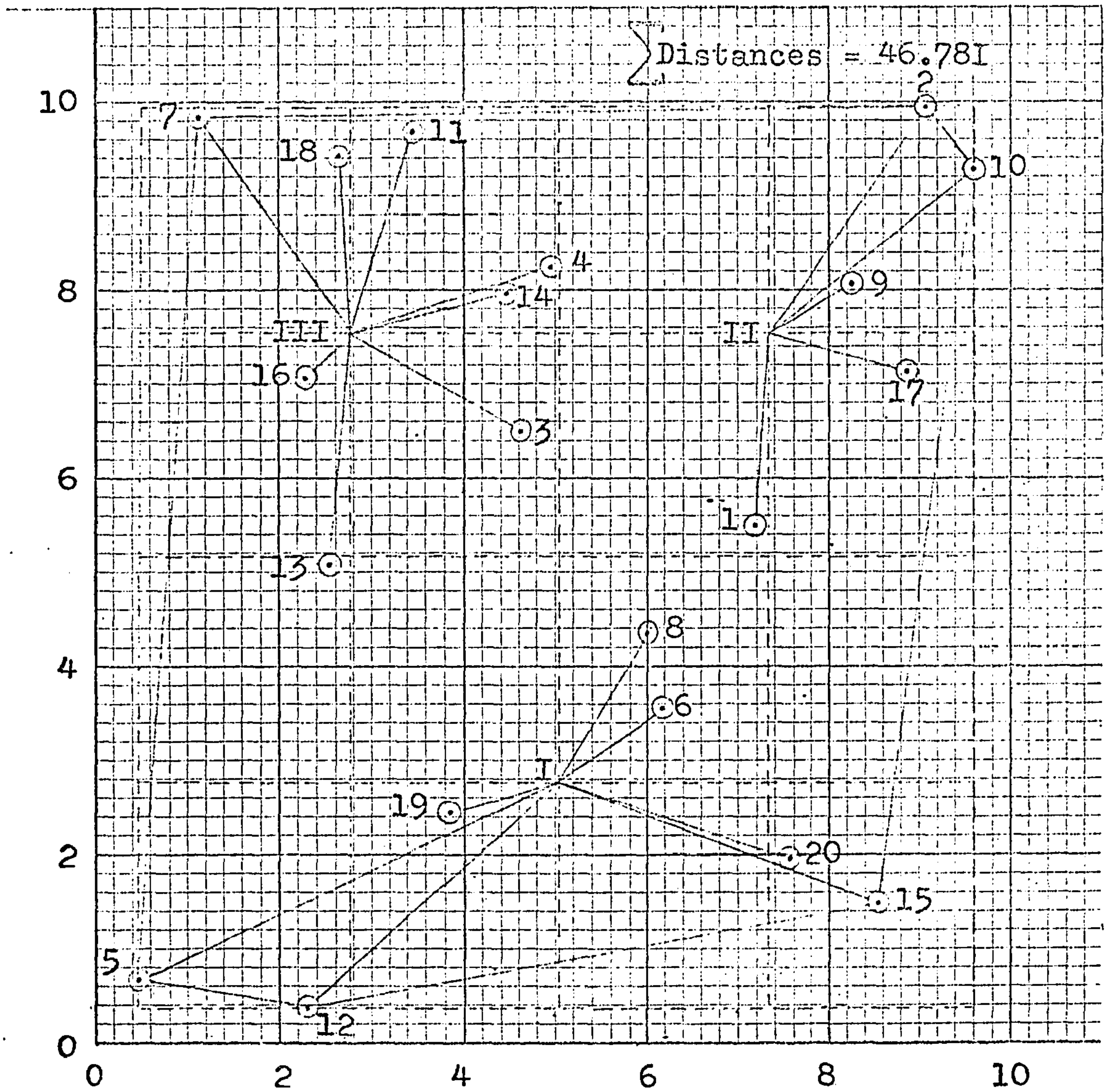


Fig. 56 Random grid with linear constraints
N = 20 , M = 3

IV.3.2. **Large System, n = 125, m = 3**

25 Samples per Grid Division

One Grid, Division in 6

Range

$$4922.980 - 3082.082 = 1840.898$$

Number of classes

$$k = 1 + 3.3 \log_{10} 25 \cong 6$$

Class interval

$$1840.898/6 = 306.8163$$

Table 13. Random Grid with Linear Constraints
Distribution of Distances
Large System

	Class Boundaries	Frequency
1	3082.082 - 3388.898	7
2	3388.899 - 3695.714	7
3	3695.715 - 4002.530	3
4	4002.531 - 4309.347	5
5	4309.348 - 4616.163	2
6	4616.164 - 4922.980	1
		25

Mean: 3754.705

Standard Deviation: 504.635

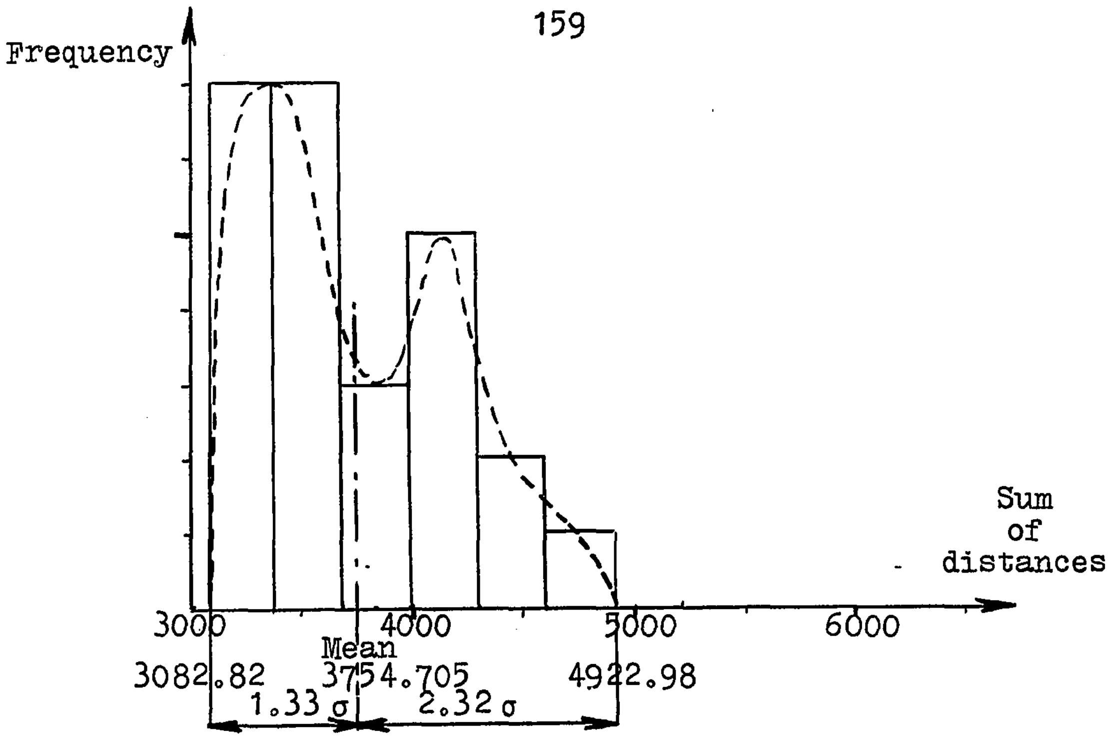


Fig. 57. Random Grid with Linear Constraints

Distribution of Distances

The optimum assignment lies at 1.33 σ from the mean, while poor assignments extend up to 2.32 σ from the mean. It is to be noted that for such a small number of samples, this method gives the best assignment of any other method. (8% better than the random destination subset algorithm). Moreover the spread of results is not so large, there is a drastic improvement in the value of the standard deviation of the sum of distances: 504 compared to 750 for the random destination subset algorithm. This improvement is easily understood by looking at Figure 40 where it is shown the possible irrationality of the sampling made with the random destination method. When the set of N facilities increases, the random grid with linear constraints will improve with it as it spreads the limited number of sampling points all over the facility space.

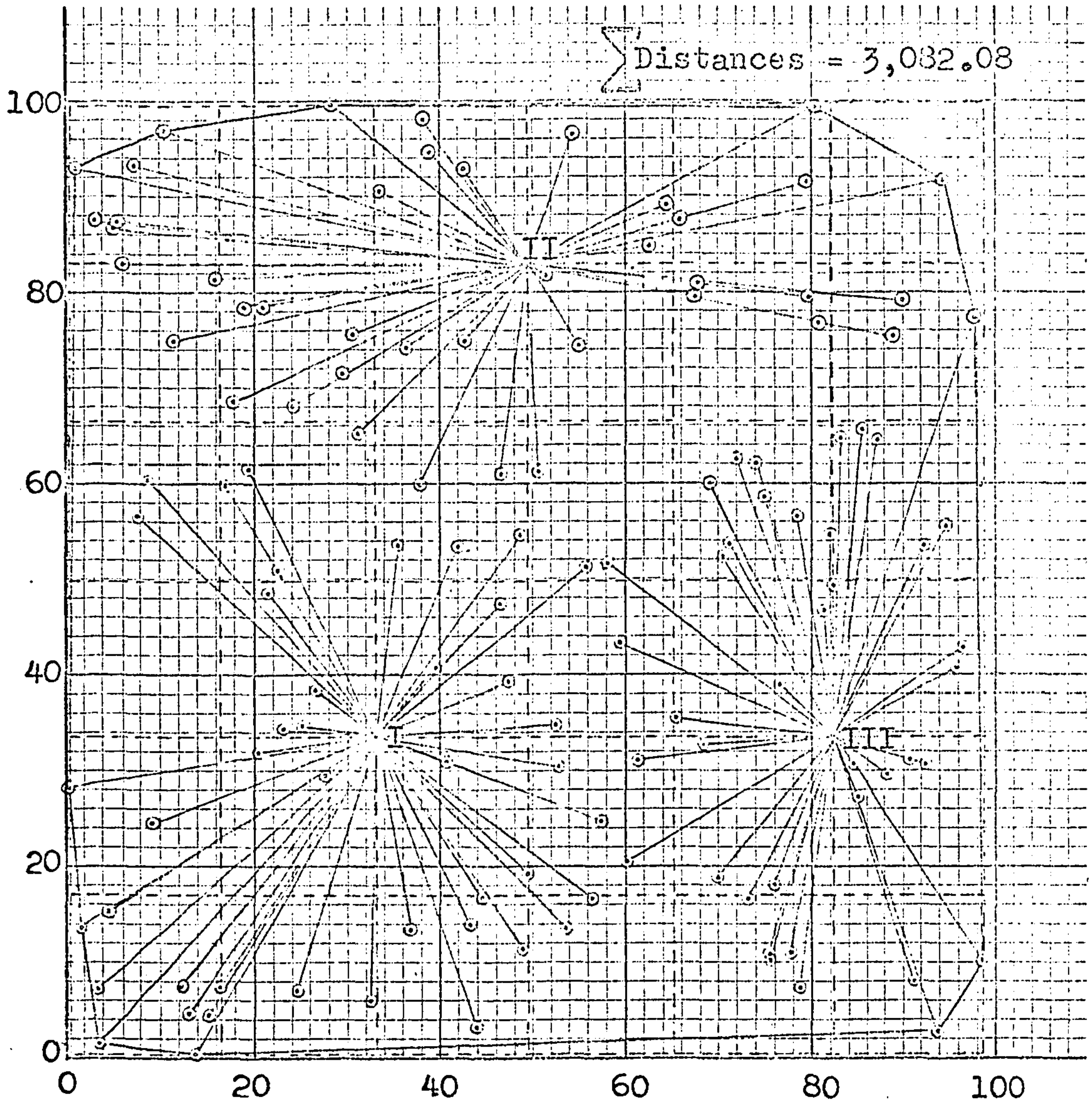


Fig. 58 Random grid with linear constraints
N = 125 , M = 3

Table 14

Random Grid with Linear Constraints versus Random Destination Subset Algorithm
Application to 4 Different Systems
$N = 125$, $M = 3$, 25 Random Samples

Computer: IBM 360/40 Printer: IBM 1403 N 1

The exact minimum sum of distances is computed by applying algorithm IV.1 to the sets of satellites defined by the corresponding algorithms.

System #		Random Destination Subset Algorithm	Random Grid with Linear Constraints
1	Minimum sum of distances	3323.594	3082.082
	Mean sum of distances	4331.586	3754.705
	Standard deviation of sum of distances	750.026	504.635
	Exact minimum sum of distances	3122.075	3000.712
2	Minimum sum of distances	3147.336	3290.181
	Mean sum of distances	4362.776	4277.913
	Standard deviation of sum of distances	938.340	790.590
	Exact minimum sum of distances	2890.957	2837.912
3	Minimum sum of distances	3048.011	3063.494
	Mean sum of distances	3726.594	3779.001
	Standard deviation of sum of distances	532.571	542.535
	Exact minimum sum of distances	3061.424	3030.233
4	Minimum sum of distances	3010.937	3050.184
	Mean sum of distances	3987.171	3609.579
	Standard deviation of sum of distances	848.243	505.884
	Exact minimum sum of distances	2966.200	2953.828

V.4. VARIABLE DISCRIMINATION ALGORITHM

The discrimination algorithm is to be used when the size of the problem is so large that it cannot be handled by the computer facility because of the memory requirement.

Using FORTRAN to program all the previous algorithms made the tools easy to use with any make of computer; however, it drastically reduces our memory space. For example, a problem with 500 facilities could not be handled with our 122K computer facility. At the national scale it is common to be faced with problems involving many thousands of facilities; the discrimination algorithm can then be used to condense this set to a smaller size of easier manipulation.

The simple variable discrimination program given on page 235 to page 237 was used to cluster a set of 125 facilities into a set of 75, using a storage space of 1062 bytes for the program and approximately 11.5 bytes per facility considered. When the facilities are not plotted it is difficult to guess correctly a good starting value for the discriminating power and its possible variations. It is wise to start with a large value of discriminant DISCR, large variation steps DINC, and wide open tolerances NERR. With a very small number of loops: NLOOP, the investigator can rapidly see how the set behaves under the clustering program; it is then easy to adjust properly the discrimination argument, its variations, and the range of tolerance for the expected subset. The program does not have a built-in system to compute the new weighted location, transport rate and amount to be transported for the generated cluster, an easily developed sub-routine could be devised to do such a computation; time was not available to develop it.

Figure 59 shows the clustering effect of the algorithm as applied to our set of 125 facilities. It would be interesting to apply any one of the previously developed algorithms to study the variation in total sum of distances and corresponding loss in accuracy (if any) created by this clustering. It is to be noted also, that with our available memory space of 122K we could easily condense sets containing up to 10,000 facilities.

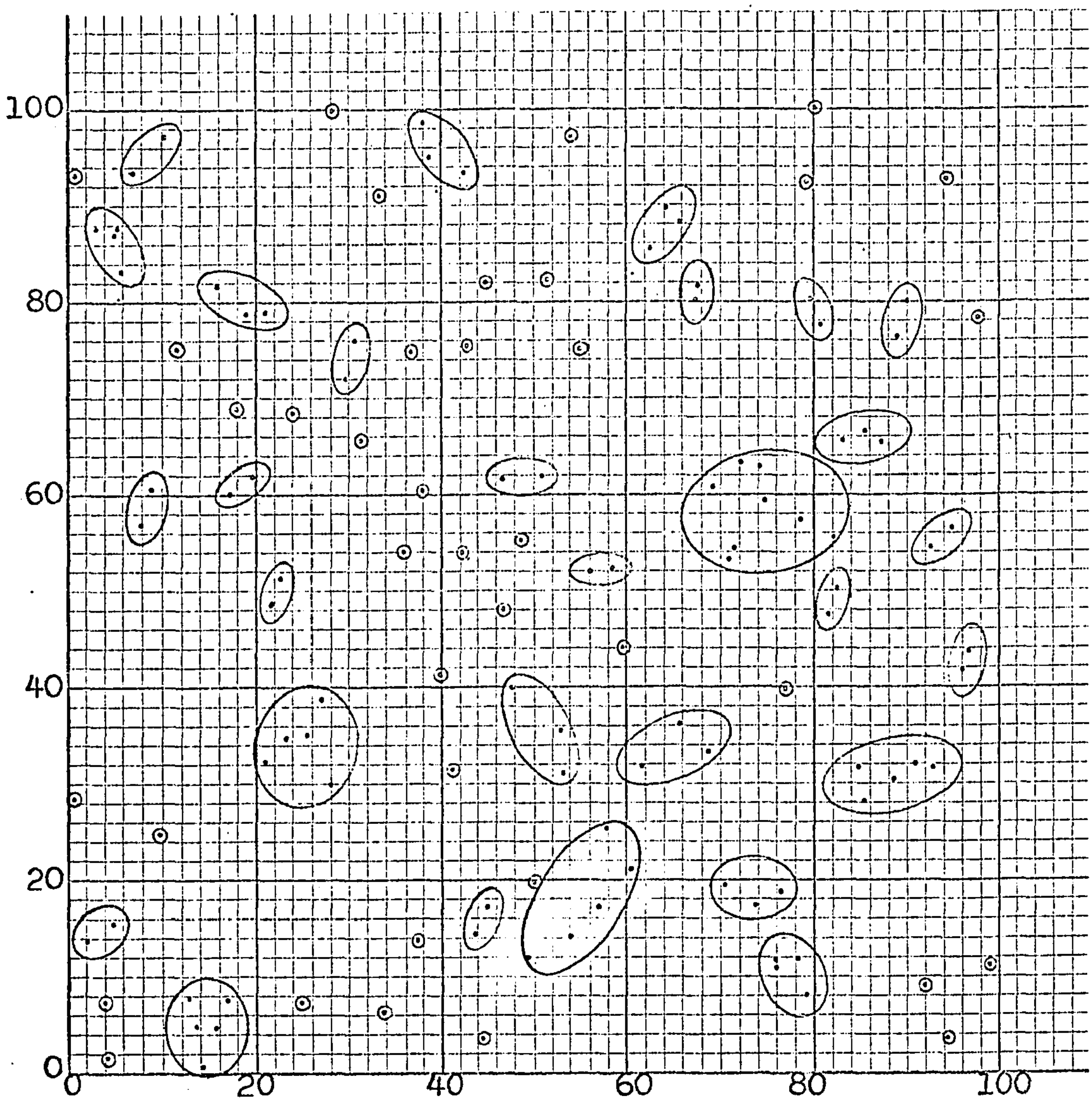


Fig. 59 Variable Discrimination Algorithm

CHAPTER VI

CONCLUSION

IV.1. Recommended Future Research

In the preceding chapters we covered some of the highlights of the state of the art in some of the methods which have been used to solve the two dimensional locational problem in Euclidean space, and we expanded some new computational methods adjustable to large systems. When reviewing these methods we find many areas which would require further research.

The analog method first appealed to the author many years ago while measuring electrical fields in oil well surveying, but it must be granted that the method involves the possibility of large errors and is relatively inflexible as the facilities must be physically plotted and fitted with mechanical or electrical devices. It is undoubtedly possible to find a better instrumentation and technique but the analog method will remain cumbersome and will have mostly a demonstration purpose.

Very little has been done with the geometric method in the case of multiple vertices, although historically it was the first one to be developed for few vertices. It is to be noted that many complex problems in the field of mechanics or electricity may be solved graphically with enough accuracy for practical purpose. The author has tried many geometrical constructions to reduce the force closure vector by a hopefully convergent procedure but to date nothing of value has been found.

The drawing of isodapanes and the mapping of costs has been undertaken on a digital computer but all the results presented in the literature show painfully hand-drawn curves from points of equal costs plotted by the computer. It seems quite feasible to devise a Calcomp plotter sub-routine for example, to trace and interpolate automatically these curves.

In the iterative algorithm of the algebraic solution as applied to one central facility we could have used the conventional approach of varying X to reduce $\partial C/\partial X$, then modify Y to reduce $\partial C/\partial Y$ and continue the process hopefully toward convergence, however, the optimum may be too long to reach or may be missed altogether if the steps are not adequate and these steps cannot be defined a priori without having an insight on how the derivatives behave.

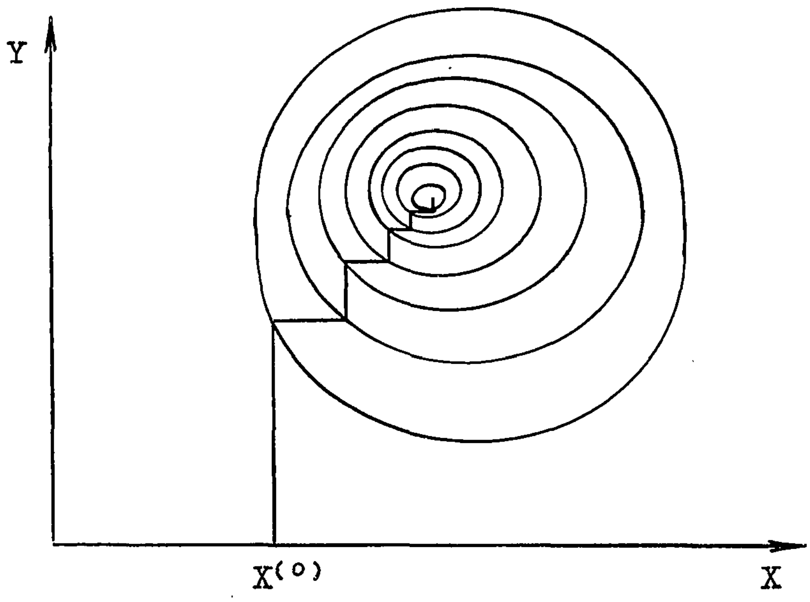


Fig. 60. Convergent Iterative Process

We preferred to use a method which starts at the weighted mean of distances and it has successfully converged all the time, however, no formal proof of this convergence is still available.

Moreover, we have found that the starting value we used may be far from the optimal location in case of weight dominance, we should then consider the use of a more heavily weighted arithmetic mean to encompass for this error. For example, it would be interesting to study the optimum value of the power $k > 1$ in the following starting values so as to obtain an accelerated convergence or even a good enough optimum location without iteration.

$$X^{(o)} = \frac{\sum_{i} w_{i}^{k} x_{i}}{\sum_{i} w_{i}^{k}}$$

$$Y^{(o)} = \frac{\sum_{i} w_{i}^{k} x_{i}}{\sum_{i} w_{i}^{k}}$$

It is probable also that the use of the Holder inequality could lead us to a more restrictive bound than the triangle inequality.

Quite a number of non-linear problems are successfully solved by the use of geometric programming. It has been previously demonstrated that the method could solve exactly the single-center locational problem. In practice, however, the method becomes rapidly infeasible because of the very large degree of difficulty encountered, even with relatively small systems.

The constrained problem has been mentioned but no exact solution computationally attractive has been found to date with the Lagrangian or Khun-Tucker methods.

In the study of multiple central facilities we tried to develop a few algorithms computationally attractive for the case of large systems. Because of time and computer expenditures limitations we barely developed and tested three algorithms and the very few cases run were not enough to statistically prove the superiority of the variable grid algorithm with linear constraints over the random destination algorithm in the case of large systems. It seemed that above 100 facilities the variable grid with linear constraints gave a better allocation in less time and with less spread in results than the random destination algorithm but this statement must be supported by future substantial statistics and further research.

VI.2. Summary

In Table 1 of the introductory chapter we had listed most of the types of locational problems we are faced with in practical life. In our study we limited ourselves to the expansion of some computational techniques applied to large systems of facilities located in a discrete space and interconnected by Euclidean distances. Even under such drastic limitations we only developed three new methods: the random grid location, the random grid with linear constraints and the variable discrimination algorithms. The testing and debugging of these programs are complete, but due to limitations in time and computer expenditure the complete statistical study of results is quite incomplete.

Our effort is not in vain however, and adds to the multiple research done in locational theory. We may look down on the analogue methods of solution but in some practical problems these are the only techniques self-explanatory to management and of recommended use by Haley [30] in 1963 or Eilon and Deziel [22] in 1966.

Some researchers, on the other hand, are very oriented to theory and try to reduce the locational problem to a more manageable mathematical model, or to connect it more closely to the extensively studied transportation and transhipment problems [3] [25] [29].

For some other researchers, an approximate answer is quite sufficient for the practical use it will be made of it. Although it has been shown that the definition of the center of mass is erroneous, it will give in practice a result with acceptable accuracy, mostly if there is no undue weight dominance or extreme locations. The optimal location in the case of one center is then found at the intersection of the weighted arithmetic means of the demand points along two orthogonal axis and the problem is identical to the definition of the center of gravity of a two-dimensional object in mechanic [19] [44].

When faced with equations of implicit nature, some researchers will try to find some heuristic algorithms, that is to say some numerical methods of iterative or simulative nature contributing to a reduction in the average search for the optimum solution. This is the method that I have adopted because it is flexible enough to adjust to all the problems and realistic enough to approach real life problems. Some of these methods are presented in the literature [10] [24] [35] [50] [53], and although they are logically complete they require extensive programming and testing to be duplicated by a potential user. Some of these heuristic methods may even attempt some non-linear cost functions or some large systems [24]. Most of these methods have some simplifying assumptions: Kuehn and Hamburger [35], as well as Feldman, Lehrer and Ray [24] read in all the potential central locations sites and use the add or "Drop" approach to eliminate the ones which are not economically interesting transportation-wise; they may even consider some unified transport rates which are far from real world situations. Nor do any of them take into account the type of road system, inaccessible areas, labor costs and site costs, etc., but even with these drastic restrictions the results could be useful, but it is the extreme exception rather than the rule when the computer program is made public [18].

Up to now, we have not even mentioned the very important case in which the elements to service are so numerous that they may be considered as part of a continuum with a given density per unit area. This is the case for example, of locating city service centers and is of very practical nature. Some approaches to this problem have been treated by Witzgall [56] in the case of a Manhattan metric of streets and perpendicular avenues, or in the case of a city beltway and radial streets. For example, a computer program in FORTRAN was devised [57] to locate a central facility serving a polygonal demand distribution that is a superposition of many distributions each bounded by a polygon with constant density inside and vertices following in counterclockwise direction.

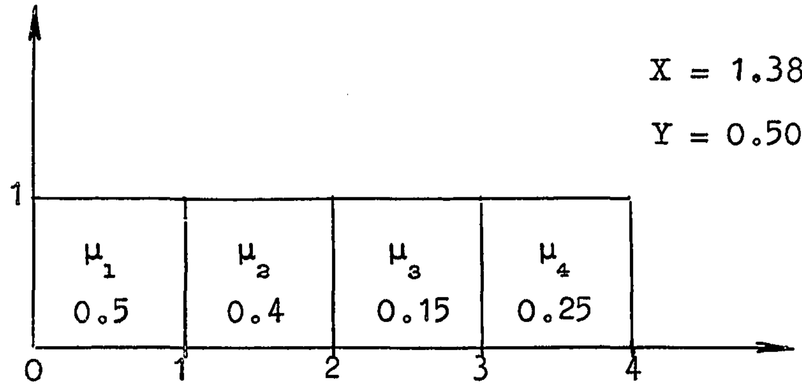


Fig. 61. Example of Polygonal Demand Distribution Location of Central Facility

The locational problem in Manhattan metric has a great potential application in optimal plant layout and has been studied, for example, by Bindschedler and Moore [5] [44] and a computer program devised by Armour [1] to mention just a few.

In our study we have always assumed a value of m, without knowing if it were the most appropriate and the one leading to an absolute minimum cost. Bender, Goldman and Levin [4] [39] have done some research in that area to find the necessary "degree of centralization" that is the optimum number of centers to be located in a given area.

Our study was considering only a two-dimensional space with each facility bearing 2 attributes of cartesian coordinate location; however, in numerous practical problems of sorting and classification it is common to find elements with multiple characteristics. In taxonomy for example, n species may be scored for m characters. In these particular problems it is desirable to cluster large number of objects, symbols or persons into smaller numbers of mutually exclusive groups, each having members that are as much alike as possible. In two dimensional space our Euclidean distances represented the minimization of weighted sum of squares about the group mean. Similarly in multi-dimensional space we will consider the minimization of the function:

$$C = \sum_{i=1}^{n} w_i \left[\sum_{j=1}^{m} (x_{ij} - x_i)^2 \right]^{\frac{1}{2}}$$

and it represents the "loss of information" as reflected by the error sum of squares. This cluster analysis is very common in social sciences, psychology, biology and marketing. Much has been written on this subject but we will mention just a few [27] [2] [20]. The theory is very nicely covered by Cooper [14] and a selection of computer programs to solve this generalized locational problem are available [6] [49] [40].

Even though the amount of literature is impressive in the field of locational theory and spans over many centuries of research little has been achieved. The geometric solution did not lead to anything of much value, the proof of a convergent algorithm has never been done completely [34] and the various investigative algorithms become rapidly impossible over few hundred facilities. The main goal of this dissertation was to decrease the computational requirement of some of the methods. The variable grid algorithms and the variable discrimination method allow the condensation of a very large set into a smaller one easier to manipulate. In the process of agglomeration some of the information is lost and the final solution may be only suboptimum; however, the result will be better than nothing at all. In multi-dimensional problems the variable grid algorithms will also be applicable, the grid points will be multi-dimensional and the linear constraints will be changed into planes and hyperplanes; similarly the discrimination algorithm will have to screen through all the attributes to condense close points. Although we limited ourselves to discrete space and Euclidean distances, it would be a great engineering achievement to compile in an orderly manner all the accomplishments in the field of locational theory into a set of tools directly and easily available to the practicing engineer, economist or social and government worker.

BIBLIOGRAPHY

1. Armour, G. C., "A Heuristic Algorithm and Simulation approach to relative location of facilities", Unpublished Ph.D. Dissertation, Department of Industrial Engineering, University of California at Los Angeles, 1961.

2. Bah, G. H., "Data Analysis in the Social Sciences: What about the Details?", Proceedings - Fall Joint Computer Conference, Stanford Research Institute, Menlo Park, California, 1965.

3. Baumol, W. J. and P. Wolfe, "A Warehouse Location Problem", Operations Research, V. G., March - April, 1958 pp. 252-258.

4. Bender, B. K. and A. J. Goldman, "Optimization of Distribution Networks. Mathematical Development", Unpublished report #6930, National Bureau of Standards, July 13, 1960.

5. Bindschedler, A. E. and J. M. Moore, "Optimal Location of New Machines in Existing Plant Layouts", Journal of Industrial Engineering, 12:1, January, 1961, pp. 41-48.

6. Bonner, R. E., "Clustering Program", International Business Machine Corporation, Yorktown Heights, New York, May, 1964.

7. Brink, E. L. and J. S. De Cani, "An Analogue Solution of the Generalized Transportation Problem with Specific Application to Marketing Location", Proceedings - First International Conference on Operational Research, Wiley, New York, 1961, pp. 123-137.

8. Burstall, R. M., R. A. Leaver and J. E. Sussams, "Evaluation of Transport Costs for Alternative Factory Sites....A Case Study", Operational Research Quarterly, 13:4, December, 1962, pp. 345-354.

9. Cooper, L., "An Extension of the Generalized Weber Problem", Unpublished paper, Washington University, School of Engineering and Applied Science, St. Louis, 1967.

10. Cooper, L., "Heuristic Methods for Location - Allocation Problems", SIAM Review, 6:1, January, 1964, pp. 37-53.

11. Cooper, L., "Location - Allocations Problems", Operations Research, 11:3, 1963, pp. 331-343.

12. Cooper, L., "The Optimal Location of Facilities", School of Engineering and Applied Science, Washington University, St. Louis, E & AS Research, Number 1, April, 1969.

13. Cooper, L., "Solutions of Generalized Locational Equilibrium Models", Journal of Regional Science, 7:1, 1967.

14. Cooper, L., "An n-Dimensional Extension of Locational Equilibrium Models", Unpublished paper, Washington University, School of Engineering and Applied Science, St. Louis, 1968.

15. Courant, R. and H. Robbins, "What is Mathematics", Oxford University Press, New York, 1963.

16. Dean, W. H. Jr., "The Theory of the Geographic Location of Economic Activities", Unpublished Ph.D. Dissertation, Ann Arbor, 1938.

17. Devine, M. D. and W. G. Lesso, "Optimization Studies in the Development of Offshore Oil Fields", Unpublished report, Department of Mechanical Engineering, The University of Texas at Austin, 1968.

18. Durbin, E. P. and D. M. Kroenke, "The Out of Kilter Algorithm: A Primer", RAND Corporation Memorandum, RM-5472-PR, December, 1967.

19. Eddison, R. T., K. Pennycuick and B. H. P. Rivett, "Operational Research in Management", English Union Press, London, 1962, Chapter 8.

20. Edwards, A. W. and L. L. Cavilli-Sforza, "A Method for Cluster Analysis", Biometrics, 21, June, 1965, pp. 362-375.

21. Efroymson, M. A. and T. L. Ray, "A Branch-Bound Algorithm for Plant Location", Journal of the Operations Research Society of America, Vol. 14, No. 3, May-June, 1966, pp. 361-368.

22. Eilon, S. and D. P. Deziel, "Sitting a Distribution Center, An Analogue Computer Application", Management Science, Vol. 12, No. 6, February, 1966, pp. B245-B254.

23. Espenshade, E. B. Jr., "Goode's World Atlas", Rand McNally & Company, Chicago, 1965.

24. Feldman, E., F. A. Lehrer and T. L. Ray, "Warehouse Location under Continuous Economics of Scale", Management Science, Vol. 12, No. 9, May, 1966, pp. 670-684.

25. Ford, L. R. and D. R. Fulkerson, "Flows in Networks", Princeton University Press, Princeton, New Jersey, 1962.

26. Friedrich, C. J., "Alfred Weber's Theory of Location of Industries", The University of Chicago Press, 1929, pp. 230-231.

27. Green, P. E., R. E. Frank and P. J. Robinson, "Cluster Analysis in Test Market Selection", Management Science, 13:8, April, 1967.

28. Greenhut, M., "Plant Location in Theory and in Practice", University of North Carolina Press, 1956.

29. Hadley, G., "Linear Programming", Reading, Massachusetts, Addison-Wesley, 1962.

30. Haley, K. B., "The Siting of Depots", International Journal of Production Research, Vol. 2, No. 1, March, 1963, pp. 41-45.

31. Hoover, E. M., "Location Theory and the Shoe and Leather Industries", Unpublished Ph.D. Dissertation, Cambridge, Massachusetts, 1937, pp. 42-52.

32. Hoover, E. M., "The Location of Economic Activity", McGraw-Hill Book Company, New York, 1948.

33. Isard, W., "Location and Space Economy", Technology Press, M.I.T., Cambridge, 1956.

34. Katz, I. N., "On the Convergence of a Numerical Scheme for Solving the Generalized Weber Problem and Some of its Extensions", Unpublished Report No. AM-68-4, Department of Applied Mathematics and Computer Science, Washington University, St. Louis, 1968.

35. Kuehn, A. A. and M. J. Hamburger, "A Heuristic Program for Locating Warehouses", Management Science, Vol. 9, No. 4, July, 1963, pp. 643-668.

36. Kuhn, H. W. and R. E. Kuenne, "An Efficient Algorithm for the Numerical Solution of the Generalized Weber Problem in Spatial Economics", Journal of Regional Science, 4:2, 1962, pp. 21-33.

37. Launhardt, W., "Matematische Begrundung der Volskswirtschaftslehre", Leipzig, 1885.

38. Lehmer, D. H., "Teaching Combinatorial Tricks to a Computer", Proceedings of Symposia in Applied Mathematics, Vol. X, Combinatorial Analysis, American Mathematical Society.

39. Levin, B. M., "Optimization of Distribution Networks", Unpublished Technical Report No. 6930 of the National Bureau of Standards to the U.S. Post Office Department, July 13, 1960.

40. Lingoes, J. C., "Multivariate Analysis of Contingencies: an IBM 7090 Program for Analyzing Metric/Non-Metric or Linear/Non-Linear Data", Unpublished Computation Report, University of Michigan Computing Center, June, 1966.

41. Manne, A. S., "Plant Location Under Economies-of-Scale-Decentralization and Computation", Management Science, Vol. 11, November, 1964.

42. McHose, A. H., "A Quadratic Formulation of the Activity Location Problem", Journal of Industrial Engineering, 12:5, September, 1961, pp. 334-337.

43. Miehle, W., "Link-length Minimization in Networks", Operations Research, March - April, 1958, pp. 232-243.

44. Moore, J. M., "Plant Layout and Design", New York, The Macmillan Company, 1962, p. 39.

45. Moore, J. M. and M. R. Mariner, "Layout Planning: New Role for Computers", Modern Materials Handling, 18:3, March, 1963, pp. 38-42.

46. Palander, T., "Beitrage zur Standortstheorie", Almquist and Wikselts, Uppsala, Chapter IX, 1935.

47. Palermo, F. P., "A Network Minimization Problem", IBM Journal of Research and Development, 5, 1961, pp. 335-337.

48. Polya, G., "Induction and Analogy in Mathematics", Princeton University Press, 1954, pp. 145-146.

49. Rubin, J., "Optimal Taxonomy Program (7090-IBM-0026)", International Business Machine Corporation, Program Information Department, Hawthorne, New York, 1965.

50. Shycon, H. N. and R. B. Maffei, "Simulation - Tool for Distribution", Harvard Business Review, Vol. 41, No. 6, November - December, 1960, pp. 65-74.

51. Stewart, D. W. Jr., "The Development of a Quantitative Method for the Optimization of the Facilities Location Problem", Unpublished M.S. Thesis, Georgia Institute of Technology, January, 1963.

52. U.S. Post Office Department, "First Class and Air Mail Volume Flow Information, Vol. 2, November 16-18, 1965, Sectional Center to Destination State, Average Daily Pounds and Pieces", Unpublished report from the Research Branch of the Bureau of Transportation and International Services, Washington, D.C., 1965.

53. Vergin, R. C. and J. D. Rogers, "An Algorithm and Computational Procedure for Locating Economic Facilities", Management Science, Vol. 13, No. 6, February, 1967, pp. B240-B253.

54. Weber, A., "Uber den Standort der Industrien", Tubingen, 1909.

55. Wimmert, R. J., "A Mathematical Model of Equipment Location", Journal of Industrial Engineering, Vol. 9, No. 6, November - December, 1958, pp. 498-510.

56. Witzgall, C., "Optimal Location of a Central Facility. Mathematical Models and Concepts", Unpublished Technical Report No. 8388 of the National Bureau of Standards to the U.S. Post Office Department, June 30, 1964.

57. Witzgall, C., "Computer Routines for Optimally Locating a Central Facility Serving a Polygonal Demand Distribution", Unpublished Working Paper No. 3-65 of the National Bureau of Standards to the U.S. Post Office Department, Project 4230450, January 7, 1965.

58. Yassen, L. C., "Plant Location", American Research Council, New York, 1958, pp. 220.

APPENDIX

COMPUTER PROGRAMS FOR THE LOCATION OF CENTRAL FACILITIES

The following programs have been written in FORTRAN IV with control cards for the IBM 360/40 or the IBM 1130 computers.

Please refer to the comment cards to modify the dimension statements according to the size of the problem. The input/output codes for R(Read) and W(Write) should also be changed to fit the available computer connections. Note also the format input, and punch the data accordingly. The terminology of variables is given on the comment cards.

The appendix includes the following programs:

1 - Location of one central facility

1A - Location of one central facility in case of weight dominance

2 - Destination subset algorithm, one facility is used as central location

3 - Random destination subset algorithm

4 - Random grid location algorithm

5 - Random grid with linear constraints

6 - Variable discrimination algorithm

ONE CENTRAL FACILITY
HEURISTIC ALGORITHM

```
C     LOCATION OF ONE CENTRAL FACILITY                                 SPRING  1969
C     DEFINITION OF MACHINE INPUT/OUTPUT :R READ
C                                         W WRITE
      INTEGER R,W
C     CHANGE THE DIMENSION CARD IF MORE THAN 1000 FACILITIES ARE CONSIDERED
      DIMENSION X(1000),Y(1000),XR(1000),XM(1000),D(1000),C(1000)
      R=5
      W=6
C     READ N   :NUMBER OF FACILITIES
C          ERR:ADMISSIBLE DISTANCE ERROR IN LOCATING A CENTRAL FACILITY
C     REMARK  :THE MACHINE WILL STOP AFTER ITER ITERATIONS IF OPTIMUM IS NOT Y
C              REACHED
      READ(R,10)N,ERR,ITER
   10 FORMAT(I10,F10.0,I10)
C     READ VARIABLES X(I),Y(I):CARTESIAN COORDINATES OF FACILITIES
C                         XR(I):TRANSPORT RATE ON ROUTE I
C                         XM(I):QUANTITY TO TRANSPORT ON ROUTE I
      READ(R,20)(X(I),Y(I),XR(I),XM(I),I=1,N)
   20 FORMAT(10F7.0)
      WRITE(W,30)N
   30 FORMAT(1H1,//,35X,15HLOCATION OF THE,I5,2X,10HFACILITIES,//,24X,
     121HCARTESIAN COORDINATES,6X,14HTRANSPORT RATE,3X,11HQUANTITY TO,/,
     269X,9HTRANSPORT,/,27X,1HX,14X,1HY,15X,1HR,14X,1HM,//)
      DO 40 I=1,N
      WRITE(W,35)I,X(I),Y(I),XR(I),XM(I)
   35 FORMAT(13X,2HI=,I5,F13.3,F15.3,F14.3,F16.3)
   40 CONTINUE
      WRITE(W,50)
   50 FORMAT(1H1,//,35X,32HLOCATION OF ONE CENTRAL FACILITY,/,30X,
     141HCARTESIAN COORDINATES  -  EUCLIDEAN SPACE,//,27X,1HX,14X,1HY,
     27X,25HTOTAL TRANSPORTATION COST,//)
      XDELT=ERR+1.0
      YDELT=ERR+1.0
      K=1
C     COMPUTE STARTING VALUES OF CENTRAL FACILITY COORDINATES: XC,YC
      SRM =0.0
      SRMX=0.0
      SRMY=0.0
      DO 60 I=1,N
      RM=XR(I)*XM(I)
      SRM=SRM+RM
      SRMX=SRMX+RM*X(I)
   60 SRMY=SRMY+RM*Y(I)
      XC=SRMX/SRM
      YC=SRMY/SRM
C     COMPUTE EUCLIDEAN DISTANCES TO CENTRAL FACILITY
   70 DO 80 I=1,N
   80 D(I)=(((XC-X(I))**2)+((YC-Y(I))**2))**0.5
C     COMPUTE TOTAL TRANSPORTATION COST: SC
      SC=0.0
      DO 90 I=1,N
      C(I)=XR(I)*XM(I)*D(I)
   90 SC=SC+C(I)
   95 IF( 100-K)106,106,100
```

```
  100 IF(XDELT-ERR)105,105,120
  105 IF(YDELT-ERR)106,106,120
  106 WRITE(W,110)XC,YC,SC,K
  110 FORMAT(3X,15HLOCATION OF THE,/,3X,16HCENTRAL FACILITY, 2X,F12.3,
     1F15.3,7X,E15.7,/,1X,6HAT THE,I4,1X,9HITERATION)
      WRITE(W,115)
  115 FORMAT(1H1,//,34X,34HDISTANCES AND TRANSPORTATION COSTS,/,40X,
     123HTO THE CENTRAL FACILITY,//,35X,8HDISTANCE,11X,
     219HTRANSPORTATION COST,//)
      DO 118 I=1,N
      WRITE(W,116)I,D(I),C(I)
  116 FORMAT(13X,2HI=,I5,F23.3,E25.7)
  118 CONTINUE
      GO TO 1000
  120 WRITE(W,130)XC,YC,SC
  130 FORMAT(21X,F12.3,F15.3,7X,E15.7)
C     COMPUTE NEW VALUES OF CENTRAL FACILITY COORDINATES: XCN,YCN
      SDENO=0.0
      SXNUM=0.0
      SYNUM=0.0
      DO 140 I=1,N
      DENO=(XR(I)*XM(I))/D(I)
      SDENO=SDENO+DENO
      XNUM=DENO*X(I)
      SXNUM=SXNUM+XNUM
      YNUM=DENO*Y(I)
  140 SYNUM=SYNUM+YNUM
      XCN=SXNUM/SDENO
      YCN=SYNUM/SDENO
      XDELT=ABS(XCN-XC)
      YDELT=ABS(YCN-YC)
      XC=XCN
      YC=YCN
      K=K+1
      GO TO 70
 1000 CALL EXIT
      END
```

LOCATION OF THE 50 FACILITIES

		CARTESIAN COORDINATES		TRANSPORT RATE	QUANTITY TO TRANSPORT
		X	Y	R	M
I=	1	25165.000	39.000	119.000	323.000
I=	2	903.000	233.000	345.000	61.000
I=	3	483.000	3621.000	5853.000	18.000
I=	4	23.000	2167.000	804.000	315.000
I=	5	242.000	989.000	1714.000	57.000
I=	6	555.000	485.000	823.000	521.000
I=	7	2289.000	669.000	23.000	148.000
I=	8	23.000	28.000	616.000	71.000
I=	9	2363.000	888.000	28.000	1386.000
I=	10	293.000	79.000	1030.000	75.000
I=	11	564.000	34.000	2927.000	1278.000
I=	12	41.000	28.000	590.000	140.000
I=	13	233.000	299.000	6.000	150.000
I=	14	8.000	4.000	114.000	19.000
I=	15	11.000	1.000	24.000	12.000
I=	16	9.000	3.000	9.000	38.000
I=	17	15.000	9.000	8.000	8.000
I=	18	6.000	4.000	11.000	13.000
I=	19	18.000	22.000	4.000	17.000
I=	20	12.000	11.000	6.000	3.000
I=	21	20.000	7.000	70.000	5.000
I=	22	5.000	53.000	12.000	57.000
I=	23	19.000	4.000	5.000	2.000
I=	24	7.000	29.000	6.000	2.000
I=	25	10.000	228.000	5.000	14.000
I=	26	74.000	9.000	5156.000	172.000
I=	27	4736.000	619.000	126.000	14.000
I=	28	182.000	197.000	84.000	12.000
I=	29	75.000	1221.000	398.000	295.000
I=	30	243.000	119.000	77.000	20.000
I=	31	144.000	201.000	450.000	350.000
I=	32	42.000	505.000	125.000	139.000
I=	33	206.000	25.000	238.000	557.000
I=	34	1253.000	62.000	55.000	628.000
I=	35	246.000	173.000	546.000	16.000
I=	36	38.000	54.000	92.000	935.000
I=	37	226.000	12.000	82.000	247.000
I=	38	58.000	280.000	248.000	5.000
I=	39	86.000	8042.000	626.000	117.000
I=	40	84.000	15.000	111.000	203.000
I=	41	215.000	4.000	13.000	1012.000
I=	42	272.000	136.000	261.000	121.000
I=	43	676.000	11.000	132.000	107.000
I=	44	410.000	165.000	374.000	1184.000
I=	45	15.000	105.000	18.000	17.000
I=	46	152.000	75.000	758.000	132.000
I=	47	11.000	439.000	797.000	50.000
I=	48	318.000	18.000	59.000	19.000
I=	49	1249.000	1619.000	14.000	2.000
I=	50	115.000	58.000	26.000	165.000

LOCATION OF ONE CENTRAL FACILITY
CARTESIAN COORDINATES - EUCLIDEAN SPACE

	X	Y	TOTAL TRANSPORTATION COST
	556.274	323.658	0.4850975E 10
	477.053	141.980	0.4170824E 10
	487.000	88.940	0.4080162E 10
	509.750	70.027	0.4038015E 10
	527.149	58.731	0.4009905E 10
	539.161	50.929	0.3991098E 10
	547.296	45.513	0.3978548E 10
	552.779	41.794	0.3970146E 10
	556.467	39.260	0.3964532E 10
	558.945	37.542	0.3960774E 10
	560.602	36.382	0.3958258E 10
	561.718	35.602	0.3956569E 10
	562.474	35.078	0.3955432E 10
	562.979	34.722	0.3954670E 10
	563.315	34.484	0.3954158E 10
	563.542	34.324	0.3953819E 10
	563.693	34.217	0.3953592E 10
	563.790	34.145	0.3953444E 10
	563.857	34.099	0.3953341E 10
	563.907	34.067	0.3953267E 10
	563.939	34.045	0.3953222E 10
	563.960	34.029	0.3953193E 10
	563.974	34.019	0.3953170E 10
LOCATION OF THE CENTRAL FACILITY AT THE 24 ITERATION	563.983	34.013	0.3953154E 10

DISTANCES AND TRANSPORTATION COSTS
TO THE CENTRAL FACILITY

		DISTANCE	TRANSPORTATION COST
I =	1	24601.000	0.9455885E 09
I =	2	393.102	0.8272822E 07
I =	3	3587.884	0.3779978E 09
I =	4	2200.523	0.5573043E 09
I =	5	1007.806	0.9846064E 08
I =	6	451.077	0.1934140E 09
I =	7	1838.176	0.6257151E 07
I =	8	541.016	0.2366189E 08
I =	9	2452.504	0.9517677E 08
I =	10	274.691	0.2121990E 08
I =	11	0.021	0.8023719E 05
I =	12	523.017	0.4320120E 08
I =	13	423.990	0.3815912E 06
I =	14	556.792	0.1206011E 07
I =	15	553.967	0.1595425E 06
I =	16	555.848	0.1901001E 06
I =	17	549.552	0.3517134E 05
I =	18	558.789	0.7990681E 05
I =	19	546.115	0.3713579E 05
I =	20	552.462	0.9944312E 04
I =	21	544.653	0.1906286E 06
I =	22	559.305	0.3825644E 06
I =	23	545.808	0.5458082E 04
I =	24	557.005	0.6684059E 04
I =	25	586.965	0.4108754E 05
I =	26	490.621	0.4350979E 09
I =	27	4212.816	0.7431408E 07
I =	28	415.302	0.4186244E 06
I =	29	1283.760	0.1507263E 09
I =	30	332.043	0.5113461E 06
I =	31	451.963	0.7118411E 08
I =	32	703.061	0.1221568E 08
I =	33	358.096	0.4747134E 08
I =	34	689.586	0.2381829E 08
I =	35	347.031	0.3031662E 07
I =	36	526.362	0.4527766E 08
I =	37	338.698	0.6859999E 07
I =	38	562.608	0.6976341E 06
I =	39	8022.195	0.5875615E 09
I =	40	480.359	0.1082393E 08
I =	41	350.271	0.4608162E 07
I =	42	309.282	0.9767426E 07
I =	43	114.357	0.1615174E 07
I =	44	202.159	0.8951931E 08
I =	45	553.553	0.1693872E 06
I =	46	414.016	0.4142482E 08
I =	47	685.423	0.2731410E 08
I =	48	246.503	0.2763302E 06
I =	49	1726.683	0.4834712E 05
I =	50	449.623	0.1928881E 07

ONE CENTRAL FACILITY

CASE OF WEIGHT DOMINANCE

LOCATION OF THE 3 FACILITIES

		CARTESIAN COORDINATES X	Y	TRANSPORT RATE R	QUANTITY TO TRANSPORT M
I=	1	11.000	2.000	1.000	10.000
I=	2	4.000	1.200	1.000	5.000
I=	3	1.500	5.500	1.000	3.000

LOCATION OF ONE CENTRAL FACILITY
CARTESIAN COORDINATES - EUCLIDEAN SPACE

	X	Y	TOTAL TRANSPORTATION COST
	7.472	2.361	74.008
	8.023	2.100	72.466
	8.516	2.091	71.272
	8.951	2.086	70.243
	9.326	2.077	69.370
	9.644	2.067	68.639
	9.910	2.056	68.034
	10.129	2.046	67.540
	10.307	2.038	67.139
	10.451	2.030	66.817
	10.567	2.024	66.559
	10.659	2.019	66.355
	10.732	2.015	66.192
	10.790	2.012	66.064
	10.835	2.009	65.964
	10.871	2.007	65.884
	10.899	2.006	65.822
	10.921	2.004	65.774
	10.938	2.003	65.736
	10.952	2.002	65.706
	10.962	2.002	65.683
LOCATION OF THE CENTRAL FACILITY AT THE 22 ITERATION	10.970	2.001	65.664

DISTANCES AND TRANSPORTATION COSTS
TO THE CENTRAL FACILITY

		DISTANCE	TRANSPORTATION COST
I=	1	0.029	0.292
I=	2	7.016	35.083
I=	3	10.096	30.288

ONE CENTRAL FACILITY

DESTINATION SUBSET ALGORITHM

ONE FACILITY IS USED AS CENTRAL LOCATION

```
C       LOCATION OF ONE CENTRAL FACILITY                           SPRING 1969
C               THE DESTINATION SUBSET ALGORITHM
C               ONE FACILITY IS USED AS CENTRAL LOCATION
C       DEFINITION OF MACHINE INPUT/OUTPUT: R READ
C                                            W WRITE
        INTEGER R,W
C       CHANGE THE DIMENSION CARD IF MORE THAN 50 FACILITIES ARE CONSIDERED
        DIMENSION X(  50),Y(  50),XR(  50),XM(  50)
        R=5
        W=6
        READ(R,10)N
     10 FORMAT(I10)
C       READ VARIABLES X(I),Y(I):CARTESIAN COORDINATES OF FACILITIES
C                             XR(I):TRANSPORT RATE ON ROUTE I
C                             XM(I):QUANTITY TO TRANSPORT ON ROUTE I
        READ(R,20)(X(I),Y(I),XR(I),XM(I),I=1,N)
     20 FORMAT(4F15.0)
        WRITE(W,30)
     30 FORMAT(1H1,//,39X,32HLOCATION OF ONE CENTRAL FACILITY,/,41X,
       128HDESTINATION SUBSET ALGORITHM,/,35X,40HONE FACILITY IS USED AS C
       2ENTRAL LOCATION,//,27X,21HCARTESIAN COORDINATES,6X,14HTRANSPORT RA
       3TE,3X,11HQUANTITY TO,/,72X,9HTRANSPORT,/,29X,1HX,14X,1HY,13X,1HR,
       415X,1HM,//)
        DO 40 I=1,N
        WRITE(W,35)I,X(I),Y(I),XR(I),XM(I)
     35 FORMAT(13X,2HI=,I5,F13.3,F15.3,F14.3,F16.3)
     40 CONTINUE
        WRITE(W,50)
     50 FORMAT(1H1,//,39X,32HLOCATION OF ONE CENTRAL FACILITY,/,35X,
       139HCARTESIAN COORDINATES - EUCLIDEAN SPACE,/,41X,28HDESTINATION SU
       2BSET ALGORITHM,/,35X,40HONE FACILITY IS USED AS CENTRAL LOCATION,
       3//,26X,28HCENTRAL FACILITY COORDINATES,6X,25HTOTAL TRANSPORTATION
       4COST,/,32X,1HX,14X,1HY,24X,1HC,//)
        CSAV=0.0
        DO 100 J=1,N
        C=0.0
        DO 60 I=1,N
        DP=(((X(J)-X(I))**2)+((Y(J)-Y(I))**2))**0.5
        DC=XR(I)*XM(I)*DP
        C=C+DC
     60 CONTINUE
        WRITE(W,70)J,X(J),Y(J),C
     70 FORMAT(15X,2HJ=,I5,F14.3,F15.3,F25.3)
        IF(J-1)80,90,80
     80 IF(C-CSAV)90,100,100
     90 CSAV=C
        JSAV=J
    100 CONTINUE
        WRITE(W,110)JSAV,X(JSAV),Y(JSAV),CSAV
    110 FORMAT(/,1X,16HCENTRAL LOCATION,/,8X,2HAT,5X,2HJ=,I5,F14.3,F15.3,
       1F25.3)
        CALL EXIT
        END
```

LOCATION OF ONE CENTRAL FACILITY
DESTINATION SUBSET ALGORITHM
ONE FACILITY IS USED AS CENTRAL LOCATION

	CARTESIAN COORDINATES X	CARTESIAN COORDINATES Y	TRANSPORT RATE R	QUANTITY TO TRANSPORT M
I= 1	7.190	5.490	1.000	2.000
I= 2	9.070	9.940	1.000	2.000
I= 3	4.610	6.490	1.000	2.000
I= 4	4.940	8.250	1.000	2.000
I= 5	0.470	0.690	1.000	2.000
I= 6	6.180	3.570	1.000	2.000
I= 7	1.130	9.810	1.000	2.000
I= 8	6.000	4.360	1.000	2.000
I= 9	8.230	8.060	1.000	2.000
I= 10	9.600	9.280	1.000	2.000
I= 11	3.460	9.680	1.000	2.000
I= 12	2.310	0.390	1.000	2.000
I= 13	2.530	4.570	1.000	2.000
I= 14	4.440	7.990	1.000	2.000
I= 15	8.530	1.490	1.000	2.000
I= 16	2.290	7.030	1.000	2.000
I= 17	8.830	7.120	1.000	2.000
I= 18	2.620	9.410	1.000	2.000
I= 19	3.820	2.420	1.000	2.000
I= 20	7.550	1.970	1.000	2.000

LOCATION OF ONE CENTRAL FACILITY
CARTESIAN COORDINATES - EUCLIDEAN SPACE
DESTINATION SUBSET ALGORITHM
ONE FACILITY IS USED AS CENTRAL LOCATION

		CENTRAL FACILITY COORDINATES X	Y	TOTAL TRANSPORTATION COS C
J=	1	7.190	5.490	169.706
J=	2	9.070	9.940	250.014
J=	3	4.610	6.490	155.955
J=	4	4.940	8.250	169.834
J=	5	0.470	0.690	302.881
J=	6	6.180	3.570	177.446
J=	7	1.130	9.810	254.922
J=	8	6.000	4.360	165.387
J=	9	8.230	8.060	198.313
J=	10	9.600	9.280	248.497
J=	11	3.460	9.680	208.244
J=	12	2.310	0.390	274.667
J=	13	2.530	4.570	190.018
J=	14	4.440	7.990	167.346
J=	15	8.530	1.490	254.922
J=	16	2.290	7.030	189.366
J=	17	8.830	7.120	203.038
J=	18	2.620	9.410	213.579
J=	19	3.820	2.420	204.816
J=	20	7.550	1.970	225.713
CENTRAL LOCATION AT J=	3	4.610	6.490	155.955

MULTIPLE CENTRAL FACILITIES

RANDOM DESTINATION SUBSET ALGORITHM

```
C     LOCATION OF MULTIPLE CENTRAL FACILITIES                     SPRING 1969
C              THE RANDOM DESTINATION SUBSET ALGORITHM
C     DEFINITION OF MACHINE INPUT/OUTPUT: R READ
C                                         W WRITE
      INTEGER R,W
C     CHANGE THE DIMENSION CARD IF MORE THAN 20 FACILITIES ARE CONSIDERED
C     CHANGE THE DIMENSION CARD IF MORE THAN  5 CENTRAL LOCATIONS ARE DESIRED
      DIMENSION X(  20),Y(  20),XR(  20),XM(  20)
      DIMENSION DOLD(  20),DNEW(  20),COLD(  20),CNEW(  20)
      DIMENSION JOSAV(  20),JNSAV(  20),IOSAV(   5),INSAV(    5)
      DIMENSION RAD(  20),XC(  5),YC(  5),D(  20,  20)
      R=5
      W=6
C     TERMINOLOGY OF VARIABLES
C         N          :NUMBER OF FACILITIES
C         M          :NUMBER OF CENTRAL FACILITIES
C         ITERA      :NUMBER OF ITERATIONS IN RANDOM SEARCH OF FACILITIES
C         X(I),Y(I):CARTESIAN COORDINATES OF FACILITIES
C         XR(I)      :TRANSPORT RATE ON ROUTE I
C         XM(I)      :QUANTITY TO TRANSPORT ON ROUTE I
C         DOLD(I),DNEW(I):OLD AND NEW EUCLIDEAN DISTANCES FROM FACILITY I TO
C                    OPTIMUM CENTRAL LOCATION
C         SDOLD,SDNEW:OLD AND NEW SUM OF DISTANCES TO THE CENTRAL FACILITIES
C         COLD(I),CNEW(I):OLD AND NEW TRANSPORTATION COSTS FROM FACILITY I TO
C                    OPTIMAL CENTRAL LOCATION
C         SCOLD,SCNEW:OLD AND NEW SUM OF TRANSPORT COST TO THE CENTRAL FACILITY
C         JOSAV(I),JNSAV(I):OLD AND NEW CODE NUMBER ALLOCATION OF THE FACILITY
C         IOSAV(J),INSAV(I):OLD AND NEW CODE NUMBER OF THE RANDOMLY SELECTED
C                    CENTRAL LOCATION
C         RAINC      :CLASS WIDTH ON CUMULATIVE DISTRIBUTION OF LOCATIONS
C         RAD(I)     :CLASS BOUNDARIES ON CUMULATIVE DISTRIBUTION OF LOCATIONS
C         KITER      :ITERATION COUNTER
C         YFL        :RANDOM NUMBER BETWEEN 0 AND 1.000
C         XC(J),YC(J) :CARTESIAN COORDINATES OF RANDOMLY SELECTED CENTRAL
C                    FACILITIES
C         D(I,J)     :EUCLIDEAN DISTANCES FROM FACILITY I TO CENTRAL LOCATION J
      READ(R,10)N,M,ITERA
   10 FORMAT(3I10)
      READ(R,20)(X(I),Y(I),XR(I),XM(I),I=1,N)
   20 FORMAT(4F15.0)
C     FIRST TABULATION: LIST GIVEN VARIABLES
      WRITE(W,30)M,N
   30 FORMAT(1H1,//,35X,39HLOCATION OF MULTIPLE CENTRAL FACILITIES,/,
     137X,35HRANDOM DESTINATION SUBSET ALGORITHM,//,43X,I3,1X,
     218HCENTRAL FACILITIES,/,46X,I5,1X,10HFACILITIES,//,24X,
     321HCARTESIAN COORDINATES,6X,14HTRANSPORT RATE,3X,11HQUANTITY TO,/,
     469X,9HTRANSPORT,/,27X,1HX,14X,1HY,15X,1HR,14X,1HM,//)
      DO 36 I=1,N
      WRITE (W,35) I,X(I),Y(I),XR(I),XM(I)
   35 FORMAT (10X,3HI =,I5,4(4X,F11.3))
   36 CONTINUE
C     ZERO ALL COMPUTATIONAL VARIABLES
      DO 40 I=1,N
      DOLD(I)=0.0
```

```
      DNEW(I)=0.0
      COLD(I)=0.0
      CNEW(I)=0.0
      JOSAV(I)=0
   40 JNSAV(I)=0
      DO 45 J=1,M
      IOSAV(J)=0
   45 INSAV(J)=0
      SDOLD=0.0
      SDNEW=0.0
      SCOLD=0.0
      SCNEW=0.0
C     ZERO ITERATION COUNTER
      KITER=0
C     DEFINITION OF CLASS INTERVALS FOR 0 TO 1 RANDUM NUMBER DEFINING THE 0 TO N
C     INTEGERS
      XN=N
      RAINC =1.0/XN
      DO 50 I=1,N
      XI=I
   50 RAD(I)=XI*RAINC
C     RANDOM GENERATOR STARTING VALUE
      IY=21735
C     STARTING OF ITERATION COUNTER
   60 KITER=KITER+1
      IF(KITER-ITERA)70,70,275
C     RANDOM CHOICE OF M CENTRAL FACILITIES
   70 DO 100 J=1,M
   75 CALL RANDU(IY,IY,YFL)
      DO 90 I=1,N
      IF (YFL-RAD(I))80,80,90
   80 XC(J)=X(I)
      YC(J)=Y(I)
      INSAV(J)=I
      IF(J-1)82,100,82
   82 K=J-1
C     CHECK THAT RANDOMLY CHOSEN FACILITY HAS NOT BEEN ALREADY PICKED AS CENTRAL
C     LOCATION
      DO 84 KJ=1,K
      IF(XC(J)-XC(KJ))100,83,100
   83 IF(YC(J)-YC(KJ))100,75,100
   84 CONTINUE
   90 CONTINUE
  100 CONTINUE
C     COMPUTE ARRAY OF EUCLIDEAN DISTANCES TO RANDOMLY CHOSEN CENTRAL FACILITIES
      DO 110 I=1,N
      DO 110 J=1,M
  110 D(I,J)=(((XC(J)-X(I))**2)+((YC(J)-Y(I))**2))**0.5
C     FOR EACH FACILITY SELECT THE CLOSER CENTRAL LOCATION (IF DISTANCE IS ZERO
C     IT MEANS THAT THE FACILITY HAS BEEN CHOSEN AS CENTRAL LOCATION)
      DO 170 I=1,N
      SHORT=D(I,1)
      J=1
      DO 160 K=2,M
```

```
      IF(D(I,K)-SHORT)150,160,160
  150 SHORT=D(I,K)
      J=K
  160 CONTINUE
      DNEW(I)=SHORT
      CNEW(I)=XR(I)*XM(I)*DNEW(I)
      JNSAV(I)=J
  170 CONTINUE
C     COMPUTE SUM OF OPTIMAL DISTANCES AND TOTAL TRANSPORTATION COST
      SDNEW=0.0
      SCNEW=0.0
      DO 180 I=1,N
      SDNEW=SDNEW+DNEW(I)
  180 SCNEW=SCNEW+CNEW(I)
      IF(KITER-1)190,190,210
C     TITLE OF SECOND TABULATION
  190 WRITE(W,200)
  200 FORMAT(1H1,//,37X,35HRANDOM DESTINATION SUBSET ALGORITHM,/,34X,
     141HDISTRIBUTION OF TOTAL DISTANCES AND COSTS,//,31X,9HITERATION,
     25X,14HSUM OF OPTIMUM,5X,16HSUM OF TRANSPORT,/,32X,6HNUMBER,9X,
     39HDISTANCES,12X,5HCOSTS,//)
C     BODY OF SECOND TABULATION
  210 WRITE(W,220)KITER,SDNEW,SCNEW
  220 FORMAT(30X,I7,8X,E13.6,7X,E13.6)
C     IF THIS IS THE FIRST ITERATION REPEAT RANDOM CHOICE ANOTHER TIME
      IF(SCOLD)230,240,230
C     CHECK IF NEW CHOICE OF CENTRAL FACILITIES GIVES BETTER RESULTS
  230 IF(SCNEW-SCOLD)240,240,270
  240 SDOLD=SDNEW
      SCOLD=SCNEW
      DO 250 I=1,N
      DOLD(I)=DNEW(I)
      COLD(I)=CNEW(I)
  250 JOSAV(I)=JNSAV(I)
      DO 260 J=1,M
  260 IOSAV(J)=INSAV(J)
  270 GO TO 60
C     THIRD TABULATION : OPTIMUM ALLOCATION
  275 WRITE(W,280)ITERA
  280 FORMAT(1H1,//,37X,35HRANDOM DESTINATION SUBSET ALGORITHM,/,35X,
     124HOPTIMUM ALLOCATION AFTER,I5,1X,10HITERATIONS,//,35X,
     239HCODE AND LOCATION OF CENTRAL FACILITIES,/,45X,4HCODE,8X,
     38HFACILITY,/,44X,6HNUMBER,8X,6HNUMBER,//)
      DO 295 J=1,M
      WRITE(W,290)J,IOSAV(J)
  290 FORMAT(44X,2HJ=,I4,8X,2HI=,I4)
  295 CONTINUE
      WRITE(W,300)
  300 FORMAT(//,46X,18HOPTIMUM ALLOCATION,//,18X,19HF A C I L I T I E S,
     110X,16HCENTRAL LOCATION,6X,11HDISTANCE TO,7X,14HTRANSPORTATION,/,
     212X,6HNUMBER,4X,21HCARTESIAN COORDINATES,9X,4HCODE,11X,
     316HCENTRAL LOCATION,9X,5HCOSTS,/,25X,1HX,13X,1HY,//)
      WRITE(W,310)(I,X(I),Y(I),JOSAV(I),DOLD(I),COLD(I),I=1,N)
  310 FORMAT(11X,3HI =,I4,2X,F11.3,3X,F11.3,7X,I3,10X,E16.6,4X,E15.6)
```

```
      WRITE(W,320)SDOLD,SCOLD
  320 FORMAT(/,62X,5HTOTAL,E14.6,4X,E15.6)
      CALL EXIT
      END
```

LOCATION OF MULTIPLE CENTRAL FACILITIES
RANDOM DESTINATION SUBSET ALGORITHM

3 CENTRAL FACILITIES
20 FACILITIES

	CARTESIAN COORDINATES		TRANSPORT RATE	QUANTITY TO TRANSPORT
	X	Y	R	M
I = 1	7.190	5.490	1.000	2.000
I = 2	9.070	9.940	1.000	2.000
I = 3	4.610	6.490	1.000	2.000
I = 4	4.940	8.250	1.000	2.000
I = 5	0.470	0.690	1.000	2.000
I = 6	6.180	3.570	1.000	2.000
I = 7	1.130	9.810	1.000	2.000
I = 8	6.000	4.360	1.000	2.000
I = 9	8.230	8.060	1.000	2.000
I = 10	9.600	9.280	1.000	2.000
I = 11	3.460	9.680	1.000	2.000
I = 12	2.310	0.390	1.000	2.000
I = 13	2.530	4.570	1.000	2.000
I = 14	4.440	7.990	1.000	2.000
I = 15	8.530	1.490	1.000	2.000
I = 16	2.290	7.030	1.000	2.000
I = 17	8.830	7.120	1.000	2.000
I = 18	2.620	9.410	1.000	2.000
I = 19	3.820	2.420	1.000	2.000
I = 20	7.550	1.970	1.000	2.000

RANDOM DESTINATION SUBSET ALGORITHM
DISTRIBUTION OF TOTAL DISTANCES AND COSTS

ITERATION NUMBER	SUM OF OPTIMUM DISTANCES	SUM OF TRANSPORT COSTS
1	0.494680E 02	0.989361E 02
2	0.622942E 02	0.124589E 03
3	0.512428E 02	0.102486E 03
4	0.522075E 02	0.104415E 03
5	0.573191E 02	0.114638E 03
6	0.629998E 02	0.126000E 03
7	0.482333E 02	0.964668E 02
8	0.626795E 02	0.125359E 03
9	0.444866E 02	0.889732E 02
10	0.622860E 02	0.124572E 03
11	0.606394E 02	0.12127[illegible]
12	0.810183E 02	[illegible]
13	0.633401E 02	
14	0.46483[illegible]	
15	[illegible]	
16		
		[illegible]E 03
		[illegible].986447E 02
		0.120742E 03
	[illegible]4E 02	0.138589E 03
	0.499599E 02	0.999198E 02
	0.597973E 02	0.119595E 03
479	0.700885E 02	0.140177E 03
480	0.599969E 02	0.119994E 03
481	0.552149E 02	0.110430E 03
482	0.583724E 02	0.116745E 03
483	0.540727E 02	0.108146E 03
484	0.499599E 02	0.999198E 02
485	0.582791E 02	0.116558E 03
486	0.542777E 02	0.108556E 03
487	0.525178E 02	0.105036E 03
488	0.655060E 02	0.131012E 03
489	0.490505E 02	0.981011E 02
490	0.622432E 02	0.124486E 03
491	0.558385E 02	0.111677E 03
492	0.530840E 02	0.106168E 03
493	0.491263E 02	0.982527E 02
494	0.716451E 02	0.143290E 03
495	0.528780E 02	0.105756E 03
496	0.536986E 02	0.107397E 03
497	0.590859E 02	0.118172E 03
498	0.600908E 02	0.120182E 03
499	0.818133E 02	0.163627E 03
500	0.702484E 02	0.140497E 03

RANDOM DESTINATION SUBSET ALGORITHM
OPTIMUM ALLOCATION AFTER 500 ITERATIONS

CODE AND LOCATION OF CENTRAL FACILITIES

CODE NUMBER	FACILITY NUMBER
J= 1	I= 14
J= 2	I= 17
J= 3	I= 19

OPTIMUM ALLOCATION

FACILITIES NUMBER	CARTESIAN COORDINATES X	Y	CENTRAL LOCATION CODE	DISTANCE TO CENTRAL LOCATION	TRANSPORTATION COSTS
I = 1	7.190	5.490	2	0.231225E 01	0.462450E 01
I = 2	9.070	9.940	2	0.283019E 01	0.566039E 01
I = 3	4.610	6.490	1	0.150960E 01	0.301921E 01
I = 4	4.940	8.250	1	0.563560E 00	0.112712E 01
I = 5	0.470	0.690	3	0.377033E 01	0.754066E 01
I = 6	6.180	3.570	3	0.262528E 01	0.525056E 01
I = 7	1.130	9.810	1	0.377737E 01	0.755473E 01
I = 8	6.000	4.360	3	0.291822E 01	0.583644E 01
I = 9	8.230	8.060	2	0.111517E 01	0.223034E 01
I = 10	9.600	9.280	2	0.229314E 01	0.458628E 01
I = 11	3.460	9.680	1	0.195359E 01	0.390717E 01
I = 12	2.310	0.390	3	0.253002E 01	0.506004E 01
I = 13	2.530	4.570	3	0.250731E 01	0.501462E 01
I = 14	4.440	7.990	1	0.0	0.0
I = 15	8.530	1.490	3	0.480093E 01	0.960187E 01
I = 16	2.290	7.030	1	0.235459E 01	0.470918E 01
I = 17	8.830	7.120	2	0.0	0.0
I = 18	2.620	9.410	1	0.230842E 01	0.461684E 01
I = 19	3.820	2.420	3	0.0	0.0
I = 20	7.550	1.970	3	0.375704E 01	0.751409E 01
			TOTAL	0.439270E 02	0.878539E 02

MULTIPLE CENTRAL FACILITIES
RANDOM GRID LOCATION ALGORITHM

```
FORTRAN IV G LEVEL 1, MOD 3  MAIN                  DATE 06/18/15                    PAGE 0001

C     LOCATION OF MULTIPLE CENTRAL FACILITIES                                 SPRING 1969
C                      THE RANDOM GRID LOCATION ALGORITHM
C     DEFINITION OF MACHINE INPUT/OUTPUT: R READ
C                                         W WRITE
      INTEGER R,W
C     CHANGE THE DIMENSION CARD IF MORE THAN 20 FACILITIES ARE CONSIDERED
      DIMENSION X(  20),Y(  20),XR(  20),XM(  20),D(  20,  20)
      DIMENSION DOLD(  20),DNEW(  20),COLD(  20),CNEW(  20)
      DIMENSION JOSAV(  20),JNSAV(  20),RAD(  20)
C     CHANGE THE DIMENSION CARD IF MORE THAN  5 CENTRAL LOCATIONS ARE DESIRED
      DIMENSION IOSAV(  5),INSAV(  5),XC(  5),YC(  5)
C     CHANGE THE DIMENSION CARD IF MORE THAN 100 GRID INTERSECT. ARE CONSIDERED
      DIMENSION XG( 100),YG( 100)
      R=5
      W=6
C     TERMINOLOGY OF VARIABLES
C           N             :NUMBER OF FACILITIES
C           M             :NUMBER OF CENTRAL FACILITIES
C           ITERA         :NUMBER OF ITERATIONS IN RANDOM SEARCH OF FACILITIES
C           IGRID         :INITIAL NUMBER OF GRID DIVISIONS ON EACH X AND Y AXIS
C           ITGRD         :NUMBER OF GRID SIZE CHANGES
C           INC           :INCREMENTAL NUMBER OF DIVISIONS ON EACH X AND Y AXIS WHEN
C                          PASSING FROM ONE GRID SIZE TO THE NEXT
C           X(I),Y(I):CARTESIAN COORDINATES OF FACILITIES
C           XR(I)         :TRANSPORT RATE ON ROUTE I
C           XM(I)         :QUANTITY TO TRANSPORT ON ROUTE I
C           DOLD(I),DNEW(I):OLD AND NEW EUCLIDEAN DISTANCES FROM FACILITY I TO
C                          OPTIMUM CENTRAL LOCATION
C           SDOLD,SDNEW:OLD AND NEW SUM OF DISTANCES TO THE CENTRAL FACILITIES
C           COLD(I),CNEW(I):OLD AND NEW TRANSPORTATION COSTS FROM FACILITY I TO
C                          OPTIMAL CENTRAL LOCATION
C           SCOLD,SCNEW:OLD AND NEW SUM OF TRANSPORT COST TO THE CENTRAL FACILITY
C           JOSAV(I),JNSAV(I):OLD AND NEW CODE NUMBER ALLOCATION OF THE FACILITY
C           IOSAV(J),INSAV(I):OLD AND NEW CODE NUMBER OF THE RANDOMLY SELECTED
C                          CENTRAL LOCATION
C           RAINC         :CLASS WIDTH ON CUMULATIVE DISTRIBUTION OF LOCATIONS
C           RAD(I)        :CLASS BOUNDARIES ON CUMULATIVE DISTRIBUTION OF LOCATIONS
C           KITER         :ITERATION COUNTER
C           YFL           :RANDOM NUMBER BETWEEN 0 AND 1.000
C           XC(J),YC(J) :CARTESIAN COORDINATES OF RANDOMLY SELECTED CENTRAL
C                          FACILITIES
C           D(I,J)        :EUCLIDEAN DISTANCES FROM FACILITY I TO CENTRAL LOCATION J
C           L             :GRID SPACING COUNTER
C           XMIN,XMAX:MINIMUM AND MAXIMUM VALUES OF X(I)
C           YMIN,YMAX:MINIMUM AND MAXIMUM VALUES OF Y(I)
      READ(R,10)N,M,ITERA,IGRID,INC,ITGRD
   10 FORMAT(6I10)
      READ(R,20)(X(I),Y(I),XR(I),XM(I),I=1,N)
   20 FORMAT(4F15.0)
C     FIRST TABULATION: LIST GIVEN VARIABLES
      WRITE(W,30)ITGRD,ITERA,IGRID,INC,M,N
   30 FORMAT(1H1,//,35X,39HLOCATION OF MULTIPLE CENTRAL FACILITIES,/,
     140X,30HRANDOM GRID LOCATION ALGORITHM,//,29X,I3,1X,20HGRID SPACING
```

```
     2 CHANGES,4X,I4,1X,19HITERATIONS PER GRID,/,15X,33HINITIAL NUMBER O
     3F GRID DIVISIONS ,I4,4X,36HDIVISION INCREMENTS PER GRID CHANGE ,I4
     4,//,43X,I3,1X,18HCENTRAL FACILITIES,/,46X,I5,1X,10HFACILITIES,//)
      WRITE(W,32)
   32 FORMAT(24X,21HCARTESIAN COORDINATES,6X,14HTRANSPORT RATE,3X,
     111HQUANTITY TO,/,69X,9HTRANSPORT,/,27X,1HX,14X,1HY,15X,1HP,14X,1HM
     2,//)
      DO 36 I=1,N
      WRITE (W,35) I,X(I),Y(I),XR(I),XM(I)
   35 FORMAT (10X,3HI =,I5,4(4X,F11.3))
   36 CONTINUE
C     GRID COUNTER SET FOR FIRST GRID INVESTIGATION
      L=1
C     DEFINE X MAXIMUM, Y MAXIMUM
      XMAX=X(1)
      DO 80 I=2,N
      IF(X(I)-XMAX)80,80,70
   70 XMAX=X(I)
   80 CONTINUE
      YMAX=Y(1)
      DO 100 I=2,N
      IF(Y(I)-YMAX)100,90,90
   90 YMAX=Y(I)
  100 CONTINUE
C     DEFINE X MINIMUM, Y MINIMUM
      XMIN=X(1)
      DO 120 I=2,N
      IF(X(I)-XMIN)110,120,120
  110 XMIN=X(I)
  120 CONTINUE
      YMIN=Y(1)
      DO 140 I=2,N
      IF(Y(I)-YMIN)130,140,140
  130 YMIN=Y(I)
  140 CONTINUE
C     DEFINE RANGE OF X AND Y
      RANGX=XMAX-XMIN
      RANGY=YMAX-YMIN
C     COMPUTE NUMBER OF GRID INTERSECTION POINTS
  150 NBER=IGRID+1
      NG=(NBER)**2
C     CHECK IF NUMBER OF GRID INTERSECTION POINTS IS LARGE ENOUGH
      IF(NG-M)160,160,170
  160 IGRID=IGRID+INC
      GO TO 150
C     DEFINE GRID SPACING
  170 GRID=IGRID
      XINC=RANGX/GRID
      YINC=RANGY/GRID
C     DEFINE GRID INTERSECTION POINTS
      DO 180 I=1,NBER
      IX=I-1
      DO 180 J=1,NBER
      XK=J-1
```

```
      K=J+IX*NBER
      XG(K)=XMIN+(XK*XINC)
  180 YG(K)=YMIN+(IX*YINC)
C     ZERO ALL COMPUTATIONAL VARIABLES
      DO 50 I=1,N
      DOLD(I)=0.0
      DNEW(I)=0.0
      COLD(I)=0.0
      CNEW(I)=0.0
      JOSAV(I)=0
   50 JNSAV(I)=0
      DO 60 J=1,M
      IOSAV(J)=0
   60 INSAV(J)=0
      SDOLD=0.0
      SDNEW=0.0
      SCOLD=0.0
      SCNEW=0.0
C     ZERO ITERATION COUNTER
      KITER=0
C     DEFINITION OF CLASS INTERVALS FOR 0 TO 1 RANDOM NUMBER DEFINING THE
C     0 TO NG INTEGERS
      XNG=NG
      RAINC=1.0/XNG
      DO 190 I=1,NG
      XI=I
  190 RAD(I)=XI*RAINC
C     RANDOM GENERATOR STARTING VALUE
      IY=21735
C     STARTING OF ITERATION COUNTER FOR A GIVEN GRID
  200 KITER=KITER+1
      IF(KITER-ITERA)210,210,390
C     RANDOM CHOICE OF M CENTRAL FACILITIES FROM THE NG GRID POINTS
  210 DO 240 J=1,M
  215 CALL RANDU(IY,IY,YFL)
      DO 230 I=1,NG
      IF (YFL-RAD(I))220,220,230
  220 XC(J)=XG(I)
      YC(J)=YG(I)
      INSAV(J)=I
      IF(J-1)222,240,222
  222 K=J-1
C     CHECK THAT RANDOMLY CHOSEN GRID POINT HAS NOT BEEN ALREADY PICKED AS
C     CENTRAL LOCATION
      DO 224 KJ=1,K
      IF(INSAV(J)-INSAV(KJ))240,215,240
  224 CONTINUE
  230 CONTINUE
  240 CONTINUE
C     COMPUTE ARRAY OF EUCLIDEAN DISTANCES TO RANDOMLY CHOSEN CENTRAL FACILITIE
      DO 250 I=1,N
      DO 250 J=1,M
  250 D(I,J)=(((XC(J)-X(I))**2)+((YC(J)-Y(I))**2))**0.5
C     FOR EACH FACILITY SELECT THE CLOSER CENTRAL LOCATION
```

```
      DO 280 I=1,N
      SHORT=D(I,1)
      J=1
      DO 270 K=2,M
      IF(D(I,K)-SHORT)260,270,270
  260 SHORT=D(I,K)
      J=K
  270 CONTINUE
      DNEW(I)=SHORT
      CNEW(I)=XR(I)*XM(I)*DNEW(I)
      JNSAV(I)=J
  280 CONTINUE
C     COMPUTE SUM OF OPTIMAL DISTANCES AND TOTAL TRANSPORTATION COST
      SDNEW=0.0
      SCNEW=0.0
      DO 290 I=1,N
      SDNEW=SDNEW+DNEW(I)
  290 SCNEW=SCNEW+CNEW(I)
      IF(KITER-1)300,300,320
C     TITLE OF SECOND TABULATION
  300 WRITE(W,310)
  310 FORMAT(1H1,//,40X,30HRANDOM GRID LOCATION ALGORITHM,/,34X,
     141HDISTRIBUTION OF TOTAL DISTANCES AND COSTS,//,31X,9HITERATION,
     25X,14HSUM OF OPTIMUM,5X,16HSUM OF TRANSPORT,/,32X,6HNUMBER,9X,
     39HDISTANCES,12X,5HCOSTS,//)
C     BODY OF SECOND TABULATION
  320 WRITE(W,330)KITER,SDNEW,SCNEW
  330 FORMAT(30X,I7,8X,E13.6,7X,E13.6)
C     IF THIS IS THE FIRST ITERATION REPEAT RANDOM CHOICE ANOTHER TIME
      IF(SCOLD)340,350,340
C     CHECK IF NEW CHOICE OF CENTRAL FACILITIES GIVES BETTER RESULTS
  340 IF(SCNEW-SCOLD)350,350,380
  350 SDOLD=SDNEW
      SCOLD=SCNEW
      DO 360 I=1,N
      DOLD(I)=DNEW(I)
      COLD(I)=CNEW(I)
  360 JOSAV(I)=JNSAV(I)
      DO 370 J=1,M
  370 IOSAV(J)=INSAV(J)
  380 GO TO 200
C     THIRD TABULATION : OPTIMUM ALLOCATION
  390 WRITE(W,400)L,ITERA
  400 FORMAT(1H1,//,40X,30HRANDOM GRID LOCATION ALGORITHM,/,40X,
     128HOPTIMUM ALLOCATION ON GRID #,I3,/,45X,5HAFTER,1X,I4,1X,
     210HITERATIONS,//,35X,39HCODE AND LOCATION OF CENTRAL FACILITIES,/,
     333X,11HCODE NUMBER,10X,21HCARTESIAN COORDINATES,/,57X,1HX,14X,1HY,
     4//)
      DO 415 J=1,M
      JJ=IOSAV(J)
      WRITE(W,410)J,XG(JJ),YG(JJ)
  410 FORMAT(34X,3HJ =,I4,9X,F11.3,4X,F11.3)
  415 CONTINUE
      WRITE(W,420)
```

```
  420 FORMAT(//,46X,18HOPTIMUM ALLOCATION,//,13X,19HF A C I L I T I E S,
     110X,16HCENTRAL LOCATION,6X,11HDISTANCE TO,7X,14HTRANSPORTATION,/,
     212X,6HNUMBER,4X,21HCARTESIAN COORDINATES,9X,4HCODE,11X,
     316HCENTRAL LOCATION,9X,5HCOSTS,/,25X,1HX,13X,1HY,//)
      WRITE(W,430)(I,X(I),Y(I),JOSAV(I),DOLD(I),COLD(I),I=1,N)
  430 FORMAT(11X,3HI =,I4,2X,F11.3,3X,F11.3,7X,I3,10X,E16.6,4X,E15.6)
      WRITE(W,435)SDOLD,SCOLD
  435 FORMAT(/,62X,5HTOTAL,E14.6,4X,F15.6)
C     CHECK IF ALL GRID CHANGES HAVE BEEN DONE
      IF(L-ITGRD)440,450,450
  440 L=L+1
      GO TO 160
  450 CALL EXIT
      END
```

LOCATION OF MULTIPLE CENTRAL FACILITIES
RANDOM GRID LOCATION ALGORITHM

5 GRID SPACING CHANGES   100 ITERATIONS PER GRID
INITIAL NUMBER OF GRID DIVISIONS 3 DIVISION INCREMENTS PER GRID CHANGE 1

3 CENTRAL FACILITIES
20 FACILITIES

	CARTESIAN COORDINATES X	Y	TRANSPORT RATE R	QUANTITY TO TRANSPORT M
I = 1	7.190	5.490	1.000	2.000
I = 2	9.070	9.940	1.000	2.000
I = 3	4.610	6.490	1.000	2.000
I = 4	4.940	8.250	1.000	2.000
I = 5	0.470	0.690	1.000	2.000
I = 6	6.180	3.570	1.000	2.000
I = 7	1.130	9.810	1.000	2.000
I = 8	6.000	4.360	1.000	2.000
I = 9	8.230	8.060	1.000	2.000
I = 10	9.600	9.280	1.000	2.000
I = 11	3.460	9.680	1.000	2.000
I = 12	2.310	0.390	1.000	2.000
I = 13	2.530	4.570	1.000	2.000
I = 14	4.440	7.990	1.000	2.000
I = 15	8.530	1.490	1.000	2.000
I = 16	2.290	7.030	1.000	2.000
I = 17	8.830	7.120	1.000	2.000
I = 18	2.620	9.410	1.000	2.000
I = 19	3.820	2.420	1.000	2.000
I = 20	7.550	1.970	1.000	2.000

RANDOM GRID LOCATION ALGORITHM
DISTRIBUTION OF TOTAL DISTANCES AND COSTS

ITERATION NUMBER	SUM OF OPTIMUM DISTANCES	SUM OF TRANSPORT COSTS
1	0.720299E 02	0.144060E 03
2	0.626835E 02	0.125367E 03
3	0.772214E 02	0.154443E 03
4	0.704148E 02	0.140830E 03
5	0.732473E 02	0.145495E 03
6	0.663145E 02	0.132629E 03
7	0.711765E 02	0.142353E 03
8	0.914100E 02	0.182820E 03
9	0.652524E 02	0.130505E 03
10	0.640542E 02	0.128109E 03
11	0.782951E 02	0.156590E 03
12	0.874874E 02	0.174975E 03
13	0.588681E 02	0.117736E 03
14	0.985630E 02	0.197126[illegible]
15	0.613379E 02	[illegible]
16	0.722191E 02	[illegible]
17	0.658111[illegible]	[illegible]
18	[illegible]	[illegible]
19	[illegible]	[illegible]
[illegible]	[illegible]	[illegible] 03
[illegible]	[illegible]	[illegible]117611E 03
[illegible]	[illegible]	0.113195E 03
[illegible]	[illegible] 02	0.155366E 03
[illegible]	[illegible].685871E 02	0.137174E 03
[illegible]	0.702514E 02	0.140503E 03
92	0.878319E 02	0.175664E 03
93	0.594903E 02	0.118981E 03
94	0.542861E 02	0.108572E 03
95	0.781114E 02	0.156223E 03
96	0.850261E 02	0.170052E 03
97	0.874333E 02	0.174867E 03
98	0.809427E 02	0.161885E 03
99	0.555136E 02	0.111027E 03
100	0.574836E 02	0.114967E 03

RANDOM GRID LOCATION ALGORITHM
OPTIMUM ALLOCATION ON GRID # 1
AFTER 100 ITERATIONS

CODE AND LOCATION OF CENTRAL FACILITIES

CODE NUMBER	CARTESIAN COORDINATES X	Y
J = 1	6.557	6.757
J = 2	3.513	0.390
J = 3	3.513	9.940

OPTIMUM ALLOCATION

F A C I L I T I E S NUMBER	CARTESIAN COORDINATES X	Y	CENTRAL LOCATION CODE	DISTANCE TO CENTRAL LOCATION	TRANSPORTATION COSTS
I = 1	7.190	5.490	1	0.141617E 01	0.283235E 01
I = 2	9.070	9.940	1	0.405592E 01	0.811184E 01
I = 3	4.610	6.490	1	0.196484E 01	0.392969E 01
I = 4	4.940	8.250	1	0.220083E 01	0.440166E 01
I = 5	0.470	0.690	2	0.305808E 01	0.611616E 01
I = 6	6.180	3.570	1	0.320885E 01	0.641769E 01
I = 7	1.130	9.810	3	0.238688E 01	0.477375E 01
I = 8	6.000	4.360	1	0.246046E 01	0.492092E 01
I = 9	8.230	8.060	1	0.212102E 01	0.424204E 01
I = 10	9.600	9.280	1	0.395336E 01	0.790673E 01
I = 11	3.460	9.680	3	0.265411E 00	0.530822E 00
I = 12	2.310	0.390	2	0.120333E 01	0.240657E 01
I = 13	2.530	4.570	2	0.429410E 01	0.858820E 01
I = 14	4.440	7.990	3	0.215898E 01	0.431796E 01
I = 15	8.530	1.490	2	0.513585E 01	0.102717E 02
I = 16	2.290	7.030	3	0.315668E 01	0.631335E 01
I = 17	8.830	7.120	1	0.230219E 01	0.460437E 01
I = 18	2.620	9.410	3	0.103872E 01	0.207744E 01
I = 19	3.820	2.420	2	0.205303E 01	0.410606E 01
I = 20	7.550	1.970	2	0.433486E 01	0.866973E 01
			TOTAL	0.527695E 02	0.105539E 03

RANDOM GRID LOCATION ALGORITHM
DISTRIBUTION OF TOTAL DISTANCES AND COSTS

ITERATION NUMBER	SUM OF OPTIMUM DISTANCES	SUM OF TRANSPORT COSTS
1	0.725670E 02	0.145134E 03
2	0.527695E 02	0.105539E 03
3	0.608446E 02	0.121689E 03
4	0.754811E 02	0.150962E 03
5	0.781114E 02	0.156223E 03
6	0.794097E 02	0.158820E 03
7	0.542861E 02	0.108572E 03
8	0.805535E 02	0.161107E 03
9	0.763510E 02	0.152702E 03
10	0.722993E 02	0.144599E 03
11	0.657194E 02	0.131439E 03
12	0.645500E 02	0.129100E 03
13	0.110114E 03	0.220228E 03
14	0.595514E 02	0.119103E [illegible]
15	0.779213E 02	[illegible]
16	0.725670E 02	[illegible]
17	0.57796[illegible]	[illegible]
18	[illegible]	[illegible]
19	[illegible]	[illegible]
[illegible]	[illegible]	[illegible] 03
[illegible]	[illegible]	[illegible]125367E 03
[illegible]	[illegible]	0.112681E 03
[illegible]	[illegible]E 02	0.131510E 03
[illegible]	[illegible].623930E 02	0.124786E 03
[illegible]	0.797754E 02	0.159551E 03
86	0.572911E 02	0.114582E 03
87	0.708918E 02	0.141784E 03
88	0.577336E 02	0.115467E 03
89	0.736609E 02	0.147322E 03
90	0.640542E 02	0.128109E 03
91	0.597877E 02	0.119575E 03
92	0.926379E 02	0.185276E 03
93	0.648323E 02	0.129665E 03
94	0.925606E 02	0.185121E 03
95	0.586081E 02	0.117216E 03
96	0.781926E 02	0.156385E 03
97	0.731762E 02	0.146353E 03
98	0.576437E 02	0.115288E 03
99	0.866447E 02	0.173290E 03
100	0.778810E 02	0.155762E 03

RANDOM GRID LOCATION ALGORITHM
OPTIMUM ALLOCATION ON GRID # 2
AFTER 100 ITERATIONS

CODE AND LOCATION OF CENTRAL FACILITIES

CODE NUMBER	CARTESIAN COORDINATES X	Y
J = 1	5.035	2.777
J = 2	5.035	7.552
J = 3	7.317	5.165

OPTIMUM ALLOCATION

F A C I L I T I E S NUMBER	CARTESIAN COORDINATES X	Y	CENTRAL LOCATION CODE	DISTANCE TO CENTRAL LOCATION	TRANSPORTATION COSTS
I = 1	7.190	5.490	3	0.349115E 00	0.698229E 00
I = 2	9.070	9.940	2	0.468843E 01	0.937686E 01
I = 3	4.610	6.490	2	0.114435E 01	0.228869E 01
I = 4	4.940	8.250	2	0.703941E 00	0.140788E 01
I = 5	0.470	0.690	1	0.501964E 01	0.100393E 02
I = 6	6.180	3.570	1	0.139251E 01	0.278502E 01
I = 7	1.130	9.810	2	0.451058E 01	0.902115E 01
I = 8	6.000	4.360	3	0.154396E 01	0.308793E 01
I = 9	8.230	8.060	3	0.303541E 01	0.607081E 01
I = 10	9.600	9.280	3	0.470564E 01	0.941128E 01
I = 11	3.460	9.680	2	0.264705E 01	0.529410E 01
I = 12	2.310	0.390	1	0.362295E 01	0.724590E 01
I = 13	2.530	4.570	1	0.308027E 01	0.616054E 01
I = 14	4.440	7.990	2	0.738532E 00	0.147706E 01
I = 15	8.530	1.490	1	0.372460E 01	0.744921E 01
I = 16	2.290	7.030	2	0.279428E 01	0.558857E 01
I = 17	8.830	7.120	3	0.247178E 01	0.494356E 01
I = 18	2.620	9.410	2	0.304672E 01	0.609344E 01
I = 19	3.820	2.420	1	0.126650E 01	0.253300E 01
I = 20	7.550	1.970	1	0.264145E 01	0.528291E 01
			TOTAL	0.531276E 02	0.106255E 03

RANDOM GRID LOCATION ALGORITHM
DISTRIBUTION OF TOTAL DISTANCES AND COSTS

ITERATION NUMBER	SUM OF OPTIMUM DISTANCES	SUM OF TRANSPORT COSTS
1	0.854613E 02	0.170923E 03
2	0.663508E 02	0.132702E 03
3	0.798092E 02	0.159619E 03
4	0.601718E 02	0.120344E 03
5	0.751462E 02	0.150293E 03
6	0.593459E 02	0.118692E 03
7	0.672455E 02	0.134491E 03
8	0.594032E 02	0.118806E 03
9	0.764203E 02	0.152841E 03
10	0.886103E 02	0.177221E 03
11	0.836693E 02	0.167339E 03
12	0.863575E 02	0.172715E 03
13	0.940181E 02	0.188036E 03
14	0.664655E 02	0.132931E [illegible]
15	0.912603E 02	0.1[illegible]
16	0.771456E 02	[illegible]
17	0.61723[illegible]	[illegible]
18	[illegible]	[illegible]
19	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]E 03
[illegible]	[illegible]	[illegible].167567E 03
[illegible]	[illegible]	0.101695E 03
[illegible]	[illegible]05E 02	0.166473E 03
[illegible]	[illegible].816075E 02	0.163215E 03
[illegible]	0.656188E 02	0.131233E 03
86	0.631860E 02	0.126372E 03
87	0.674561E 02	0.134912E 03
88	0.691661E 02	0.138332E 03
89	0.807124E 02	0.161425E 03
90	0.764203E 02	0.152841E 03
91	0.765754E 02	0.153151E 03
92	0.9[illegible]6000E 02	0.199200E 03
93	0.694505E 02	0.138901E 03
94	0.658127E 02	0.131626E 03
95	0.670770E 02	0.134154E 03
96	0.618181E 02	0.123636E 03
97	0.831058E 02	0.166212E 03
98	0.634162E 02	0.126832E 03
99	0.804886E 02	0.160977E 03
100	0.620358E 02	0.124072E 03

RANDOM GRID LOCATION ALGORITHM
OPTIMUM ALLOCATION ON GRID # 3
AFTER 100 ITERATIONS

CODE AND LOCATION OF CENTRAL FACILITIES

CODE NUMBER	CARTESIAN COORDINATES X	Y
J = 1	0.470	9.940
J = 2	4.122	2.300
J = 3	7.774	8.030

OPTIMUM ALLOCATION

F A C I L I T I E S NUMBER	CARTESIAN COORDINATES X	Y	CENTRAL LOCATION CODE	DISTANCE TO CENTRAL LOCATION	TRANSPORTATION COSTS
I = 1	7.190	5.490	3	0.260627E 01	0.521254E 01
I = 2	9.070	9.940	3	0.230819E 01	0.461637E 01
I = 3	4.610	6.490	3	0.351887E 01	0.703774E 01
I = 4	4.940	8.250	3	0.284252E 01	0.568505E 01
I = 5	0.470	0.690	2	0.399114E 01	0.798227E 01
I = 6	6.180	3.570	2	0.241832E 01	0.483664E 01
I = 7	1.130	9.810	1	0.672680E 00	0.134536E 01
I = 8	6.000	4.360	2	0.278756E 01	0.557512E 01
I = 9	8.230	8.060	3	0.456989E 00	0.913978E 00
I = 10	9.600	9.280	3	0.221287E 01	0.442574E 01
I = 11	3.460	9.680	1	0.300128E 01	0.600256E 01
I = 12	2.310	0.390	2	0.263276E 01	0.526552E 01
I = 13	2.530	4.570	2	0.277261E 01	0.554521E 01
I = 14	4.440	7.990	3	0.333424E 01	0.666847E 01
I = 15	8.530	1.490	2	0.448180E 01	0.896360E 01
I = 16	2.290	7.030	1	0.343227E 01	0.686454E 01
I = 17	8.830	7.120	3	0.139400E 01	0.278800E 01
I = 18	2.620	9.410	1	0.221436E 01	0.442872E 01
I = 19	3.820	2.420	2	0.324966E 00	0.649932E 00
I = 20	7.550	1.970	2	0.344385E 01	0.688770E 01
			TOTAL	0.508474E 02	0.101695E 03

RANDOM GRID LOCATION ALGORITHM
DISTRIBUTION OF TOTAL DISTANCES AND COSTS

ITERATION NUMBER	SUM OF OPTIMUM DISTANCES	SUM OF TRANSPORT COSTS
1	0.656294E 02	0.131259E 03
2	0.710376E 02	0.142075E 03
3	0.629839E 02	0.125968E 03
4	0.715650E 02	0.143130E 03
5	0.666213E 02	0.133243E 03
6	0.539547E 02	0.107909E 03
7	0.634637E 02	0.126927E 03
8	0.728946E 02	0.145789E 03
9	0.716677E 02	0.143336E 03
10	0.710172E 02	0.142034E 03
11	0.716880E 02	0.143376E 03
12	0.693906E 02	0.138781E 03
13	0.612919E 02	0.122584E 03
14	0.683918E 02	0.136784E [illegible]
15	0.539547E 02	0.[illegible]
16	0.654310E 02	[illegible]
17	0.71729[illegible]	[illegible]
18	0.[illegible]	[illegible]
19	[illegible]	[illegible]
[illegible]	[illegible]	[illegible] 03
[illegible]	[illegible]	[illegible].143162E 03
[illegible]	[illegible]	0.115648E 03
[illegible]	[illegible] 02	0.153768E 03
[illegible]	[illegible].635449E 02	0.127090E 03
[illegible]	0.730849E 02	0.146170E 03
86	0.717667E 02	0.143534E 03
87	0.712331E 02	0.142466E 03
88	0.613199E 02	0.122640E 03
89	0.738285E 02	0.147657E 03
90	0.716677E 02	0.143336E 03
91	0.717667E 02	0.143534E 03
92	0.720453E 02	0.144091E 03
93	0.646060E 02	0.129212E 03
94	0.654171E 02	0.130834E 03
95	0.731503E 02	0.146301E 03
96	0.814409E 02	0.162882E 03
97	0.716134E 02	0.143227E 03
98	0.634033E 02	0.126807E 03
99	0.731541E 02	0.146308E 03
100	0.585288E 02	0.117058E 03

RANDOM GRID LOCATION ALGORITHM
OPTIMUM ALLOCATION ON GRID # 4
AFTER 100 ITERATIONS

CODE AND LOCATION OF CENTRAL FACILITIES

CODE NUMBER	CARTESIAN COORDINATES X	Y
J = 1	3.513	6.757
J = 2	0.959	0.390
J = 3	6.557	3.573

OPTIMUM ALLOCATION

FACILITIES NUMBER	CARTESIAN COORDINATES X	Y	CENTRAL LOCATION CODE	DISTANCE TO CENTRAL LOCATION	TRANSPORTATION COSTS
I = 1	7.190	5.490	3	0.201859E 01	0.403719E 01
I = 2	9.070	9.940	1	0.640392E 01	0.128078E 02
I = 3	4.610	6.490	1	0.112862E 01	0.225724E 01
I = 4	4.940	8.250	1	0.206529E 01	0.413058E 01
I = 5	0.470	0.690	2	0.573847E 00	0.114769E 01
I = 6	6.180	3.570	3	0.376681E 00	0.753362E 00
I = 7	1.130	9.810	1	0.387339E 01	0.774677E 01
I = 8	6.000	4.360	3	0.963703E 00	0.192740E 01
I = 9	8.230	8.060	3	0.478855E 01	0.957709E 01
I = 10	9.600	9.280	3	0.646745E 01	0.129349E 02
I = 11	3.460	9.680	1	0.292382E 01	0.584764E 01
I = 12	2.310	0.390	2	0.135082E 01	0.270163E 01
I = 13	2.530	4.570	1	0.239759E 01	0.479518E 01
I = 14	4.440	7.990	1	0.154267E 01	0.308534E 01
I = 15	8.530	1.490	3	0.286955E 01	0.573910E 01
I = 16	2.290	7.030	1	0.125350E 01	0.250699E 01
I = 17	8.830	7.120	3	0.421271E 01	0.842542E 01
I = 18	2.620	9.410	1	0.279968E 01	0.559936E 01
I = 19	3.820	2.420	3	0.296976E 01	0.593953E 01
I = 20	7.550	1.970	3	0.188610E 01	0.377221E 01
			TOTAL	0.528661E 02	0.105732E 03

RANDOM GRID LOCATION ALGORITHM
DISTRIBUTION OF TOTAL DISTANCES AND COSTS

ITERATION NUMBER	SUM OF OPTIMUM DISTANCES	SUM OF TRANSPORT COSTS
1	0.641018E 02	0.128204E 03
2	0.794937E 02	0.158988E 03
3	0.777695E 02	0.155539E 03
4	0.795612E 02	0.159123E 03
5	0.777540E 02	0.155508E 03
6	0.705469E 02	0.141094E 03
7	0.793529E 02	0.158706E 03
8	0.678613E 02	0.135723E 03
9	0.831784E 02	0.166357E 03
10	0.832447E 02	0.166490E 03
11	0.614358E 02	0.122872E 03
12	0.119858E 03	0.239717E 03
13	0.113220E 03	0.226441E 03
14	0.671465E 02	0.134293E [illegible]
15	0.705469E 02	0.1[illegible]
16	0.657394E 02	[illegible]
17	0.832713[illegible]	[illegible]
18	0.[illegible]	[illegible]
19	[illegible]	[illegible]
[illegible]	[illegible]	[illegible] 03
[illegible]	[illegible]	0.133795E 03
[illegible]	[illegible]	0.166406E 03
[illegible]	[illegible]52E 02	0.141251E 03
[illegible]	0.730675E 02	0.146135E 03
[illegible]	0.832283E 02	0.166457E 03
86	0.755538E 02	0.151108E 03
87	0.749800E 02	0.149960E 03
88	0.118726E 03	0.237452E 03
89	0.698867E 02	0.139774E 03
90	0.778401E 02	0.155680E 03
91	0.119693E 03	0.239387E 03
92	0.657542E 02	0.131508E 03
93	0.754223E 02	0.150845E 03
94	0.805656E 02	0.161131E 03
95	0.834971E 02	0.176994E 03
96	0.752929E 02	0.150586E 03
97	0.657327E 02	0.131466E 03
98	0.714335E 02	0.142867E 03
99	0.717689E 02	0.143538E 03
100	0.777695E 02	0.155539E 03

RANDOM GRID LOCATION ALGORITHM
OPTIMUM ALLOCATION ON GRID # 5
AFTER 100 ITERATIONS

CODE AND LOCATION OF CENTRAL FACILITIES

CODE NUMBER	CARTESIAN COORDINATES X	Y
J = 1	0.719	0.390
J = 2	0.470	9.940
J = 3	8.296	4.483

OPTIMUM ALLOCATION

FACILITIES NUMBER	CARTESIAN COORDINATES X	Y	CENTRAL LOCATION CODE	DISTANCE TO CENTRAL LOCATION	TRANSPORTATION COSTS
I = 1	7.190	5.490	3	0.149564E 01	0.299128E 01
I = 2	9.070	9.940	3	0.551180E 01	0.110236E 02
I = 3	4.610	6.490	3	0.419679E 01	0.839358E 01
I = 4	4.940	8.250	2	0.477880E 01	0.955760E 01
I = 5	0.470	0.690	1	0.389713E 00	0.779427E 00
I = 6	6.180	3.570	3	0.230424E 01	0.460849E 01
I = 7	1.130	9.810	2	0.672680E 00	0.134536E 01
I = 8	6.000	4.360	3	0.229899E 01	0.459799E 01
I = 9	8.230	8.060	3	0.357774E 01	0.715549E 01
I = 10	9.600	9.280	3	0.497129E 01	0.994258E 01
I = 11	3.460	9.680	2	0.300128E 01	0.600256E 01
I = 12	2.310	0.390	1	0.159125E 01	0.318250E 01
I = 13	2.530	4.570	1	0.455554E 01	0.911109E 01
I = 14	4.440	7.990	2	0.442305E 01	0.884610E 01
I = 15	8.530	1.490	3	0.300201E 01	0.600402E 01
I = 16	2.290	7.030	2	0.343227E 01	0.686454E 01
I = 17	8.830	7.120	3	0.269072E 01	0.538145E 01
I = 18	2.620	9.410	2	0.221436E 01	0.442872E 01
I = 19	3.820	2.420	1	0.370657E 01	0.741314E 01
I = 20	7.550	1.970	3	0.262117E 01	0.524234E 01
			TOTAL	0.614358E 02	0.122872E 03

MULTIPLE CENTRAL FACILITIES

RANDOM GRID WITH LINEAR CONSTRAINTS

```
C       LOCATION OF MULTIPLE CENTRAL FACILITIES                      SPRING 1969
C                         THE RANDOM GRID LOCATION ALGORITHM
C                         WITH LINEAR CONSTRAINTS
C       DEFINITION OF MACHINE INPUT/OUTPUT: R READ
C                                           W WRITE
        INTEGER R,W
C       CHANGE THE DIMENSION CARD IF MORE THAN 20 FACILITIES ARE CONSIDERED
        DIMENSION X(  20),Y(  20),XR(  20),XM(  20),D(  20,  20)
        DIMENSION DOLD(  20),DNEW(  20),COLD(  20),CNEW(  20)
        DIMENSION JOSAV(  20),JNSAV(  20),RAD(  20)
C       CHANGE THE DIMENSION CARD IF MORE THAN  5 CENTRAL LOCATIONS ARE DESIRED
        DIMENSION IOSAV(  5),INSAV(  5),XC(  5),YC(  5)
C       CHANGE THE DIMENSION CARD IF MORE THAN 100 GRID INTERSECT. ARE CONSIDERED
        DIMENSION XG( 100),YG( 100)
C       CHANGE DIMENSION CARD IF MORE THAN   10 CONSTRAINTS ARE CONSIDERED
        DIMENSION A( 10),B( 10)
        R=5
        W=6
C       TERMINOLOGY OF VARIABLES
C             N          :NUMBER OF FACILITIES
C             M          :NUMBER OF CENTRAL FACILITIES
C             ITERA      :NUMBER OF ITERATIONS IN RANDOM SEARCH OF FACILITIES
C             IGRID      :INITIAL NUMBER OF GRID DIVISIONS ON EACH X AND Y AXIS
C             ITGRD      :NUMBER OF GRID SIZE CHANGES
C             INC        :INCREMENTAL NUMBER OF DIVISIONS ON EACH X AND Y AXIS WHEN
C                         PASSING FROM ONE GRID SIZE TO THE NEXT
C             X(I),Y(I):CARTESIAN COORDINATES OF FACILITIES
C             XR(I)      :TRANSPORT RATE ON ROUTE I
C             XM(I)      :QUANTITY TO TRANSPORT ON ROUTE I
C             NC         :TOTAL NUMBER OF LINEAR CONSTRAINTS ON CONVEX HULL
C             MC         :NUMBER OF LINEAR CONSTRAINTS WITH LESS THAN OR EQUAL TO
C             K          :CODE NUMBER OF LINEAR CONSTRAINT
C             I1,I2      :CODE NUMBER OF THE 2 FACILITIES DEFINING THE LINE CONSTRAIN
C             A(KC)      :ANGULAR COEFFICIENT OF LINEAR CONSTRAINT
C             B(KC)      :ORDINATE AT ORIGIN OF LINEAR CONSTRAINT
C             DOLD(I),DNEW(I):OLD AND NEW EUCLIDEAN DISTANCES FROM FACILITY I TO
C                         OPTIMUM CENTRAL LOCATION
C             SDOLD,SDNEW:OLD AND NEW SUM OF DISTANCES TO THE CENTRAL FACILITIES
C             COLD(I),CNEW(I):OLD AND NEW TRANSPORTATION COSTS FROM FACILITY I TO
C                         OPTIMAL CENTRAL LOCATION
C             SCOLD,SCNEW:OLD AND NEW SUM OF TRANSPORT COST TO THE CENTRAL FACILITY
C             JOSAV(I),JNSAV(I):OLD AND NEW CODE NUMBER ALLOCATION OF THE FACILITY
C             IOSAV(J),INSAV(I):OLD AND NEW CODE NUMBER OF THE RANDOMLY SELECTED
C                         CENTRAL LOCATION
C             RAINC      :CLASS WIDTH ON CUMULATIVE DISTRIBUTION OF LOCATIONS
C             RAD(I)     :CLASS BOUNDARIES ON CUMULATIVE DISTRIBUTION OF LOCATIONS
C             KITER      :ITERATION COUNTER
C             YFL        :RANDOM NUMBER BETWEEN 0 AND 1.000
        READ(R,10)N,M,ITERA,IGRID,INC,ITGRD
     10 FORMAT(6I10)
        READ(R,20)(X(I),Y(I),XR(I),XM(I),I=1,N)
     20 FORMAT(4F15.0)
        READ(R,21)NC,MC
```

```
   21 FORMAT(2I10)
      DO 23 KC=1,MC
      READ(R,22)KC,I1,I2
   22 FORMAT(3I10)
C     ANGULAR COEFFICIENT OF LINEAR CONSTRAINT
      A(KC)=(Y(I1)-Y(I2))/(X(I1)-X(I2))
C     ORDINATE AT ORIGIN OF LINE CONSTRAINT
   23 B(KC)=Y(I1)-A(KC)*X(I1)
      MC1=MC+1
C     FIRST TABULATION: LIST GIVEN VARIABLES
      WRITE(W,30)ITGRD,ITERA,IGRID,INC,M,N
   30 FORMAT(1H1,//,35X,39HLOCATION OF MULTIPLE CENTRAL FACILITIES,/,
     140X,30HRANDOM GRID LOCATION ALGORITHM,/,43X,23HWITH LINEAR CONSTRA
     2INTS,//,29X,I3,1X,20HGRID SPACING CHANGES,
     3           4X,I4,1X,19HITERATIONS PER GRID,/,15X,33HINITIAL NUMBER O
     4F GRID DIVISIONS ,I4,4X,35HDIVISION INCREMENTS PER GRID CHANGE ,I4
     5,//,43X,I3,1X,18HCENTRAL FACILITIES,/,46X,I5,1X,10HFACILITIES,//)
      WRITE(W,32)
   32 FORMAT(24X,21HCARTESIAN COORDINATES,6X,14HTRANSPORT RATE,3X,
     111HQUANTITY TO,/,69X,9HTRANSPORT,/,27X,1HX,14X,1HY,15X,1HR,14X,1HM
     2,//)
      DO 36 I=1,N
      WRITE (W,35) I,X(I),Y(I),XR(I),XM(I)
   35 FORMAT (10X,3HI =,I5,4(4X,E11.3))
   36 CONTINUE
      WRITE(W,40)
   40 FORMAT(///,46X,18HLINEAR CONSTRAINTS,//)
      DO 43 KC=1,MC
      WRITE(W,42)KC,A(KC),KC,B(KC)
   42 FORMAT  (22X,2HY(,I3,3H) -,E14.6,5H * X(,I3,23H) LESS THAN OR EQUA
     1L TO,E17.6)
   43 CONTINUE
      DO 45 KC=MC1,NC
      WRITE(W,44)KC,A(KC),KC,B(KC)
   44 FORMAT  (22X,2HY(,I3,3H) -,E14.6,5H * X(,I3,26H) GREATER THAN OR E
     1QUAL TO,E14.6)
   45 CONTINUE
C     GRID COUNTER SET FOR FIRST GRID INVESTIGATION
      L=1
C     DEFINE X MAXIMUM, Y MAXIMUM
      XMAX=X(1)
      DO 80 I=2,N
      IF(X(I)-XMAX)80,80,70
   70 XMAX=X(I)
   80 CONTINUE
      YMAX=Y(1)
      DO 100 I=2,N
      IF(Y(I)-YMAX)100,90,90
   90 YMAX=Y(I)
  100 CONTINUE
C     DEFINE X MINIMUM, Y MINIMUM
      XMIN=X(1)
      DO 120 I=2,N
      IF(X(I)-XMIN)110,120,120
```

```
  110 XMIN=X(I)
  120 CONTINUE
      YMIN=Y(1)
      DO 140 I=2,N
      IF(Y(I)-YMIN)130,140,140
  130 YMIN=Y(I)
  140 CONTINUE
C     DEFINE RANGE OF X AND Y
      RANGX=XMAX-XMIN
      RANGY=YMAX-YMIN
C     COMPUTE NUMBER OF GRID INTERSECTION POINTS
  150 NBER=IGRID+1
      NG=(NBER)**2
C     CHECK IF NUMBER OF GRID INTERSECTION POINTS IS LARGE ENOUGH
      IF(NG-M)160,160,170
  160 IGRID=IGRID+INC
      GO TO 150
C     DEFINE GRID SPACING
  170 GRID=IGRID
      XINC=RANGX/GRID
      YINC=RANGY/GRID
C     DEFINE GRID INTERSECTION POINTS
      DO 130 I=1,NBER
      IX=I-1
      DO 180 J=1,NBER
      XK=J-1
      K=J+IX*NBER
      XG(K)=XMIN+(XK*XINC)
  180 YG(K)=YMIN+(IX*YINC)
C     ZERO ALL COMPUTATIONAL VARIABLES
      DO 50 I=1,N
      DOLD(I)=0.0
      DNEW(I)=0.0
      COLD(I)=0.0
      CNEW(I)=0.0
      JOSAV(I)=0
   50 JNSAV(I)=0
      DO 60 J=1,M
      IOSAV(J)=0
   60 INSAV(J)=0
      SDOLD=0.0
      SDNEW=0.0
      SCOLD=0.0
      SCNEW=0.0
C     ZERO ITERATION COUNTER
      KITER=0
C     DEFINITION OF CLASS INTERVALS FOR 0 TO 1 RANDUM NUMBER DEFINING THE
C     0 TO NG INTEGERS
      XNG=NG
      RAINC=1.0/XNG
      DO 190 I=1,NG
      XI=I
  190 RAD(I)=XI*RAINC
C     RANDOM GENERATOR STARTING VALUE
```

```
      IY=21735
C     STARTING OF ITERATION COUNTER FOR A GIVEN GRID
  200 KITER=KITER+1
      IF(KITER-ITERA)210,210,390
C     RANDUM CHOICE OF M CENTRAL FACILITIES FROM THE NG GRID POINTS
  210 DO 240 J=1,M
  215 CALL RANDU(IY,IY,YFL)
      DO 230 I=1,NG
      IF (YFL-RAD(I))220,220,230
C     CHECK THAT RANDOMLY CHOSEN GRID POINTS SATISFY THE LINEAR CONSTRAINTS
C     CONSTRAINTS LESS THAN OR EQUAL TO
  220 DO 1220 KC=1,MC
      ERROR=(YG(I)-A(KC)*XG(I))-B(KC)
      IF(ERROR)1220,1220,215
 1220 CONTINUE
C     CONSTRAINTS GREATER THAN OR EQUAL TO
      DO 1222 KC=MC1,NC
      ERROR=(YG(I)-A(KC)*XG(I))-B(KC)
      IF(ERROR)215,1222,1222
 1222 CONTINUE
      XC(J)=XG(I)
      YC(J)=YG(I)
      INSAV(J)=I
      IF(J-1)222,240,222
  222 K=J-1
C     CHECK THAT RANDOMLY CHOSEN GRID POINT HAS NOT BEEN ALREADY PICKED AS
C     CENTRAL LOCATION
      DO 224 KJ=1,K
      IF(INSAV(J)-INSAV(KJ))240,215,240
  224 CONTINUE
  230 CONTINUE
  240 CONTINUE
C     COMPUTE ARRAY OF EUCLIDEAN DISTANCES TO RANDOMLY CHOSEN CENTRAL FACILITIE
      DO 250 I=1,N
      DO 250 J=1,M
  250 D(I,J)=(((XC(J)-X(I))**2)+((YC(J)-Y(I))**2))**0.5
C     FOR EACH FACILITY SELECT THE CLOSER CENTRAL LOCATION
      DO 280 I=1,N
      SHORT=D(I,1)
      J=1
      DO 270 K=2,M
      IF(D(I,K)-SHORT)260,270,270
  260 SHORT=D(I,K)
      J=K
  270 CONTINUE
      DNEW(I)=SHORT
      CNEW(I)=XR(I)*XM(I)*DNEW(I)
      JNSAV(I)=J
  280 CONTINUE
C     COMPUTE SUM OF OPTIMAL DISTANCES AND TOTAL TRANSPORTATION COST
      SDNEW=0.0
      SCNEW=0.0
      DO 290 I=1,N
      SDNEW=SDNEW+DNEW(I)
```

```
  290 SCNEW=SCNEW+CNEW(I)
      IF(KITER-1)300,300,320
C     TITLE OF SECOND TABULATION
  300 WRITE(W,310)
  310 FORMAT(1H1,//,40X,30HRANDOM GRID LOCATION ALGORITHM,/,43X,
     123HWITH LINEAR CONSTRAINTS,//,34X,
     241HDISTRIBUTION OF TOTAL DISTANCES AND COSTS,//,31X,9HITERATION,
     35X,14HSUM OF OPTIMUM,5X,16HSUM OF TRANSPORT,/,32X,6HNUMBER,9X,
     49HDISTANCES,12X,5HCOSTS,//)
C     BODY OF SECOND TABULATION
  320 WRITE (W,330)KITER,SDNEW,SCNEW
  330 FORMAT(30X,I7,8X,E13.6,7X,E13.6)
C     IF THIS IS THE FIRST ITERATION REPEAT RANDOM CHOICE ANOTHER TIME
      IF(SCOLD)340,350,340
C     CHECK IF NEW CHOICE OF CENTRAL FACILITIES GIVES BETTER RESULTS
  340 IF(SCNEW-SCOLD)350,350,380
  350 SDOLD=SDNEW
      SCOLD=SCNEW
      DO 360 I=1,N
      DOLD(I)=DNEW(I)
      COLD(I)=CNEW(I)
  360 JOSAV(I)=JNSAV(I)
      DO 370 J=1,M
  370 IOSAV(J)=INSAV(J)
  380 GO TO 200
C     THIRD TABULATION : OPTIMUM ALLOCATION
  390 WRITE(W,400)L,ITERA
  400 FORMAT(1H1,//,40X,30HRANDOM GRID LOCATION ALGORITHM,/,43X,
     123HWITH LINEAR CONSTRAINTS,//,40X,
     228HOPTIMUM ALLOCATION ON GRID #,I3,/,45X,5HAFTER,1X,I4,1X,
     310HITERATIONS,//,35X,39HCODE AND LOCATION OF CENTRAL FACILITIES,/,
     433X,11HCODE NUMBER,10X,21HCARTESIAN COORDINATES,/,57X,1HX,14X,1HY,
     5//)
      DO 415 J=1,M
      JJ=IOSAV(J)
      WRITE(W,410)J,XG(JJ),YG(JJ)
  410 FORMAT(34X,3HJ =,I4,9X,F11.3,4X,F11.3)
  415 CONTINUE
      WRITE(W,420)
  420 FORMAT(//,46X,18HOPTIMUM ALLOCATION,//,18X,19HF A C I L I T I E S,
     110X,16HCENTRAL LOCATION,6X,11HDISTANCE TO,7X,14HTRANSPORTATION,/,
     212X,6HNUMBER,4X,21HCARTESIAN COORDINATES,9X,4HCODE,11X,
     316HCENTRAL LOCATION,9X,5HCOSTS,/,25X,1HX,13X,1HY,//)
      WRITE(W,430)(I,X(I),Y(I),JOSAV(I),DOLD(I),COLD(I),I=1,N)
  430 FORMAT(11X,3HI =,I4,2X,F11.3,3X,F11.3,7X,I3,10X,E16.6,4X,E15.6)
      WRITE(W,435)SDOLD,SCOLD
  435 FORMAT(/,62X,5HTOTAL,E14.6,4X,E15.6)
C     CHECK IF ALL GRID CHANGES HAVE BEEN DONE
      IF(L-ITGRD)440,450,450
  440 L=L+1
      GO TO 160
  450 CALL EXIT
      END
```

LOCATION OF MULTIPLE CENTRAL FACILITIES
RANDOM GRID LOCATION ALGORITHM
WITH LINEAR CONSTRAINTS

5 GRID SPACING CHANGES   100 ITERATIONS PER GRID
INITIAL NUMBER OF GRID DIVISIONS   3   DIVISION INCREMENTS PER GRID CHANGE   1

3 CENTRAL FACILITIES
20 FACILITIES

	CARTESIAN COORDINATES		TRANSPORT RATE	QUANTITY TO TRANSPORT
	X	Y	R	M
I = 1	7.190	5.490	1.000	2.000
I = 2	9.070	9.940	1.000	2.000
I = 3	4.610	6.490	1.000	2.000
I = 4	4.940	8.250	1.000	2.000
I = 5	0.470	0.690	1.000	2.000
I = 6	6.180	3.570	1.000	2.000
I = 7	1.130	9.810	1.000	2.000
I = 8	6.000	4.360	1.000	2.000
I = 9	8.230	8.060	1.000	2.000
I = 10	9.600	9.280	1.000	2.000
I = 11	3.460	9.680	1.000	2.000
I = 12	2.310	0.390	1.000	2.000
I = 13	2.530	4.570	1.000	2.000
I = 14	4.440	7.390	1.000	2.000
I = 15	8.530	1.490	1.000	2.000
I = 16	2.290	7.030	1.000	2.000
I = 17	8.830	7.120	1.000	2.000
I = 18	2.620	9.410	1.000	2.000
I = 19	3.820	2.420	1.000	2.000
I = 20	7.550	1.970	1.000	2.000

LINEAR CONSTRAINTS

```
Y( 1) -  0.163728E-01 * X( 1) LESS THAN OR EQUAL TO      0.979150E 01
Y( 2) - -0.124528E 01 * X( 2) LESS THAN OR EQUAL TO      0.212347E 02
Y( 3) -  0.138182E 02 * X( 3) LESS THAN OR EQUAL TO     -0.580455E 01
Y( 4) -  0.728038E 01 * X( 4) GREATER THAN OR EQUAL TO -0.606116E 02
Y( 5) -  0.176849E 00 * X( 5) GREATER THAN OR EQUAL TO -0.185194E-01
Y( 6) - -0.163043E 00 * X( 6) GREATER THAN OR EQUAL TO  0.766630E 00
```

RANDOM GRID LOCATION ALGORITHM
WITH LINEAR CONSTRAINTS

DISTRIBUTION OF TOTAL DISTANCES AND COSTS

ITERATION NUMBER	SUM OF OPTIMUM DISTANCES	SUM OF TRANSPORT COSTS
1	0.644550E 02	0.128910E 03
2	0.629839E 02	0.125968E 03
3	0.652284E 02	0.130457E 03
4	0.652284E 02	0.130457E 03
5	0.529413E 02	0.105883E 03
6	0.529413E 02	0.105883E 03
7	0.652284E 02	0.130457E 03
8	0.529413E 02	0.105883E 03
9	0.529413E 02	0.105883E 03
10	0.652284E 02	0.130457E 03
11	0.644550E 02	0.128910E 03
12	0.652284E 02	0.130457E 03
13	0.529413E 02	0.10588[illegible]
14	0.629839E 02	
15	0.529413E 02	
16	0.55[illegible]	
17		
[illegible]		
		[illegible]3E 03
		0.111526E 03
		0.125968E 03
	[illegible]9413E 02	0.105883E 03
	0.529413E 02	0.105883E 03
[illegible]	0.529413E 02	0.105883E 03
85	0.557631E 02	0.111526E 03
86	0.644550E 02	0.128910E 03
87	0.656294E 02	0.131259E 03
88	0.529413E 02	0.105883E 03
89	0.556484E 02	0.111297E 03
90	0.529413E 02	0.105883E 03
91	0.656294E 02	0.131259E 03
92	0.656294E 02	0.131259E 03
93	0.557631E 02	0.111526E 03
94	0.557631E 02	0.111526E 03
95	0.545271E 02	0.109054E 03
96	0.557631E 02	0.111526E 03
97	0.656294E 02	0.131259E 03
98	0.529413E 02	0.105883E 03
99	0.545271E 02	0.109054E 03
100	0.545271E 02	0.109054E 03

RANDOM GRID LOCATION ALGORITHM
WITH LINEAR CONSTRAINTS

OPTIMUM ALLOCATION ON GRID # 1
AFTER 100 ITERATIONS

CODE AND LOCATION OF CENTRAL FACILITIES

CODE NUMBER	CARTESIAN COORDINATES X	Y
J = 1	3.513	6.757
J = 2	6.557	3.573
J = 3	6.557	6.757

OPTIMUM ALLOCATION

F A C I L I T I E S NUMBER	CARTESIAN COORDINATES X	Y	CENTRAL LOCATION CODE	DISTANCE TO CENTRAL LOCATION	TRANSPORTATION COSTS
I = 1	7.190	5.490	3	0.141617E 01	0.283235E 01
I = 2	9.070	9.940	3	0.405592E 01	0.811184E 01
I = 3	4.610	6.490	1	0.112862E 01	0.225724E 01
I = 4	4.940	8.250	1	0.206529E 01	0.413058E 01
I = 5	0.470	0.690	2	0.673506E 01	0.134701E 02
I = 6	6.180	3.570	2	0.376681E 00	0.753362E 00
I = 7	1.130	9.810	1	0.387339E 01	0.774677E 01
I = 8	6.000	4.360	2	0.963703E 00	0.192740E 01
I = 9	8.230	8.060	3	0.212102E 01	0.424204E 01
I = 10	9.600	9.280	3	0.395336E 01	0.790673E 01
I = 11	3.460	9.680	1	0.292382E 01	0.584764E 01
I = 12	2.310	0.390	2	0.530733E 01	0.106147E 02
I = 13	2.530	4.570	1	0.239759E 01	0.479518E 01
I = 14	4.440	7.990	1	0.154267E 01	0.308534E 01
I = 15	8.530	1.490	2	0.286955E 01	0.573910E 01
I = 16	2.290	7.030	1	0.125350E 01	0.250699E 01
I = 17	8.830	7.120	3	0.230219E 01	0.460437E 01
I = 18	2.620	9.410	1	0.279968E 01	0.559936E 01
I = 19	3.820	2.420	2	0.296976E 01	0.593953E 01
I = 20	7.550	1.970	2	0.188610E 01	0.377221E 01
			TOTAL	0.529413E 02	0.105883E 03

RANDOM GRID LOCATION ALGORITHM
WITH LINEAR CONSTRAINTS

DISTRIBUTION OF TOTAL DISTANCES AND COSTS

ITERATION NUMBER	SUM OF OPTIMUM DISTANCES	SUM OF TRANSPORT COSTS
1	0.625649E 02	0.125130E 03
2	0.522644E 02	0.104529E 03
3	0.635649E 02	0.127130E 03
4	0.551998E 02	0.110400E 03
5	0.558664E 02	0.111733E 03
6	0.565647E 02	0.113129E 03
7	0.635649E 02	0.127130E 03
8	0.486947E 02	0.973895E 02
9	0.585199E 02	0.117040E 03
10	0.522878E 02	0.104576E 03
11	0.681104E 02	0.136221E 03
12	0.605081E 02	0.121016E 03
13	0.603747E 02	0.1207[illegible]
14	0.565647E 02	[illegible]
15	0.574391E 02	[illegible]
16	0.52[illegible]	[illegible]
17	[illegible]	[illegible]
18	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]74E 03
[illegible]	[illegible]	0.968786E 02
[illegible]	[illegible]	0.109433E 03
[illegible]	[illegible]295E 02	0.117659E 03
[illegible]	0.647847E 02	0.129570E 03
[illegible]	0.623605E 02	0.124721E 03
85	0.648323E 02	0.129665E 03
86	0.569812E 02	0.113962E 03
87	0.564376E 02	0.112875E 03
88	0.619273E 02	0.123855E 03
89	0.569812E 02	0.113962E 03
90	0.585917E 02	0.117184E 03
91	0.564917E 02	0.112983E 03
92	0.576692E 02	0.115338E 03
93	0.657648E 02	0.131530E 03
94	0.494325E 02	0.988651E 02
95	0.504782E 02	0.100957E 03
96	0.541615E 02	0.108323E 03
97	0.539387E 02	0.107877E 03
98	0.537629E 02	0.107526E 03
99	0.608083E 02	0.121617E 03
100	0.592391E 02	0.118478E 03

RANDOM GRID LOCATION ALGORITHM
WITH LINEAR CONSTRAINTS

OPTIMUM ALLOCATION ON GRID # 2
AFTER 100 ITERATIONS

CODE AND LOCATION OF CENTRAL FACILITIES

CODE NUMBER	CARTESIAN COORDINATES X	Y
J = 1	5.035	2.777
J = 2	7.317	7.552
J = 3	2.752	7.552

OPTIMUM ALLOCATION

FACILITIES NUMBER	CARTESIAN COORDINATES X	Y	CENTRAL LOCATION CODE	DISTANCE TO CENTRAL LOCATION	TRANSPORTATION COSTS
I = 1	7.190	5.490	2	0.206644E 01	0.413287E 01
I = 2	9.070	9.940	2	0.296166E 01	0.592332E 01
I = 3	4.610	6.490	3	0.213991E 01	0.427982E 01
I = 4	4.940	8.250	3	0.229601E 01	0.459202E 01
I = 5	0.470	0.690	1	0.501964E 01	0.100393E 02
I = 6	6.180	3.570	1	0.139251E 01	0.278502E 01
I = 7	1.130	9.810	3	0.278007E 01	0.556015E 01
I = 8	6.000	4.360	1	0.185352E 01	0.370704E 01
I = 9	8.230	8.060	2	0.104413E 01	0.208827E 01
I = 10	9.600	9.280	2	0.286253E 01	0.572506E 01
I = 11	3.460	9.680	3	0.224206E 01	0.448411E 01
I = 12	2.310	0.390	1	0.362295E 01	0.724590E 01
I = 13	2.530	4.570	3	0.299079E 01	0.598157E 01
I = 14	4.440	7.990	3	0.174329E 01	0.348658E 01
I = 15	8.530	1.490	1	0.372460E 01	0.744921E 01
I = 16	2.290	7.030	3	0.697789E 00	0.139558E 01
I = 17	8.830	7.120	2	0.157312E 01	0.314625E 01
I = 18	2.620	9.410	3	0.186222E 01	0.372444E 01
I = 19	3.820	2.420	1	0.126650E 01	0.253300E 01
I = 20	7.550	1.970	1	0.264145E 01	0.528291E 01
			TOTAL	0.467811E 02	0.935623E 02

RANDOM GRID LOCATION ALGORITHM
WITH LINEAR CONSTRAINTS

DISTRIBUTION OF TOTAL DISTANCES AND COSTS

ITERATION NUMBER	SUM OF OPTIMUM DISTANCES	SUM OF TRANSPORT COSTS
1	0.556597E 02	0.111319E 03
2	0.536016E 02	0.107203E 03
3	0.594032E 02	0.118806E 03
4	0.758892E 02	0.151778E 03
5	0.632144E 02	0.126429E 03
6	0.576732E 02	0.115347E 03
7	0.532543E 02	0.106509E 03
8	0.886603E 02	0.177321E 03
9	0.587778E 02	0.117556E 03
10	0.500512E 02	0.100103E 03
11	0.831058E 02	0.166212E 03
12	0.612416E 02	0.122483E 03
13	0.632144E 02	0.1264[illegible]
14	0.633477E 02	
15	0.539263E 02	
16	0.58[illegible]	
17		
18		
		[illegible]E 03
		0.132327E 03
		0.144980E 03
	[illegible]585E 02	0.134477E 03
	0.725148E 02	0.145030E 03
	0.572565E 02	0.114513E 03
85	0.507136E 02	0.101427E 03
86	0.630383E 02	0.126077E 03
87	0.673329E 02	0.134666E 03
88	0.547111E 02	0.109422E 03
89	0.765031E 02	0.153006E 03
90	0.560851E 02	0.112170E 03
91	0.619687E 02	0.123937E 03
92	0.560851E 02	0.112170E 03
93	0.724897E 02	0.144980E 03
94	0.594032E 02	0.118806E 03
95	0.553295E 02	0.110659E 03
96	0.579937E 02	0.115988E 03
97	0.594032E 02	0.118806E 03
98	0.846596E 02	0.169319E 03
99	0.547111E 02	0.109422E 03
100	0.672767E 02	0.134553E 03

RANDOM GRID LOCATION ALGORITHM
WITH LINEAR CONSTRAINTS

OPTIMUM ALLOCATION ON GRID # 3
AFTER 100 ITERATIONS

CODE AND LOCATION OF CENTRAL FACILITIES

CODE NUMBER	CARTESIAN COORDINATES X	Y
J = 1	7.774	8.030
J = 2	2.296	6.120
J = 3	5.948	2.300

OPTIMUM ALLOCATION

F A C I L I T I E S NUMBER	CARTESIAN COORDINATES X	Y	CENTRAL LOCATION CODE	DISTANCE TO CENTRAL LOCATION	TRANSPORTATION COSTS
I = 1	7.190	5.490	1	0.260627E 01	0.521254E 01
I = 2	9.070	9.940	1	0.230819E 01	0.461637E 01
I = 3	4.610	6.490	2	0.234340E 01	0.468679E 01
I = 4	4.940	8.250	1	0.284252E 01	0.568505E 01
I = 5	0.470	0.690	3	0.570968E 01	0.114194E 02
I = 6	6.180	3.570	3	0.129102E 01	0.258203E 01
I = 7	1.130	9.810	2	0.336984E 01	0.773967E 01
I = 8	6.000	4.360	3	0.206066E 01	0.412131E 01
I = 9	8.230	8.060	1	0.456989E 00	0.913978E 00
I = 10	9.600	9.280	1	0.221287E 01	0.442574E 01
I = 11	3.460	9.680	2	0.374546E 01	0.749092E 01
I = 12	2.310	0.390	3	0.410890E 01	0.821781E 01
I = 13	2.530	4.570	2	0.156756E 01	0.313513E 01
I = 14	4.440	7.990	2	0.284493E 01	0.568986E 01
I = 15	8.530	1.490	3	0.270607E 01	0.541215E 01
I = 16	2.290	7.030	2	0.910021E 00	0.182004E 01
I = 17	8.830	7.120	1	0.139400E 01	0.278800E 01
I = 18	2.620	9.410	2	0.330591E 01	0.661183E 01
I = 19	3.820	2.420	3	0.213138E 01	0.426276E 01
I = 20	7.550	1.970	3	0.163564E 01	0.327127E 01
			TOTAL	0.500512E 02	0.100103E 03

RANDOM GRID LOCATION ALGORITHM
WITH LINEAR CONSTRAINTS

DISTRIBUTION OF TOTAL DISTANCES AND COSTS

ITERATION NUMBER	SUM OF OPTIMUM DISTANCES	SUM OF TRANSPORT COSTS
1	0.656294E 02	0.131259E 03
2	0.543087E 02	0.108618E 03
3	0.562535E 02	0.112507E 03
4	0.676936E 02	0.135387E 03
5	0.616904E 02	0.123381E 03
6	0.620587E 02	0.124117E 03
7	0.633277E 02	0.126655E 03
8	0.738285E 02	0.147657E 03
9	0.738285E 02	0.147657E 03
10	0.597720E 02	0.119544E 03
11	0.708013E 02	0.141603E 03
12	0.633277E 02	0.126655E 03
13	0.669555E 02	0.1339[illegible]
14	0.628177E 02	[illegible]
15	0.588689E 02	
16	0.6970[illegible]	
17		
18		
		[illegible]43E 03
		0.129558E 03
		0.162382E 03
	[illegible]2756E 02	0.118551E 03
	0.664855E 02	0.132971E 03
[illegible]	0.636705E 02	0.127341E 03
85	0.665534E 02	0.133107E 03
86	0.597720E 02	0.119544E 03
87	0.696991E 02	0.139398E 03
88	0.603119E 02	0.120624E 03
89	0.605868E 02	0.121174E 03
90	0.596583E 02	0.119317E 03
91	0.809727E 02	0.161946E 03
92	0.571633E 02	0.114327E 03
93	0.589596E 02	0.117919E 03
94	0.542832E 02	0.108566E 03
95	0.598216E 02	0.119643E 03
96	0.696991E 02	0.139398E 03
97	0.734318E 02	0.146864E 03
98	0.603119E 02	0.120624E 03
99	0.604502E 02	0.120901E 03
100	0.577034E 02	0.115407E 03

RANDOM GRID LOCATION ALGORITHM
WITH LINEAR CONSTRAINTS

OPTIMUM ALLOCATION ON GRID # 4
AFTER 100 ITERATIONS

CODE AND LOCATION OF CENTRAL FACILITIES

CODE NUMBER	CARTESIAN COORDINATES X	Y
J = 1	3.513	6.757
J = 2	1.000	1.982
J = 3	6.557	3.573

OPTIMUM ALLOCATION

F A C I L I T I E S NUMBER	CARTESIAN COORDINATES X	Y	CENTRAL LOCATION CODE	DISTANCE TO CENTRAL LOCATION	TRANSPORTATION COSTS
I = 1	7.190	5.490	3	0.201859E 01	0.403719E 01
I = 2	9.070	9.940	1	0.640392E 01	0.128078E 02
I = 3	4.610	6.490	1	0.112862E 01	0.225724E 01
I = 4	4.940	8.250	1	0.206529E 01	0.413058E 01
I = 5	0.470	0.690	2	0.139617E 01	0.279235E 01
I = 6	6.180	3.570	3	0.376681E 00	0.753362E 00
I = 7	1.130	9.810	1	0.387339E 01	0.774677E 01
I = 8	6.000	4.360	3	0.963703E 00	0.192740E 01
I = 9	8.230	8.060	3	0.478855E 01	0.957709E 01
I = 10	9.600	9.280	3	0.646745E 01	0.129349E 02
I = 11	3.460	9.680	1	0.292382E 01	0.584764E 01
I = 12	2.310	0.390	2	0.206143E 01	0.412286E 01
I = 13	2.530	4.570	1	0.239759E 01	0.479518E 01
I = 14	4.440	7.990	1	0.154267E 01	0.308534E 01
I = 15	8.530	1.490	3	0.286955E 01	0.573910E 01
I = 16	2.290	7.030	1	0.125350E 01	0.250699E 01
I = 17	8.830	7.120	3	0.421271E 01	0.842542E 01
I = 18	2.620	9.410	1	0.279968E 01	0.559936E 01
I = 19	3.820	2.420	2	0.285386E 01	0.570772E 01
I = 20	7.550	1.970	3	0.188610E 01	0.377221E 01
			TOTAL	0.542832E 02	0.108566E 03

RANDOM GRID LOCATION ALGORITHM
WITH LINEAR CONSTRAINTS

DISTRIBUTION OF TOTAL DISTANCES AND COSTS

ITERATION NUMBER	SUM OF OPTIMUM DISTANCES	SUM OF TRANSPORT COSTS
1	0.641018E 02	0.128204E 03
2	0.754149E 02	0.150830E 03
3	0.754223E 02	0.150845E 03
4	0.779967E 02	0.155994E 03
5	0.668643E 02	0.133729E 03
6	0.779967E 02	0.155994E 03
7	0.678829E 02	0.135766E 03
8	0.668584E 02	0.133717E 03
9	0.754960E 02	0.150992E 03
10	0.755897E 02	0.151180E 03
11	0.804386E 02	0.160377E 03
12	0.806761E 02	0.161352E 03
13	0.678399E 02	0.135[illegible]
14	0.717151E 02	
15	0.779476E 0[illegible]	
16	0.757[illegible]	
17		
1[illegible]		
		[illegible]15E 03
		0.161122E 03
		0.133754E 03
	[illegible]511E 02	0.151302E 03
	0.668643E 02	0.133729E 03
	0.806317E 02	0.161264E 03
85	0.668768E 02	0.133754E 03
86	0.677577E 02	0.135515E 03
87	0.805705E 02	0.161141E 03
88	0.756726E 02	0.151345E 03
89	0.668904E 02	0.133781E 03
90	0.755477E 02	0.151096E 03
91	0.668835E 02	0.133767E 03
92	0.668584E 02	0.133717E 03
93	0.753817E 02	0.150763E 03
94	0.754223E 02	0.150845E 03
95	0.805656E 02	0.161131E 03
96	0.717344E 02	0.143469E 03
97	0.754223E 02	0.150845E 03
98	0.779967E 02	0.155994E 03
99	0.717950E 02	0.143590E 03
100	0.755669E 02	0.151134E 03

RANDOM GRID LOCATION ALGORITHM
WITH LINEAR CONSTRAINTS

UPTIMUM ALLOCATION ON GRID # 5
AFTER 100 ITERATIONS

CODE AND LOCATION OF CENTRAL FACILITIES

CODE NUMBER	CARTESIAN COORDINATES X	Y
J = 1	3.079	7.211
J = 2	8.296	4.483
J = 3	8.296	4.483

OPTIMUM ALLOCATION

F A C I L I T I E S NUMBER	CARTESIAN COORDINATES X	Y	CENTRAL LOCATION CODE	DISTANCE TO CENTRAL LOCATION	TRANSPORTATION COSTS
I = 1	7.190	5.490	2	0.149564E 01	0.299128E 01
I = 2	9.070	9.940	2	0.551180E 01	0.110236E 02
I = 3	4.610	6.490	1	0.169285E 01	0.338569E 01
I = 4	4.940	8.250	1	0.213156E 01	0.426312E 01
I = 5	0.470	0.690	1	0.702379E 01	0.140476E 02
I = 6	6.180	3.570	2	0.230424E 01	0.460849E 01
I = 7	1.130	9.810	1	0.324800E 01	0.649600E 01
I = 8	6.000	4.360	2	0.229899E 01	0.459799E 01
I = 9	8.230	8.060	2	0.357774E 01	0.715549E 01
I = 10	9.600	9.280	2	0.497129E 01	0.994258E 01
I = 11	3.460	9.680	1	0.249787E 01	0.499573E 01
I = 12	2.310	0.390	1	0.686458E 01	0.137292E 02
I = 13	2.530	4.570	1	0.269779E 01	0.539558E 01
I = 14	4.440	7.990	1	0.156833E 01	0.313667E 01
I = 15	8.530	1.490	2	0.300201E 01	0.600402E 01
I = 16	2.290	7.030	1	0.809171E 00	0.161834E 01
I = 17	8.830	7.120	2	0.269072E 01	0.538145E 01
I = 18	2.620	9.410	1	0.224589E 01	0.449177E 01
I = 19	3.820	2.420	1	0.484845E 01	0.969690E 01
I = 20	7.550	1.970	2	0.262117E 01	0.524234E 01
			TOTAL	0.641018E 02	0.128204E 03

VARIABLE DISCRIMINATION ALGORITHM

AGGLOMERATION OF A LARGE SYSTEM

```
// JOB T
// FOR
**   R.CHAPELLE RS-02399
*IOCS(CARD, 1132 PRINTER, DISK)
*ONE WORD INTEGERS
*LIST SOURCE PROGRAM
C     VARIABLE DISCRIMINATION ALGORITHM                        SPRING 1969
C     AGGREGATION OF A LARGE SYSTEM OF N1 FACILITIES INTO A SET OF N2
C     TERMINOLOGY OF VARIABLES
C         R            .READ/INPUT
C         W            .WRITE/OUTPUT
C         N1           .NUMBER OF ORIGINAL FACILITIES
C         N2           .EXPECTED NUMBER OF CLUSTERED FACILITIES
C         NERR         .ACCEPTABLE VARIATION OF N2
C         N            .EFFECTIVE NUMBER OF CLUSTERED FACILITIES
C         DISCR        .RANGE OF AGGLOMERATION,X(I) OR Y(I) + OR - DISCR
C         DINC         .INCREMENTAL DECREASE OR INCREASE OF DISCR TO OBTAIN N IN
C                       THE NEIGHBORHOOD OF N2
C         X(I),Y(I).CARTESIAN COORDINATES OF FACILITIES
C         XR(I)        .TRANSPORT RATE ON ROUTE I
C         XM(I)        .QUANTITY TO TRANSPORT ON ROUTE I
C         NLOOP        .MAXIMUM NUMBER OF TIMES WE WANT THE PROGRAM TO RUN
C                       THROUGH A CLUSTERING LOOP. IF N2 + OR - NERR IS NOT REACHE
C                       DURING THESE NLOOP, MODIFY INC OR NERR BY CHECKING BEHAVIO
C                       OF OUTPUT N
C         KN           .MAXIMUM NUMBER OF POINTS IN A CLUSTER
      INTEGER R,W
C     CHANGE THE DIMENSION CARD IF MORE THAN  125 FACILITIES ARE CONSIDERED
      DIMENSION X( 125),Y( 125),XR( 125),XM( 125),ISAV( 125)
C     DIMENSION OF JCH=KN=(N1/N2)*5, OF JSAV=NIC=2*N1
      DIMENSION JCH( 30),JSAV( 250)
      R=2
      W=3
      READ(R,10)N1,N2,NERR,NLOOP,DISCR,DINC
   10 FORMAT(4I10,2F10.0)
      DO 25 I=1,N1
      READ(R,20)X(I),Y(I),XR(I),XM(I)
   20 FORMAT(4F15.0)
   25 CONTINUE
C     FIRST TABULATION. LISTING OF KNOWN VARIABLES
      WRITE(W,30)N1
   30 FORMAT(1H1,//,35X,33HVARIABLE DISCRIMINATION ALGORITHM,/,31X,
     124HLOCATION OF THE ORIGINAL,I5,2X,10HFACILITIES,//,24X,21HCARTESIA
     2N COORDINATES,6X,14HTRANSPORT RATE,3X,11HQUANTITY TO,/,69X,9HTRANS
     3PORT,/,27X,1HX,14X,1HY,15X,1HR,14X,1HM,//)
      WRITE(W,40)(I,X(I),Y(I),XR(I),XM(I),I=1,N1)
   40 FORMAT(13X,3HI =,I5,F13.3,F15.3,F14.3,F16.3)
C     SECOND TABULATION. HEADING
      WRITE(W,50)N1,N2,NERR
   50  FORMAT(1H1,33X,33HVARIABLE DISCRIMINATION ALGORITHM,/,35X,
     113HCLUSTERING OF,I5,2X,10HFACILITIES,/,39X,4HINTO,I5,7H + OR -,I5,
     2//)
      LOOP =0
   60 LOOP=LOOP+1
      DO 61 I=1,N1
```

```
   61 ISAV(I)=I
C     CLUSTERING OF FACILITIES
      DO 120 I=1,N1
      XST=X(I)-DISCR
      XFIN=X(I)+DISCR
      YST=Y(I)-DISCR
      YFIN=Y(I)+DISCR
      DO 110 J=1,N1
      IF(I-J)65,110,65
   65 IF(X(J)-XST)110,70,70
   70 IF(X(J)-XFIN)80,80,110
   80 IF(Y(J)-YST)110,90,90
   90 IF(Y(J)-YFIN)100,100,110
  100 ISAV(J)=I
  110 CONTINUE
  120 CONTINUE
C     COUNTING PROCEDURE OF CLUSTERS, UP TO STATEMENT 186
  150 N=0
      IC=0
      NIC=2*N1
      DO 152 IBC=1,NIC
  152 JSAV(IBC)=0
  153 KN=(N1/N2)*5
C     EXAMINATION OF EACH CLUSTER AND CHECK THE CHAIN LINKS
      DO 186 J=1,N1
      DO 154 KM=1,KN
  154 JCH(KM)=0
C     CHECK THAT J IS NOT ALREADY PART OF A CHAIN, IF IT IS, DO NOT COUNT IT
      DO 156 NS=1,NIC
      IF(J-JSAV(NS))156,186,156
  156 CONTINUE
  158 K=1
      JCH(K)=J
      JN=J
  160 JCH(K+1)=ISAV(JN)
      IF(K-1)166,162,166
  162 IF(JCH(1)-JCH(K+1))164,174,164
  164 GO TO 168
  166 IF(JCH(K-1)-JCH(K+1))168,174,168
  168 K=K+1
      IF(K-KN)172,172,169
  169 WRITE(W,170)
  170 FORMAT(10X,69HERROR. INCREASE DIMENSION OF JCH AND MODIFY STATEMEN
     1T 153 ACCORDINGLY)
  172 JN=JCH(K)
      GO TO 160
  174 N=N+1
C     SAVE ALL JCH LARGER THAN J, INTO JSAV
      K1=K+1
      DO 184 IN=1,K1
      IF(JCH(IN)-J)184,184,176
  176 IC=IC+1
      IF(IC-NIC)182,182,178
```

```
  178 WRITE(W,180)
  180 FORMAT(10X,34HERROR. INCREASE DIMENSION FOR JSAV)
  182 JSAV(IC)=JCH(IN)
  184 CONTINUE
  186 CONTINUE
C     CHECK IF PROPER CLUSTERING HAS BEEN OBTAINED
      INCN=IABS(N2-N)
      IF(INCN-NERR)213,213,210
  210 IF(NLOOP-LOOP)211,211,220
  211 WRITE(W,212)
  212 FORMAT(31X,39HDESIRED CLUSTERING HAS NOT BEEN REACHED,/,27X,
     146HMODIFY THE VALUES OF DISCR OR DINC ACCORDINGLY,//)
      GO TO 215
  213 WRITE(W,214)
  214 FORMAT(33X,35HDESIRED CLUSTERING HAS BEEN REACHED,//)
  215 WRITE(W,216)
  216 FORMAT(21X,8HFACILITY,10X,21HCARTESIAN COORDINATES,11X,9HCLUSTERED
     1,/,23X,4HCODE,15X,1HX,14X,1HY,11X,13HWITH FACILITY,//)
      WRITE(W,217)(J,X(J),Y(J),ISAV(J),J=1,N1)
  217 FORMAT(21X,3HI =,I5,F17.3,F15.3,I16)
      WRITE(W,218)N,DISCR
  218 FORMAT (//,29X,32HTOTAL NUMBER OF CLUSTERED POINTS,I9,/,29X,33HOBT
     1AINED WITH A DISCRIMINATION OF,F8.3)
      GO TO 250
C     MODIFY DISCRIMINATING POWER ACCORDING TO THE VALUE OF N
  220 IF(N2-N)230,213,240
  230 DISCR=DISCR+DINC
      GO TO 60
  240 DISCR=DISCR-DINC
      GO TO 60
  250 CALL EXIT
      END

UNREFERENCED STATEMENTS
 150    153    158

FEATURES SUPPORTED
 ONE WORD INTEGERS
 IOCS

CORE REQUIREMENTS FOR
 COMMON      0  VARIABLES   1442  PROGRAM   1062

END OF COMPILATION
```

VARIABLE DISCRIMINATION ALGORITHM
LOCATION OF THE ORIGINAL 125 FACILITIES

	CARTESIAN COORDINATES X	Y	TRANSPORT RATE R	QUANTITY TO TRANSPORT M
I = 1	0.277	28.174	1.000	2.000
I = 2	1.074	93.066	1.000	2.000
I = 3	1.747	13.836	1.000	2.000
I = 4	3.334	87.345	1.000	2.000
I = 5	3.673	7.112	1.000	2.000
I = 6	3.894	1.525	1.000	2.000
I = 7	4.330	15.335	1.000	2.000
I = 8	5.174	86.615	1.000	2.000
I = 9	5.397	87.284	1.000	2.000
I = 10	6.045	82.979	1.000	2.000
I = 11	7.374	93.216	1.000	2.000
I = 12	7.718	56.749	1.000	2.000
I = 13	8.870	60.205	1.000	2.000
I = 14	9.381	24.237	1.000	2.000
I = 15	10.700	96.945	1.000	2.000
I = 16	11.709	74.970	1.000	2.000
I = 17	12.191	7.668	1.000	2.000
I = 18	13.162	4.329	1.000	2.000
I = 19	13.912	0.331	1.000	2.000
I = 20	15.249	4.341	1.000	2.000
I = 21	15.891	81.267	1.000	2.000
I = 22	16.232	7.289	1.000	2.000
I = 23	17.108	59.919	1.000	2.000
I = 24	17.914	68.586	1.000	2.000
I = 25	19.045	78.228	1.000	2.000
I = 26	19.730	61.533	1.000	2.000
I = 27	20.304	32.007	1.000	2.000
I = 28	21.146	78.281	1.000	2.000
I = 29	21.614	48.435	1.000	2.000
I = 30	22.485	51.049	1.000	2.000
I = 31	23.155	34.308	1.000	2.000
I = 32	23.984	68.045	1.000	2.000
I = 33	24.497	7.085	1.000	2.000
I = 34	25.401	34.796	1.000	2.000
I = 35	26.589	38.362	1.000	2.000
I = 36	27.813	29.532	1.000	2.000
I = 37	28.200	99.444	1.000	2.000
I = 38	29.390	71.763	1.000	2.000
I = 39	30.697	75.686	1.000	2.000
I = 40	31.306	65.012	1.000	2.000
I = 41	32.488	6.057	1.000	2.000
I = 42	33.522	90.409	1.000	2.000
I = 43	35.837	53.606	1.000	2.000
I = 44	36.428	74.129	1.000	2.000
I = 45	36.989	13.310	1.000	2.000
I = 46	37.928	59.878	1.000	2.000

I =	47	38.122	97.961	1.000	2.000
I =	48	39.000	94.343	1.000	2.000
I =	49	39.859	40.672	1.000	2.000
I =	50	40.759	30.872	1.000	2.000
I =	51	41.993	53.322	1.000	2.000
I =	52	42.646	92.783	1.000	2.000
I =	53	42.890	74.764	1.000	2.000
I =	54	43.467	13.996	1.000	2.000
I =	55	44.015	3.139	1.000	2.000
I =	56	44.348	16.637	1.000	2.000
I =	57	45.184	81.644	1.000	2.000
I =	58	46.254	47.356	1.000	2.000
I =	59	46.631	60.987	1.000	2.000
I =	60	47.720	39.255	1.000	2.000
I =	61	48.729	54.781	1.000	2.000
I =	62	48.925	11.619	1.000	2.000
I =	63	49.418	19.348	1.000	2.000
I =	64	50.859	61.171	1.000	2.000
I =	65	51.528	81.927	1.000	2.000
I =	66	52.484	34.795	1.000	2.000
I =	67	53.038	30.206	1.000	2.000
I =	68	53.817	13.795	1.000	2.000
I =	69	54.209	96.220	1.000	2.000
I =	70	55.251	74.346	1.000	2.000
I =	71	55.964	51.487	1.000	2.000
I =	72	56.811	16.526	1.000	2.000
I =	73	57.471	24.758	1.000	2.000
I =	74	58.101	51.645	1.000	2.000
I =	75	59.486	43.301	1.000	2.000
I =	76	60.131	20.236	1.000	2.000
I =	77	61.636	31.000	1.000	2.000
I =	78	62.898	84.788	1.000	2.000
I =	79	64.300	88.995	1.000	2.000
I =	80	65.281	35.686	1.000	2.000
I =	81	65.936	87.652	1.000	2.000
I =	82	67.288	79.207	1.000	2.000
I =	83	67.849	80.892	1.000	2.000
I =	84	68.460	32.725	1.000	2.000
I =	85	69.222	60.011	1.000	2.000
I =	86	70.068	18.799	1.000	2.000
I =	87	70.850	52.394	1.000	2.000
I =	88	71.326	53.823	1.000	2.000
I =	89	72.066	62.293	1.000	2.000
I =	90	73.376	16.223	1.000	2.000
I =	91	74.074	62.067	1.000	2.000
I =	92	74.892	58.270	1.000	2.000
I =	93	75.505	10.108	1.000	2.000
I =	94	75.616	10.442	1.000	2.000
I =	95	76.083	18.093	1.000	2.000
I =	96	76.792	38.968	1.000	2.000
I =	97	77.804	10.756	1.000	2.000
I =	98	78.551	56.747	1.000	2.000
I =	99	78.845	7.628	1.000	2.000
I =	100	79.736	91.551	1.000	2.000
I =	101	79.853	79.402	1.000	2.000

I = 102	80.241	99.316	1.000	2.000
I = 103	80.962	76.480	1.000	2.000
I = 104	81.397	46.533	1.000	2.000
I = 105	82.071	54.807	1.000	2.000
I = 106	82.304	49.257	1.000	2.000
I = 107	83.214	64.486	1.000	2.000
I = 108	84.436	30.651	1.000	2.000
I = 109	85.342	27.120	1.000	2.000
I = 110	85.744	65.825	1.000	2.000
I = 111	87.325	64.317	1.000	2.000
I = 112	88.250	29.594	1.000	2.000
I = 113	89.058	75.768	1.000	2.000
I = 114	90.167	79.096	1.000	2.000
I = 115	90.824	31.067	1.000	2.000
I = 116	91.503	8.102	1.000	2.000
I = 117	92.115	53.690	1.000	2.000
I = 118	92.770	30.653	1.000	2.000
I = 119	93.921	2.857	1.000	2.000
I = 120	94.357	91.666	1.000	2.000
I = 121	94.810	55.524	1.000	2.000
I = 122	96.087	40.605	1.000	2.000
I = 123	96.813	42.783	1.000	2.000
I = 124	97.891	77.267	1.000	2.000
I = 125	98.426	10.120	1.000	2.000

VARIABLE DISCRIMINATION ALGORITHM
CLUSTERING OF 125 FACILITIES
INTO 50 + OR - 30

DESIRED CLUSTERING HAS BEEN REACHED

FACILITY CODE	CARTESIAN COORDINATES X	Y	CLUSTERED WITH FACILITY
I = 1	0.277	28.174	1
I = 2	1.074	93.066	2
I = 3	1.747	13.836	7
I = 4	3.334	87.345	10
I = 5	3.673	7.112	5
I = 6	3.894	1.525	6
I = 7	4.330	15.335	3
I = 8	5.174	86.615	10
I = 9	5.397	87.284	10
I = 10	6.045	82.979	9
I = 11	7.374	93.216	15
I = 12	7.718	56.749	13
I = 13	8.870	60.205	12
I = 14	9.381	24.237	14
I = 15	10.700	96.945	11
I = 16	11.709	74.970	16
I = 17	12.191	7.668	22
I = 18	13.162	4.329	22
I = 19	13.912	0.331	20
I = 20	15.249	4.341	22
I = 21	15.891	81.267	25
I = 22	16.232	7.289	20
I = 23	17.108	59.919	26
I = 24	17.914	68.586	24
I = 25	19.045	78.228	28
I = 26	19.730	61.533	23
I = 27	20.304	32.007	31
I = 28	21.146	78.281	25
I = 29	21.614	48.435	30
I = 30	22.485	51.049	29
I = 31	23.155	34.308	36
I = 32	23.984	68.045	32
I = 33	24.497	7.085	33
I = 34	25.401	34.796	35
I = 35	26.589	38.362	34
I = 36	27.813	29.532	31
I = 37	28.200	99.444	37
I = 38	29.390	71.763	39
I = 39	30.697	75.686	38
I = 40	31.306	65.012	40
I = 41	32.488	6.057	41
I = 42	33.522	90.409	42
I = 43	35.837	53.606	43

I = 44	36.428	74.129	44
I = 45	36.989	13.310	45
I = 46	37.928	59.878	46
I = 47	38.122	97.961	48
I = 48	39.000	94.343	52
I = 49	39.859	40.672	49
I = 50	40.759	30.872	50
I = 51	41.993	53.322	51
I = 52	42.646	92.783	48
I = 53	42.890	74.764	53
I = 54	43.467	13.996	56
I = 55	44.015	3.139	55
I = 56	44.348	16.637	54
I = 57	45.184	81.644	57
I = 58	46.254	47.356	58
I = 59	46.631	60.987	64
I = 60	47.720	39.255	66
I = 61	48.729	54.781	61
I = 62	48.925	11.619	68
I = 63	49.418	19.348	63
I = 64	50.859	61.171	59
I = 65	51.528	81.927	65
I = 66	52.484	34.795	67
I = 67	53.038	30.206	66
I = 68	53.817	13.795	72
I = 69	54.209	96.220	69
I = 70	55.251	74.346	70
I = 71	55.964	51.487	74
I = 72	56.811	16.526	76
I = 73	57.471	24.758	76
I = 74	58.101	51.645	71
I = 75	59.486	43 301	75
I = 76	60.131	20.236	73
I = 77	61.636	31.000	80
I = 78	62.898	84.788	83
I = 79	64.300	88.995	81
I = 80	65.281	35.686	84
I = 81	65.936	87.652	79
I = 82	67.288	79.207	83
I = 83	67.849	80.892	82
I = 84	68.460	32.725	80
I = 85	69.222	60.011	91
I = 86	70.068	18.799	90
I = 87	70.850	52.394	88
I = 88	71.326	53.823	92
I = 89	72.066	62.293	92
I = 90	73.376	16.223	95
I = 91	74.074	62.067	92
I = 92	74.892	58.270	98
I = 93	75.505	10.108	99
I = 94	75.616	10.442	99
I = 95	76.083	18.093	90
I = 96	76.792	38.968	96
I = 97	77.804	10.756	99
I = 98	78.551	56.747	105

I =	99	78.845	7.628	97
I =	100	79.736	91.551	100
I =	101	79.853	79.402	103
I =	102	80.241	99.316	102
I =	103	80.962	76.480	101
I =	104	81.397	46.533	106
I =	105	82.071	54.807	98
I =	106	82.304	49.257	104
I =	107	83.214	64.486	111
I =	108	84.436	30.651	112
I =	109	85.342	27.120	112
I =	110	85.744	65.825	111
I =	111	87.325	64.317	110
I =	112	88.250	29.594	118
I =	113	89.058	75.768	114
I =	114	90.167	79.096	113
I =	115	90.824	31.067	118
I =	116	91.503	8.102	116
I =	117	92.115	53.690	121
I =	118	92.770	30.653	115
I =	119	93.921	2.857	119
I =	120	94.357	91.666	120
I =	121	94.810	55.524	117
I =	122	96.087	40.605	123
I =	123	96.813	42.783	122
I =	124	97.891	77.267	124
I =	125	98.426	10.120	125

TOTAL NUMBER OF CLUSTERED POINTS 75
OBTAINED WITH A DISCRIMINATION OF 5.000

I =	99	78.845	7.628	97
I =	100	79.736	91.551	100
I =	101	79.853	79.402	103
I =	102	80.241	99.316	102
I =	103	80.962	76.480	101
I =	104	81.397	46.533	106
I =	105	82.071	54.807	98
I =	106	82.304	49.257	104
I =	107	83.214	64.486	111
I =	108	84.436	30.651	112
I =	109	85.342	27.120	112
I =	110	85.744	65.825	111
I =	111	87.325	64.317	110
I =	112	88.250	29.594	118
I =	113	89.058	75.768	114
I =	114	90.167	79.096	113
I =	115	90.824	31.067	118
I =	116	91.503	8.102	116
I =	117	92.115	53.690	121
I =	118	92.770	30.653	115
I =	119	93.921	2.857	119
I =	120	94.357	91.666	120
I =	121	94.810	55.524	117
I =	122	96.087	40.605	123
I =	123	96.813	42.783	122
I =	124	97.891	77.267	124
I =	125	98.426	10.120	125

TOTAL NUMBER OF CLUSTERED POINTS 75
OBTAINED WITH A DISCRIMINATION OF 5.000