A DISTRIBUTION FREE EMPIRICAL METHOD FOR CLASSIFICATION OR CLUSTERING OF VECTOR-VALUED SAMPLES

Daniel Albert Rohrback

NAVAL POSTGRADUATE SCHOOL

Monterey, California

THESIS

A Distribution Free Empirical Method
for Classification or Clustering
of Vector-Valued Samples

by

Daniel Albert Rohrback

Thesis Advisor: B. O. Shubert

March 1974

Approved for public release; distribution unlimited.

A Distribution Free Empirical Method for Classification or Clustering of Vector-Valued Samples

by

Daniel Albert Rohrback
Lieutenant, United States Navy
B.S., Oregon State University, 1966

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN OPERATIONS RESEARCH

[illegible]
R [illegible]
c.

ABSTRACT

An algorithm pertinent to pattern classification and cluster analysis of n-dimensional (n>2) data is presented. The algorithm simplifies the dimensionality of the sample set by a perpendicular projection of each point onto a plane which is determined by two orthonormal vectors. The operator views the plane on a computer graphics screen, selects a point on the screen with a lightpen, and attempts to achieve separation of classes by choosing a "direction of movement." Each iteration, a movement of a single point in a certain direction, is actually a reorientation of the plane into which the data points have been projected. These iterations continue until the operator decides successful separation has been accomplished. The method is neither restricted by statistical distribution assumptions nor the dimensions of the sample set. From the investigation of seven case studies, this algorithm emerges as a tremendous assistance in the analysis of multidimensional data.

TABLE OF CONTENTS

I. INTRODUCTION-- 4

II. DISCUSSION OF THE PREVIOUS ALGORITHM----------- 7

III. DISCUSSION OF THE NEW ALGORITHM---------------- 11

IV. ANALYSIS OF THE SEVEN CASES-------------------- 17

V. CONCLUSIONS AND RECOMMENDATIONS---------------- 34

APPENDIX I. COMPUTER PROGRAM LISTING---------------- 37

APPENDIX II. PROGRAM OPERATING INSTRUCTIONS---------- 55

APPENDIX III.DATA LISTING------------------------------ 62

APPENDIX IV. TABLE OF A AND B VECTORS---------------- 92

APPENDIX V. TWO DIMENSIONAL PLOTS------------------ 93

BIBLIOGRAPHY--153

INITIAL DISTRIBUTION LIST---------------------------154

FORM DD 1473---155

I. INTRODUCTION

The algorithm presented in this paper is an improvement on the separation algorithm presented in the thesis by Gilbert Lauzon [1]. He developed a separation algorithm to use in pattern classification and cluster analysis of n-dimensional data.

The algorithm was written in Fortran IV for the XDS 9300 computer installation at the Naval Postgraduate School. The Adage Graphics and Text (AGT) system is linked directly to the digital computer. The Fortran callable graphics subroutines in the XDS 9300 allow for communications between the two systems. Also, by using the library program "MAD" of the AGT system, the lightpen and sixteen function switches permit an alteration of the display on the graphics screen.

The computer program projects each point into a plane which decreases the dimensionality of the analysis from the original n (>2) to 2. The points on the plane are presnted to the operator on a cathode ray tube. The user reorients the plane on each iteration by selecting a point and choosing a "direction of movement." Using the XDS 9300/AGT-10 system, the operator makes the decision whether to proceed with a different orientation of the plane or terminate the analysis. The human decision is based on the pattern of points presented on the screen. The interaction of the man-machine system is emphasized.

Many classification problems can be approached in a graphical diagram. A two-dimensional plot of the features is drawn. From this, the human must suggest a rule that partitions the feature space into regions, one for each category. The rule can be a series of straight or curved lines depending on the number of categories. Since the decision is based on the perception of the operator, the rule is quite arbitrary. Therefore, the analyst must insure on a proper number of samples in each category. In addition, the human tries to determine the rule which according to his subjective judgement minimizes the probability of error. In essence, this means that the probability of a random observation from a specific category falling outside its respective region defined by the rule should be made as small as possible. The preceding analogy is the basic characteristics that prevail in the separation algorithm. The graphical diagram is the display screen with the projections from the computations of the digital computer. The rule is solely determined by the operator from the video presentation of the graphical diagram.

The first modification incorporated into G. Lauzon's algorithm was to insure that the A and B vectors were orthonormal on each iteration through the computer program. Basically, this means that the two vectors are perpendicular and each vector has a length of one.

Secondly, the existing "X-Y" video display was changed to a more discernible pair of symobls consisting of the "X-O"

scheme. This improvement greatly enhanced the ability of the operator to visually identify the two groups on the graphics screen.

Thirdly, supplementary video information was added to the display on the screen. Each iteration of the algorithm involves the selection of a point and a "direction of movement." These operations are controlled by a sequence of steps utilizing the sixteen function switches and the lightpen. The function switches have been numbered left to right and top to bottom for easy reference. A record of the use of these function switches is most beneficial to the operator. So, the numbers of two function switches have been displayed in the upper left hand corner of the screen. The one number records the function switch corresponding to the type of symbol selected and the second keeps a record of the other function switches. The modification was incorporated to merely assist the operator and did not contribute to improving the separation algorithm.

II. DISCUSSION OF THE PREVIOUS ALGORITHM

The previous separation algorithm was presented in the thesis by Lt. Gilbert Lauzon [1]. The algorithm is applicable to use in pattern classification and cluster analysis of n-dimensional data. The computer graphics system projects each point of a data set into a plane. The projection is not necessarily orthonormal. The method reduces the dimensionality of the problem from n to 2.

First, a description of the algorithm will be presented for a pattern classification problem. Assume there are two classes of objects, P_1 and P_2. These two groups are comprised of L-dimensional vectors of numbers. These numbers may be measurements or merely some assigned values of nature. The goal of the analysis is to determine the existence of a partition of the two groups. Once a separation has been achieved and a discriminant function has been determined, it can be used to classify new objects by the rule:

$$f(Z) > 0 \rightarrow Z \text{ is a member of the set } P_1;$$

$$f(Z) < 0 \rightarrow Z \text{ is a member of the set } P_2.$$

The two groups are considered separable if an appropriate discriminant function can be specified. Otherwise, the classes are inseparable and no discriminant function can be assigned.

The L-dimensional vectors of the measurements on objects from the classes P_1 and P_2 will be denoted by X and Y, respectively. The algorithm commences by calculating the average value, DAV, of each of the L-components of the data set.

$$DAV(K) = \left(\frac{1}{M + N}\right) \left(\sum_{I=1}^{M} Y(I,K) + \sum_{J=1}^{N} X(J,K)\right) \; ; \; K = 1,\ldots,L$$

Here M and N are the number of vectors in P_1 and P_2, respectively. Using the L-component vector DAV, the data points are centered as follows:

$$X'(J,K) = X(J,K) - DAV(K) \; ; \; J=1,\ldots,M \text{ and } K=1,\ldots,L$$

$$Y'(I,K) = Y(I,K) - DAV(K) \; ; \; I=1,\ldots,N \text{ and } K=1,\ldots,L$$

Two arbitrary non-zero vectors, A and B, in Euclidean L-space are chosen. A linear transformation is computed for each point using the two vectors. This amounts to projecting each vector to a point in Euclidean 2-space with coordinates

$$T_{A,B}(Z) = (A{\cdot}Z,\; B{\cdot}Z)$$

where $A{\cdot}Z = \sum_{K=1}^{L} a_K z_K$, $B{\cdot}Z = \sum_{K=1}^{L} b_K z_K$, Z is an arbitrary point from the X or Y set. The coordinates are scaled so the points can be displayed on the AGT screen. The procedure determines the maximum absolute value of both coordinates as shown:

$$\text{TMAX} = \max\left(\max_{J} |A\cdot X| , \max_{I} |A\cdot Y|\right) \qquad I = 1,\ldots,N$$

$$\text{SMAX} = \max\left(\max_{J} |B\cdot X| , \max_{I} |B\cdot Y|\right) \qquad J = 1,\ldots,M$$

Each coordinate is divided by the respective maximum value.

$$T_{A,B}(Z) = (A\cdot Z/\text{TMAX},\ B\cdot Z/\text{SMAX})$$

At this point, the video display will appear depicting the projections of the X and Y points. The operator selects a point and chooses a "direction of movement." For example, if the tenth X point is picked and a move right is chosen, a new vector, A', is computed.

$$A'(K) = A(K) + X(10,K)\ ;\ K = 1,\ldots,L$$

The new values of the A and B vectors are dependent on the "direction of movement" and the selected point as follows:

Direction of Move	New A	New B
RIGHT	A + Z	B
LEFT	A - Z	B
UP	A	B + Z
DOWN	A	B - Z

The computer returns to the linear transformation calculations and repeats the procedure using the new vectors.

The goal of the procedure is a separation of X's and Y's into disjoint areas of the plane (screen). The operator proceeds using the lightpen and sixteen function switches until

an appropriate separation is attained or discontinues the analysis. If the two classes have been successfully separated, the discriminant function f(Z) can be specified. For instance, if the X's and Y's on the screen can be separated by a straight line, the discriminant function would be of the form:

$$f(Z) = d\ (A \cdot Z/TMAX) + e\ (B \cdot Z/SMAX) + f$$

where d, e, and f are scalars. The function need not be unique. The pattern classification is completed once the discriminant function has been determined.

The separation algorithm for the cluster analysis problem is very much like that in the pattern classification. Normally, the set of data is not broken down into two classes. The entire set is presented as a set of Y's. The concept of this method is the determination of groupings from the sample set. It is normally used on data where the goal of the analysis is to detect clusters. The final groupings of the data points into the appropriate number of disjoint subsets is the decision of the operator. Each of the subsets represents a cluster.

III. DISCUSSION OF THE NEW ALGORITHM

As mentioned in Section I, the separation algorithm was modified using orthonormal vectors, A and B. This modification is important for both the pattern classification and the cluster analysis problems. The following paragraphs will describe in detail the concept of the orthonormal projection algorithm.

First, let us assume the analysis is conducted on a pattern classification data set. The data has been divided into N vectors in P_1 and M vectors in P_2. The P_1 set is read by the computer as the Y points and the P_2 set as the X points. The data points have been centered by calculating the L-dimensioned vector, DAV, by averaging each component of the entire data set.

Next, the operator must select the A and B vectors to be inserted into the computer memory. If the inner product between two vectors is zero, then the two are perpendicular. The length of any vector is determined by taking the square root of the inner product with itself. For example, the length of a vector A, denoted by ||A||, is given by

$$||A|| = (A \cdot A)^{\frac{1}{2}} = (\sum_{K=1}^{L} a_K{}^2)^{\frac{1}{2}}$$

Now the linear transformation is performed on each point as follows:

$$T_{A,B}\ (Z) = (A \cdot Z,\ B \cdot Z);\ A \cdot Z = \sum_{K=1}^{L} a_K z_K,\ B \cdot Z = \sum_{K=1}^{L} b_K z_K$$

This transformation is an orthonormal projection of the point Z (an arbitrary X or Y) into the plane defined by the orthonormal vectors, A and B. The two sets of projections are presented to the operator by the video display on the screen.

The scaling method described in Section II is performed on the data points in order for them to be displayed on the screen. By using only one scale factor, the same perspective of the two projections will be retained. This scaling term, VMAX, is determined as follows:

$$\mathrm{VMAX} = \max\left(\max_{J} (|A \cdot X|, |B \cdot X|), \max_{I} (|A \cdot Y|, |B \cdot Y|)\right)$$

where J = 1,...,M and I = 1,...,N. The final values of the coordinates from the linear transformation will be

$$T_{A,B}(Z) = (A \cdot Z/\mathrm{VMAX}, B \cdot Z/\mathrm{VMAX})$$

The first iteration of the algorithm commences when the operator picks a point on the screen and chooses a "direction of movement." Actually, the computations do not "move" the point selected but reorients the plane upon which the entire data set is projected. The reorientation of plane is accomplished by computing two new orthonormal vectors. Suppose that the user has selected an arbitrary point Z and desires to move it right. The first step in the computation is that of recomputing the A vector,

$$A'(K) = A(K) + Z(K) \ ; \ K = 1,\ldots,L$$

For the projection of Z to shift horizontally to the right, the new vector, B', should be the same as B. But this is impossible if A' and B' are to be perpendicular. Hence, the projection of Z shifts also up and down. We want this vertical shift to be minimal. It is best to follow the remainder of the concept to finding such a B vector with mathematical expressions.

The appropriate symbols will be defined as:

A and B = old orthonormal vectors

A' and B' = new orthonormal vectors

$|A \cdot B|$ = absolute value of the inner product of the vectors A and B

The new vector A' must be changed so it remains constant with a unit length. This is done by dividing the vector (A + Z) by its length as such:

$$A' = (A + Z)/||(A + Z)||$$

This is the new vector which will be used in recomputation of the new projections.

The new B vector will be determined using Lagrangian minimization. This technique is applicable because the requirement to determine the minimum value of the objective function, $|Z \cdot (B' - B)|$, has the two restrictions imposed on it. Now, to find the B' which minimizes the difference between $Z \cdot B'$ and $Z \cdot B$, the following problem will be solved:

$$\text{Minimize} \quad |Z\cdot(B' - B)| \tag{1}$$

$$\text{subject to} \quad B'\cdot A' = 0 \tag{2}$$

$$||B'||^2 = 1 \tag{3}$$

Using the Lagrangian method to find the solution, let

$$F = \pm Z\cdot(B'-B) - \lambda_1 (B'\cdot A') - \lambda_2 (||B'||^2 - 1) \tag{4}$$

where λ_1 and λ_2 are the Lagrangian multipliers. Taking the first derivative with respect to B' and setting the result equal to zero:

$$\frac{\partial F}{\partial B'} = \pm Z - \lambda_1 A' - 2\lambda_2 B' = 0 \tag{5}$$

Using equations (2) and (5), the value for λ_1 is:

$$\lambda_1 = \pm Z\cdot A' \tag{6}$$

This result is true if and only if λ_2 has a non-zero value. Using equations (3) and (5), the value for λ_2 can be determined as:

$$\lambda_2 = \pm \frac{||Z - (Z\cdot A')\ A'||}{2} \tag{7}$$

Substituting the values of λ_1 and λ_2 into equation (5), the final solution to the minimization problem is:

$$B' = \pm \frac{(Z - (Z\cdot A')\ A')}{||(Z - (Z\cdot A')\ A')||} \tag{8}$$

The final step is the determination of the sign to use for the B' vector. By using the inner product of Z with both sides of

equation (8) and the second derivative of equation (4), the following statements can be shown to be true:

(1) Use + sign if sign $(Z \cdot B') \neq$ sign $(Z - (Z \cdot A'))$

(2) Use - sign if sign $(Z \cdot B') =$ sign $(Z - (Z \cdot A'))$

The same reasoning can be extended to "moving" an arbitrary point Z to the left. However, in selecting a point Z and "moving" it up, the new vector B' will be determined first. This time the minimization problem will be:

Minimize $|Z \cdot (A' - A)|$

Subject to $B' \cdot A' = 0$

$||A'||^2 = 1$

The following table gives a guide to the computations of the new vectors using an arbitrary point Z.

Direction of Move	New A	New B
RIGHT	$(A + Z)/\|\|A + Z\|\|$	B*
LEFT	$(A - Z(/\|\|A - Z\|\|$	B*
UP	A*	$(B + Z)/\|\|B + Z\|\|$
DOWN	A*	$(B - Z)/\|\|B - Z\|\|$

*Denotes the use of the minimization technique.

These new A and B vectors are used in the recomputation of the linear transformation T on the data set. After the projections have been calculated, a new display of the plane depicting the two subsets will appear on the screen. The operator can make

another selection or terminate the analysis depending on the presentation of the points.

The "X-Y" letters in the old routine were changed to the "X-O" symbols. This was done to make the presentation easier for the user to discriminate the two data sets. The similarity between an X and Y made it perplexing when the two different symbols were in close proximity. The two letters are sketched on the screen through a set of draw and move instructions. These computer instructions are stored in a vector called IPACK.

Video information of the function switches was also incorporated into the computer program. Now, when the operator depresses one of the sixteen function switches, the number of the switch is displayed in the upper left corner of the screen. The number in the first row indicates the function switch for selecting either an X or a Y point. The value of these commands will be an 11 or 3, respectively. The remaining function switches will be indicated on the second row. This additional video display is controlled by a series of ENCODE and CALL TEXTO instructions in the computer program. The role of the function switches is to communicate between the Fortran graphics package "MAD" and the 9300 computer. The display of the function switch numbers assist the operator in efficient execution of the program.

IV. ANALYSIS OF THE SEVEN CASES

The operation of the separation algorithm using the orthonormal vectors was first examined using four sets of artificially generated coordinates. The first set was just to test the modified program. The remaining three were generated for the sake of testing the applicability of the method to pattern classification. Each set of points filled various geometrical shapes. This quality allowed for ensuring proper appraisal of the procedure using multi-dimensional data. Finally, three cases of measurements from "real world" applications were analyzed to evaluate the performance of the new algorithm. The seven cases that will be discussed are:

TESTS:

(1) One three-dimensional pyramid

(2) Two three-dimensional cubes

(3) Two six-dimensional cubes

(4) One four-dimensional shell enclosing a four-space sphere

APPLICATIONS:

(5) Iris flowers

(6) Fossils

(7) Coronary heart disease

Prior to exemplifying on each case, a few comments are appropriate. The output from the computer printer comprises a sixty-one by thirty-seven dimensional matrix. The printout is a six by six inch square. The outline of the square is shown by asterisks and the coordinate axes are indicated by four zeroes. The symbols denoting the data points on the printout correspond to the plot on the AGT graphics screen. The points will appear as an X, O, 8, E, or J. The "X" and "O" symbols match the positions of the X and Y projections, respectively. An "8" symbol represents two or more Y's in a cell; and "E" refers to two or more X's. A cell denotes a matrix element of the printout array. The matrix is stored in the computer memory. For example, PIC (31,18) in the computer program would refer to the symbol in the cell occupying row 31 and column 18 of this matrix. Lastly, the "J" letter indicates a mixture of X's and Y's occupying the same cell. The coordinate axes have been drawn in by hand. The encircled points signify the X or Y selected in the next iteration and the arrow represents the direction of movement.

For each case, the initial values of the A and B were extracted from the table in Appendix IV. This table represents only one group of an infinite set of possibilities. This particular group was chosen because the absolute values in any dimension are nearly equal. The author felt the use of nearly equal components would initially bring all measurements into the projection by the same relative proportion. It was thought

that this should give a better intitial presentation of the projections than any other choice of values. However, it should be pointed out that the decision on the particular set of values to insert is left up to the user.

Polaroid prints of the AGT plot of the projections have been included with the computer printout. The close resemblance between the printout and the Polaroid pictures is quite apparent. Of course, the Polaroid picture is a true representation of the projections. Any discrepancies in the plot from the computer printout would be due to the restrictions in setting up the matrix array. Adjustments of the vertical and horizontal controls were necessary to center the plot on the screen so a Polaroid print may be distorted but the relative distance between points has been maintained. The elaboration and the conclusion of each case will now be presented.

CASE 1

The first case involved the coordinates of a three-dimensional rectangular pyramid. The points represent the outline of this figure. The data set consisted of twenty-one vectors. Each consecutive iteration will show the rotation of the geometrical shape. The change in the A and B vectors merely reoriented the perspective of the figure. Also the shape of the figure remains true with each iteration. This example was used in evaluating the successful operation of the orthonormal projections.

Figures IV-1-1 through IV-1-5 in Appendix V show a five-step sequence in the application of the orthonormal projection concept. Since this case was done only as a test of the algorithm, no printout of the appropriate terms is given. The outline of this geometrical shape was drawn in by hand in the figures. The Polaroid picture of the AGT display, Figure IV-1-5P in Appendix V, is included to compare with the printout given in Figure IV-1-5.

CASE 2

Two three-dimensional cubes were generated for analysis using the pattern classification technique. The X and Y subsets comprised twenty-tuples depicting the outline of the two geometrical figures. The Y subset has its closest vertex to the origin at the point (3,2,5). Each of its sides has a length of two units. The other cube (X subset) had its closest vertex at (7,6,9) and also two units on a side. The two cubes were made disjoint. This made it feasible to specify a discriminant function depicting the partitioned subsets. Again, this was another test in evaluating the procedure but involved two disjoint figures.

Figures IV-2-1 through IV-2-5 in Appendix V show a six step sequence in the application of the procedure on the two cubes. Separation was maintained throughout the sequence. The values of the A and B vectors have been indicated below the figures. The initial figure used the values from the table in Appendix IV.

A possible discriminant function in Figure IV-2-5 can be specified as

$$f(Z) = A\cdot(Z - DAV)/VMAX$$

where A has the values printed from Figure IV-2-5. If $f(Z) > 0$, Z is a member of the subset of Y's; if $f(Z) < 0$, Z is an entity in the X's. These results can be checked by the following example. The printout indicated the values for VMAX and DAV;

DAV(1) = 6.000	DAV(2) = 5.000
DAV(3) = 8.000	VMAX = 4.333

Assume that Z has the coordinates of the point Y(1) in Appendix III. The calculations indicate the number for f(Z) is + 0.662. The computation has shown that Z is, in fact, a member of the subset of Y's. Also any other point from the Y's will satisfy the condition that f(Z) will be positive.

One of the two cubes in Figures IV-2-4 and IV-2-5 have been drawn in by hand. Also a figure of the Polaroid print is included to compare with Figure IV-2-5.

CASE 3

For the third application using the new separation algorithm, the dimensional space of the cube was increased from three to six. Seventy-tuples were generated for each subset, X and Y. The unit cubes were generated on the IBM 360 computer as uniform distributed random numbers. A point from the Y

subset would be denoted by $Y(I) = (Y(I,1), Y(I,2), Y(I,3), Y(I,4), Y(I,5), Y(I,6))$ where $I = 1,\ldots,N$ and $1 \leq Y(I,K) \leq 2$, $K = 1,\ldots,6$; and for the X subset $X(J) = (X(J,1), X(J,2), X(J,3), X(J,4), X(J,5), X(J,6))$ where $J = 1,\ldots,M$ and $0 \leq X(J,K) \leq 1$, $K = 1,\ldots,6$. Again, the two cubes were generated so the two samples were disjoint. First, if one of the iterations showed a combination of "X's" and "O's" in an area of the plane, it would be easier to separate the two groups on subsequent steps. Secondly, once separation between the two groups was attained, a discriminant function could be defined.

The printout of the graphical display is shown in Figures IV-3-1 through IV-3-6 in Appendix V. The degree of complexity for the two samples was increased from that of Case 2, but a partition of the two groups was still achieved. Since this case is so similar to Case 2, the discriminant function will not be determined. Consequently, the A and B vectors will also be omitted. In Figure IV-3-2, the two cubes shown by their projections can not be separated because of the mixture of "X's" and "O's" around the center of the graph. It was easy, however, to separate the two as shown in the next step, Figure IV-3-3. The similarity between the printout and the Polaroid snapshot is shown in Figures IV-3-5 and IV-3-5P.

CASE 4

The previous two cases were applications of the partitioning of geometrical shapes which were separable by a hyperplane. A sphere and a shell in four-space were created for the purpose

of making the division of two groups more complex. Twenty-tuples were generated for the X subset and forty-tuples for the Y subset. A typical position of a vector in the Y set is $Y(I) = (Y(I,1), Y(I,2), Y(I,3), Y(I,4))$, where $I = 1,\ldots,N$. The components must satisfy the requirement $2 \leq Y(I,1)^2 + Y(I,2)^2 + Y(I,3)^2 + Y(I,4)^2 \leq 4$. Similarly, an X point must fulfill the condition that $X(J,1)^2 + X(J,2)^2 + X(J,3)^2 + X(J,4)^2 \leq 1$. The two subsets are again disjoint and they form a sphere surrounded by a shell separated by one unit in radial distance. Two different trials on the sample set were conducted. The first run was finally terminated after twenty iterations without achieving successful separation. It should be noted that the arrangement of the two groups made it impossible for an appropriate partition. Of course, this is assuming that the shell has the proper density of points and no voids exist in it. However, if the procedure is modified, separation can be feasible. The second run was carried out by including the square of each component in each entity. The dimension of the sample set becomes twice the original. To clarify this last remark, if there was an arbitrary point, $Z = (Z_1,Z_2,Z_3,Z_4)$ in the original set, it will now become $Z = (Z_1,Z_2,Z_3,Z_4,Z_1^2, Z_2^2,Z_3^2,Z_4^2)$. The second trial was conducted using this eight component vector on both sets.

The first trial on case four appears in Figures IV-4-1 through IV-4-5 in Appendix V. These figures confirm that the separation was never attained. Only the first six steps have been included in the figures. The remaining steps were omitted

since they did not improve the pattern of the projections. Thus, a discriminant function can not be specified. Consequently, the printout of the A and B vectors will be eliminated from the discussion.

The situation improves in the second run using the eight component tuples. This is depicted in Figures IV-4-6 through IV-4-10 in Appendix V. Separation was attained in a mere three steps of the algorithm. This modification of the procedure can be attempted on samples that can not be successfully partitioned using the basic method. The one subset may be enclosed inside a large sphere-like shape including the other subset. This is one approach to obtaining a division of the two groups. The components of the A and B vectors in the second run are printed on the bottom of each of the figures.

CASE 5

This sample set represents the first of three "real world" cases. The data set for this case portrays multiple measurements on three species of Iris flowers. The data was originally used by Ronald A. Fisher in his classic paper on discriminant analysis [3]. It has become widely used in illustrating cluster analysis procedures. The four measurements collected are (1) sepal length, (2) sepal width, (3) petal length, and (4) petal width. The sample set has been reduced from the original 150 to 75 samples for this particular analysis.

This particular sample set has the characteristics for either a cluster analysis or pattern classification problem. The author will approach the investigation using the pattern classification technique. It is quite possible from looking at the values of the three species that the one, Iris setosa, will have projections in an area of the plane by itself. The projections of the other two species appear to be nearly the same. Hence, it may be somewhat difficult to partition these latter two species. Therefore, the observations from Iris versicolor and Iris virginica will be divided into two different subsets. The Y subset will be composed of Iris setosa and Iris versicolor; while the X subset of Iris virginica. Each species of flowers was composed of twenty-five vectors in four dimensional space.

The complexities in measurements is apparent in the partitioning of these three groups. Outliers may exist which indicate an observation is far removed from the main body of the data. They result from the fact that the sample under investigation may contain a certain proportion of extreme cases or from a population other than the sample from which the rest of the data came. An observation from the Iris versicolor species presents itself as a possible outlier in the analysis of the graphic plot. This point is Y(28) in Appendix III which coincides with the projection of X(24). The two appear as one point in the last two figures as a "J" symobl. The point Y(28) could not be separated from the group of X's as it is projected to the left of the line denoting the discriminant function.

Figures IV-5-1 through IV-5-6 in Appendix V show the twelve step sequence in the analysis of the three Iris flowers. Even though a discriminant function is generally determined in partitioning two groups, two linear functions will be specified which divide the three types of flowers. This will be done using Figure IV-5-6. The first function will separate the X subset from the Y subset. Then the second will divide the two Iris species comprising the Y subset. First, the values of the A and B vectors from the bottom of the figure must be recorded.

Now, a linear discriminant function has the form given in Section II as;

$$f(Z) = \frac{d(A(Z - DAV))}{VMAX} + \frac{e(B(Z - DAV))}{VMAX} + f$$

where d, e, and f are scalars. The f scalar is the negative sign of the ordinate intercept while the term d has the negative sign of the slope of the line. Finally, the value of the coefficient e will be one. Using the line in Figure IV-5-6 separating the X and Y groups, an appropriate discriminant function has the form:

$$f(Z) = \frac{6(A(Z - DAV))}{VMAX} + \frac{B(Z - DAV)}{VMAX} + 2.25$$

The magnitude for DAV and VMAX from this figure are:

DAV(1) = 5.872 DAV(2) = 3.061

DAV(3) = 3.804 DAV(4) = 1.212

VMAX = 3.033

Using these figures, the value for f(Z) is - 0.054 when the components of X(24) from Appendix III are substituted into Z. Also carrying out the same computations, the result for f(Z) is + 0.252 for the point Y(26). These calculations have merely emphasized that a positive value for f(Z) implies Z is a member of the Y subset. If f(Z) is negative, then Z is an entity of the X subset.

Now, a description of the line splitting the Y subset must be resolved. The form of the discriminant function has been determined as:

$$f(U) = \frac{A\cdot(U - DAV)}{VMAX} - 0.370$$

where the number 0.370 is the value of the intercept on the abscissa axis. If the coordinates of Y(38) are substituted for the arbitrary point U, the result of f(U) is - 0.030. This calculation indicates that any point from the set of Y's, where $f(U) < 0$, represents an entity from the Iris versicolor species. Any U where $f(U) > 0$ corresponds to an observation from the Iris setosa.

CASE 6

The sample set for this analysis problem is quantitative measurements of morphologic parameters of specific fossils. The data set was extracted from Wright and Switzer's paper titled, "Numerical Classification Applied to Certain Jamaican Eocene Nummulitids" [4]. Eighty-two eight component vectors

were used in the investigation. Six of the original observations had missing parameters and were therefore omitted from the sample set.

Again like Case 5, this sample will be divided into two groups for a pattern classification problem. The author feels the separation will be easier to attain using two subsets. It should be noted that Lauzon investigated these specimens using cluster analysis procedures [1]. The Y subset was initially composed of the first thirty-eight observations. It is comprised of specimens from the Operculinoides wilcoxi species. This species will be denoted by Group A. The X subset was composed of the other three species: Operculinoides jennyi, Operculinoides cushmani, and Nummulites striatoreticulatus. These three types of Nummulitids will be called Groups B, C, and D, respectively, Using these subsets, it was extremely difficult to pick out the projections from Group B and those from Group C. In order to eliminate this confusion, the two subsets were rearranged. Observations from Groups A and B were combined to form the Y subset; Groups C and D joined to make up the X subset. The second run was conducted using the above groupings. Also to make the investigation similar to the discussion in Wright and Switzer's paper, only the first six parameters were used. The results from the analysis were quite remarkable. Even on the first step, three different groups were apparent; Group A, Groups B and C, and Group D. In Wright and Switzer's

investigation, ninety-six iterations were conducted to determine the coefficients of a linear discriminant function. They were able to find two different groupings, Groups A and D and Groups B and C. The same procedure using 96 iterations was repeated to find another linear function dividing Group A and D. At last, a linear function splitting Group B and C was determined. Then the two researchers utilized these two final divisions as input to a standard discriminant analysis program. The output resulted in two canonical variables. The plot of these two variables revealed three groupings: Group A, Groups B and C, and Group D. The two concluded that the specimens from Groups B and C could not be partitioned. The results in this thesis are in agreement with these findings. The procedure used by this author appears to be superior in two ways. First, there are substantially fewer computations, hence the computer time to accomplish the analysis is considerably less. Secondly, the plot of the results is readily available from the printout of the computer. It should also be noted that G. Lauzon's investigation of this sample set showed only two groupings.

Figures IV-6-1 through IV-6-3 in Appendix V show the graphical display obtained on the first run. These figures depict the eight-dimensional analysis of the fossil measurements. In Figure IV-6-2, the three groups have been labeled. There is no visual method to distinguish between the B and C groups. The displays of the second run are given in Figures

IV-6-4 through IV-6-7. The values of the A and B components of these figures are given below each.

The two discriminant functions separating the three groupings will be specified using Figure IV-6-7. First, the tabulation of the numbers for VMAX and DAV are as follows:

DAV(1) =	163.220	DAV(2) =	41.43
DAV(3) =	9.390	DAV(4) =	25.012
DAV(5) =	88.866	DAV(6) =	677.817
VMAX =	587.954		

A description of the line splitting Groups A and D must be determined. The form of the discriminant function has been resolved as

$$f(Z) = \frac{B \cdot (Z - DAV)}{VMAX} + 0.635$$

If $f(Z) > 0$, then the arbitrary point Z is a member of Group A, B, or C. On the other hand, if $f(Z) < 0$, then Z is an entity of Group D. The equation of the discriminant function describing the upper line is given as:

$$f(U) = -\frac{0.6(A \cdot (U - DAV))}{VMAX} + \frac{B \cdot (U - DAV)}{VMAX}$$

U is an observation from Group A or D if $f(U) > 0$. Likewise, U is a vector from Group B or C if the result of $f(U) < 0$.

CASE 7

The final analysis was conducted on a thirteen-dimensional pattern classification problem. Coronary patients from Letterman Hospital comprised the sixty-tuples making up the Y subset [1]. The other subset was composed of fifty Army personnel at Silas B. Hays Hospital who had undergone annual physical examinations. Those patients from Letterman had been diagnosed as having some type of coronary heart disease. The X subset consisted of thirteen features from a random sample of the Army personnel. There were no indications from the latter group to either classify or not as having heart disease. If successful separation is obtained and a discriminant function is computed, then it could be used to classify new patients.

A few of the components in each thirteen-tuple represented measurable entities such as weight and age. Other characteristics such as race and smoking behavior were assigned arbitrary values rather than numbers to signify a measurement. The analysis will show that these classifiers are not really appropriate to use. The actual measurements exhibit at least the characteristics of an interval scale. This is a minimum requirement in computing a meaningful average value of a parameter. Those components which had an arbitrary value assigned do not meet this criteria for an interval scale. In fact, they have the characteristics of an ordinal scale. Therefore, when the average values of the components, DAV,

are computed in the program, it is inappropriate to use these classifiers in the analysis.

The first run was conducted on the thirteen-dimensional sample set. The displays of the projections are given in Figures IV-7-1 through IV-7-6. The last figure had the best presentation with respect to partitioning the two groups. A line dividing the "X's" and "O's" was drawn in the figure. If a discriminant function had been specified for this line, thirty-six of the fifty X points and forty-nine of the sixty Y points would have been properly classified. This result means that approximately 25% of the total sample have been projected out of their respective regions. This last statement does not seem satisfactory for any prediction model or separation.

The values of the A and B vectors used by G. Lauzon of this data indicate a definite pattern. For those assigned values, the number in both vectors for each component are less than or equal to the absolute value of two. On the other hand, one of these two numbers is very much greater than 2 for every measured component. For this reason, the components of assigned values will be deleted in the second run. In addition, this algorithm is more appropriate for separation of data where the sample set is on an interval scale.

In the second run, an arbitrary observation Z will be denoted by

$$Z(H) = (Z(H,2),\ Z(H,3),\ Z(H,4),\ Z(H,12),\ Z(H,13))$$

where H = 1,...,M for an X and H = 1,...,N for a Y. Figures IV-7-7 through IV-7-12 show the printout of the five-dimensional plot. The figure with the best partitioning is IV-7-12. This time the discriminant function would result in a smaller probability of error. There are fifty-four Y projections in the left region and thirty-six X's to the right. Now, the result is less than 80% of the projections fell into their respective areas. The author feels that this second run has also led to an unsatisfactory separation of the two groups. The reason for this conclusion is the high density of points near the dividing line. In addition, the distance is rather insignificant for the different points of the two subsets across this dividing line. Therefore, the author feels that an attempt to specify a discriminant function for this data would be futile.

V. CONCLUSIONS AND RECOMMENDATIONS

It can be concluded that the improvements on the algorithm have resulted in achieving greater success than the previous application. First of all, the revised algorithm provided a partitioning of the generated sample sets in the first four cases. The new separation algorithm was also applied to three "real world" applications. The first investigation obtained a satisfactory partitioning of the three Iris species. The three groups were divided by specifying two discriminant functions. The second sample set involved a more complex analysis problem. The observations consisted of an eight-dimensional array. In addition, it had been hypothesized that there existed four species in the data. The display for this analysis indicated the presence of only three species. The same conclusion was also reached by Wright and Switzer [4] in their approach to this discriminant analysis problem. However, their computations in reaching this result were much more involved than the algorithm in this paper. The investigation by G. Lauzon showed the presence of only two such separate groupings. From this, the improved algorithm using the orthonormal vectors appears to achieve a more definite partitioning. Finally, the employment of this method on the thirteen-dimensional medical data resulted in unsatisfactory separation. The rejection of a partition may be due to the use of inappropriate parameters in analyzing coronary disease.

All separations are really approximations. The final classifications must be subjected to a variety of sample sets for a confirmation of the discriminant function. Even after this has been accomplished, the presence of outliers will result in some finite value for the probability of error. The description by the discriminant function is only the optimal division of the groupings.

The use of the values from the table in Appendix IV should be mentioned. All investigations in this paper were conducted choosing the initial orthonormal vectors as listed in the table (Appendix IV). However, if different values had been substituted in the components, the final separation may be better. In the future, the operator should utilize a variety of unit orthonormal vectors in the attempt of achieving a successful partition.

This algorithm has provided a relatively easy and simple approach to the classification problem. It is possible that the merit of this method could be incorporated into a portable unit. A computer module designed for the appropriate computations would provide a means for obtaining the required projections. A television cathode ray tube could act as the display system. An interface system could be developed to handle the input/output operations. The outstanding feature using a color t.v. set is the ability to display a variety of colors. The various subsets of the sample set could be linked to a respective hue. The combination of these two systems would make up an ideal graphics/computer system in which the algorithm would be a major feature.

The development of this algorithm could be applied in an unlimited variety of fields. The analysis of medical and taxonomy samples has already been mentioned. The method is pertinent to any field in which there is mutivariate data to be separated. For instance, measurements of meteorological parameters may be subjected to this method of analysis. The motivation of the investigation could be to determine the existence of conditions pertaining to the development of a tropical storm and those present during the infancy of a hurricane. This algorithm may be able to distinguish the two different groupings. If this can be accomplished, then the model can be used as a basis in predicting future storm activity.

The final comments are directed to future changes in the computer routine. First, the A and B vectors should be punched on computer cards and read in with the data deck. This would eliminate the possibility of typing in an incorrect component. Secondly, a video presentation of the number of the selected point would be most helpful to the operator. In addition, the printout of the plot should indicate the selected point by a different set of letters than the present "X-O" scheme. These two modifications would make it possible to repeat the same iterations on subsequent runs. Lastly, it would be beneficial to the operator if a series of sample sets could be read. This could be arranged by adding the use of another function switch. The operator would now be able to conduct a series of different investigations without leaving the graphics console.

```
C
C                           APPENDIX I
C
C                    CØMPUTER PRØGRAM LISTING
C
C
C     SETTING UP ØF ARRAYS, INITIALIZING
C     CØNSTANTS,AND INPUT DATA VALUES
C
      INTEGER ALPH(8),PIC(61,37)
      DIMENSIØN LXT(20)
      DIMENSIØN C(13),II(3),ARR(20)
      DIMENSIØN ITDIR(4),IGDIR(3),TØT(13),DAV(13)
      DIMENSIØN IMAG(481),IFRAM(401),A(13),B(13),JMAG(481),JFRAM(401)
      DIMENSIØN X(080,13),Y(080,13),XX1(080),XX2(080),YY1(080),YY2(080)
      DATA (ALPH(I),I=1,8)/1HO,1H8,1HX,1HE,1HJ,1H ,1H*,1HØ/
      NAMELIST A,B,M,N,L,IDEV,NPFLG,IER,ICC
      IDEV=1
      ØUTPUT(101) 'TYPE IDEV=2* AND C/R IF USING AGT 2'
      INPUT(101)
      CALL DTINIT(IDEV,ITDIR,4,IER)
      IF(IER.NE.0) ØUTPUT(101) IER,'DTINIT'
      TT=0.01
      DØ 2 I=1,3
      READ (5,1) LXT
    2 WRITE (6,1) LXT
      READ (5,1) ARR
      READ(5,ARR) ((Y(I,K),K=1,L),I=1,N)
      IF(M.EQ.0) GØ TØ 5
      READ(5,ARR) ((X(J,K),K=1,L),J=1,M)
C
C     CENTRALIZING DATA PØINTS
C
    5 DØ 9 K=1,L
    9 TØT(K)=0.0
```

```
 12 DØ 14 I=1,N
    DØ 14 K=1,L,2
    IKX=K+1
 14 WRITE (6,15) I,K,Y(I,K),I,IKX,Y(I,IKX)
    II(3)=60606060B
    DØ 13  I=1,N
    DØ 13  K=1,L
 13 TØT(K)=TØT(K)+Y(I,K)
    DØ 17 J=1,M
    DØ 17   K=1,L
 17 TØT(K)=TØT(K)+X(J,K)
    DEN=FLØAT(M+N)
    DØ 21   K=1,L
    DAV(K)=TØT(K)/DEN
 21 WRITE(6,22) K,DAV(K)
    DØ 29 I=1,N
    DØ 29 K=1,L
 29 Y(I,K)=Y(I,K)-DAV(K)
    DØ 33 J=1,M
    DØ 33 K=1,L
 33 X(J,K)=X(J,K)-DAV(K)
    WRITE(6,39)
    DØ 41 I=1,N
    DØ 41 K=1,L
    IF (NPFLG.EQ.0) GØ TØ 46
    IKX=K+1
 41 WRITE (6,42) I,K,Y(I,K),I,IKX,Y(I,IKX)
 46 DØ 49 J=1,M
    DØ 49 K=1,L,2
    IF (NPFLG.EQ.0) GØ TØ 51
    IKX=K+1
 49 WRITE (6,50) J,K,X(J,K),J,IKX,X(J,IKX)
 51 WRITE(6,52)
C
C     INPUT CØMPØNENT VALUES FØR THE A AND B VECTØRS
```

```
C
   57 CALL GINPUT(IDEV,ITDIR,1)
      ITEXT=11
      ENCΘDE (4,58,II)ITEXT
      CALL TEXTΘ (IDEV,II,3,2,1,1,3,IER)
      IF (IER.NE.0) ΘUTPUT(101) IER,'FNS 11'
      DΘ 61  K=1,L
   61 WRITE(6,62) K,A(K),K,B(K)
      KST=0
      KRT=0
   69 CALL DGINIT (IDEV,IGDIR,3,IER)
      IF(IER.NE.0) ΘUTPUT (101) IER,'DGINIT'
C
C     CALCULATING ΘF THE DATA PΘINTS PRΘJECTIΘNS
C
      VMAX=0.0
      SUMY1=0.0
      SUMY2=0.0
      DΘ 73 I=1,N
      YY1(I)=0.0
      YY2(I)=0.0
      DΘ 77 K=1,L
      YY1(I)=YY1(I)+A(K)*Y(I,K)
   77 YY2(I)=YY2(I)+B(K)*Y(I,K)
      SUMY1=SUMY1+YY1(I)
   73 SUMY2=SUMY2+YY2(I)
      SUMX1=0.0
      SUMX2=0.0
      DΘ 85 J=1,M
      XX1(J)=0.0
      XX2(J)=0.0
      DΘ 81 K=1,L
      XX1(J)=XX1(J)+A(K)*X(J,K)
   81 XX2(J)=XX2(J)+B(K)*X(J,K)
      SUMX1=SUMX1+XX1(J)
```

```
   85 SUMX2=SUMX2+XX2(J)
      AV1=(SUMY1+SUMX1)/DEN
      AV2=(SUMY2+SUMX2)/DEN
      IF (NPFLG.EQ.0) GØ TØ 87
      WRITE(6,86)  AV1,AV2
C
C     SCALING THE PRØJECTIØNS
C
   87 DØ 89 I=1,N
      YY1(I)=YY1(I)-AV1
      YY2(I)=YY2(I)-AV2
      GA=ABS(YY1(I))
      GB=ABS(YY2(I))
      IF(GA.GT.VMAX) VMAX=GA
   89 IF(GB.GT.VMAX) VMAX=GB
      DØ 93 J=1,M
      XX1(J)=XX1(J)-AV1
      XX2(J)=XX2(J)-AV2
      ZA=ABS(XX1(J))
      ZB=ABS(XX2(J))
      IF(ZA.GT.VMAX) VMAX=ZA
   93 IF(ZB.GT.VMAX) VMAX=ZB
      WRITE(6,95) VMAX
C
C     DRAWING THE Y PØINTS ØN THE AGT DISPLAY SCREEN
C
      KN=5*N
      KK=KN+1
      JL=0
      IFRAM(1)=IHEAD(0,10)
      DØ 101 I=2,KN,5
      JL=JL+1
      FA=YY1(JL)/VMAX
      FB=YY2(JL)/VMAX
      IF (NPFLG.EQ.0) GØ TØ 98
```

```
      WRITE(6,96)    JL,FA,JL,FB
   98 FC=FA+TT
      FD=FA-TT
      FE=FB+TT
      FF=FB-TT
      IFRAM(I)=IPACK(FC,FE,0)
      IFRAM(I+1)=IPACK(FC,FF,1)
      IFRAM(I+2)=IPACK(FD,FF,1)
      IFRAM(I+3)=IPACK(FD,FE,1)
  101 IFRAM(I+4)=IPACK(FC,FE,1)
      CALL GRAPH8 (IDEV,IFRAM,KK,1,IER)
      IF(IER.NE.0) ØUTPUT (101) IER,'GBLK1'
C
C     DRAWING THE X PØINTS ØN THE AGT DISPLAY SCREEN
C
      KM=6*M
      IF(KM.EQ.0) GØ TØ 117
      JJ=KM+1
      JN=0
      IMAG(1)=IHEAD(0,10)
      DØ 109 I=2,KM,6
      JN=JN+1
      TA=XX1(JN)/VMAX
      TB=XX2(JN)/VMAX
      IF (NPFLG.EQ.0) GØ TØ 102
      WRITE(6,103)   JN,TA,JN,TB
  102 TC=TA+TT
      TD=TA-TT
      TE=TB+TT
      TF=TB-TT
      IMAG(I)=IPACK(TA,TB,0)
      IMAG(I+1)=IPACK(TC,TE,1)
      IMAG(I+2)=IPACK(TC,TF,0)
      IMAG(I+3)=IPACK(TD,TE,1)
      IMAG(I+4)=IPACK(TD,TF,0)
```

```
  109 IMAG(I+5)=IPACK(TA,TB,1)
      CALL GRAPHØ (IDEV,IMAG,JJ,2,IER)
      IF(IER.NE.0) ØUTPUT(101) IER,'GBLK2'
C
C     CHECKING FØR THE DEPRESSED FUNCTIØN SWITCHES
C
  117 CALL FNS(IDEV,ISW,IER)
      IF(IER.NE.0) ØUTPUT(101) IER,'GBLK3'
      IF(ISW.GE.0) GØ TØ 117
      NPFLG=0
      IF (LLS(ISW,9).LT.0) NPFLG=1
      IF(LLS(ISW,11).LT.0) GØ TØ 57
      IF(LLS(ISW,3).LT.0) GØ TØ 129
      IF(LLS(ISW,13).LT.0) GØ TØ 149
      IF(LLS(ISW,14).LT.0) GØ TØ 297
      IF(LLS(ISW,15).LT.0) GØ TØ 349
      IF(KRT.EQ.1) GØ TØ 121
      IF(KST.EQ.1) GØ TØ 125
      IF(LLS(ISW,16).LT.0) GØ TØ 117
      IF(LLS(ISW,17).LT.0) GØ TØ 117
      IF(LLS(ISW,18).LT.0) GØ TØ 117
      GØ TØ 117
C
C     CHØØSING THE DIRECTIØN ØF MØVE WITH THE Y PØINT PICKED
C
  121 CALL FNS(IDEV,ISW,IER)
      IF(IER.NE.0) ØUTPUT(101) IER,'GBLK7'
      IF(ISW.GE.0) GØ TØ 117
      IF(LLS(ISW,5).LT.0) GØ TØ 169
      IF(LLS(ISW,6).LT.0) GØ TØ 181
      IF(LLS(ISW,7).LT.0) GØ TØ 193
      IF(LLS(ISW,8).LT.0) GØ TØ 204
      GØ TØ 117
C
C     CHØØSING THE DIRECTIØN ØF MØVE WITH THE X PØINT PICKED
```

```
C
  125 CALL FNS(IDEV,ISW,IER)
      IF(IER.NE.0) ØUTPUT(101) IER,'GBLK8'
      IF(ISW.GE.0) GØ TØ 117
      IF(LLS(ISW,5).LT.0) GØ TØ 217
      IF(LLS(ISW,6).LT.0) GØ TØ 229
      IF(LLS(ISW,7).LT.0) GØ TØ 241
      IF(LLS(ISW,8).LT.0) GØ TØ 253
      GØ TØ 117
C
C     PICKING A Y PØINT
C
  129 ITEXT=3
      ENCØDE (4,58,II)ITEXT
      CALL TEXTØ (IDEV,II,3,1,1,1,3,IER)
      IF (IER.NE.0) ØUTPUT(101) IER,'FNS 3'
      CALL GRAPHR(IDEV,IFRAM,KK,1,IER)
      IF(IER.NE.0) ØUTPUT(101) IER,'GBLK3'
  133 IF(MØD(IGDIR(1),8).EQ.0) GØ TØ 133
      CALL GRAPHI(IDEV,JFRAM,1,IER)
      IF(IER.NE.0) ØUTPUT(101) IER,'IGBLK'
      KEDS=0
      KST=0
      KRT=1
      DØ 137 I=1,N
      IY=5*I-3
      IZ=5*I+1
      DØ 141 J=IY,IZ
  141 IF(JFRAM(J).NE.IFRAM(J)) GØ TØ 145
  137 CØNTINUE
  145 JED=I
      WRITE (6,147) JED
      GØ TØ 117
C
C     PICKING A X PØINT
```

```
C
  149 ITEXT=13
      ENCØDE (4,58,II)ITEXT
      CALL TEXTØ (IDEV,II,3,1,1,1,3,IER)
      IF (IER.NE.0) ØUTPUT(101) IER,'FNS 13'
      CALL GRAPHR(IDEV,IMAG,JJ,2,IER)
      IF(IER.NE.0) ØUTPUT(101) IER,'GBLK4'
  153 IF(MØD(IGDIR(2),8).EQ.0) GØ TØ 153
      CALL GRAPHI(IDEV,JMAG,2,IER)
      IF(IER.NE.0) ØUTPUT(101) IER,'JGBLK'
      KEDS=0
      KST=1
      KRT=0
      DØ 157 J=1,M
      JY=6*J-4
      JZ=6*J+1
      DØ 161 I=JY,JZ
  161 IF(JMAG(I).NE.IMAG(I)) GØ TØ 165
  157 CØNTINUE
  165 LED=J
      WRITE (6,167) LED
      GØ TØ 117
C
C     ITERATING BY MØVING Y RIGHT
C
  169 ITEXT=5
      ENCØDE (4,58,II)ITEXT
      CALL TEXTØ (IDEV,II,3,2,1,1,3,IER)
      IF (IER.NE.0) ØUTPUT(101) IER,'FNS 5'
      IF(JED.EQ.0) GØ TØ 117
      WRITE(6,170)  JED
      DØ 171  K=1,L
  171 A(K)=A(K)+Y(JED,K)
      FF=0.0
      DØ 172  K=1,L
```

```
  172 FF=FF+A(K)**2
      DØ 173  K=1,L
  173 A(K)=A(K)/SQRT(FF)
      KS=0
  175 E=0.0
      EE=0.0
      DØ 176  K=1,L
      EE=EE+Y(JED,K)*A(K)
  176 E=E+Y(JED,K)*B(K)
      DØ 177  K=1,L
  177 C(K)=Y(JED,K)-EE*A(K)
      F=0.0
      G=0.0
      DØ 178  K=1,L
      F=F+C(K)**2
  178 G=G+Y(JED,K)**2
      G=G-EE**2
      IF ((G.GT.0.0).AND.(E.GT.0.0).ØR.(G.LT.0.0).AND.(E.LT.0.0)) E=-1.0
      IF ((G.LT.0.0).AND.(E.GT.0.0).ØR.(G.GT.0.0).AND.(E.LT.0.0)) E=1.0
      DØ 179  K=1,L
  179 B(JK)=E*C(K)/SQRT(F)
      IF (KS.EQ.1) GØ TØ 187
      KEDS=1
      GØ TØ 265
C
C     ITERATING BY MØVING Y LEFT
C
  181 ITEXT=6
      ENCØDE (4,58,II)ITEXT
      CALL TEXTØ (IDEV,II,3,2,1,1,3,IER)
      IF (IER.NE.0) ØUTPUT(101) IER,'FNS 6'
      IF(JED.EQ.0) GØ TØ 117
      WRITE(6,182)  JED
      DØ 183  K=1,L
  183 A(K)=A(K)-Y(JED,K)
```

```
      FF=0.0
      Dθ 184  K=1,L
  184 FF=FF+A(K)**2
      Dθ 185  K=1,L
  185 A(K)=A(K)/SQRT(FF)
      KS=1
      Gθ Tθ 175
  187 KEDS=2
      Gθ Tθ 265
C
C     ITERATING BY MθVING Y UP
C
  193 ITEXT=7
      ENCθDE (4,58,II)ITEXT
      CALL TEXTθ (IDEV,II,3,2,1,1,3,IER)
      IF (IER.NE.0) θUTPUT(101) IER,'FNS 7'
      IF(JED.EQ.0) Gθ Tθ 117
      WRITE(6,194)  JED
      Dθ 195  K=1,L
  195 B(K)=B(K)+Y(JED,K)
      FF=0.0
      Dθ 196  K=1,L
  196 FF=FF+B(K)**2
      Dθ 197  K=1,L
  197 B(K)=B(K)/SQRT(FF)
      KS=0
  198 E=0.0
      EE=0.0
      Dθ 199  K=1,L
      EE=EE+Y(JED,K)*B(K)
  199 E=E+Y(JED,K)*A(K)
      Dθ 200  K=1,L
  200 C(K)=Y(JED,K)-EE*B(K)
      F=0.0
      G=0.0
```

```
      DØ 201  K=1,L
      F=F+C(K)**2
  201 G=G+Y(JED,K)**2
      G=G-EE**2
      IF ((G.GT.0.0).AND.(E.GT.0.0).ØR.(G.LT.0.0).AND.(E.LT.0.0)) E=-1.0
      IF ((G.LT.0.0).AND.(E.GT.0.0).ØR.(G.GT.0.0).AND.(E.LT.0.0)) E=1.0
      DØ 202  K=1,L
  202 A(K)=E*C(K)/SQRT(F)
      IF (KS.EQ.1) GØ TØ 216
      KEDS=3
      GØ TØ 281
C
C     ITERATING BY MØVING Y DØWN
C
  204 ITEXT=8
      ENCØDE (4,58,II)ITEXT
      CALL TEXTØ (IDEV,II,3,2,1,1,3,IER)
      IF (IER.NE.0) ØUTPUT(101) IER,'FNS 8'
      IF(JED.EQ.0) GØ TØ 117
      WRITE(6,209)  JED
      DØ 213  K=1,L
  213 B(K)=B(K)-Y(JED,K)
      FF=0.0
      DØ 214  K=1,L
  214 FF=FF+B(K)**2
      DØ 215  K=1,L
  215 B(K)=B(K)/SQRT(FF)
      KS=1
      GØ TØ 198
  216 KEDS=4
      GØ TØ 281
C
C     ITERATING BY MØVING X RIGHT
C
  217 ITEXT=5
```

```
      ENCθDE (4,58,II)ITEXT
      CALL TEXTθ (IDEV,II,3,2,1,1,3,IER)
      IF (IER.NE.0) θUTPUT(101) IER,'FNS 5'
      IF(LED.EQ.0) Gθ Tθ 117
      WRITE(6,218)  LED
      Dθ 219  K=1,L
  219 A(K)=A(K)+X(LED,K)
      FF=0.0
      Dθ 220  K=1,L
  220 FF=FF+A(K)**2
      Dθ 221  K=1,L
  221 A(K)=A(K)/SQRT(FF)
      KS=0
  224 E=0.0
      EE=0.0
      Dθ 225  K=1,L
      EE=EE+X(LED,K)*A(K)
  225 E=E+X(LED,K)*B(K)
      Dθ 226  K=1,L
  226 C(K)=X(LED,K)-EE*A(K)
      F=0.0
      G=0.0
      Dθ 227  K=1,L
      F=F+C(K)**2
  227 G=G+X(LED,K)**2
      G=G-EE**2
      IF ((G.GT.0.0).AND.(E.GT.0.0).θR.(G.LT.0.0).AND.(E.LT.0.0)) E=-1.0
      IF ((G.LT.0.0).AND.(E.GT.0.0).θR.(G.GT.0.0).AND.(E.LT.0.0)) E=1.0
      Dθ 228  K=1,L
  228 B(K)=E*C(K)/SQRT(F)
      IF (KS.EQ.1) Gθ Tθ 239
      KEDS=5
      Gθ Tθ 265
C
C     ITERATING BY MθVING X LEFT
```

```
C
  229 ITEXT=6
      ENCØDE (4,58,II)ITEXT
      CALL TEXTØ (IDEV,II,3,2,1,1,3,IER)
      IF (IER.NE.0) ØUTPUT(101) IER,'FNS 6'
      IF(LED.EQ.0) GØ TØ 117
      WRITE(6,233)  LED
      DØ 234  K=1,L
  234 A(K)=A(K)-X(LED,K)
      FF=0.0
      DØ 235  K=1,L
  235 FF=FF+A(K)**2
      DØ 236  K=1,L
  236 A(K)=A(K)/SQRT(FF)
      KS=1
      GØ TØ 224
  239 KEDS=6
      GØ TØ 265
C
C     ITERATING BY MØVING X UP
C
  241 ITEXT=7
      ENCØDE (4,58,II)ITEXT
      CALL TEXTØ (IDEV,II,3,2,1,1,3,IER)
      IF (IER.NE.0) ØUTPUT(101) IER,'FNS 7'
      IF(LED.EQ.0) GØ TØ 117
      WRITE(6,242)  LED
      DØ 243  K=1,L
  243 B(K)=B(K)+X(LED,K)
      FF=0.0
      DØ 244  K=1,L
  244 FF=FF+B(K)**2
      DØ 245  K=1,L
  245 B(K)=B(K)/SQRT(FF)
      KS=0
```

```
  247 E=0.0
      EE=0.0
      DØ 248  K=1,L
      EE=EE+X(LED,K)*B(K)
  248 E=E+X(LED,K)*A(K)
      DØ 249  K=1,L
  249 C(K)=X(LED,K)-EE*B(K)
      F=0.0
      G=0.0
      DØ 250  K=1,L
      F=F+C(K)**2
  250 G=G+X(LED,K)**2
      G=G-EE**2
      IF ((G.GT.0.0).AND.(E.GT.0.0).ØR.(G.LT.0.0).AND.(E.LT.0.0)) E=-1.0
      IF ((G.LT.0.0).AND.(E.GT.0.0).ØR.(G.GT.0.0).AND.(E.LT.0.0)) E=1.0
      DØ 251  K=1,L
  251 A(K)=E*C(K)/SQRT(F)
      IF (KS.EQ.1) GØ TØ 264
      KEDS=7
      GØ TØ 281
C
C     ITERATING BY MØVING X DØWN
C
  253 ITEXT=8
      ENCØDE (4,58,II)ITEXT
      CALL TEXTØ (IDEV,II,3,2,1,1,3,IER)
      IF (IER.NE.0) ØUTPUT(101) IER,'FNS 8'
      IF(LED.EQ.0) GØ TØ 117
      WRITE(6,257)  LED
      DØ 261  K=1,L
  261 B(K)=B(K)-X(LED,K)
      FF=0.0
      DØ 262  K=1,L
  262 FF=FF+B(K)**2
      DØ 263  K=1,L
```

```
  263 B(K)=B(K)/SQRT(FF)
      KS=1
      GØ TØ 247
  264 KEDS=8
      GØ TØ 281
  265 IF (NPFLG.EQ.0) GØ TØ 269
      DØ 273 K=1,L
  273 WRITE (6,274)  K,A(K),K,B(K)
  269 GØ TØ 69
  281 IF (NPFLG.EQ.0) GØ TØ 285
      DØ 289 K=1,L
  289 WRITE (6,290)  K,A(K),K,B(K)
  285 GØ TØ 69
C
C     PLØTTING ØUTPUT ØN THE LINE PRINTER
C
  297 ITEXT=14
      ENCØDE (4,58,II)ITEXT
      CALL TEXTØ (IDEV,II,3,2,1,1,3,IER)
      IF (IER.NE.0) ØUTPUT(101) IER,'FNS 14'
      DØ 301 I=2,60
      DØ 305 J=2,36
  305 PIC(I,J)=ALPH(6)
  301 CØNTINUE
      DØ 306 J=1,18
      PIC(1,J)=ALPH(7)
  306 PIC(61,J)=ALPH(7)
      PIC(1,19)=ALPH(8)
      PIC(61,19)=ALPH(8)
      DØ 309 J=20,37
      PIC(1,J)=ALPH(7)
  309 PIC(61,J)=ALPH(7)
      DØ 313 I=1,30
      PIC(I,1)=ALPH(7)
  313 PIC(I,37)=ALPH(7)
```

```
      PIC(31,1)=ALPH(8)
      PIC(31,37)=ALPH(8)
      DØ 317 I=32,61
      PIC(I,1)=ALPH(7)
  317 PIC(I,37)=ALPH(7)
      DØ 321 I=1,N
       JH=(YY2(I)/VMAX)*17.0+19.0
       KH=(YY1(I)/VMAX)*29.0+31.0
      IF(KH.GT.60) KH=60
      IF(JH.GT.36) JH=36
      JH=38-JH
      IF(PIC(KH,JH).NE.ALPH(6)) PIC(KH,JH)=ALPH(2)
  321 IF(PIC(KH,JH).EQ.ALPH(6)) PIC(KH,JH)=ALPH(1)
      DØ 325  J=1,M
       LC=(XX1(J)/VMAX)*29.0+31.0
       LD=(XX2(J)/VMAX)*17.0+19.0
      IF(LC.GT.60) LC=60
      IF(LD.GT.36) LD=36
      LD=38-LD
      IF(PIC(LC,LD).EQ.ALPH(6)) GØ TØ 329
      IF(PIC(LC,LD).EQ.ALPH(1)) PIC(LC,LD)=ALPH(5)
      IF(PIC(LC,LD).EQ.ALPH(2)) PIC(LC,LD)=ALPH(5)
      IF(PIC(LC,LD).EQ.ALPH(3)) PIC(LC,LD)=ALPH(4)
      GØ TØ 325
  329 PIC(LC,LD)=ALPH(3)
  325 CØNTINUE
      WRITE(6,333)
      DØ 337 J=1,37
  337 WRITE(6,338) (PIC(I,J),I=1,61)
      WRITE(6,333)
      GØ TØ 117
C
C     REGRESSING TØ THE PREVIØUS ITERATIØN
C
  349 ITEXT=15
```

```
      ENCØDE (4,58,II)ITEXT
      CALL TEXTØ (IDEV,II,3,2,1,1,3,IER)
      IF (IER.NE.0) ØUTPUT(101) IER,'FNS 15'
      WRITE(6,352)
      GØ TØ (181,169,204,193,229,217,253,241),KEDS
    1 FØRMAT (20A4)
   15 FØRMAT (5X,'Y(',I2,',',I2,')=',F10.3,15X,'Y(',I2,',',I2,') =',F10.
     13)
   22 FØRMAT(5X,'DATA CØMP. AVE NØ.',I2,'=',F10.3)
   39 FØRMAT(5X,'CENTRALIZED DATA',/)
   42 FØRMAT (5X,'Y(',I2,',',I2,')=',F10.3,15X,'Y(',I2,',',I2,') =',F10.
     13)
   50 FØRMAT (5X,'X(',I2,',',I2,')=',F10.3,15X,'X(',I2,',',I2,') =',F10.
     13)
   52 FØRMAT(5X,///)
   58 FØRMAT (I4)
   62 FØRMAT(5X,'A(',I2,')=',F6.3,' ;B(',I2,')=',F6.3)
   86 FØRMAT(5X,'SCAL. AVE 1=',F10.6,'  SCAL. AVE 2=',F10.6)
   95 FØRMAT(5X,'VMAX=',F10.6)
   96 FØRMAT(5X,'YY1(',I2,')=',F10.4,10X,'YY2(',I2,')=',F10.4)
  103 FØRMAT(5X,'XX1(',I2,')=',F10.4,10X,'XX2(',I2,')=',F10.4)
  147 FØRMAT('PICKING PØINT Y(',I2,')')
  167 FØRMAT('PICKING PØINT X(',I2,')')
  170 FØRMAT(5X,'MØVING RIGHT USING Y(',I2,')')
  182 FØRMAT(5X,'MØVING LEFT USING Y(',I2,')')
  194 FØRMAT(5X,'MØVING UP USING Y(',I2,')')
  209 FØRMAT(5X,'MØVING DØWN USING Y(',I2,')')
  218 FØRMAT(5X,'MØVING RIGHT USING X(',I2,')')
  233 FØRMAT(5X,'MØVING LEFT USING X(',I2,')')
  242 FØRMAT(5X,'MØVING UP USING X(',I2,')')
  257 FØRMAT(5X,'MØVING DØWN USING X(',I2,')')
  274 FØRMAT(5X,'NEW VALUE ØF A(',I2,')=',F10.4,10X,'B(',I2,')=',F10.4)
  290 FØRMAT(5X,'NEW VALUE ØF A(',I2,')=',F10.4,10X,'B(',I2,')=',F10.4)
  333 FØRMAT('1')
  338 FØRMAT(20X,61A1)
```

```
352 FØRMAT(5X,'REGRESSING TØ LAST STEP')
    STØP
    END
```

APPENDIX II. PROGRAM OPERATING INSTRUCTIONS

There exists two methods by which this program can be executed on the XDS 9300. The first and fastest method is utilizing a magnetic tape. The memory dump of the compilation from the digital computer has been recorded onto a magnetic tape. The elapsed time until execution commences would be approximately five minutes using this tape. The other method is simply reading the Fortran program from the computer cards. It would take roughly thirty minutes until the program is ready to execute using the cards. This time factor is very crucial if more than one set of data cards must be read.

In executing the program from the magnetic tape, the operator may consult the publication titled "Electrical Engineering Computer Laboratory" by R. D. Delaura [2]. A brief guide in using the magnetic tape mechanism applicable to the object language filed on this tape follows:

(1) Load the magnetic tape

(2) Unit select to position "1"

(3) Automatic/manual switch to "MANUAL"

(4) Depress forward drive (tape will stop at the LOAD POINT)

(5) Automatic/manual switch to "AUTOMATIC"

The magnetic tape is now ready for instructions from the XDS 9300. The operator must incorporate five control cards into the data deck. These control cards are instructions

to the various systems used in the execution of this program. The deck of cards will be as follows:

```
(1)  BØØT
(2)  JØB
(3)  AGT
(4)  ASSIGN X1 = MT1A
(5)  RERUN
(6)  (DATA DECK)
```

The deck is now ready to load into the card reader and execution can commence.

The following instructions will be applicable to the use of computer cards. The list of control cards, that must precede the source program printed in Appendix I, are given below:

```
(1)  BØØT
(2)  $ DATA
(3)  $PATCH
(4)  007067 001 06711
(5)  $END
(6)  JØB
(7)  AGT
(8)  FØRTRAN LS, GØ
```

The data deck is placed at the end of the Fortran program. The deck of cards is now ready to load and execution can commence.

In preparing the computer program using either method, four cards must be punched and inserted into the data cards.

The first three cards can be used for the purpose of verbally describing the data to be investigated. This information can be punched using any part or all of the eighty column card. The operator must insure in insert these three cards even though one or more cards may be blank. The fourth card is the instruction by which the computer reads the data from the punched cards. For illustrative purposes, The statement might read, " (4F7,2,3X,3F9.2)." This indicates that a total of seven parameters will be read from each card. The first parenthesis must be punched in column seven of the card. The actual data cards are inserted behind these four cards. The four cards plus the cards of data will be considered the data deck.

A guide for operating the XDS 9300 computer and the AGT console using either a magnetic tape or cards will be given. First though, assume that there are M number of X points and N number of Y points. The value of M and N must be less than or equal to eighty. In addition, all observations in both sets constitute an L-dimensional vector. The number of L must be less than or equal to thirteen. Now, the instructions will be presented.

(1) Use the "POWER ON" procedure for the XDS 9300 and the AGT graphic systems.

(2) Executing the program

(a) At the XDS 9300 control panel, depress "SENSE 2" if using cards. Do not depress if using the tape. Then depress "IDLE", "CLEAR" and "CLEAR

FLAGS" simultaneously, "RESET", "RUN", and "CARDS." The card reader will commence reading the punched cards. The line printer and teletypewriter (TTY) will simultaneously print "JOB." The computer will now start compiling the program.

(b) At the time execution of the program starts, the TTY will print the message: "TYPE IDEV = 2* AND A CARRIAGE RETURN IF USING AGT2." The input light will illuminate to alert the operator. The operator must type the number of X's and Y's plus the dimension of the two subsets of data. In addition, the operator has the option of printing all WRITE statements prior to statement #57 in the computer program. This is done by typing "NPFLG = 1" if a printout is desired; NPFLG = 0" for no printout. As an example, suppose there are 15 X's and 20 Y's in 10 dimensions. If the operator is using AGT 2 and requests no printout, type "M = 15, N = 20, L = 10, NPFLG = 0, IDEV = 2, *" and a carriage return at the computer TTY. If using AGT 1, simply delete "IDEV = 2." The card reader will begin the process of reading the card.

(c) The word "NAMELIST" and a blinking cursor will be displayed on the bottom left corner of the AGT screen. This will occur when computer reaches

statement #57 in the program. The operator must type in the components of the A and B vectors. The values from Appendix IV may be used or the operator may use any two orthonormal unit vectors. Thirteen numbers for each vector must be inputted. If the sample is 10-dimensional, zeroes will be inserted for the last three components. A representation of the above example would look like: "A = .31628, .31628,..., .31628, 0,0,0" and a carriage return, "B = .31628,..., -.31628,0,0,0" and a carriage return, "*" and a carriage return. The XDS 9300 will compute the projections and relay the data to the AGT for display on the screen. The "X's" and "O's" denote the original "X's and Y's, respectively.

(3) Interations and output

(a) To change the orientation of the plane, the operator chooses either the "PICK X" or "PICK Y" function switches. The location of these buttons on the overlay are shown in Figure A-2-1. A video message, "TEXT EDIT REQUEST," will appear at the bottom of the screen. The operator depresses the button on the lightpen and places the light over the desired symbol (point) to be moved. Upon releasing the button, a portion of the letter will disappear to signify the selected point. The user pushes

the two switches "COMPLETE CHOICE" and "END EDIT". The video message will disappear. The iteration will commence after one of the movement buttons (function switches) is depressed: "RIGHT", "LEFT", "UP", "DOWN." The 9300 recomputes the projections from the inner product with the orthonormal vectors and displays the new results on the screen.

(b) A printout of the new projections can be obtained by simultaneously depressing the "PRINT" and one of the movement function switches. The movement switch must be released first. A copy of the new projections on the plane will be printed at the line printer. If the plot of the computer projections is desired, then depress the "PLOT" function switch.

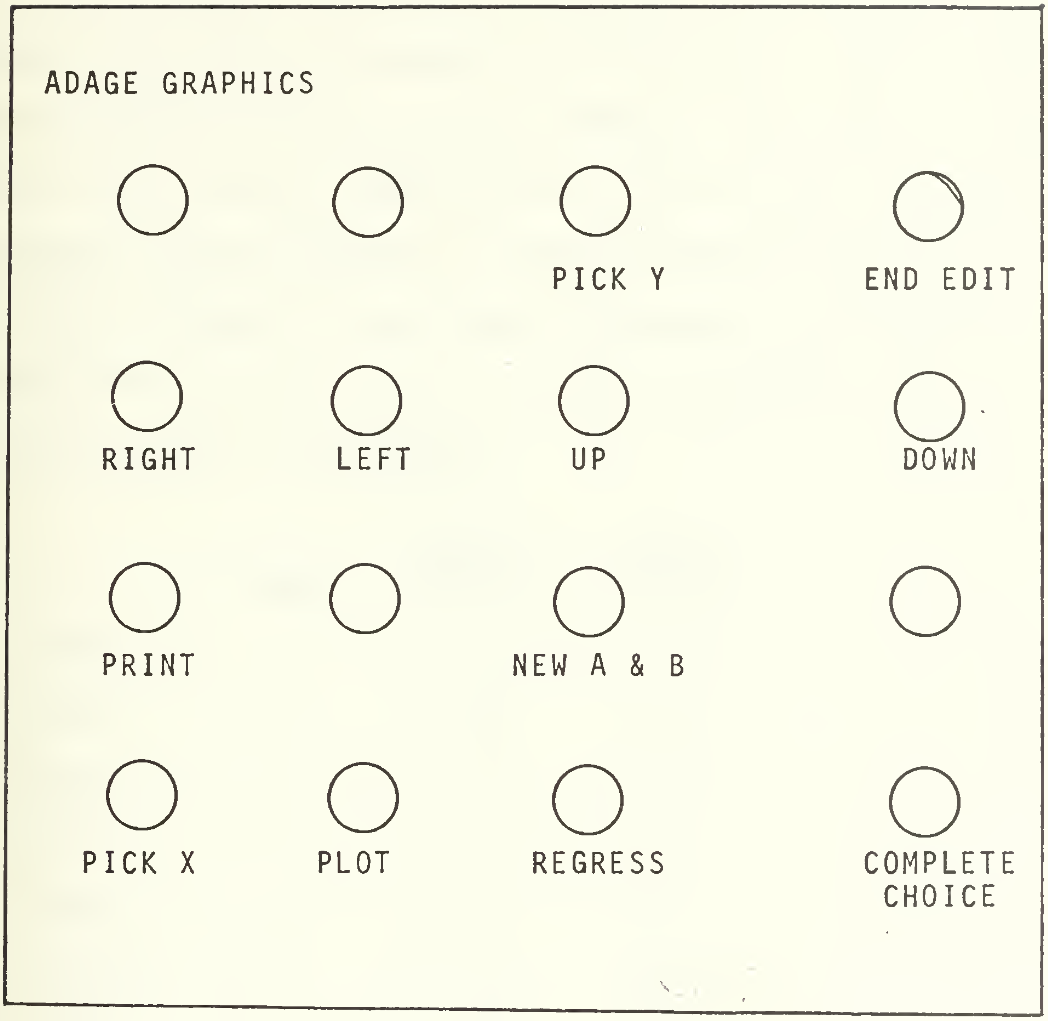

Figure A-2-1. FUNCTION SWITCH OVERLAY

APPENDIX III. DATA LISTING

This appendix presents a listing of the data sets used in the seven cases discussed in Section IV. The data has been arranged so that a row of numbers represents a single vector. For example, the data set for Case one is presented in three columns denoting a three-dimensional entity. The thirteen columns in Case Seven represents a thirteen-dimensional problem. As an aid for reference purposes, the following index is provided:

	PAGE	NUMBER OF Y'S	NUMBER OF X'S
CASE 1		21	0
CASE 2		20	20
CASE 3		70	70
CASE 4		40	20
CASE 5		50	25
CASE 6		38	44
CASE 7		60	50

CASE 1 DATA

I	Y(I,1)	Y(I,2)	Y(I,3)
1	1.0000	1.0000	1.0000
2	1.0000	1.0000	2.0000
3	2.5000	1.0000	1.0000
4	2.5000	1.0000	2.0000
5	4.0000	1.0000	1.0000
6	4.0000	1.0000	2.0000
7	1.0000	3.0000	1.0000
8	1.0000	3.0000	2.0000
9	2.5000	3.0000	1.0000
10	2.5000	3.0000	2.0000
11	4.0000	3.0000	1.0000
12	4.0000	3.0000	2.0000
13	1.0000	2.0000	1.0000
14	1.0000	2.0000	2.0000
15	4.0000	2.0000	1.0000
16	4.0000	2.0000	2.0000
17	2.5000	2.0000	3.3229
18	1.7500	1.5000	2.6614
19	1.7500	2.5000	2.6614
20	3.2500	1.5000	2.6614
21	3.2500	2.5000	2.6614

CASE 2 DATA

I	Y(I,1)	Y(I,2)	Y(I,3)
1	3.0000	2.0000	5.0000
2	4.0000	2.0000	5.0000
3	5.0000	2.0000	5.0000
4	3.0000	4.0000	5.0000
5	4.0000	4.0000	5.0000
6	5.0000	4.0000	5.0000
7	5.0000	3.0000	5.0000
8	3.0000	3.0000	5.0000
9	3.0000	2.0000	7.0000
10	4.0000	2.0000	7.0000
11	5.0000	2.0000	7.0000
12	3.0000	3.0000	7.0000
13	3.0000	4.0000	7.0000
14	4.0000	4.0000	7.0000
15	5.0000	4.0000	7.0000
16	5.0000	3.0000	7.0000
17	5.0000	2.0000	6.0000
18	5.0000	4.0000	6.0000
19	3.0000	4.0000	6.0000
20	3.0000	2.0000	6.0000

CASE 2 (CØNT)

J	X(J,1)	X(J,2)	X(J,3)
1	7.0000	6.0000	9.000
2	8.0000	6.0000	9.000
3	9.0000	6.0000	9.000
4	7.0000	8.0000	9.000
5	8.0000	8.0000	9.000
6	9.0000	8.0000	9.000
7	9.0000	7.0000	9.000
8	7.0000	7.0000	9.000
9	7.0000	6.0000	11.000
10	8.0000	6.0000	11.000
11	9.0000	6.0000	11.000
12	7.0000	7.0000	11.000
13	7.0000	8.0000	11.000
14	8.0000	8.0000	11.000
15	9.0000	8.0000	11.000
16	9.0000	7.0000	11.000
17	9.0000	6.0000	10.000
18	9.0000	8.0000	10.000
19	7.0000	8.0000	10.000
20	7.0000	6.0000	10.000

CASE 3 DATA

I	Y(I,1)	Y(I,2)	Y(I,3)	Y(I,4)	Y(I,5)	Y(I,6)
1	1.180	1.718	1.100	1.892	1.617	1.703
2	1.597	1.621	1.526	1.358	1.552	1.746
3	1.197	1.597	1.540	1.590	1.733	1.484
4	1.506	1.561	1.035	1.176	1.892	1.899
5	1.588	1.328	1.639	1.625	1.754	1.684
6	1.018	1.094	1.289	1.646	1.284	1.520
7	1.517	1.627	1.317	1.812	1.197	1.137
8	1.701	1.452	1.342	1.452	1.870	1.917
9	1.124	1.239	1.711	1.215	1.546	1.209
10	1.883	1.120	2.000	1.072	1.344	1.397
11	1.649	1.420	1.998	1.861	1.498	1.810
12	1.032	1.365	1.913	1.283	1.460	1.825
13	1.929	1.224	1.880	1.747	1.405	1.080
14	1.064	1.996	1.541	1.837	1.046	1.039
15	1.069	1.291	1.979	1.638	1.170	1.195
16	1.092	1.226	1.174	1.806	1.244	1.802
17	1.524	1.337	1.919	1.166	1.532	1.948
18	1.176	1.690	1.487	1.687	1.422	1.413
19	1.585	1.648	1.962	1.225	1.378	1.627
20	1.632	1.175	1.615	1.571	1.724	1.216
21	1.171	1.659	1.643	1.252	1.842	1.958
22	1.684	1.932	1.437	1.162	1.378	1.827

```
            CASE  3  (CØNT)

I        Y(I,1)     Y(I,2)     Y(I,3)     Y(I,4)     Y(I,5)     Y(I,6)


23       1.208      1.206      1.544      1.267      1.497      1.653

24       1.968      1.342      1.531      1.487      1.153      1.074

25       1.490      1.133      1.575      1.929      1.084      1.715

26       1.692      1.328      1.899      1.830      1.106      1.222

27       1.530      1.909      1.522      1.227      1.805      1.239

28       1.003      1.158      1.080      1.096      1.783      1.462

29       1.344      1.059      1.792      1.204      1.515      1.290

30       1.649      1.484      1.656      1.244      1.874      1.487

31       1.461      1.181      1.636      1.397      1.924      1.152

32       1.701      1.317      1.784      1.409      1.747      1.171

33       1.745      1.655      1.551      1.389      1.368      1.882

34       1.386      1.039      1.643      1.236      1.158      1.472

35       1.757      1.371      1.398      1.550      1.861      1.213

36       1.122      1.396      1.045      1.647      1.294      1.450

37       1.289      1.639      1.797      1.641      1.825      1.483

38       1.848      1.171      1.978      1.774      1.545      1.693

39       1.519      1.047      1.679      1.418      1.915      1.388

40       1.312      1.741      1.703      1.528      1.054      1.775

41       1.777      1.019      1.415      1.232      1.442      1.839

42       1.504      1.072      1.080      1.236      1.507      1.236

43       1.489      1.230      1.244      1.393      1.287      1.450

44       1.599      1.176      1.485      1.616      1.422      1.633
```

```
          CASE  3  (CØNT)

I       Y(I,1)     Y(I,2)     Y(I,3)     Y(I,4)     Y(I,5)     Y(I,6)


45      1.997      1.119      1.400      1.364      1.144      1.194

46      1.239      1.355      1.864      1.233      1.052      1.657

47      1.842      1.195      1.941      1.176      1.052      1.950

48      1.724      1.840      1.602      1.833      1.907      1.756

49      1.007      1.528      1.540      1.648      1.532      1.372

50      1.952      1.902      1.775      1.218      1.930      1.344

51      1.668      1.392      1.636      1.575      1.292      1.937

52      1.395      1.263      1.838      1.016      1.905      1.538

53      1.492      1.989      1.514      1.500      1.607      1.803

54      1.629      1.333      1.027      1.040      1.741      1.282

55      1.977      1.646      1.145      1.948      1.372      1.585

56      1.474      1.988      1.088      1.578      1.882      1.557

57      1.904      1.334      1.104      1.974      1.088      1.021

58      1.184      1.770      1.259      1.116      1.551      1.654

59      1.029      1.262      1.112      1.501      1.344      1.350

60      1.582      1.701      1.598      1.656      1.025      1.647

61      1.592      1.496      1.963      1.020      1.805      1.703

62      1.296      1.101      1.369      1.092      1.789      1.297

63      1.059      1.146      1.531      1.058      1.992      1.649

64      1.165      1.174      1.485      1.724      1.140      1.005

65      1.430      1.907      1.089      1.360      1.180      1.028

66      1.746      1.939      1.344      1.296      1.626      1.260
```

```
                CASE  3  (CØNT)

I        Y(I,1)     Y(I,2)     Y(I,3)     Y(I,4)     Y(I,5)     Y(I,6)


67       1.203      1.601      1.525      1.610      1.575      1.151

68       1.431      1.050      1.242      1.270      1.146      1.874

69       1.286      1.170      1.939      1.011      1.486      1.768

70       1.089      1.497      1.623      1.972      1.532      1.876
```

CASE 3 (CONT)

J	X(J,1)	X(J,2)	X(J,3)	X(J,4)	X(J,5)	X(J,6)
1	.528	.236	.840	.990	.761	.730
2	.063	.843	.799	.273	.804	.882
3	.251	.784	.924	.409	.471	.011
4	.466	.538	.499	.784	.751	.413
5	.293	.315	.990	.536	.259	.084
6	.023	.178	.375	.947	.090	.274
7	.239	.141	.505	.949	.500	.762
8	.402	.277	.659	.664	.139	.676
9	.335	.318	.463	.551	.079	.469
10	.573	.189	.935	.751	.155	.920
11	.971	.905	.657	.175	.275	.628
12	.949	.428	.674	.400	.400	.171
13	.967	.284	.305	.450	.688	.055
14	.928	.429	.543	.477	.709	.642
15	.112	.149	.884	.582	.252	.825
16	.694	.080	.420	.715	.068	.300
17	.171	.265	.371	.528	.462	.282
18	.543	.909	.612	.064	.572	.150
19	.233	.987	.346	.305	.748	.877
20	.439	.327	.231	.763	.198	.816
21	.182	.594	.759	.751	.420	.027
22	.334	.007	.058	.455	.079	.571

CASE 3 (CØNT)

J	X(J,1)	X(J,2)	X(J,3)	X(J,4)	X(J,5)	X(J,6)
23	.366	.942	.747	.010	.371	.237
24	.067	.288	.326	.129	.143	.353
25	.304	.483	.219	.195	.713	.821
26	1.000	.257	.693	.116	.194	.278
27	.724	.505	.323	.180	.806	.718
28	.702	.384	.307	.345	.351	.636
29	.506	.967	.298	.029	.102	.656
30	.766	.787	.584	.131	.092	.119
31	.038	.807	.826	.044	.988	.259
32	.643	.534	.796	.098	.963	.982
33	.296	.412	.109	.692	.836	.105
34	.337	.672	.070	.882	.448	.305
35	.398	.659	.515	.768	.378	.627
36	.476	.091	.688	.552	.897	.989
37	.477	.892	.514	.633	.048	.520
38	.166	.004	.062	.839	.412	.591
39	.596	.112	.665	.498	.376	.041
40	.366	.208	.867	.493	.592	.738
41	.060	.130	.542	.230	.975	.796
42	.433	.373	.658	.486	.854	.466
43	.955	.541	.412	.861	.513	.525
44	.112	.364	.799	.807	.380	.097

CASE 3 (CØNT)

J	X(J,1)	X(J,2)	X(J,3)	X(J,4)	X(J,5)	X(J,6)
45	.325	.321	.120	.010	.338	.971
46	.134	.692	.961	.127	.770	.188
47	.767	.734	.132	.436	.145	.476
48	.234	.679	.315	.683	.080	.980
49	.318	.306	.583	.455	.137	.658
50	.890	.423	.538	.126	.138	.626
51	.621	.712	.449	.148	.108	.278
52	.421	.105	.245	.284	.934	.728
53	.718	.146	.622	.702	.523	.575
54	.826	.635	.266	.808	.058	.599
55	.940	.189	.933	.178	.540	.429
56	.636	.978	.629	.380	.292	.874
57	.780	.202	.703	.072	.150	.824
58	.981	.188	.637	.680	.416	.818
59	.884	.289	.694	.340	.876	.109
60	.742	.621	.901	.104	.638	.532
61	.165	.094	.425	.477	.698	.248
62	.558	.528	.688	.278	.953	.264
63	.570	.175	.717	.916	.320	.253
64	.511	.066	.864	.498	.872	.186
65	.559	.044	.175	.705	.016	.562
66	.403	.931	.299	.654	.600	.733

CASE 3 (CØNT)

J	X(J,1)	X(J,2)	X(J,3)	X(J,4)	X(J,5)	X(J,6)
67	.463	.481	.116	.074	.870	.213
68	.739	.094	.651	.967	.349	.173
69	.504	.663	.638	.712	.513	.822
70	.212	.056	.592	.658	.391	.566

CASE 4 DATA

I	Y(I,1)	Y(I,2)	Y(I,3)	Y(I,4)
1	-.2500	-1.2000	.7000	-1.0000
2	-1.2000	-.0500	-1.2000	-.0500
3	-.5000	-1.0000	-.5000	-1.0000
4	1.0000	-.7000	-.6000	1.0000
5	1.0000	.5000	-1.0000	.8000
6	1.2000	.7000	-.7000	-1.1000
7	-.5000	1.0000	.6000	1.2000
8	.5250	1.0540	1.1500	.0550
9	1.1000	1.1000	.2000	.2000
10	1.1000	-1.1000	-.2000	.2000
11	-1.1000	1.1000	.2000	-.2000
12	-1.1000	-1.1000	-.2000	-.2000
13	1.5000	.5000	.8000	.6000
14	1.5000	-.5000	.7500	-.7500
15	.0470	-1.5000	-.3000	.9400
16	-1.2000	.5000	1.0000	.5000
17	-1.2000	-.5000	1.0000	-.5000
18	1.0000	-.4000	-1.4000	.4000
19	1.0000	-.4000	-1.4000	-.4000
20	.2000	-.1000	1.0000	1.3000
21	-.6820	-.3590	-.1320	-1.8380
22	-.7650	1.4370	.6250	.8080

```
              CASE  4  (CONT)

I           Y(I,1)          Y(I,2)          Y(I,3)          Y(I,4)


23          .5220          -.4680          -.2980         -1.8480

24         -.7580          -.5810          1.7140           .3730

25        -1.9940           .0860          -.1110          -.0550

26          .7020          -.7940          1.6710           .2390

27          .3690          -.8250          1.6220           .5890

28         -.8440          -.2490           .0820         -1.7860

29         -.7320           .0710          -.4260         -1.7740

30         -.7350           .0330          -.0700         -1.8580

31        -1.9130           .3420           .1820           .2230

32         -.8180          1.4470          -.9770           .0940

33         -.3800          1.4660         -1.1350          -.6260

34         -.5510          1.5800          -.9870           .4580

35        -1.9410          -.3660           .1630           .0220

36          .1130          1.5680         -1.0680           .3450

37          .0980          1.5600           .3780         -1.1680

38        -1.9060           .2180          -.4960           .1410

39        -1.0050          -.4050          1.6320           .1690

40        -1.9770          -.2900           .0450          -.0510
```

CASE 4 (CØNT)

J	X(J,1)	X(J,2)	X(J,3)	X(J,4)
1	.1370	.6680	-.5700	-.4500
2	-.2450	-.7640	-.5690	-.0990
3	-.1140	-.8740	.2970	.3620
4	.4360	.5580	.4460	.2870
5	.1970	.5910	.4740	-.5380
6	.0770	-.9320	-.2750	-.1710
7	.2170	.6450	-.6260	.1710
8	.0970	.5710	-.7950	.1120
9	.2930	.6520	.2490	.2060
10	-.2080	-.7060	-.4710	-.3580
11	.1990	-.8330	.3580	.1330
12	.2670	-.7540	.3010	.2510
13	.0280	-.9320	-.3580	-.0480
14	-.4860	-.7850	.1990	.2100
15	.2540	.6090	.2680	-.5980
16	.1630	-.8880	.2430	.2220
17	.4100	.3490	.3970	.3460
18	.2020	-.7230	-.5960	.1500
19	-.3550	.5350	-.6690	.3320
20	.2280	.5960	.4750	-.5370

```
          CASE  5  DATA

I         Y(I,1)        Y(I,2)        Y(I,3)        Y(I,4)


 1        5.1000        3.5000        1.4000         .2000

 2        4.9000        3.0000        1.4000         .2000

 3        4.7000        3.2000        1.3000         .2000

 4        4.6000        3.1000        1.5000         .2000

 5        5.0000        3.6000        1.4000         .2000

 6        5.4000        3.9000        1.7000         .4000

 7        4.6000        3.4000        1.4000         .3000

 8        5.0000        3.4000        1.5000         .2000

 9        4.4000        2.9000        1.4000         .2000

10        4.9000        3.1000        1.5000         .1000

11        5.4000        3.7000        1.5000         .2000

12        4.8000        3.4000        1.6000         .2000

13        4.8000        3.0000        1.4000         .1000

14        4.3000        3.0000        1.1000         .1000

15        5.8000        4.0000        1.2000         .2000

16        5.7000        4.4000        1.5000         .4000

17        5.4000        3.9000        1.3000         .4000

18        5.1000        3.5000        1.4000         .3000

19        5.7000        3.8000        1.7000         .3000

20        5.1000        3.8000        1.5000         .3000

21        5.4000        3.4000        1.7000         .2000

22        5.1000        3.7000        1.5000         .4000
```

```
          CASE  5  (CØNT)

I         Y(I,1)       Y(I,2)       Y(I,3)       Y(I,4)


23        4.6000       3.6000       1.0000        .2000

24        5.1000       3.3000       1.7000        .5000

25        4.8000       3.4000       1.9000        .2000

26        5.9000       3.2000       4.8000       1.8000

27        6.1000       2.8000       4.0000       1.3000

28        6.3000       2.5000       4.9000       1.5000

29        6.1000       2.8000       4.7000       1.2000

30        6.4000       2.9000       4.3000       1.3000

31        7.0000       3.2000       4.7000       1.4000

32        6.4000       3.2000       4.5000       1.5000

33        6.9000       3.1000       4.9000       1.5000

34        5.5000       2.3000       4.0000       1.3000

35        6.5000       2.8000       4.6000       1.5000

36        5.7000       2.8000       4.5000       1.3000

37        6.3000       3.3000       4.7000       1.6000

38        4.9000       2.4000       3.3000       1.0000

39        6.6000       2.9000       4.6000       1.3000

40        5.2000       2.7000       3.9000       1.4000

41        5.0000       2.0000       3.5000       1.0000

42        5.9000       3.0000       4.2000       1.5000

43        6.0000       2.2000       4.0000       1.0000

44        6.1000       2.9000       4.7000       1.4000
```

```
          CASE  5  (CØNT)

I         Y(I,1)        Y(I,2)        Y(I,3)        Y(I,4)


45        5.6000        2.9000        3.6000        1.3000

46        6.7000        3.1000        4.4000        1.4000

47        5.6000        3.0000        4.5000        1.5000

48        5.8000        2.7000        4.1000        1.0000

49        6.2000        2.2000        4.5000        1.5000

50        5.6000        2.5000        3.9000        1.1000
```

```
      CASE  5  (CØNT)

 J        X(J,1)      X(J,2)      X(J,3)      X(J,4)


 1        6.5000      3.2000      5.1000      2.0000
 2        6.4000      2.7000      5.3000      1.9000
 3        6.8000      3.0000      5.5000      2.1000
 4        5.7000      2.5000      5.0000      2.0000
 5        5.8000      2.8000      5.1000      2.4000
 6        6.4000      3.2000      5.3000      2.3000
 7        6.5000      3.0000      5.5000      1.8000
 8        7.7000      3.8000      6.7000      2.2000
 9        7.7000      2.6000      6.9000      2.3000
10        6.0000      2.2000      5.0000      1.5000
11        6.3000      3.3000      6.0000      2.5000
12        5.8000      2.7000      5.1000      1.9000
13        7.1000      3.0000      5.9000      2.1000
14        6.3000      2.9000      5.6000      1.8000
15        6.5000      3.0000      5.8000      2.2000
16        7.6000      3.0000      6.6000      2.1000
17        4.9000      2.5000      4.5000      1.7000
18        7.3000      2.9000      6.3000      1.8000
19        6.7000      2.5000      5.8000      1.8000
20        7.2000      3.6000      6.1000      2.5000
21        6.9000      3.2000      5.7000      2.3000
22        5.6000      2.8000      4.9000      2.0000
```

```
           CASE  5  (CØNT)

      J          X(J,1)          X(J,2)          X(J,3)          X(J,4)


      23         7.7000          2.8000          6.7000          2.0000

      24         6.3000          2.7000          4.9000          1.8000

      25         6.7000          3.3000          5.7000          2.1000
```

CASE 6 DATA

I	Y(I,1)	Y(I,2)	Y(I,3)	Y(I,4)	Y(I,5)	Y(I,6)	Y(I,7)	Y(I,8)
1	160.	5.1	10.0	28.0	70.	450.	4.10	21.0
2	155.	5.2	8.0	27.0	85.	400.	3.12	17.0
3	130.	5.0	10.0	26.0	75.	560.	2.96	19.0
4	161.	5.0	10.0	27.0	70.	665.	4.04	21.0
5	135.	5.0	12.0	27.0	88.	570.	3.05	23.0
6	165.	5.0	11.0	23.0	95.	675.	3.72	24.0
7	150.	5.0	9.0	29.0	90.	580.	3.83	25.0
8	148.	4.8	8.0	26.0	85.	390.	3.75	28.0
9	150.	4.5	7.0	31.0	60.	435.	3.05	20.0
10	120.	4.0	6.0	33.0	55.	440.	2.78	18.0
11	120.	5.1	8.0	32.0	56.	650.	2.89	23.0
12	190.	4.2	8.0	30.0	55.	640.	3.46	27.0
13	100.	4.4	9.0	35.0	48.	430.	2.98	26.0
14	150.	4.0	7.0	29.0	65.	650.	2.84	20.0
15	90.	4.6	8.0	30.0	70.	655.	2.96	17.0
16	120.	4.7	7.0	35.0	67.	645.	2.78	20.0
17	200.	4.3	9.0	30.0	62.	660.	3.20	23.0
18	120.	4.1	8.0	28.0	63.	530.	3.05	19.0
19	105.	5.0	7.0	27.0	64.	435.	3.55	27.0
20	210.	5.2	8.0	26.0	67.	440.	3.10	20.0
21	90.	4.0	10.0	25.0	68.	430.	3.41	21.0
22	110.	5.2	11.0	25.0	60.	530.	2.97	25.0

CASE 6 (CONT)

I	Y(I,1)	Y(I,2)	Y(I,3)	Y(I,4)	Y(I,5)	Y(I,6)	Y(I,7)	Y(I,8)
23	100.	4.3	9.0	25.0	70.	440.	3.62	17.0
24	70.	4.5	8.0	23.0	64.	450.	3.40	19.0
25	100.	4.8	9.0	27.0	65.	355.	3.10	24.0
26	130.	5.2	9.0	25.0	70.	380.	2.80	29.0
27	90.	4.5	11.0	37.0	74.	350.	2.95	31.0
28	80.	4.6	10.0	32.0	78.	450.	2.12	32.0
29	95.	4.9	10.0	25.0	82.	260.	3.11	23.0
30	70.	4.4	12.0	30.0	85.	262.	3.22	24.0
31	80.	4.2	13.0	36.0	69.	260.	3.35	26.0
32	95.	5.1	15.0	31.0	70.	265.	3.40	29.0
33	100.	4.6	11.0	24.0	76.	270.	3.54	30.0
34	95.	4.8	10.0	27.0	74.	355.	3.63	31.0
35	85.	4.7	12.0	25.0	73.	360.	3.17	29.0
36	70.	4.8	11.0	26.0	78.	365.	3.18	30.0
37	80.	5.4	10.0	21.0	80.	370.	2.90	32.0
38	85.	5.5	13.0	33.0	81.	355.	3.12	35.0

CASE 6 (CØNT)

J	X(J,1)	X(J,2)	X(J,3)	X(J,4)	X(J,5)	X(J,6)	X(J,7)	X(J,8)
1	200.	3.4	10.0	24.0	98.	210.	4.10	40.0
2	260.	3.1	8.0	21.0	110.	220.	3.89	36.0
3	195.	3.0	9.0	20.0	105.	130.	2.98	44.0
4	195.	3.2	9.0	19.0	110.	010.	3.10	36.0
5	220.	3.3	10.0	24.0	95.	205.	3.54	29.0
6	220.	3.0	8.0	25.0	90.	210.	3.78	34.0
7	190.	3.4	9.0	26.0	96.	070.	3.36	30.0
8	285.	3.0	11.0	19.0	100.	990.	4.23	31.0
9	300.	3.0	9.0	20.0	102.	120.	3.75	40.0
10	225.	3.0	10.0	22.0	105.	985.	3.80	25.0
11	260.	3.4	8.0	22.0	97.	090.	2.94	38.0
12	280.	3.0	8.0	20.0	112.	200.	2.70	41.0
13	300.	3.4	10.0	20.0	108.	835.	3.20	45.0
14	310.	3.0	11.0	19.0	106.	055.	3.45	37.0
15	290.	3.1	12.0	26.0	94.	240.	3.50	41.0
16	260.	3.0	8.0	22.0	98.	015.	4.60	29.0
17	290.	3.3	9.0	25.0	100.	010.	3.56	32.0
18	160.	3.1	11.0	20.0	79.	170.	4.26	34.0
19	240.	3.5	11.0	20.0	88.	990.	4.40	42.0
20	195.	3.1	8.0	21.0	81.	975.	3.58	44.0
21	290.	3.4	10.0	19.0	94.	860.	3.40	40.0
22	210.	3.5	9.0	22.0	96.	950.	2.70	47.0

CASE 6 (CONT)

J	X(J,1)	X(J,2)	X(J,3)	X(J,4)	X(J,5)	X(J,6)	X(J,7)	X(J,8)
23	180.	3.0	11.0	22.0	97.	990.	3.83	28.0
24	205.	2.9	11.0	23.0	90.	805.	3.25	38.0
25	215.	3.4	8.0	21.0	100.	700.	4.05	32.0
26	270.	3.1	8.0	20.0	111.	170.	4.60	31.0
27	290.	3.0	9.0	23.0	102.	350.	4.42	33.0
28	320.	3.2	10.0	19.0	87.	160.	3.64	38.0
29	210.	3.0	9.0	18.0	112.	010.	3.81	45.0
30	210.	3.0	8.0	21.0	95.	190.	4.00	40.0
31	185.	3.4	9.0	25.0	96.	055.	4.18	41.0
32	200.	3.2	8.0	26.0	98.	980.	3.46	38.0
33	170.	2.9	9.0	20.0	95.	095.	4.15	40.0
34	140.	3.0	9.0	20.0	98.	990.	4.25	32.0
35	110.	4.9	9.0	22.0	130.	220.	4.51	19.0
36	100.	5.6	8.0	19.0	128.	216.	2.60	14.0
37	95.	4.9	8.0	24.0	124.	218.	2.45	10.0
38	65.	6.2	9.0	30.0	134.	200.	2.61	12.0
39	55.	5.0	10.0	27.0	128.	205.	2.62	11.0
40	70.	5.3	7.0	28.0	118.	204.	2.35	12.0
41	85.	4.9	11.0	19.0	117.	206.	2.90	14.0
42	115.	5.0	10.0	21.0	122.	198.	3.25	17.0
43	95.	4.8	8.0	27.0	114.	228.	2.65	11.0
44	120.	6.1	9.0	24.0	120.	244.	2.82	18.0

CASE 7 DATA

I	Y1	Y2	Y3	Y4	Y5	Y6	Y7	Y8	Y9	Y10	Y11	Y12	Y13
1	1	365	152	90	4	2	4	3	1	1	5	250	39
2	1	517	126	70	1	2	3	2	2	2	2	326	38
3	1	343	180	86	1	1	3	2	2	3	2	278	42
4	1	432	110	65	4	1	3	2	4	4	2	278	42
5	1	376	115	80	1	1	3	2	1	3	2	278	41
6	1	410	110	70	1	1	3	2	2	3	2	255	43
7	1	408	105	65	1	1	3	2	2	3	2	278	50
8	1	410	150	100	1	2	4	2	2	4	2	278	44
9	1	315	118	76	1	2	3	2	2	4	5	315	28
10	1	546	150	70	4	1	3	2	2	3	1	212	47
11	1	437	120	70	4	2	3	2	2	4	2	372	39
12	1	506	106	72	4	2	3	2	2	3	1	145	37
13	1	338	110	70	2	2	3	2	1	3	1	278	39
14	1	446	120	70	1	1	2	2	1	4	1	278	50
15	1	426	135	80	1	1	3	2	1	4	1	278	48
16	1	410	140	85	1	1	3	2	2	4	1	250	35
17	1	483	120	70	1	2	3	2	2	4	1	278	48
18	3	377	114	70	2	2	2	2	2	4	5	350	38
19	3	410	130	85	2	1	3	2	2	4	2	264	39
20	3	414	120	80	4	2	1	2	1	3	5	246	49
21	1	369	140	90	4	1	3	2	1	4	1	278	38
22	1	335	132	82	3	1	3	1	1	4	1	278	38

CASE 7 (CONT)

I	Y1	Y2	Y3	Y4	Y5	Y6	Y7	Y8	Y9	Y10	Y11	Y12	Y13
23	1	410	140	80	3	1	3	1	2	4	2	268	40
24	1	422	152	92	1	2	2	2	2	3	2	278	49
25	1	517	110	80	1	2	3	2	2	4	5	250	53
26	1	250	130	70	1	2	3	2	2	4	2	315	45
27	1	466	114	74	1	1	2	2	2	4	3	250	38
28	1	366	120	90	1	1	2	2	2	4	3	278	47
29	1	271	110	70	1	1	3	2	2	4	5	295	36
30	1	467	130	70	4	1	3	2	2	4	5	278	41
31	1	372	128	92	4	1	3	2	2	4	5	353	43
32	1	300	140	75	4	2	3	2	2	4	5	200	38
33	1	348	110	80	4	1	2	1	2	4	2	335	40
34	1	288	130	80	1	1	4	2	2	4	2	264	47
35	1	477	115	80	1	1	2	2	2	4	2	245	43
36	1	406	140	110	2	2	2	2	2	4	5	300	33
37	1	489	110	64	1	2	2	2	2	3	5	278	42
38	1	406	130	80	4	2	4	2	2	3	5	278	51
39	1	367	135	85	1	2	3	2	1	3	3	350	40
40	1	444	125	80	1	2	3	2	1	3	1	302	45
41	1	402	122	70	1	1	1	2	2	3	1	278	46
42	1	462	120	78	1	2	3	1	2	4	1	278	41
43	1	511	140	95	4	1	2	2	1	3	1	350	45
44	1	410	140	90	4	1	2	2	1	3	2	245	52

CASE 7 (CØNT)

I	Y1	Y2	Y3	Y4	Y5	Y6	Y7	Y8	Y9	Y10	Y11	Y12	Y13
45	1	446	90	60	1	1	4	1	1	3	2	202	40
46	1	468	110	70	1	2	3	2	1	4	1	314	37
47	1	410	130	80	1	2	3	2	2	4	5	351	40
48	1	410	140	88	1	2	3	2	2	4	1	242	42
49	1	355	120	70	1	1	3	2	2	4	3	366	23
50	2	394	116	80	1	2	1	2	2	4	2	283	42
51	2	312	100	60	1	2	1	2	2	4	2	310	47
52	1	383	100	60	1	2	1	2	2	4	2	310	47
53	1	489	140	80	1	2	2	2	1	4	2	278	46
54	1	410	130	82	1	2	2	2	1	4	2	250	41
55	1	404	140	98	1	2	2	2	1	4	2	278	43
56	1	378	140	90	1	2	3	1	2	3	1	267	43
57	1	369	130	70	2	1	4	2	1	3	1	278	44
58	1	388	133	90	3	1	3	2	3	4	5	288	43
59	1	483	110	80	4	1	2	2	2	4	2	277	39
60	3	461	140	90	1	2	4	1	1	2	1	330	52

CASE 7 (CØNT)

J	X1	X2	X3	X4	X5	X6	X7	X8	X9	X10	X11	X12	X13
1	1	404	160	78	4	2	1	2	1	4	1	200	67
2	1	418	118	76	1	1	3	3	1	1	2	298	43
3	1	288	128	86	1	2	4	3	1	1	2	209	37
4	2	349	170	110	3	2	1	1	4	1	5	343	40
5	1	396	118	72	3	1	2	3	1	1	2	221	37
6	3	411	180	120	1	2	4	1	1	1	2	196	52
7	1	443	160	90	4	2	3	3	1	1	2	271	37
8	1	479	140	90	1	2	2	1	1	1	1	260	40
9	3	388	140	90	1	2	3	1	1	1	5	281	40
10	2	445	130	88	4	2	4	3	1	1	1	215	42
11	1	389	110	78	3	2	4	3	1	1	2	249	32
12	1	406	114	70	2	2	3	1	1	1	5	248	40
13	1	431	100	60	4	2	3	1	1	1	1	224	34
14	2	413	144	86	4	2	4	3	1	1	1	247	41
15	1	411	110	50	4	1	4	1	1	1	2	240	40
16	1	406	110	76	1	1	3	3	1	4	2	248	23
17	1	444	120	80	1	2	1	3	1	1	1	151	42
18	2	375	120	70	1	2	3	3	1	1	3	248	41
19	1	383	120	80	2	1	3	3	1	1	1	256	36
20	2	439	114	80	1	2	4	3	1	1	2	248	40
21	2	489	104	70	1	2	3	3	1	3	1	221	44
22	1	387	118	90	1	1	4	3	1	1	2	279	43

CASE 7 (CØNT)

J	X1	X2	X3	X4	X5	X6	X7	X8	X9	X10	X11	X12	X13
23	3	351	120	90	4	2	3	3	2	5	5	258	36
24	1	350	110	66	1	1	3	3	1	1	3	248	41
25	3	536	124	84	4	2	2	3	1	1	5	282	33
26	2	432	110	70	1	1	4	1	1	2	1	240	55
27	3	393	150	90	3	1	3	1	1	1	5	245	46
28	1	369	132	80	1	2	2	3	1	1	1	237	40
29	2	418	120	80	3	2	3	3	1	3	2	236	40
30	1	383	126	84	1	1	4	3	1	0	1	221	36
31	1	325	135	80	1	2	4	3	1	5	5	175	30
32	1	395	120	78	4	1	4	1	1	1	1	178	40
33	1	375	117	75	1	1	4	3	1	1	1	229	39
34	1	432	142	80	4	2	4	3	1	1	1	235	43
35	2	267	120	84	4	2	4	3	1	1	1	213	40
36	1	276	158	100	1	2	4	3	1	1	2	167	41
37	3	489	188	70	1	2	3	3	1	1	3	248	44
38	1	369	120	84	1	1	4	3	1	1	1	294	43
39	1	398	150	90	4	2	4	3	1	1	1	223	42
40	1	442	118	70	3	2	2	3	1	1	5	240	44
41	1	394	130	80	1	2	3	3	1	1	5	288	31
42	1	444	120	78	3	2	2	3	1	2	1	256	42
43	1	389	140	95	1	1	4	3	2	2	5	273	44
44	1	359	130	100	1	1	2	3	1	1	5	195	35

CASE 7 (CONT)

J	X1	X2	X3	X4	X5	X6	X7	X8	X9	X10	X11	X12	X13
45	2	232	142	100	1	1	4	1	1	1	5	325	55
46	1	379	106	72	4	2	4	3	1	1	1	264	34
47	1	378	140	100	4	1	4	1	1	1	1	309	39
48	1	381	150	90	4	1	4	1	1	1	1	282	54
49	2	364	128	82	4	1	4	3	1	1	5	250	33
50	2	437	140	90	4	2	4	3	1	1	5	248	36

APPENDIX IV. TABLE OF A AND B VECTORS

The following table is the values of the various components of the A and B vectors used in the seven examples. Only six of the thirteeen components have been listed. To use the table, repeat the pattern of the values until the number of components equals the dimension of the problem. For the remaining components, use a zero value until one has filled the thirteen components.

TABLE OF A AND B VALUES

DIMENSION		COMPONENTS					
		1	2	3	4	5	6
3	A	.50000	.70711	.50000	.00000	.00000	.00000
	B	.50000	-.70711	.50000	.00000	.00000	.00000
4	A	.50000	.50000	.50000	.50000	.00000	.00000
	B	.50000	-.50000	.50000	-.50000	.00000	.00000
5	A	.40825	.50000	.40825	.50000	.40825	.00000
	B	.40825	-.50000	.40825	-.50000	.40825	.00000
6	A	.40825	.40825	.40825	.40825	.40825	.40825
	B	.40825	-.40825	.40825	-.40825	.40825	-.40825
7	A	.35355	.40825	.35355	.40825	.35355	.40825
	B	.35355	-.40825	.35355	-.40825	.35355	-.40825
8	A	.35355	.35355	.35355	.35355	.35355	.35355
	B	.35355	-.35355	.35355	-.35355	.35355	-.35355
9	A	.31628	.35355	.31628	.35355	.31628	.35355
	B	.31628	-.35355	.31628	-.35355	.31628	-.35355
10	A	.31628	.31628	.31628	.31628	.31628	.31628
	B	.31628	-.31628	.31628	-.31628	.31628	-.31628
11	A	.28867	.31628	.28867	.31628	.28867	.31628
	B	.28867	-.31628	.28867	-.31628	.28867	-.31628
12	A	.28867	.28867	.28867	.28867	.28867	.28867
	B	.28867	-.28867	.28867	-.28867	.28867	-.28867
13	A	.26726	.28867	.26726	.28867	.26726	.28867
	B	.26726	-.28867	.26726	-.28867	.26726	-.28867

APPENDIX V. TWO DIMENSIONAL PLOTS

The following figures represent the plots of the computer printout. The values of the A and B vectors are printed below the plot if a discriminant function was determined. The components for the initial projections can be extracted from Appendix IV. A polaroid print is also included for each case. This allows for a comparison between the AGT display and the computer plot to be obtained.

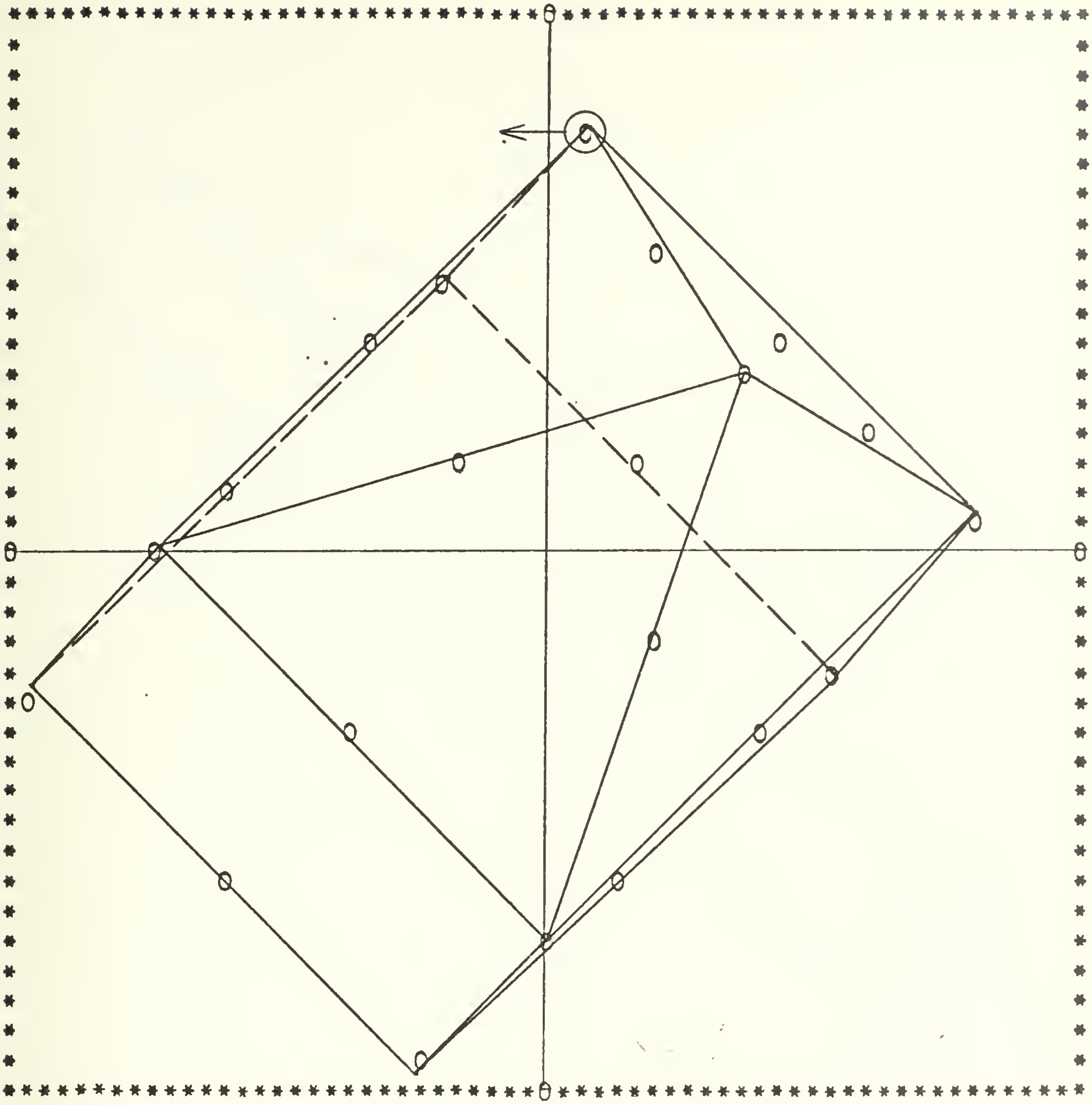

Figure IV-1-1, CASE 1, STEP 1

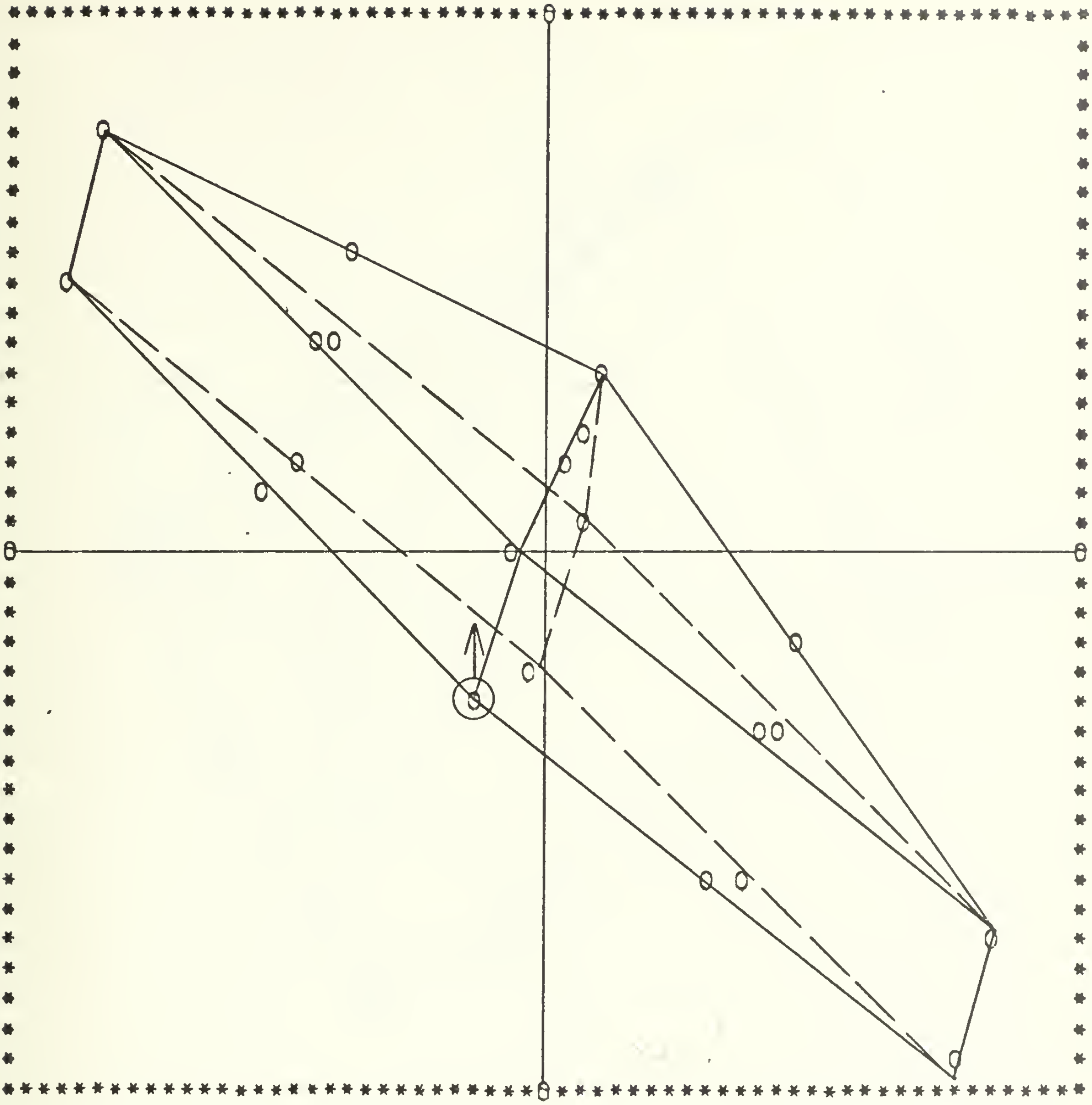

Figure IV-1-2. CASE 1, STEP 2

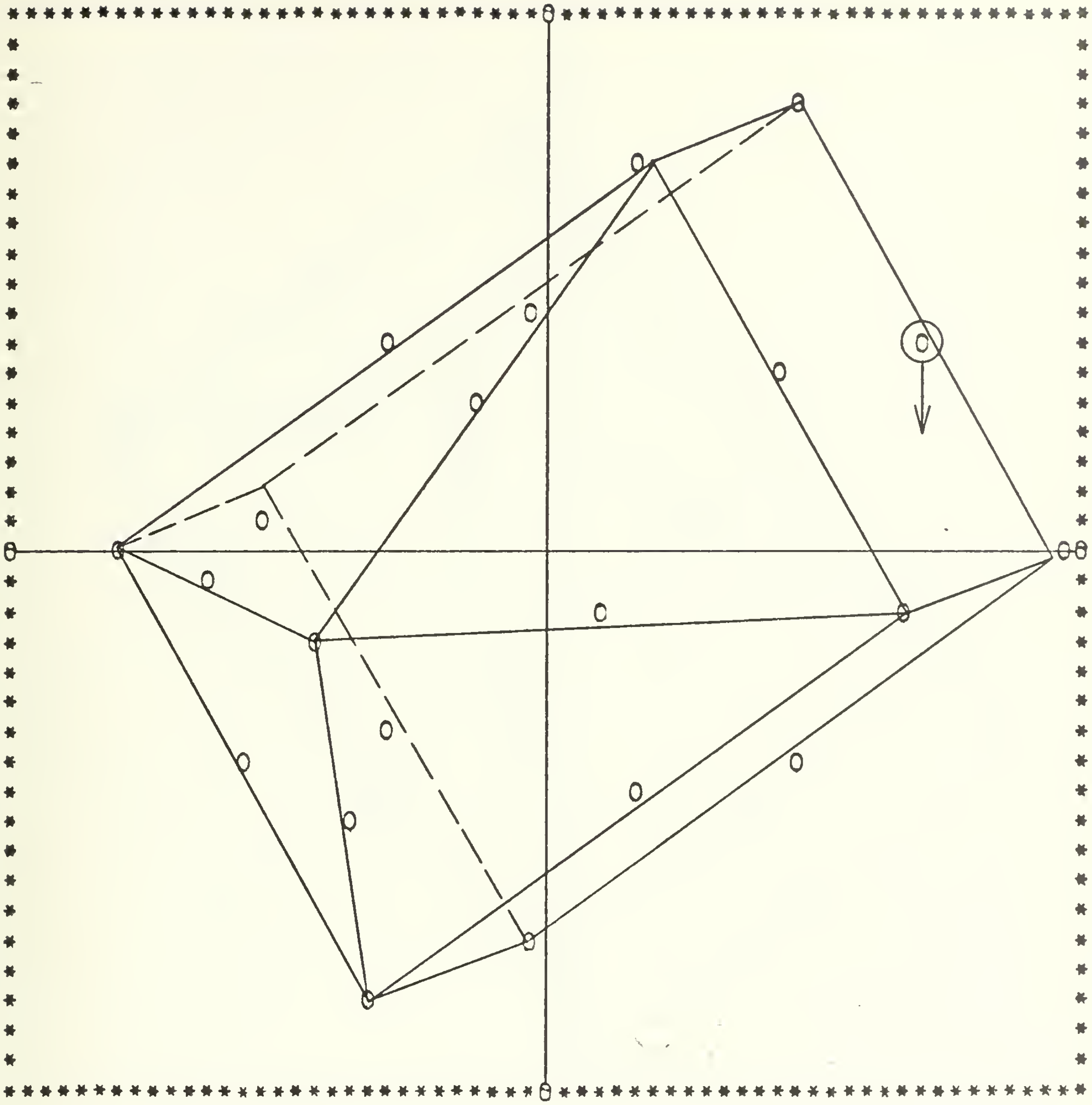

Figure IV-1-3. Case 1, STEP 3

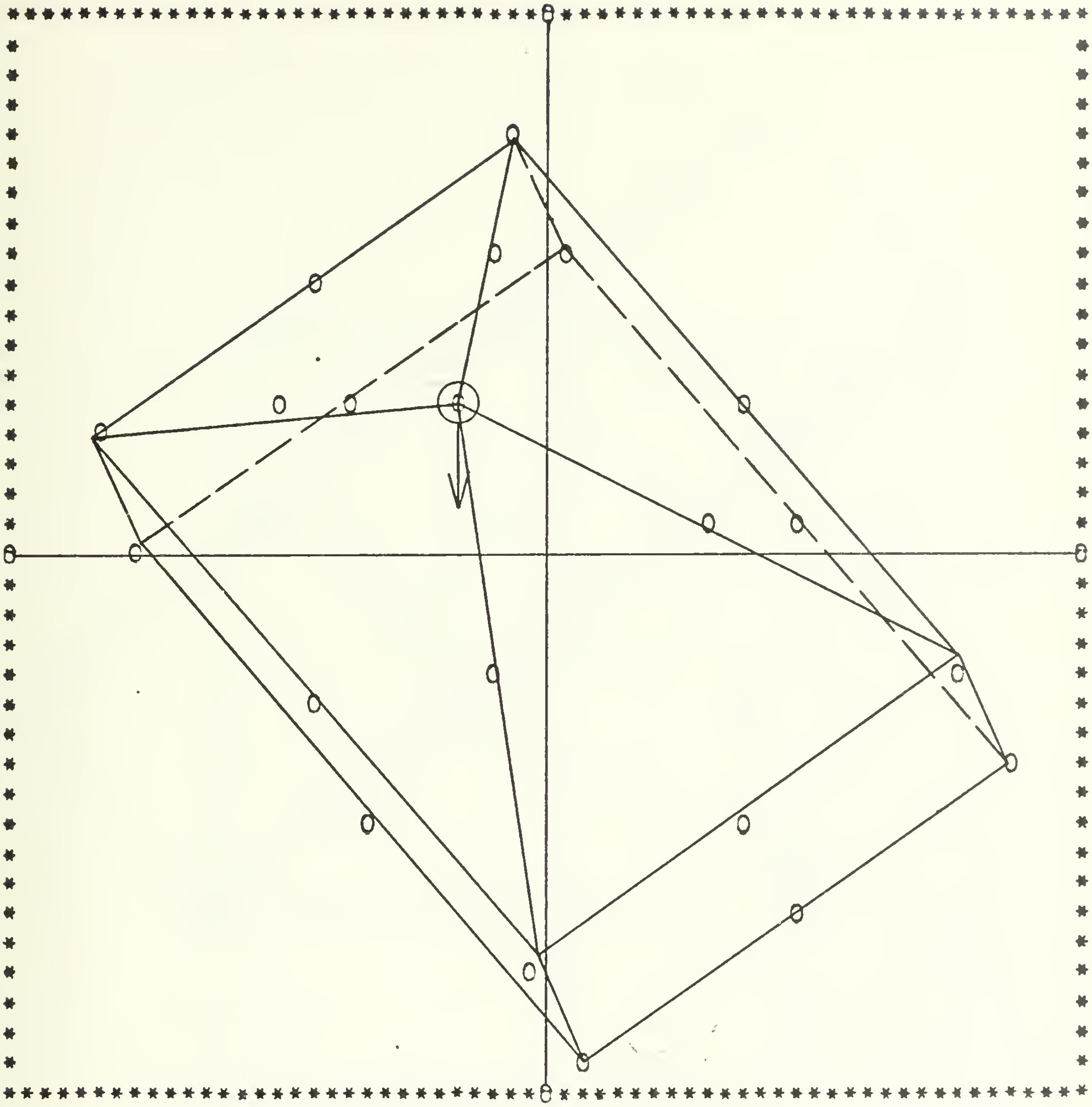

Figure IV-1-4. CASE 1, STEP 4

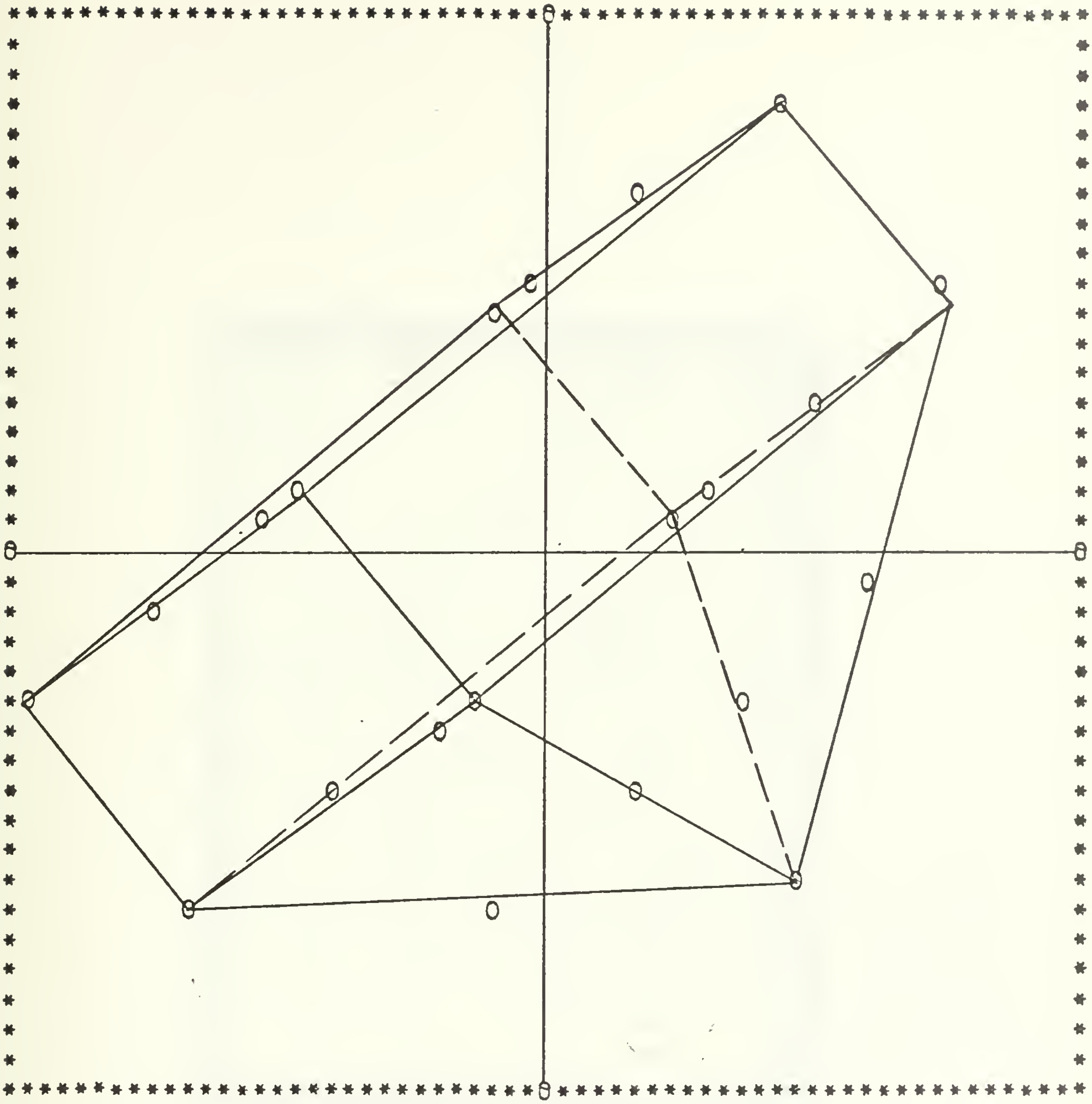

Figure IV-1-5. CASE 1, STEP 5

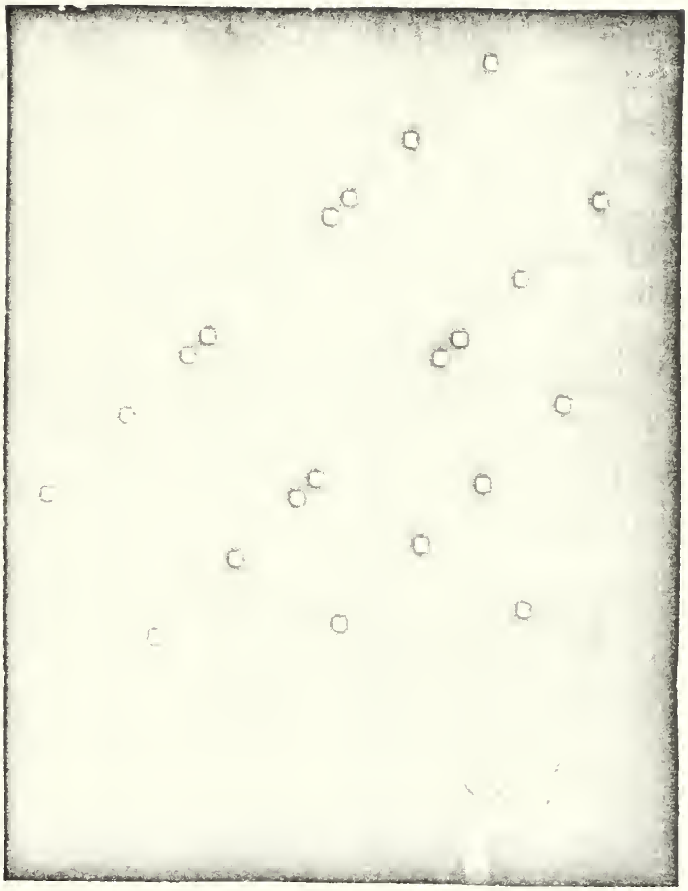

Figure IV-1-5P. CASE 1, STEP 5

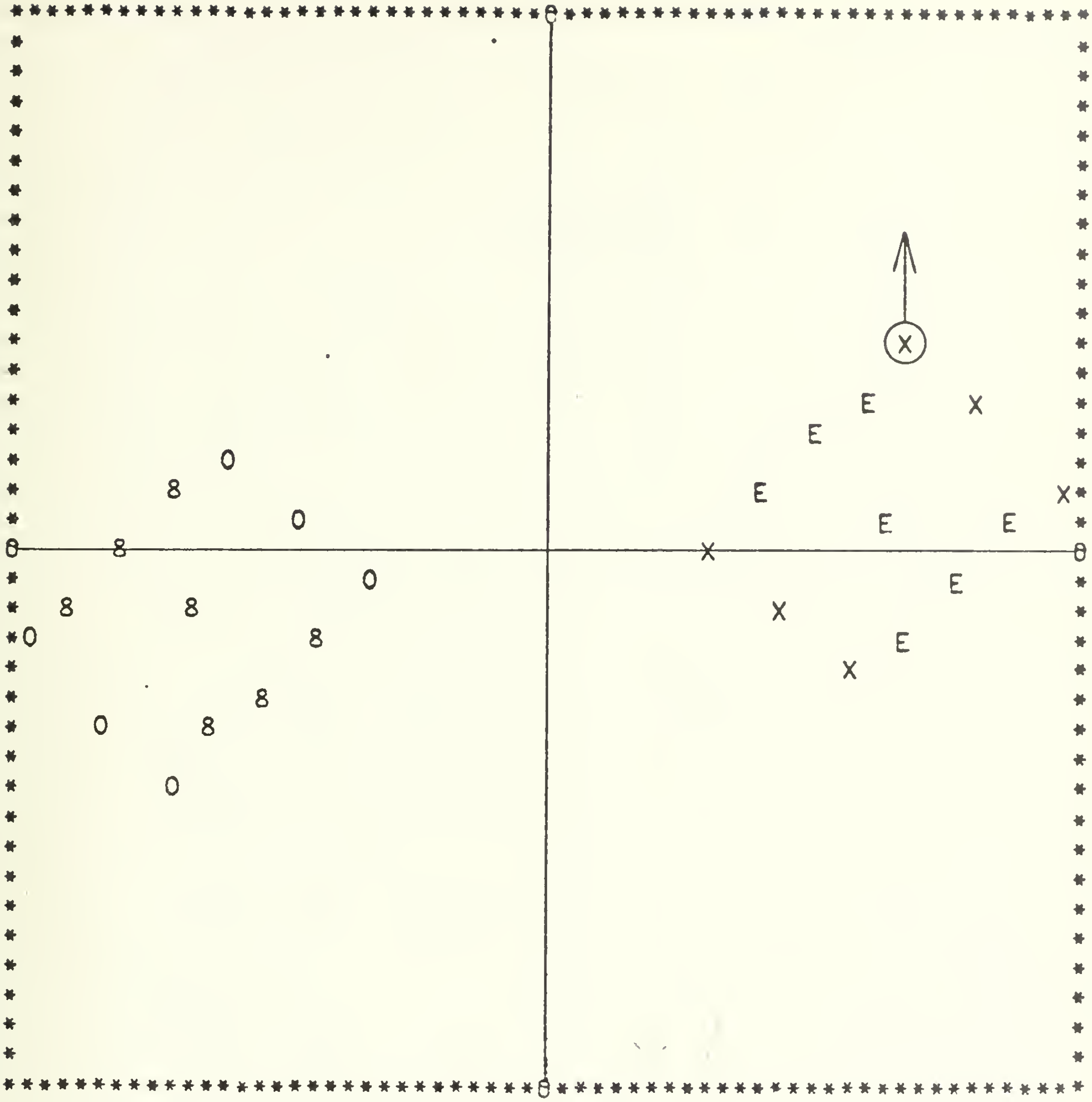

Figure IV-2-1. CASE 2, STEP 1

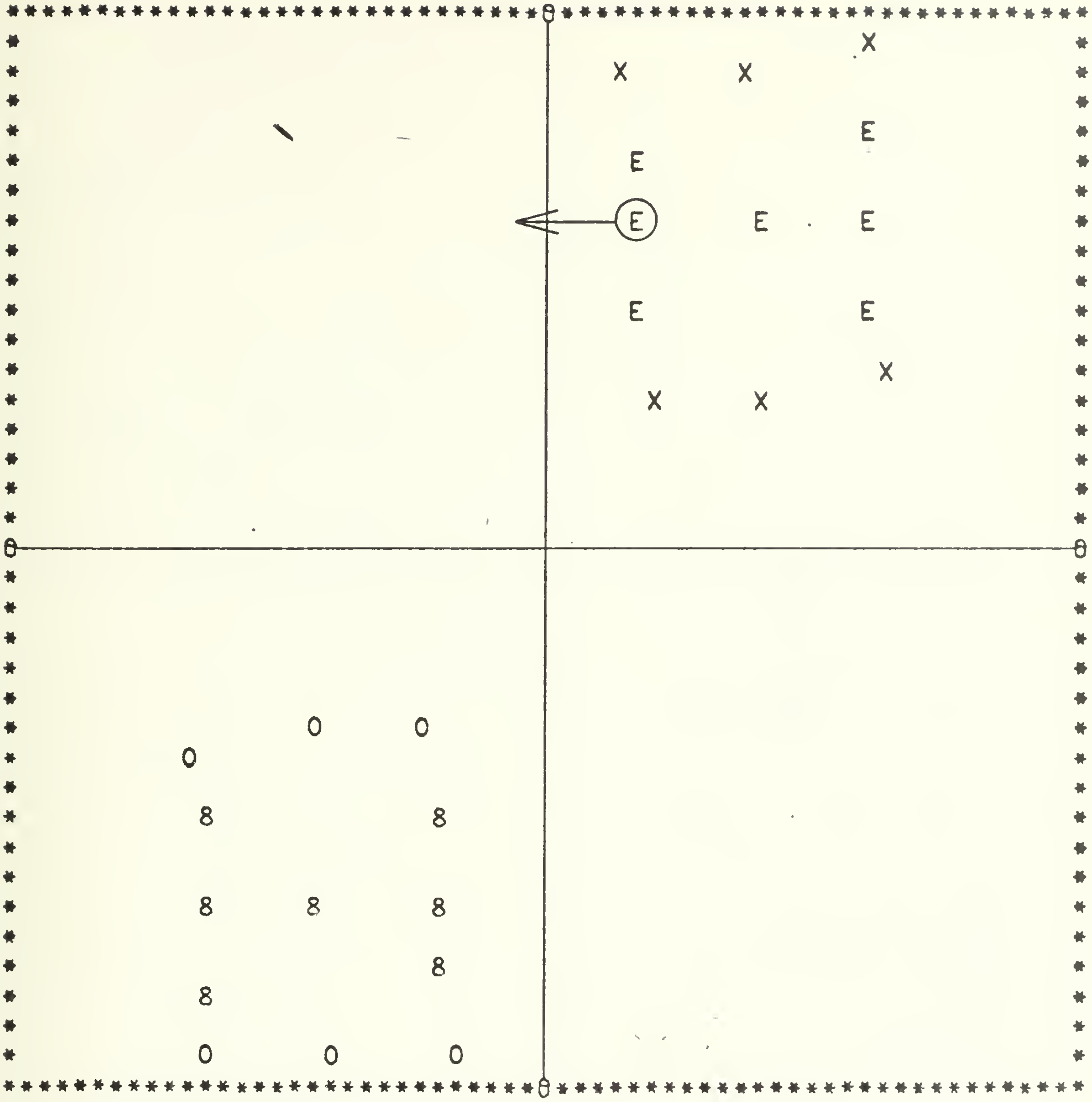

Figure IV-2-2. CASE 2, STEP 2

A(1) =-.0418	B(1) = .7059
A(2) = .9983	B(2) = .0591
A(3) =-.0418	B(3) = .7059

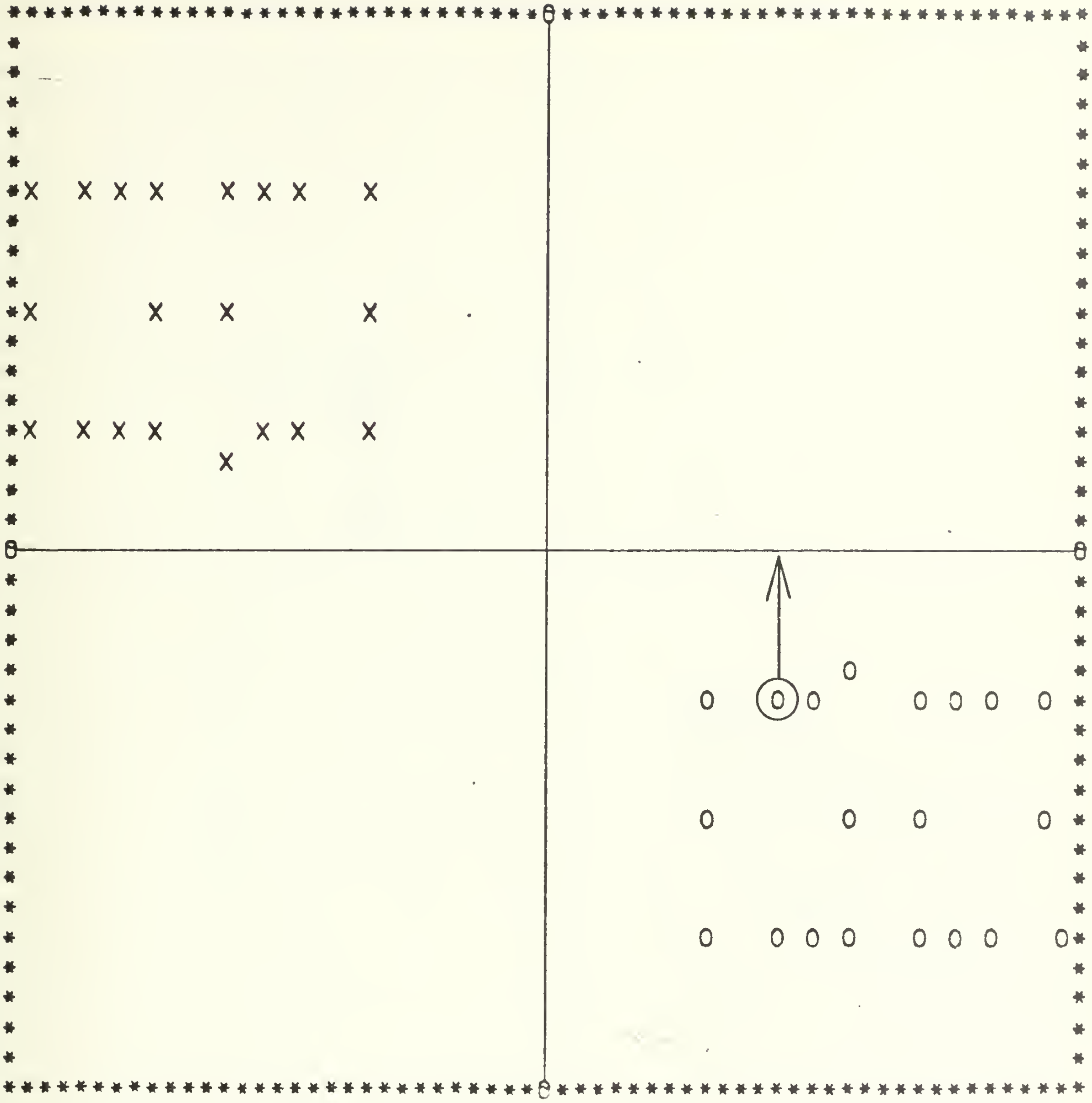

Figure IV-2-3. CASE 2, STEP 3

A(1) =-.3240	B(1) =-.0248
A(2) =-.0005	B(2) = .9997
A(3) =-.9461	B(3) = .0079

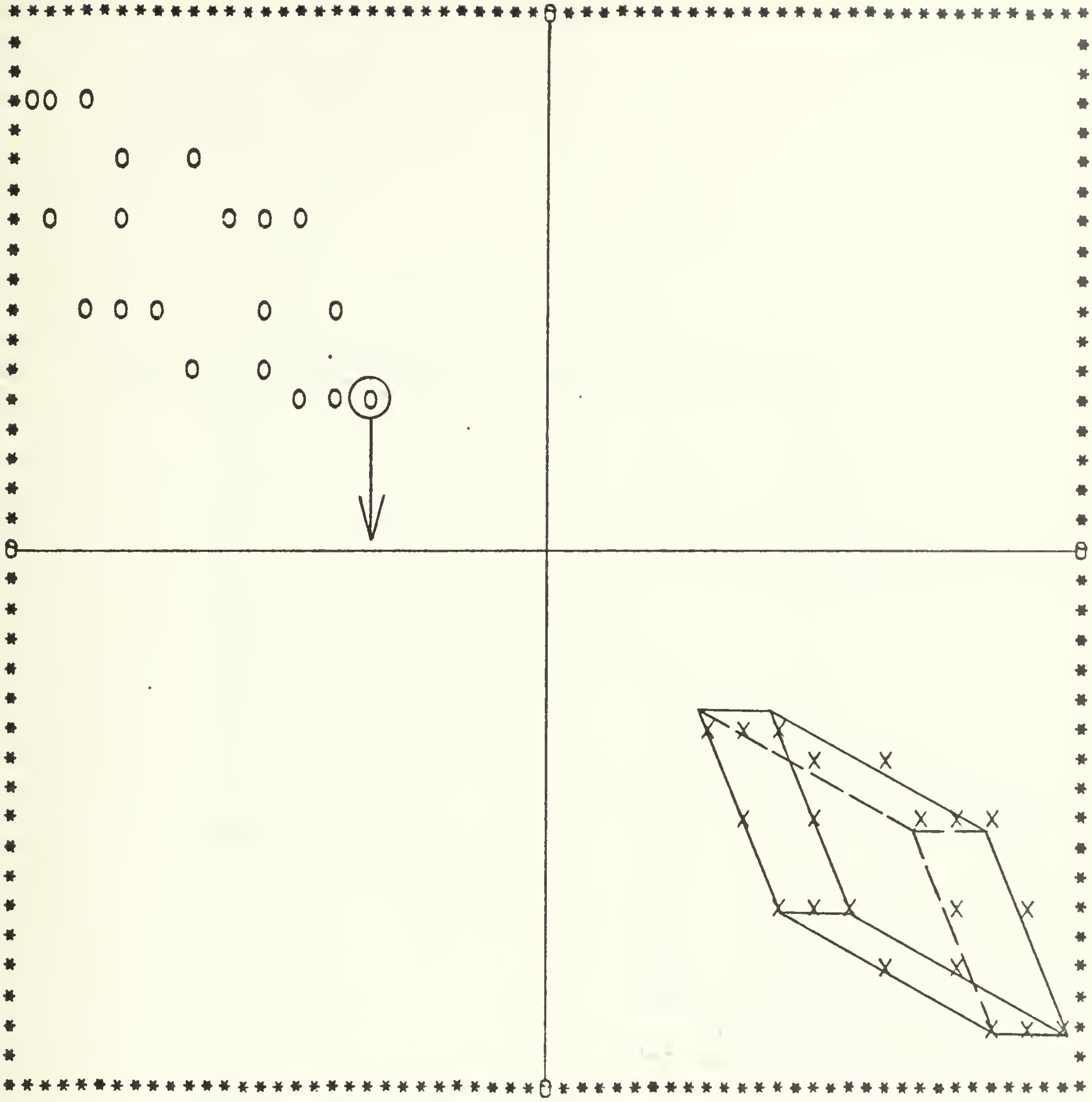

Figure IV-2-4. CASE 2, STEP 5

A(1) = .0081	B(1) =-.8980
A(2) =-.9998	B(2) =-.0002
A(3) =-.0161	B(3) =-.4400

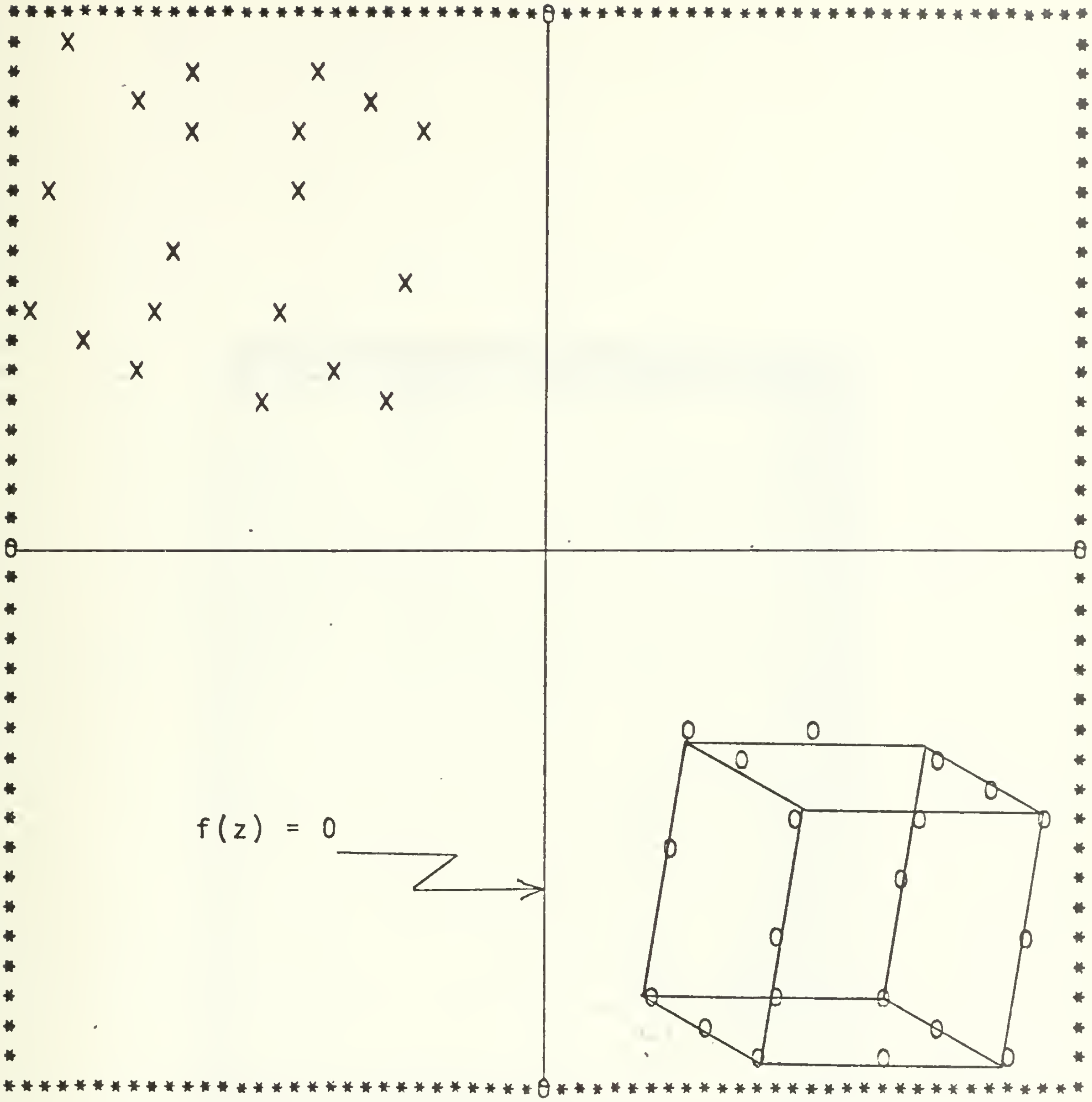

Figure IV-2-5. CASE 2, STEP 6

A(1) =-.9122	B(1) = .0887
A(2) = .2671	B(2) = .8690
A(3) =-.3107	B(3) = .4867

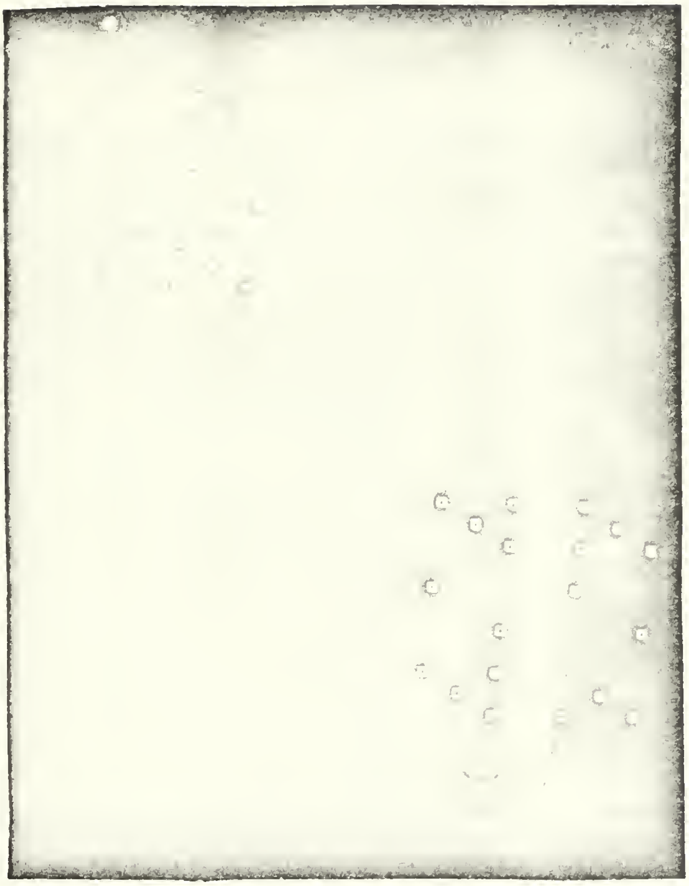

Figure IV-2-5P, CASE 2, STEP 6

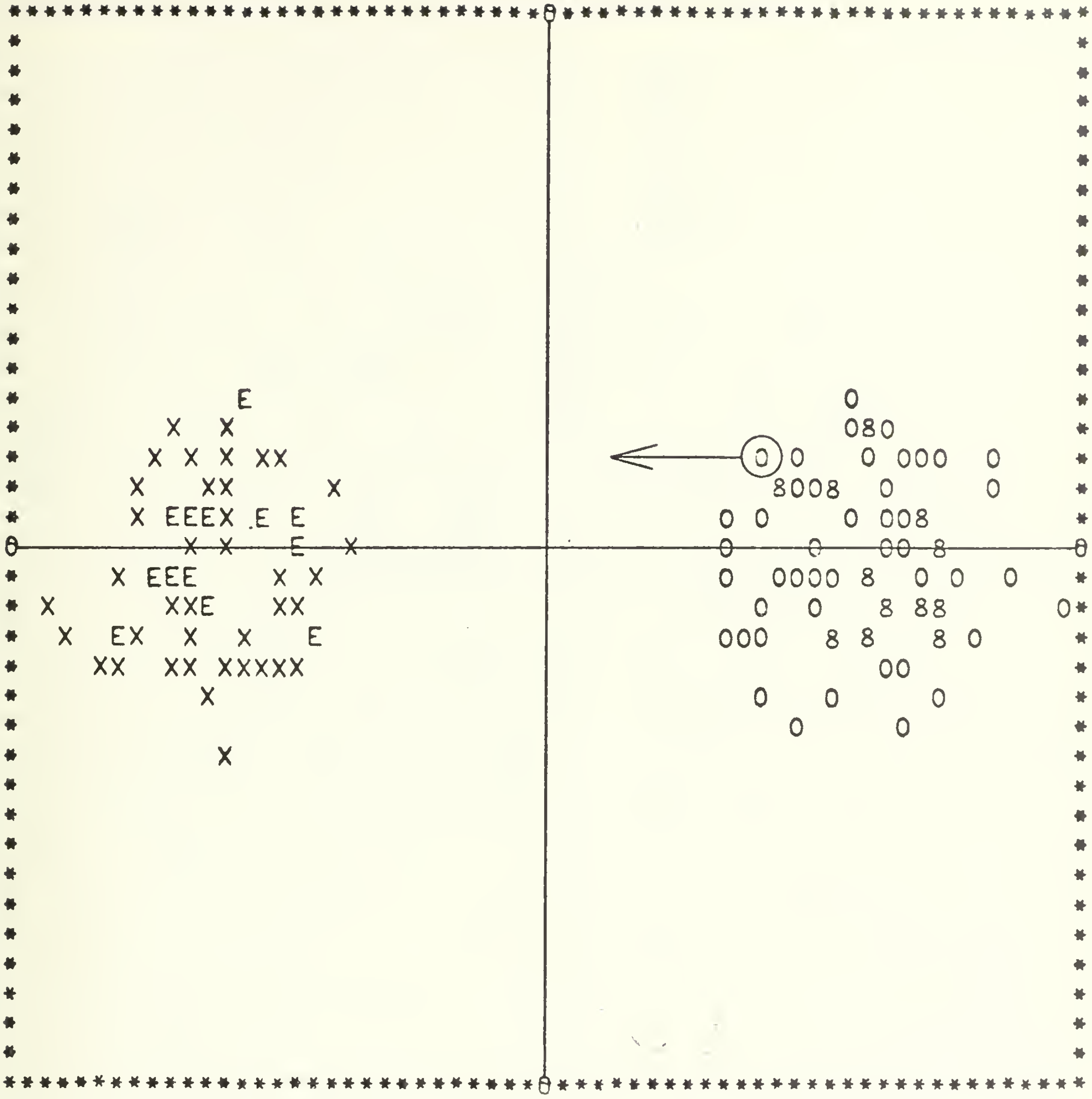

Figure IV-3-1. CASE 3, STEP 1

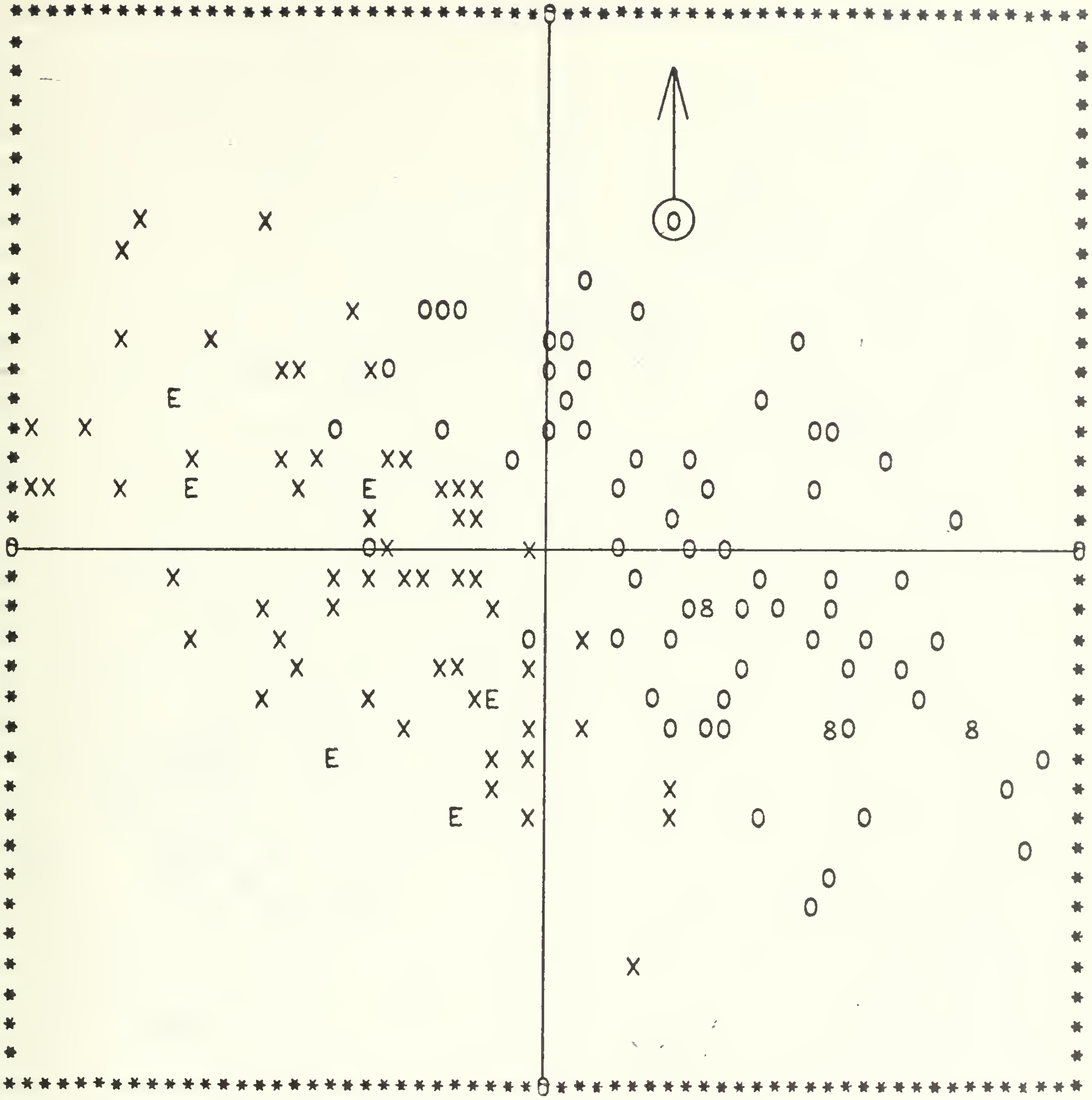

Figure IV-3-2. CASE 3, STEP 2

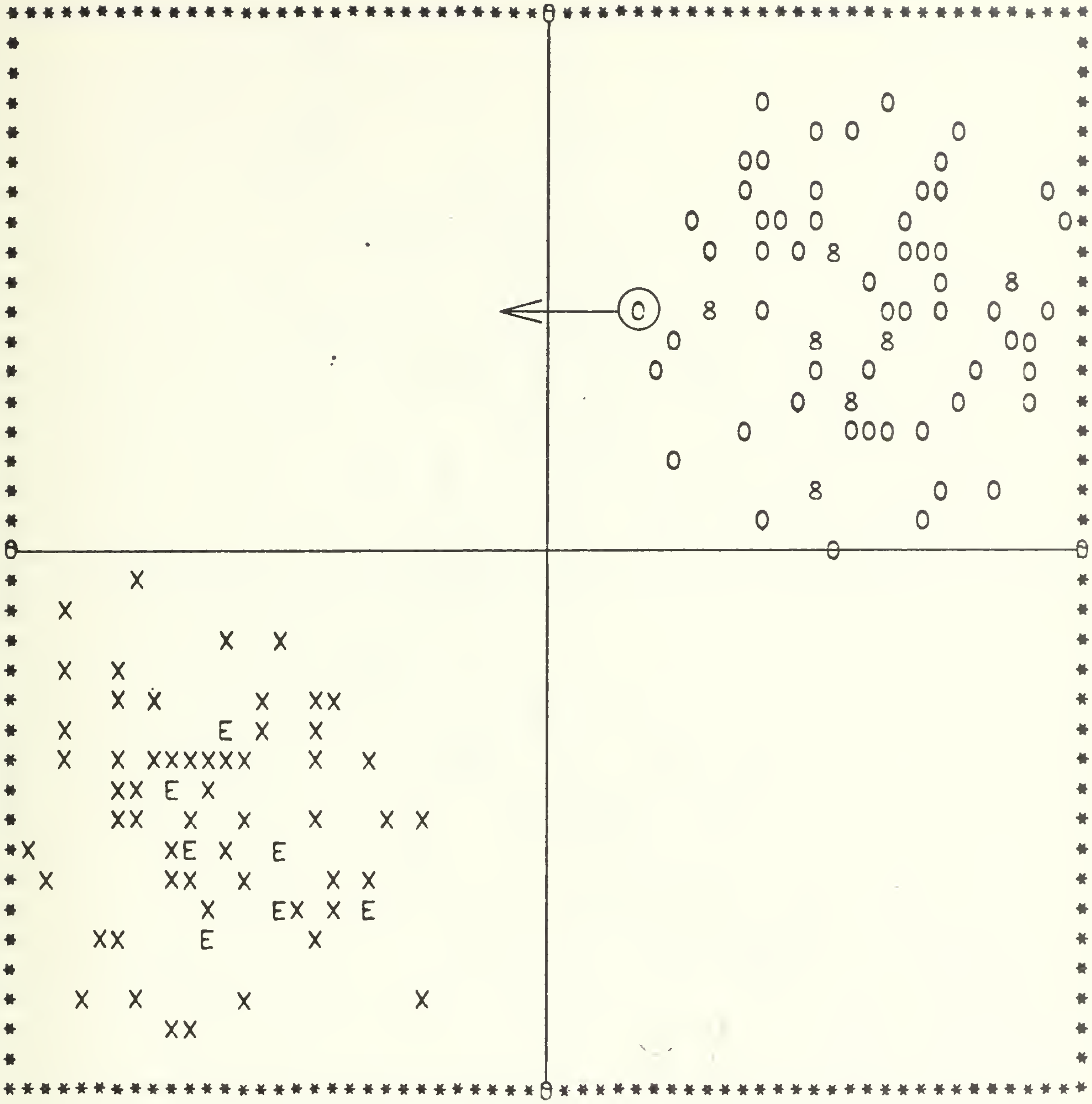

Figure IV-3-3. CASE 3, STEP 3

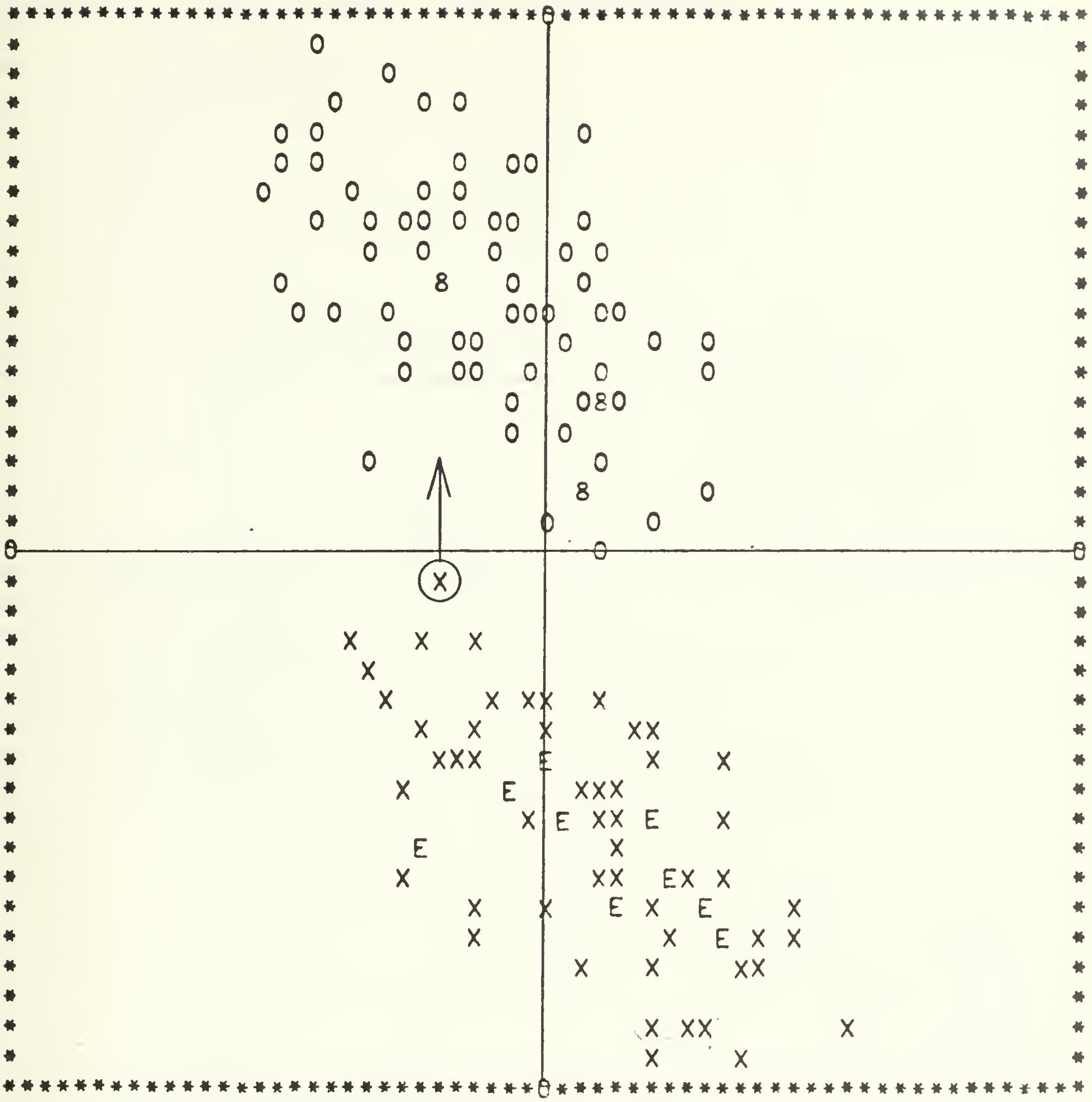

Figure IV-3-4. CASE 3, STEP 4

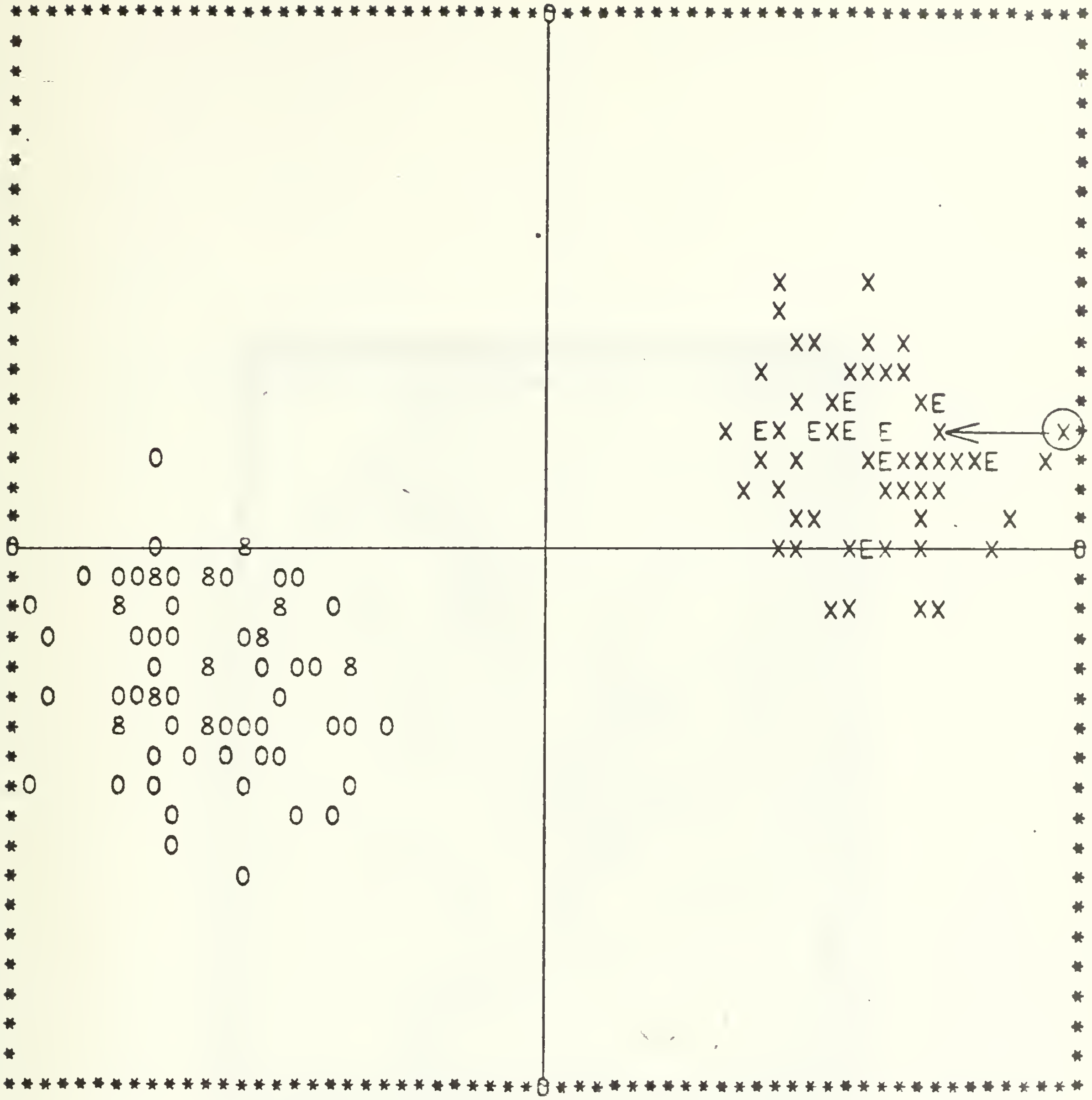

Figure IV-3-5. CASE 3, STEP 5

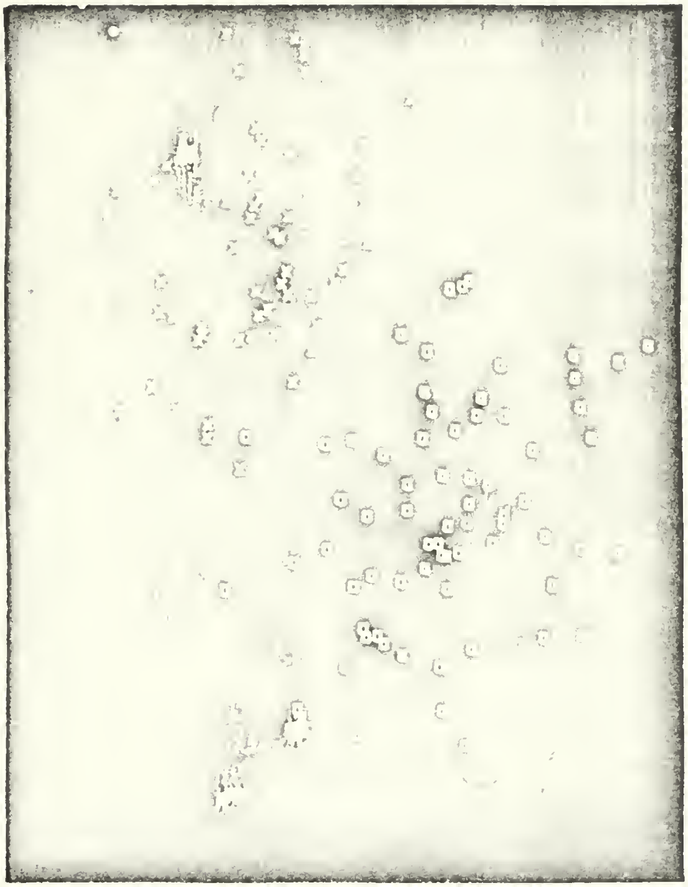

Figure IV-3-6P. CASE 3, STEP 7

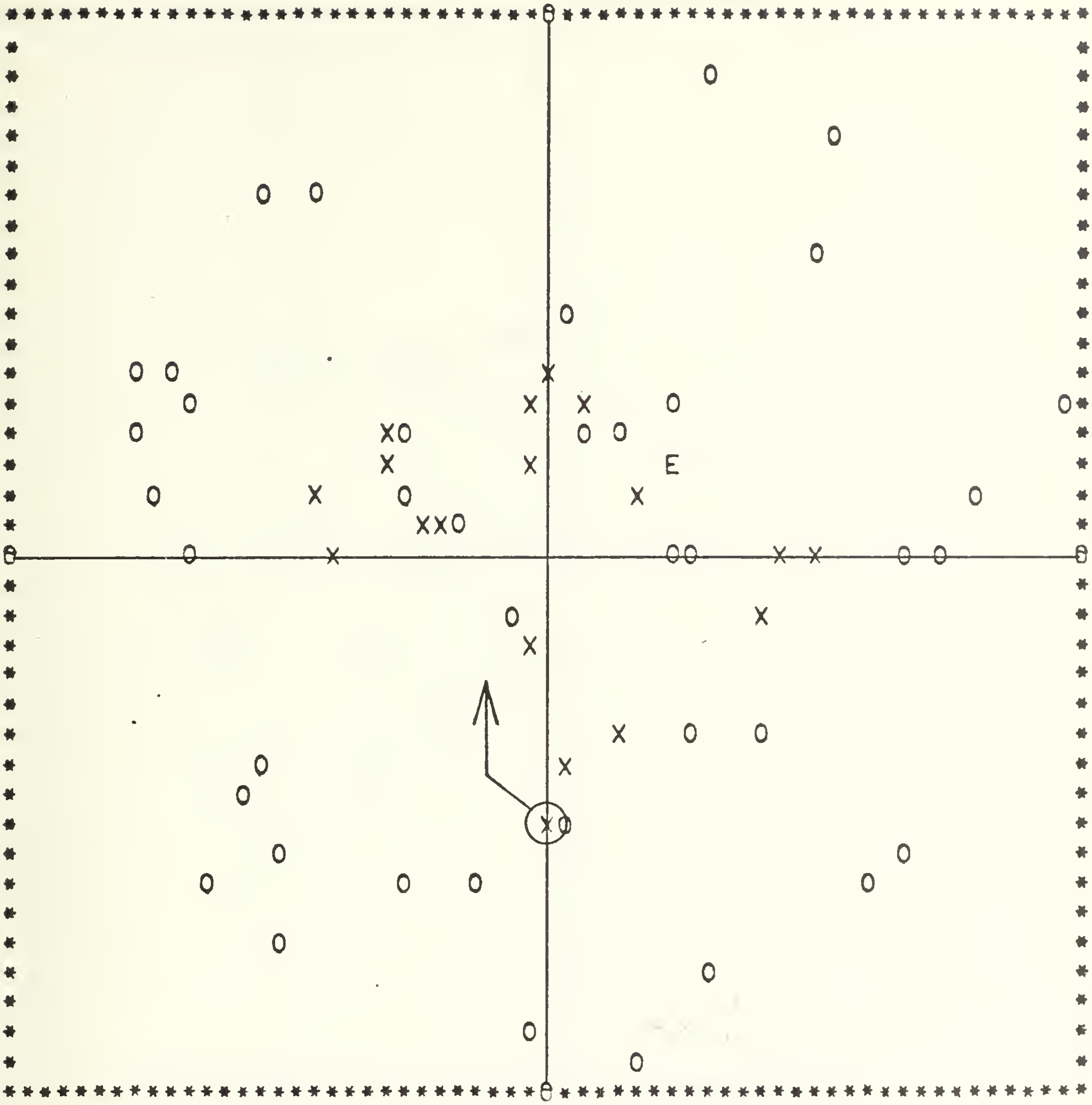

Figure IV-4-1. CASE 4, RUN 1, STEP 1

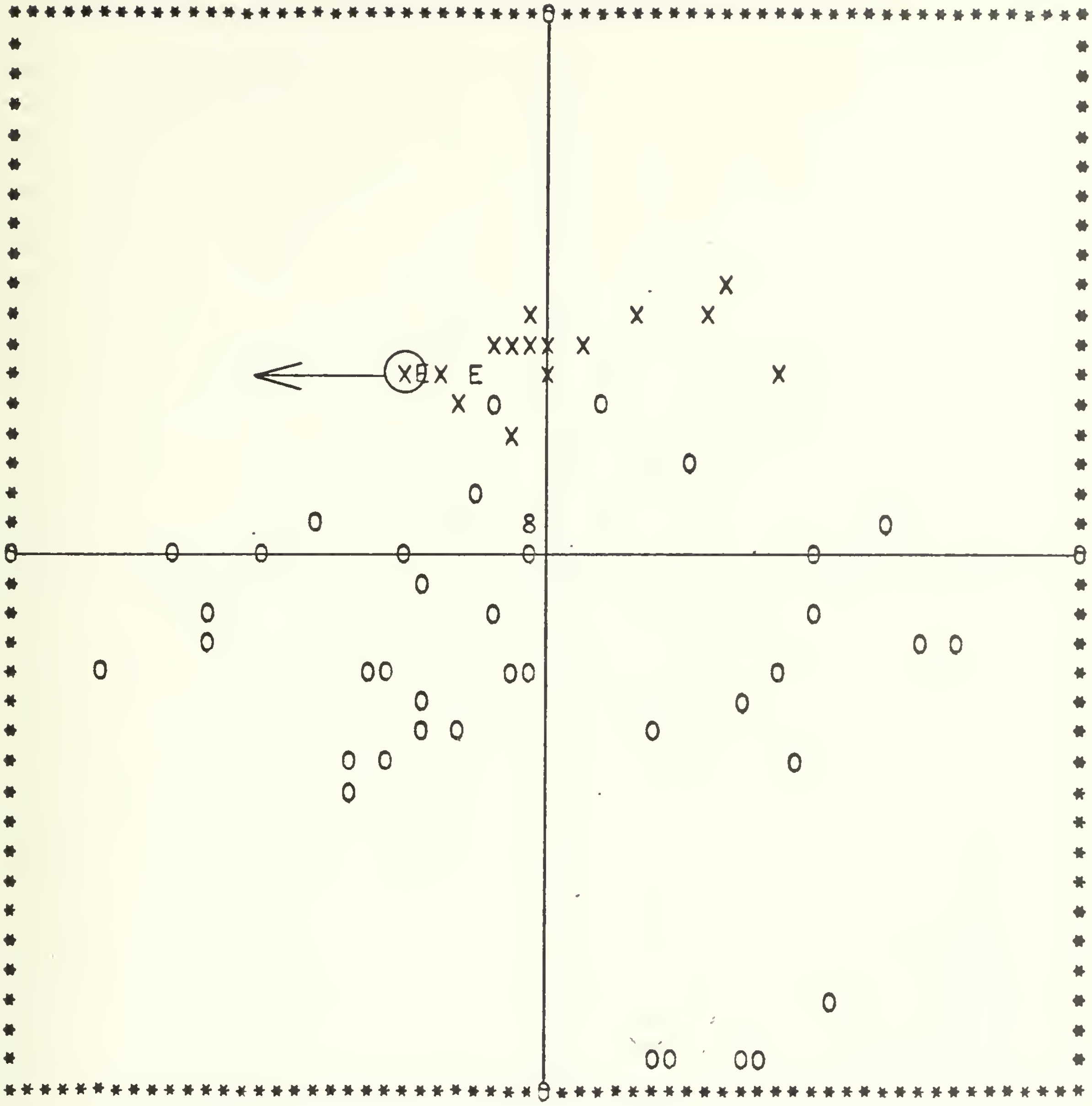

Figure IV-4-2. CASE 4, RUN 1, STEP 2

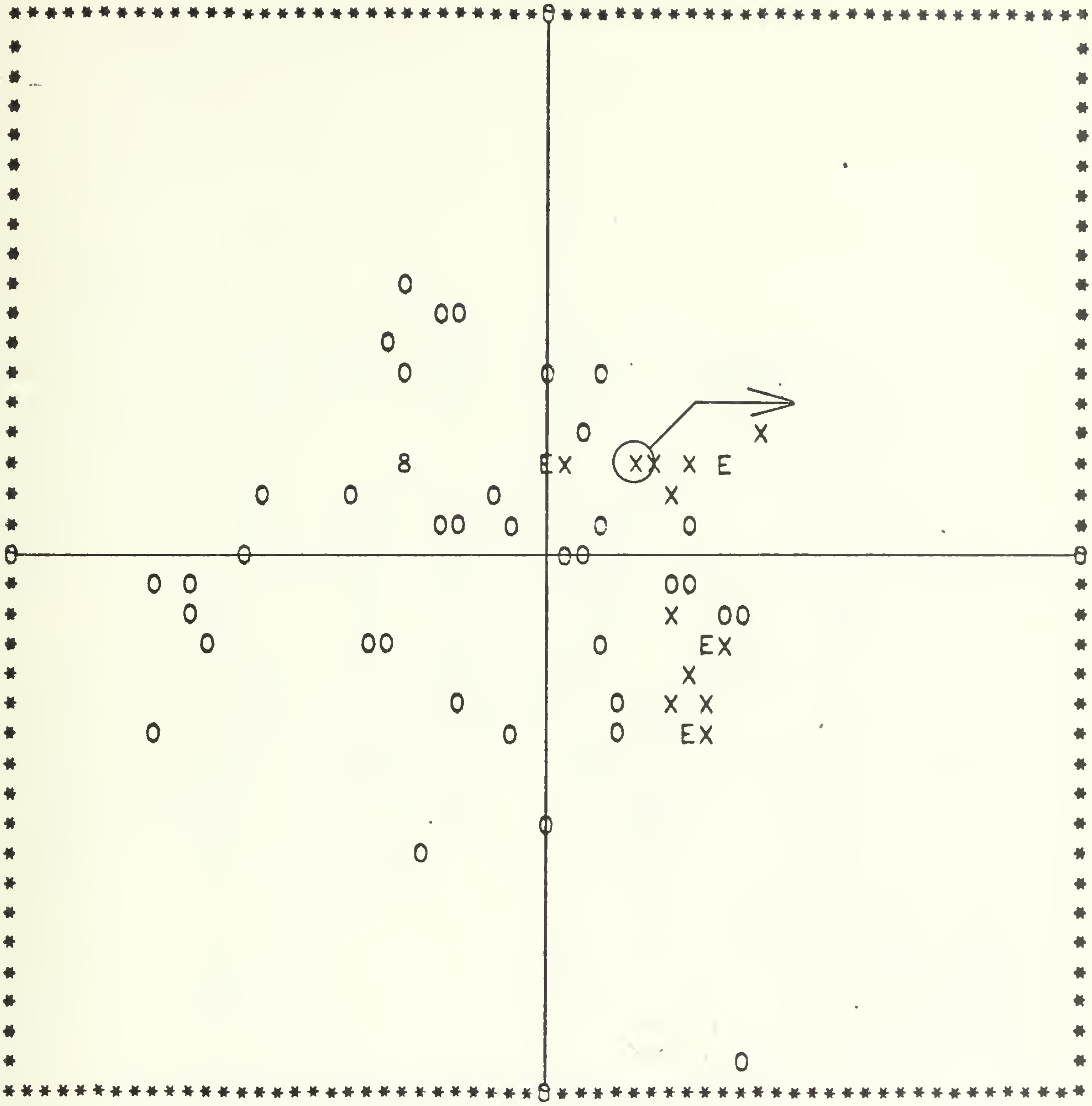

Figure IV-4-3. CASE 4, RUN 1, STEP 4

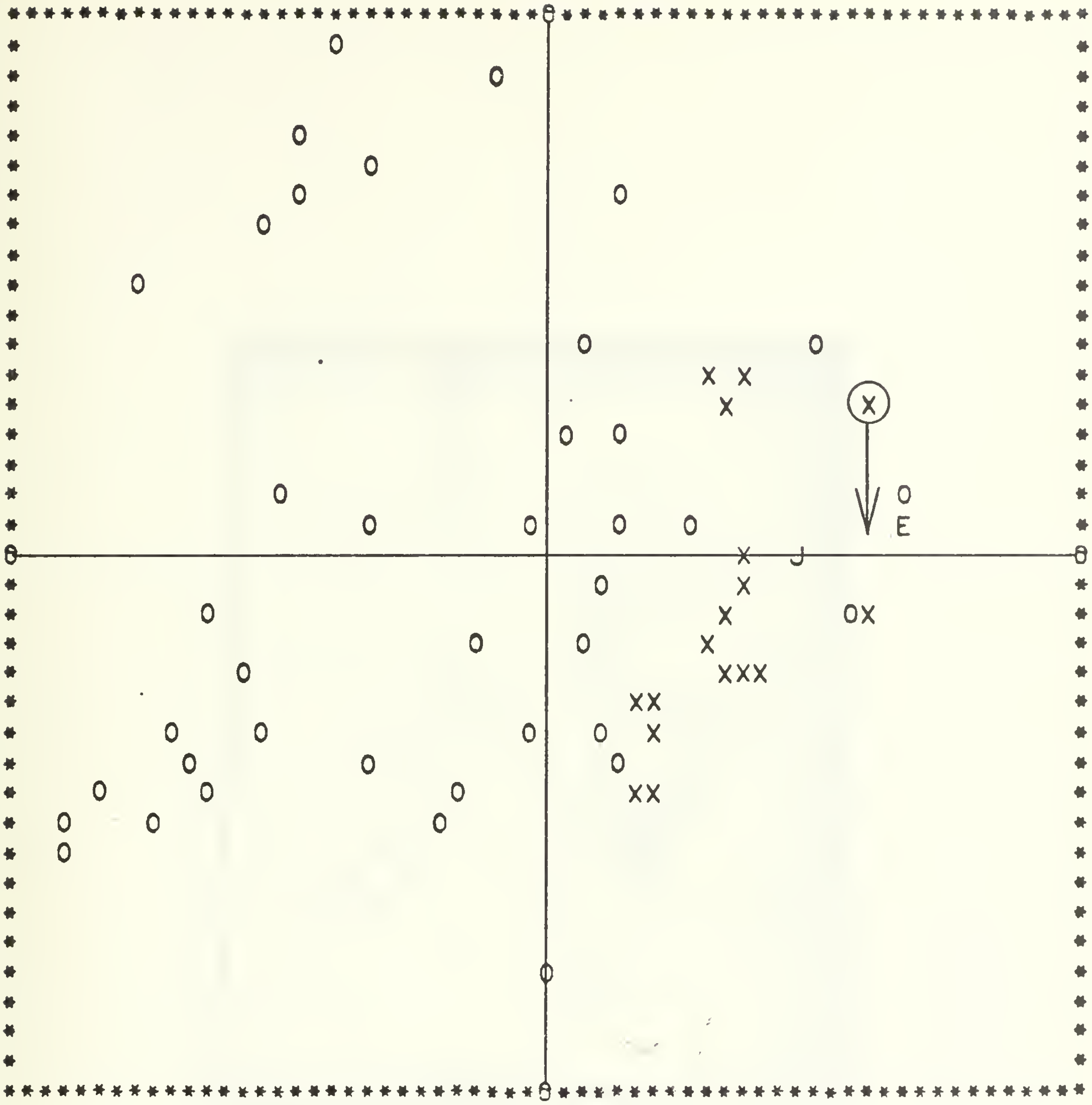

Figure IV-4-4. CASE 4, RUN 1, STEP 5

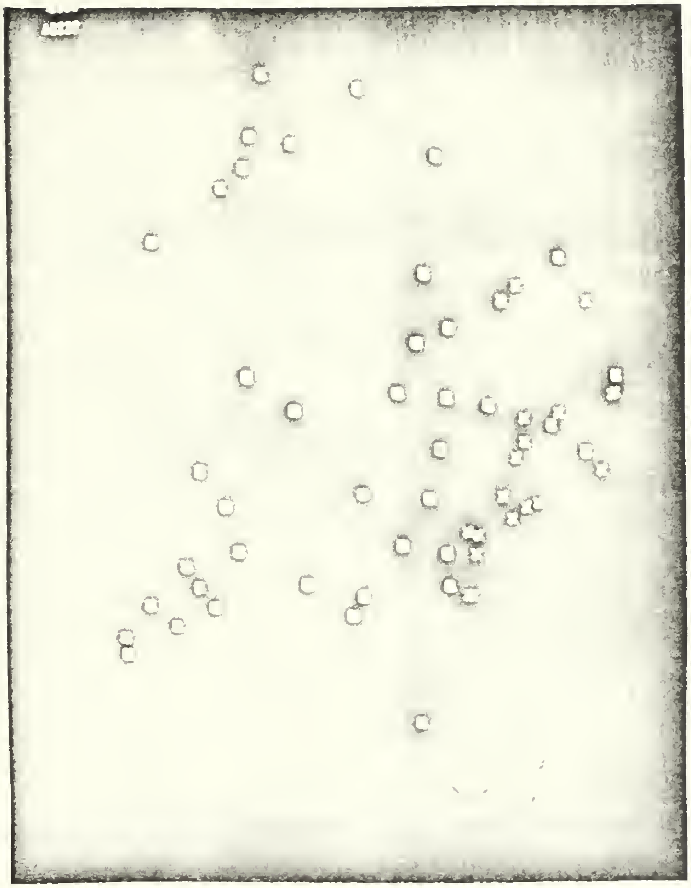

Figure IV-4-4P. CASE 4, RUN 1, STEP 5

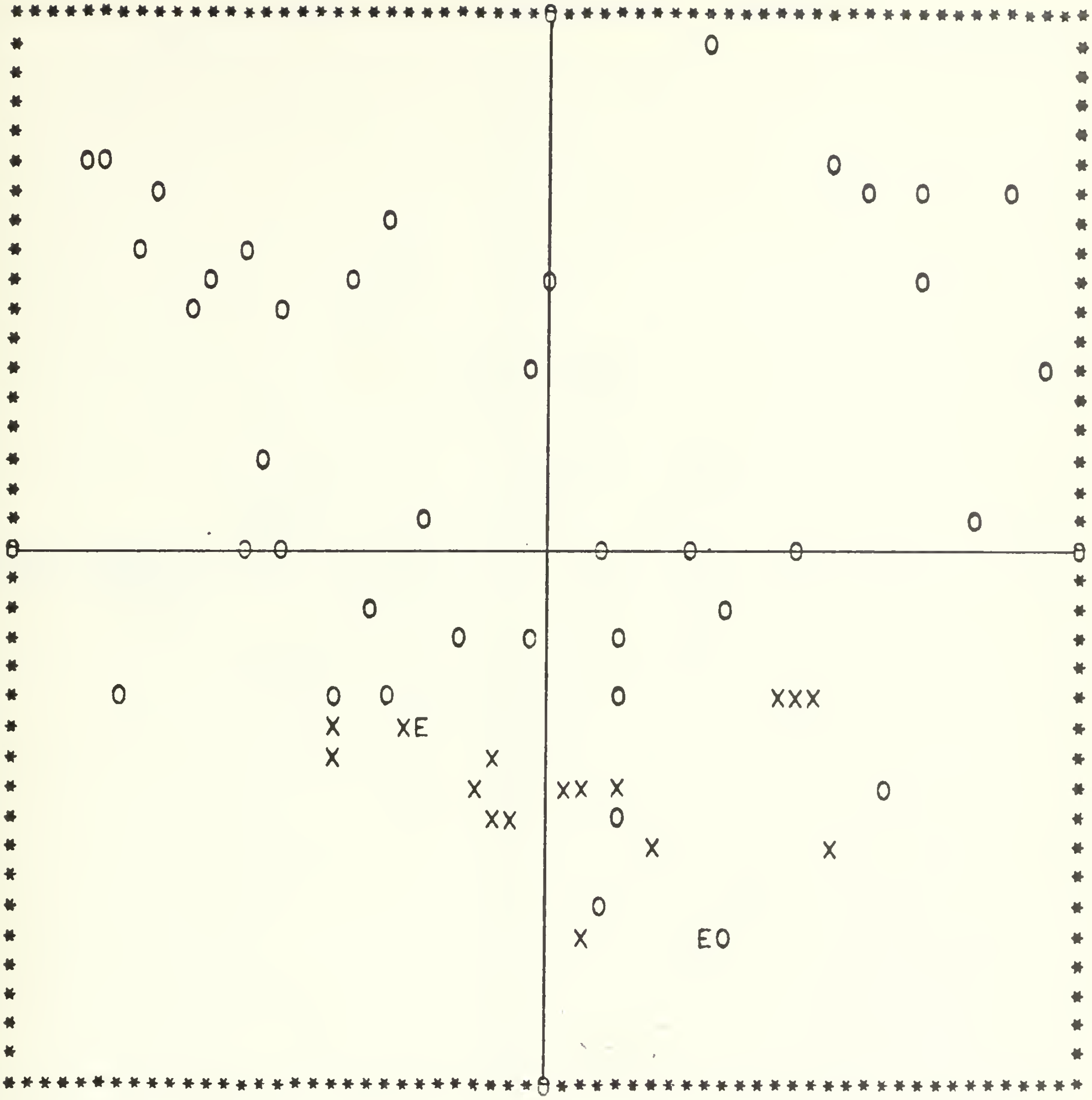

Figure IV-4-5. CASE 4, RUN 1, STEP 6

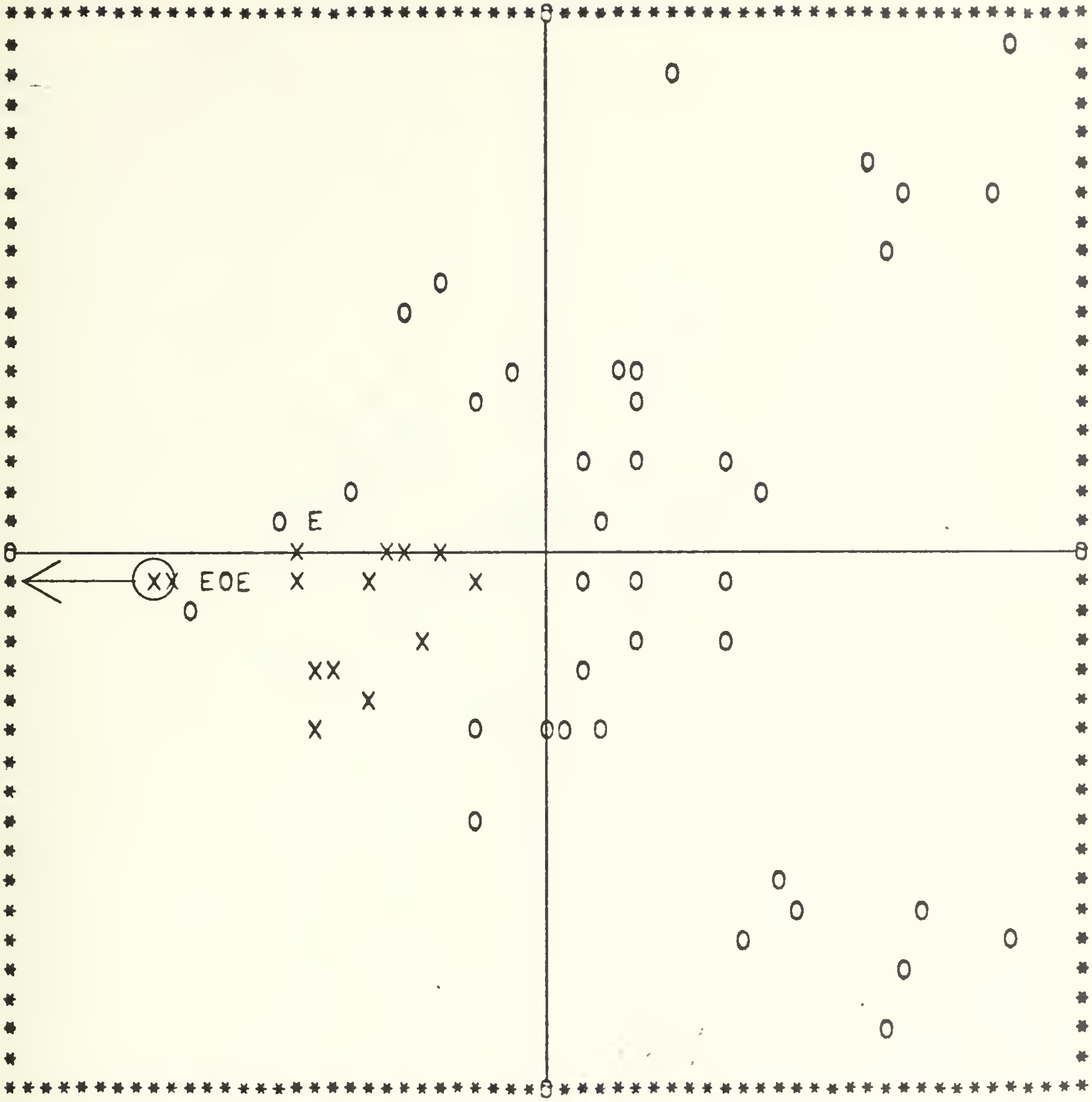

Figure IV-4-6. CASE 4, RUN 2, STEP 1

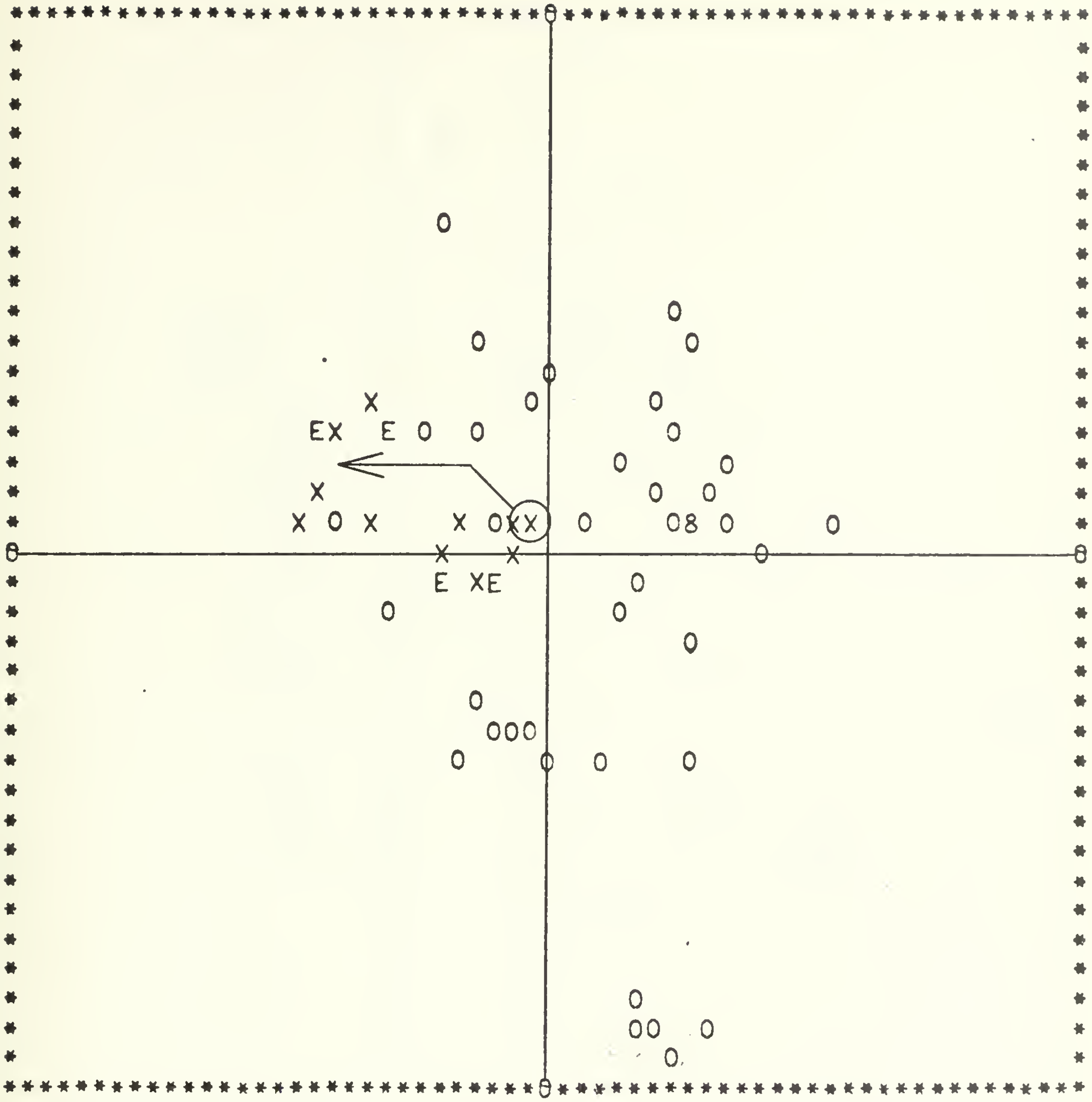

Figure IV-4-7. CASE 4, RUN 2, STEP 2

A(1) = .1822	A(5) = .4875	B(1) = .6226	B(5) =-.3942
A(2) = .4815	A(6) = .2388	B(2) =-.3740	B(6) = .4340
A(3) = .3773	A(7) = .3191	B(3) =-.0270	B(7) = .1666
A(4) = .2770	A(8) = .3455	B(4) = .3068	B(8) = .0788

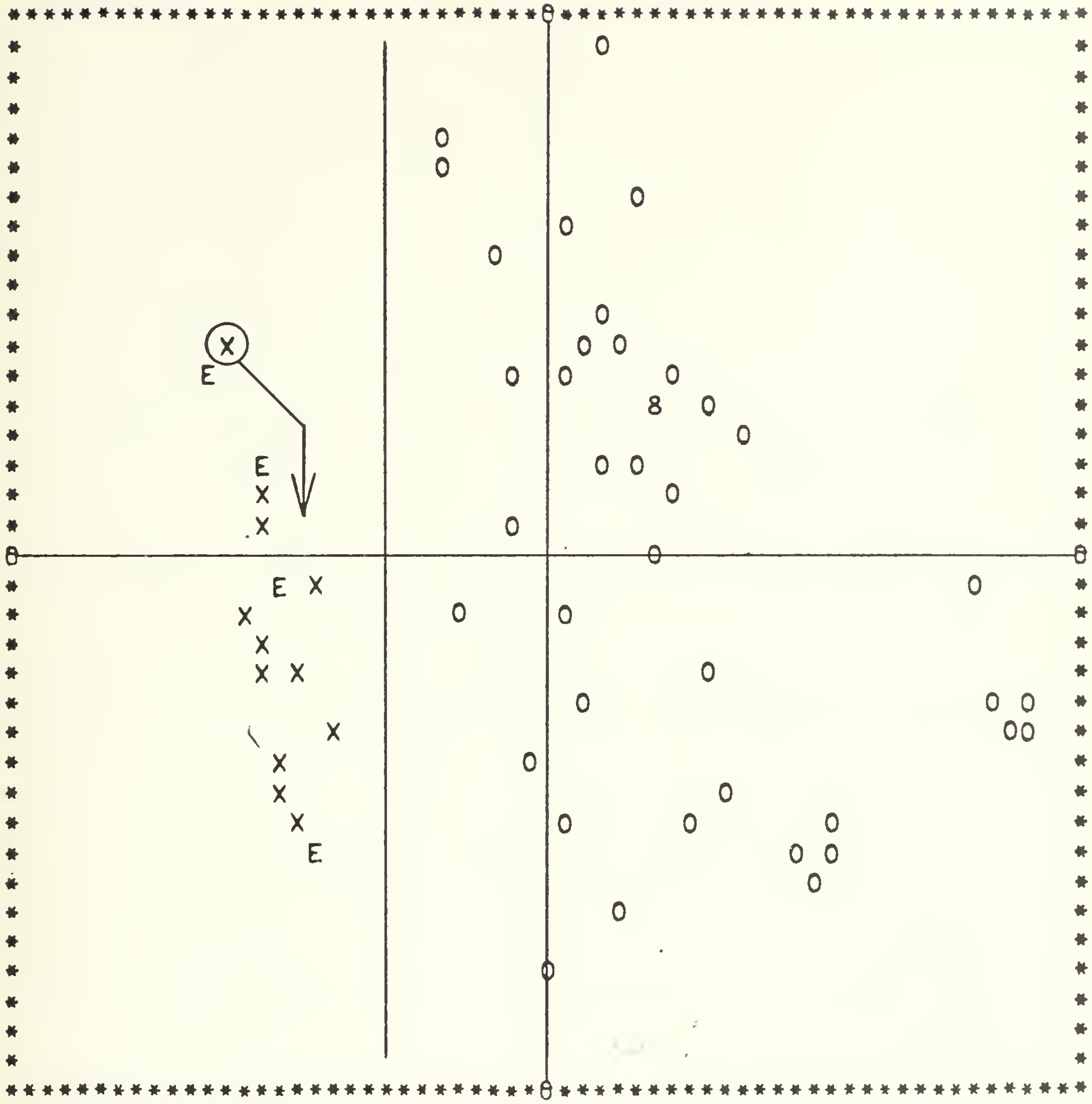

Figure IV-4-8. CASE 4, RUN 2, STEP 3

A(1) =-.2417	A(5) = .6368	B(1) = .4122	B(5) = .1287
A(2) =-.0358	A(6) = .3585	B(1) = .6317	B(6) = .0277
A(3) =-.0299	A(7) = .4150	B(3) = .4963	B(7) = .0856
A(4) =-.0608	A(8) = .4790	B(4) = .3939	B(8) = .0702

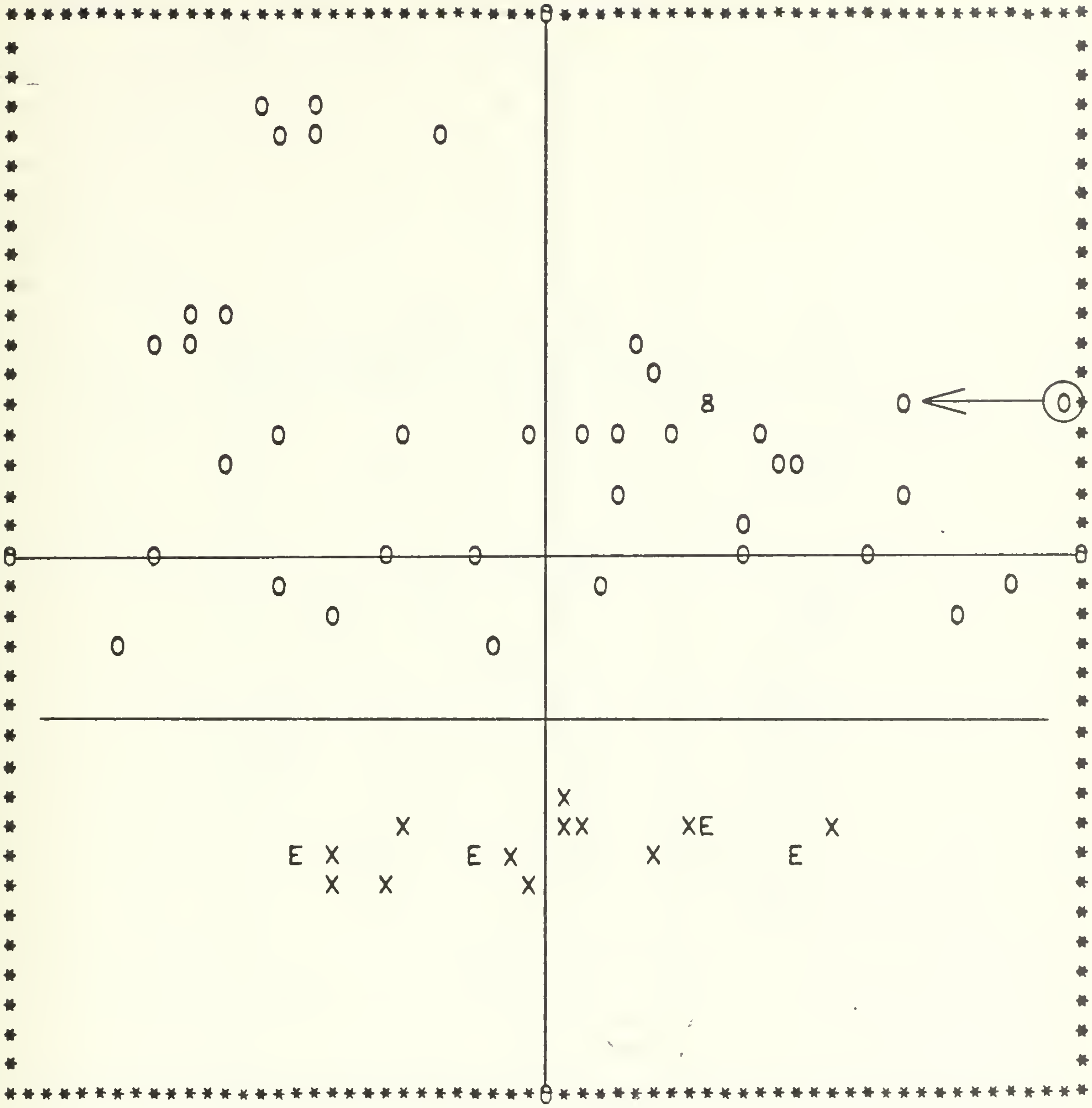

Figure IV-4-9. CASE 4, RUN 2, STEP 4

A(1) = .4496	A(5) = .0091	B(1) =-.1616	B(5) = .6497
A(2) = .6275	A(6) =-.0388	B(2) = .0812	B(6) = .3574
A(3) = .4934	A(7) = .0077	B(3) = .0621	B(7) = .4237
A(4) = .3984	A(8) =-.0193	B(4) = .0128	B(8) = .4837

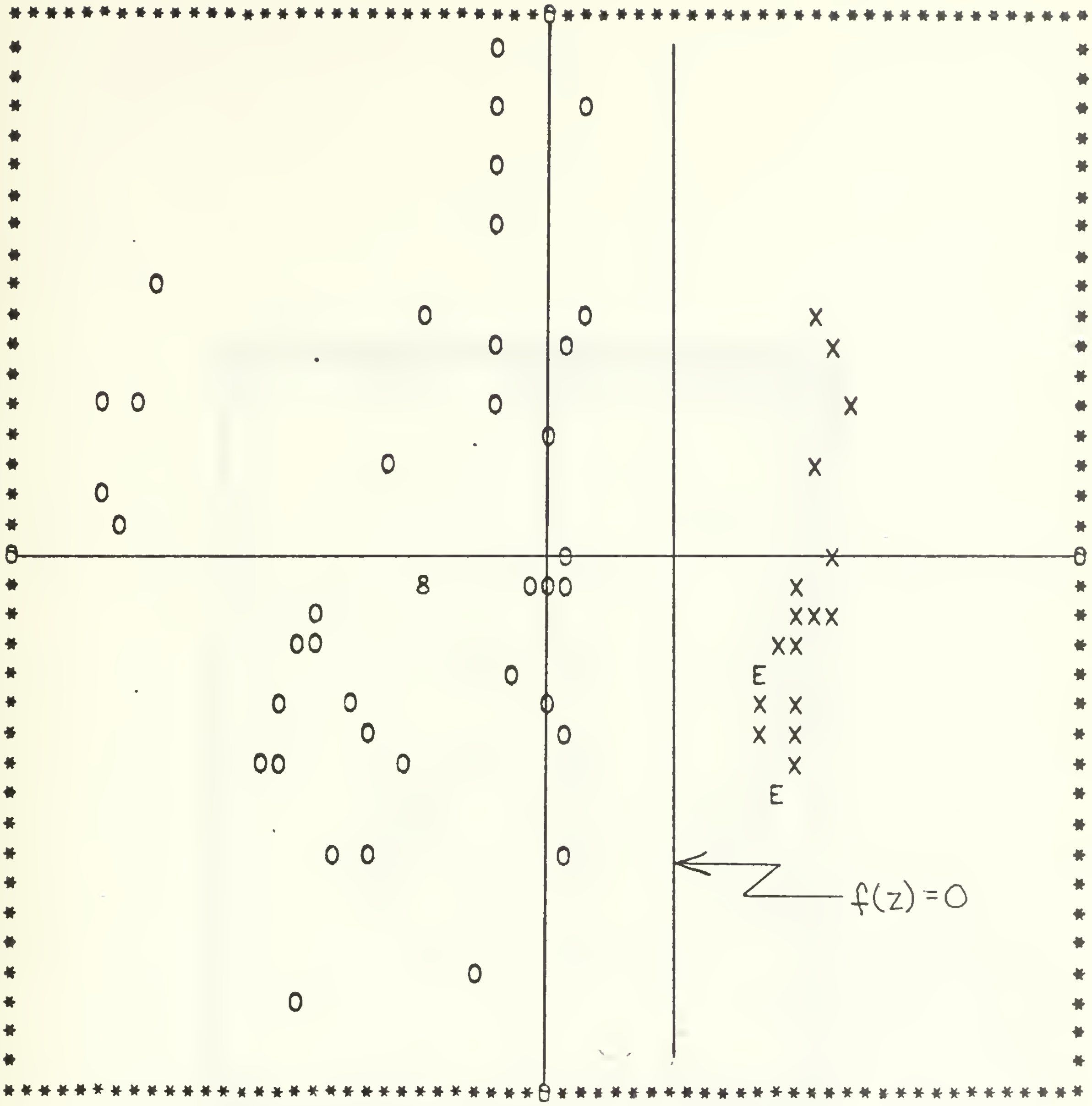

Figure IV-4-10. CASE 4, RUN 2, STEP 6

A(1) = .2460	A(5) =-.6436	B(1) = .0580	B(5) = .1556
A(2) =-.2372	A(6) =-.1104	B(2) =-.8950	B(6) = .2618
A(3) = .1528	A(7) =-.3966	B(3) = .2344	B(7) = .1001
A(4) = .2460	A(8) =-.4644	B(4) =-.1534	B(8) = .1202

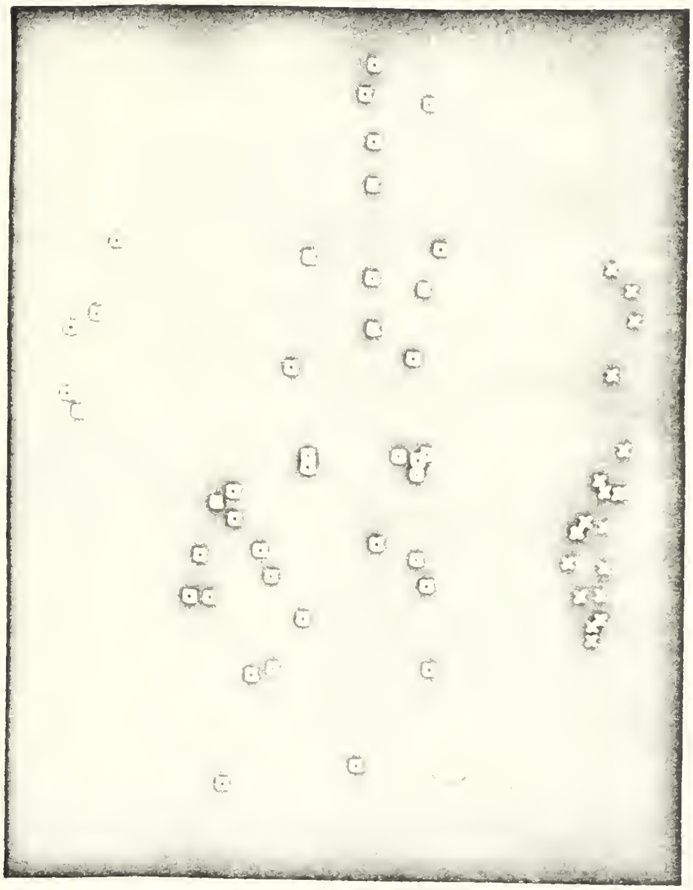

Figure IV-4-10P. CASE 4, RUN 2, STEP 6

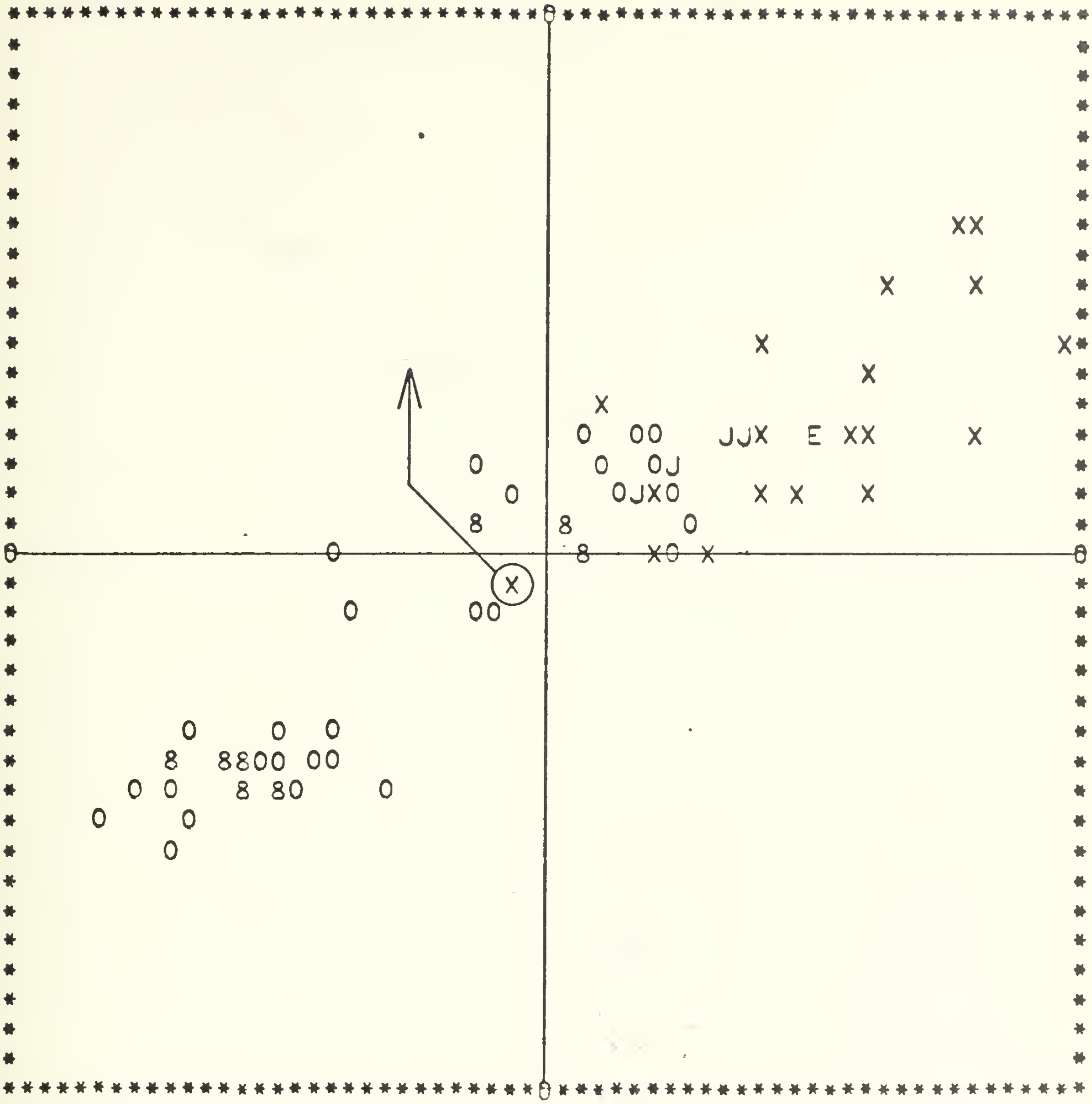

Figure IV-5-1. CASE 5, STEP 1

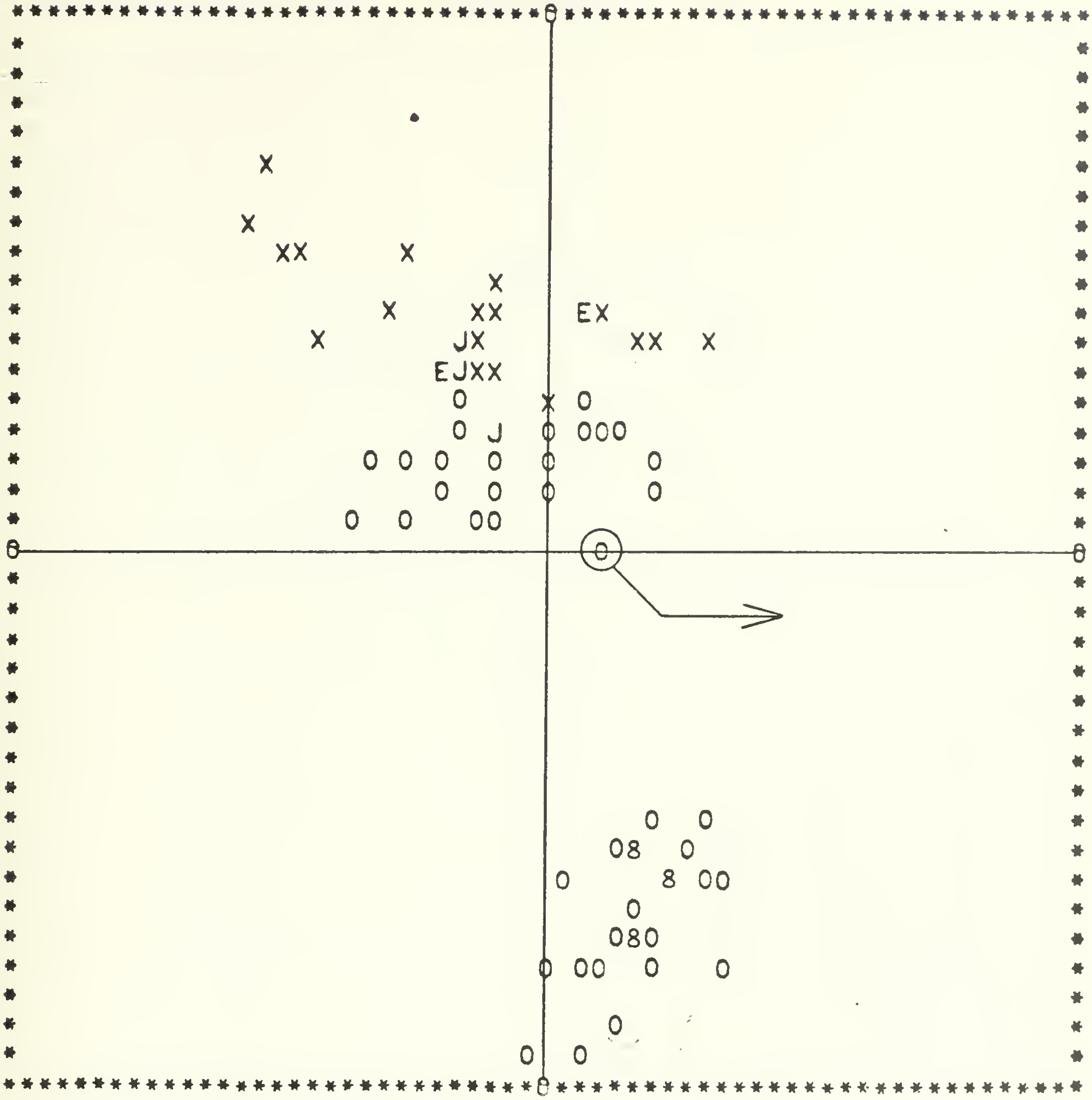

Figure IV-5-2. CASE 5, STEP 2

A(1) =-.7748	B(1) =-.2831
A(2) = .1863	B(2) =-.6366
A(3) =-.1346	B(3) = .7173
A(4) = .5890	B(4) =-.0072

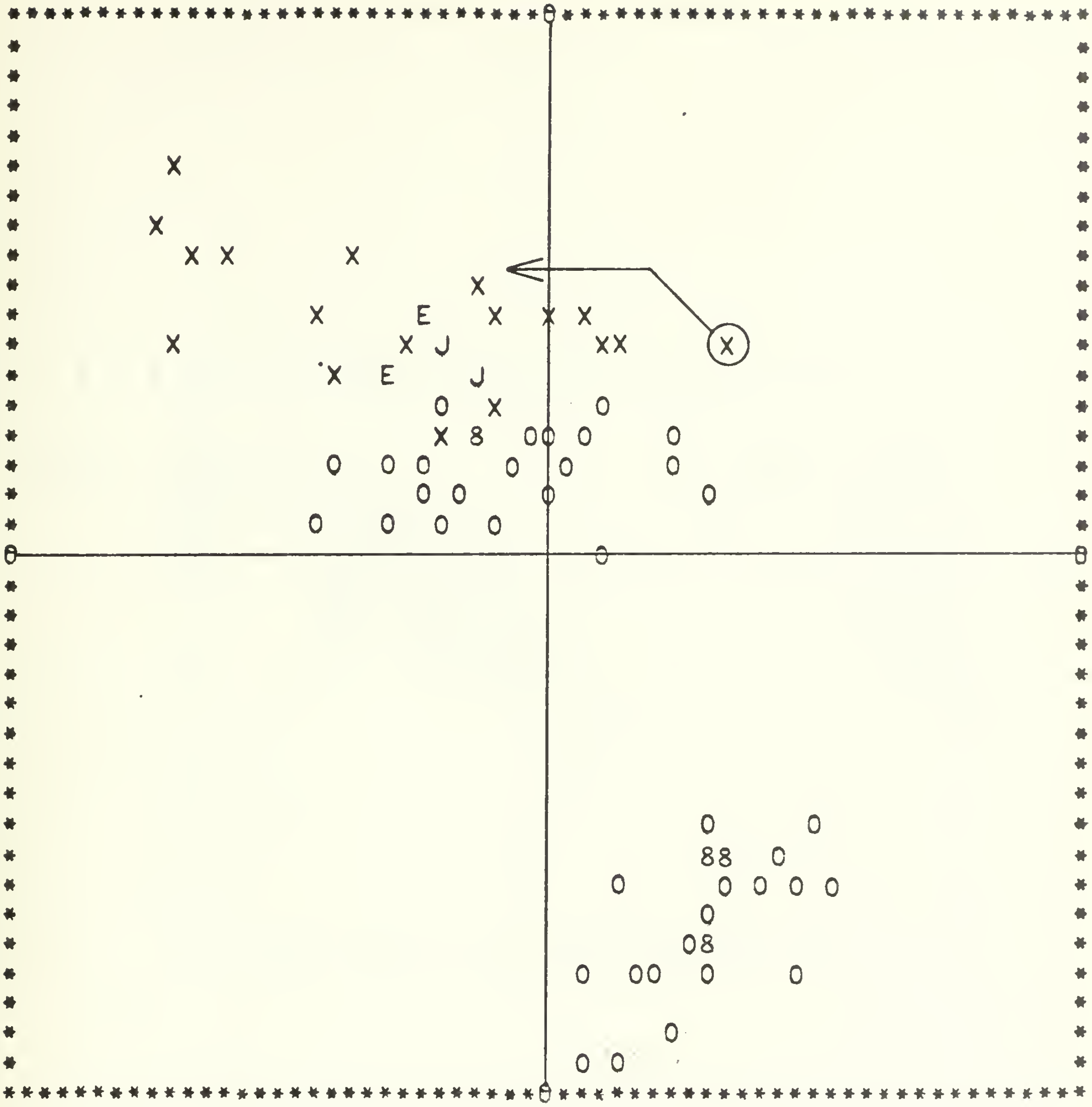

Figure IV-5-3. CASE 5, STEP 3

```
A(1) = .1429    B(1) =-.8365

A(2) = .5127    B(2) =-.0491

A(3) =-.8460    B(3) =-.1513

A(4) = .0318    B(4) = .5244
```

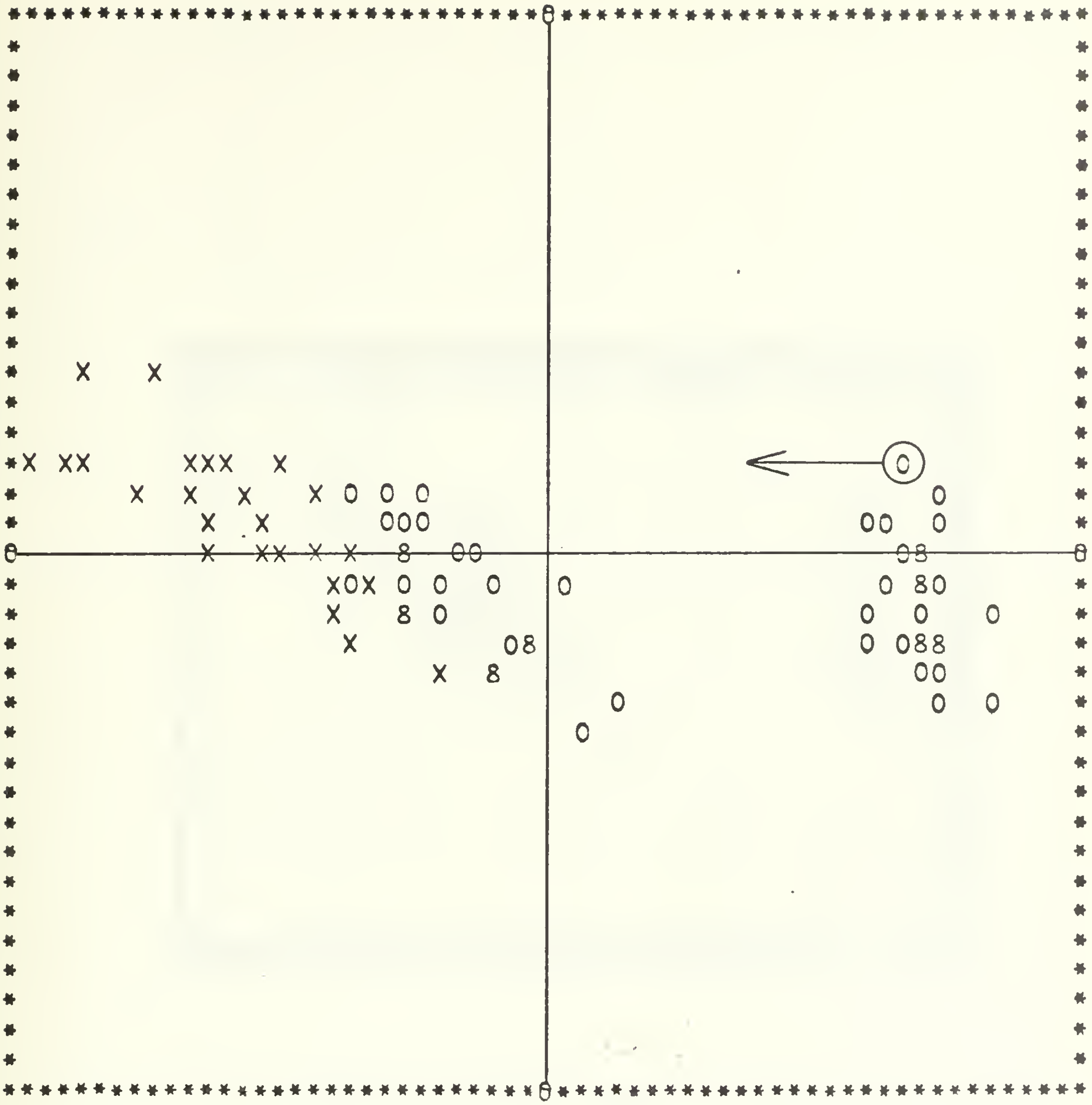

Figure IV-5-4. CASE 5, STEP 5

A(1) =-.0571 B(1) = .8120

A(2) = .1522 B(2) = .5511

A(3) =-.8827 B(3) =-.0502

A(4) =-.4409 B(4) = .1856

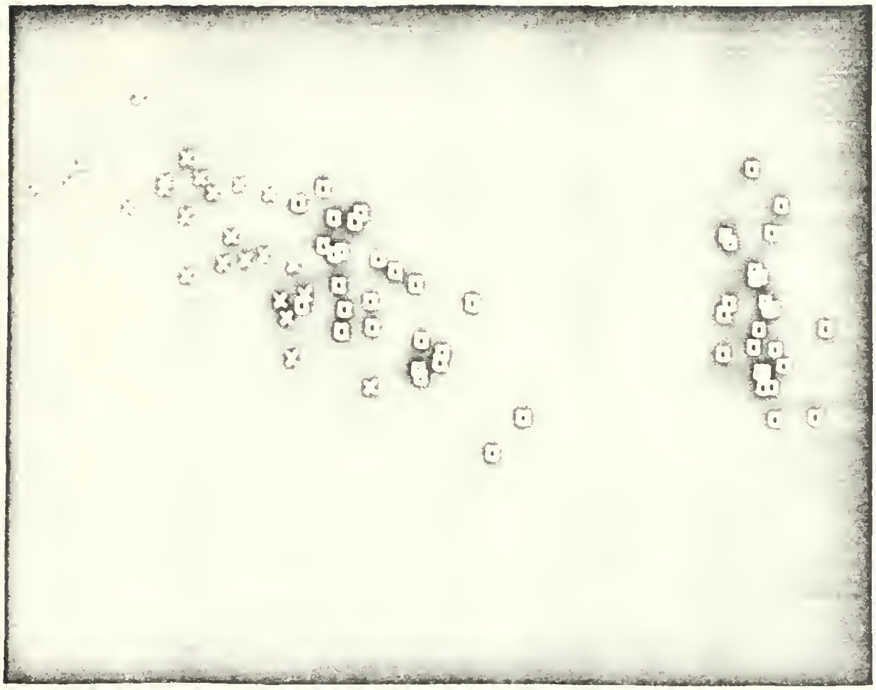

Figure IV-5-4P, CASE 5, STEP 5

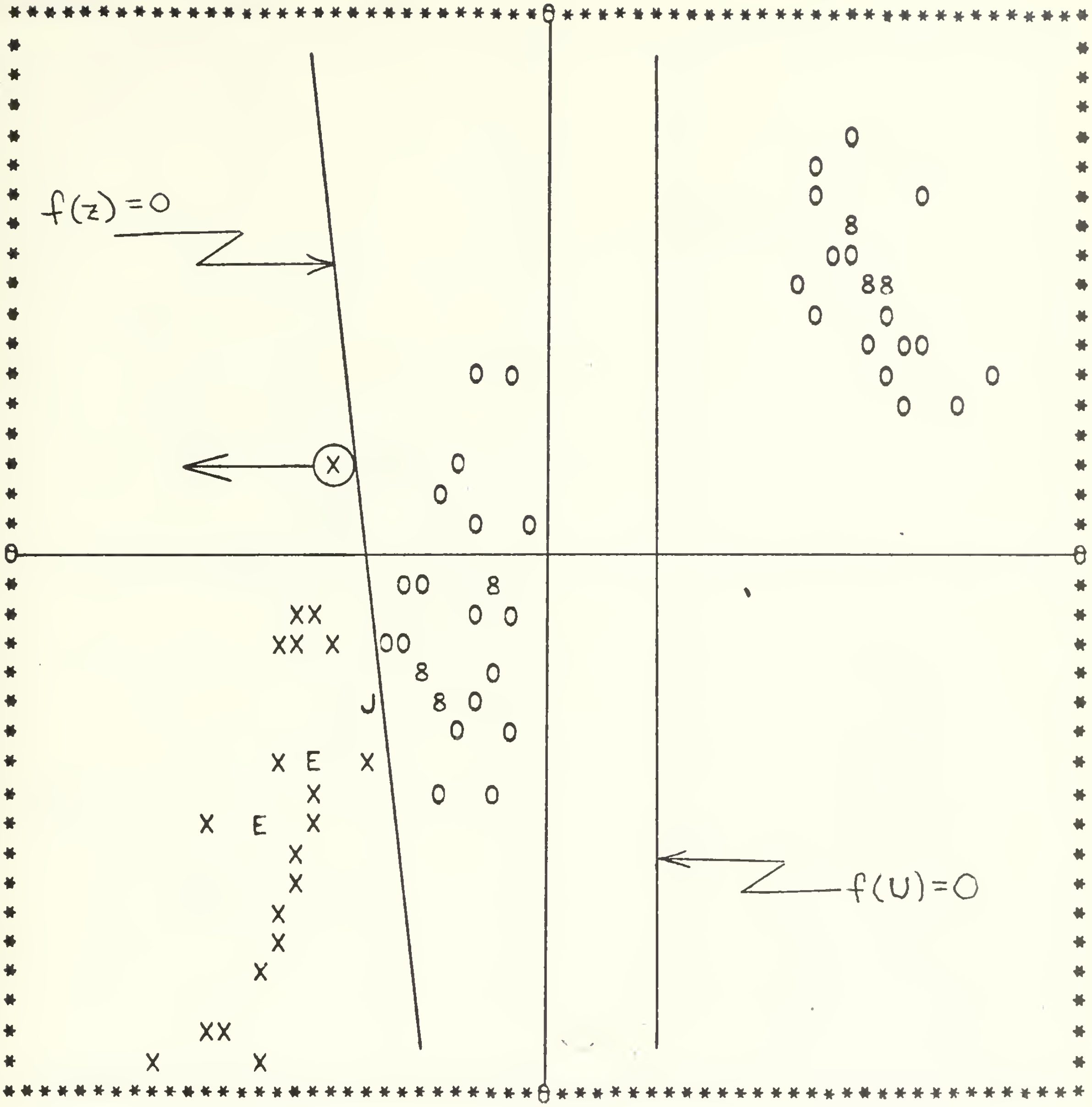

Figure IV-5-5. CASE 5, STEP 6

A(1) = .4220	B(1) =-.8145
A(2) = .3291	B(2) =-.2528
A(3) =-.7281	B(3) =-.5022
A(4) =-.4284	B(4) =-.1429

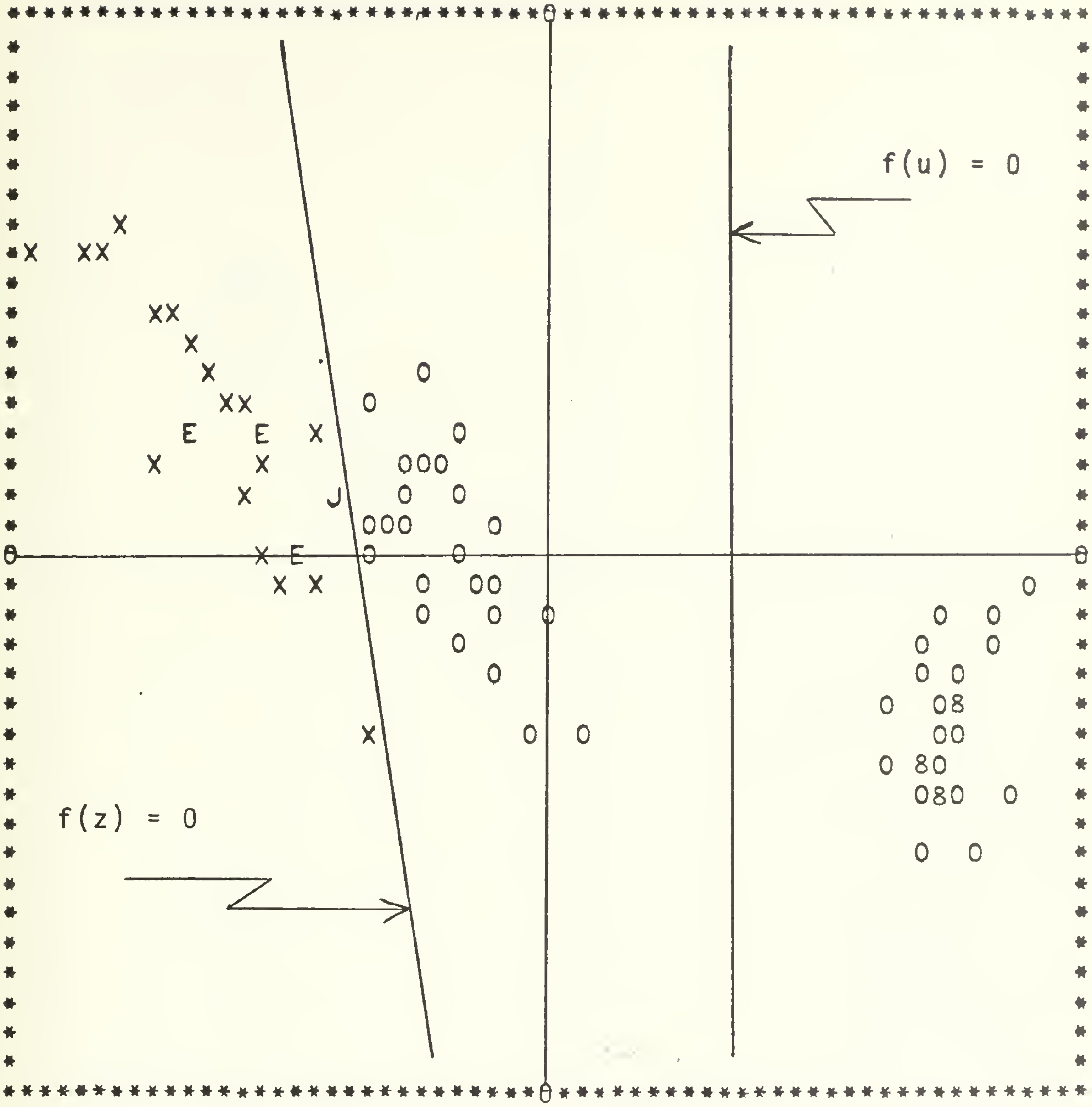

Figure IV-5-6. CASE 5, STEP 12

A(1) = .0767	B(1) = .9772
A(2) = .3515	B(2) = .1482
A(3) =-.8091	B(3) = .1515
A(4) =-.4647	B(4) = .0095

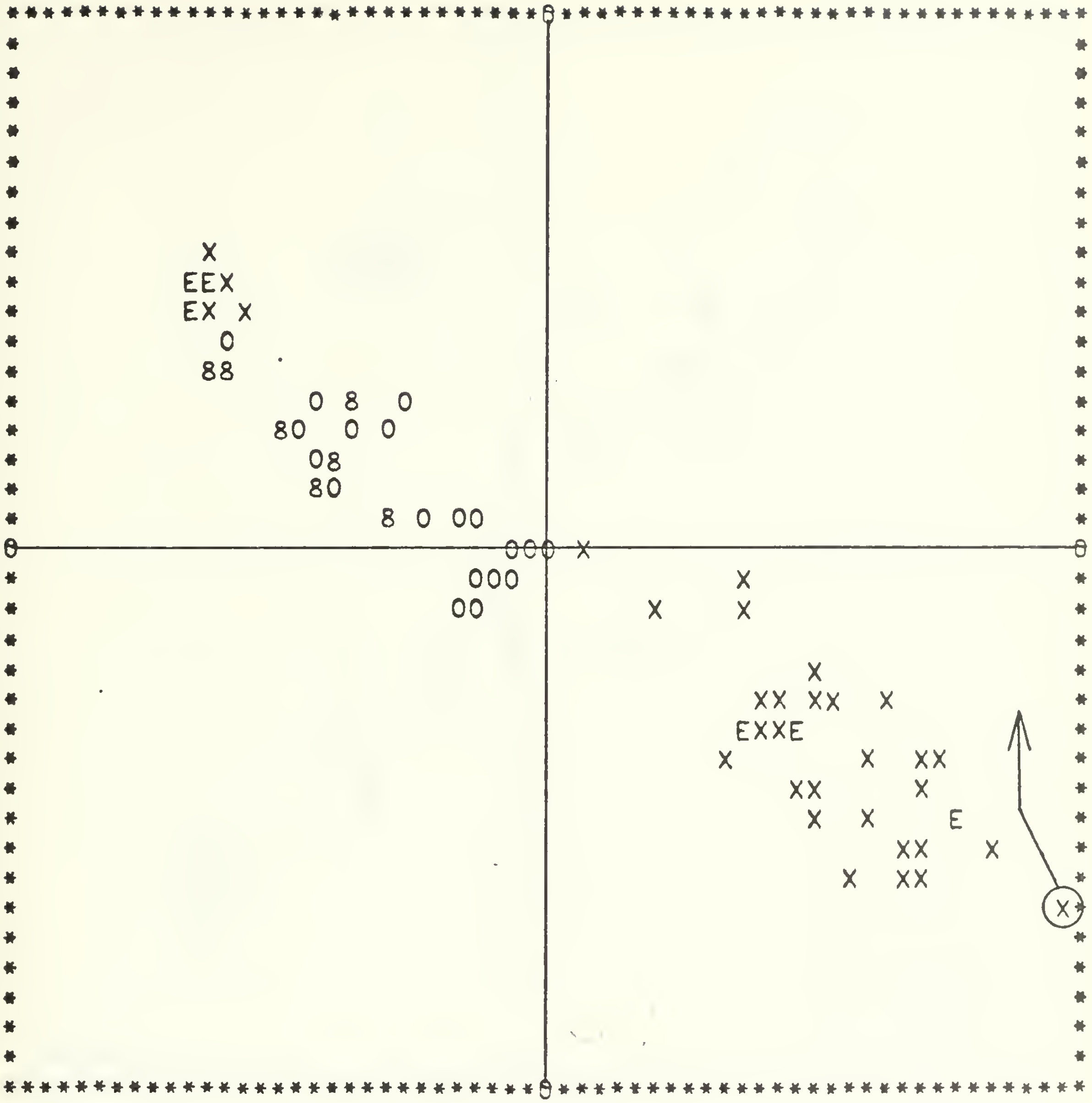

Figure IV-6-1. CASE 6, RUN 1, STEP 1

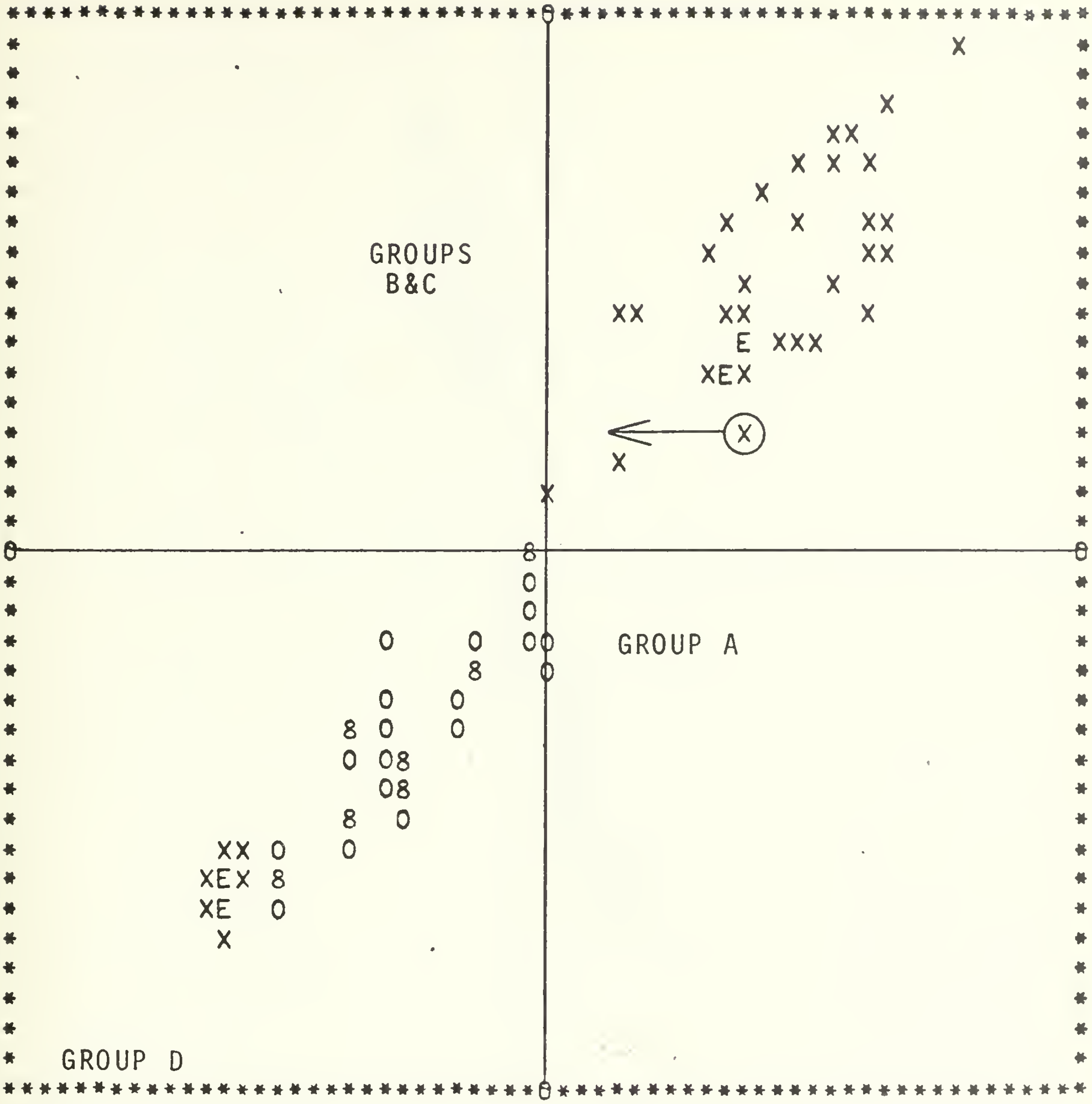

Figure IV-6-2. CASE 6, RUN 1, STEP 2

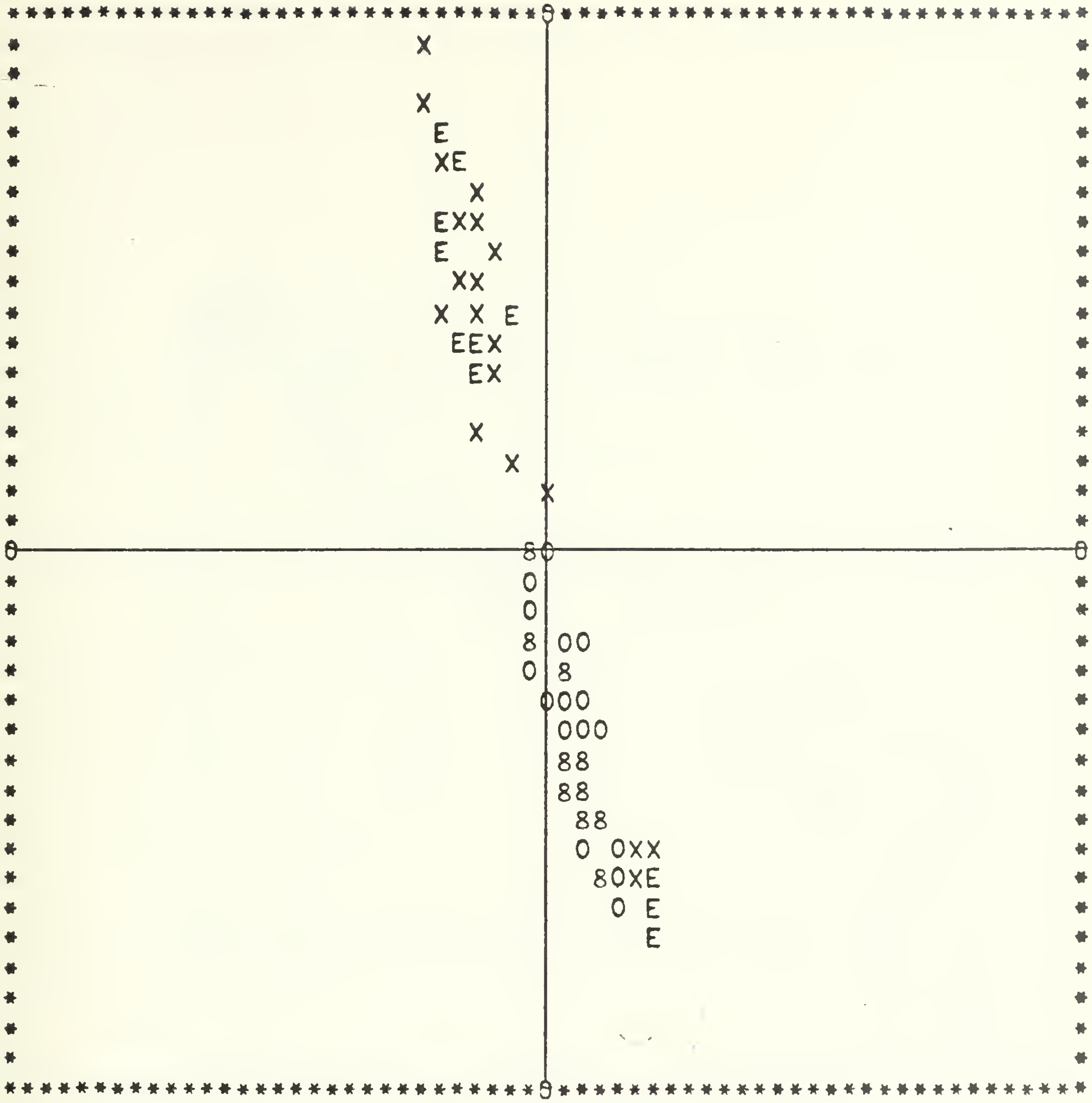

Figure IV-6-3. CASE 6, RUN 2, STEP 4

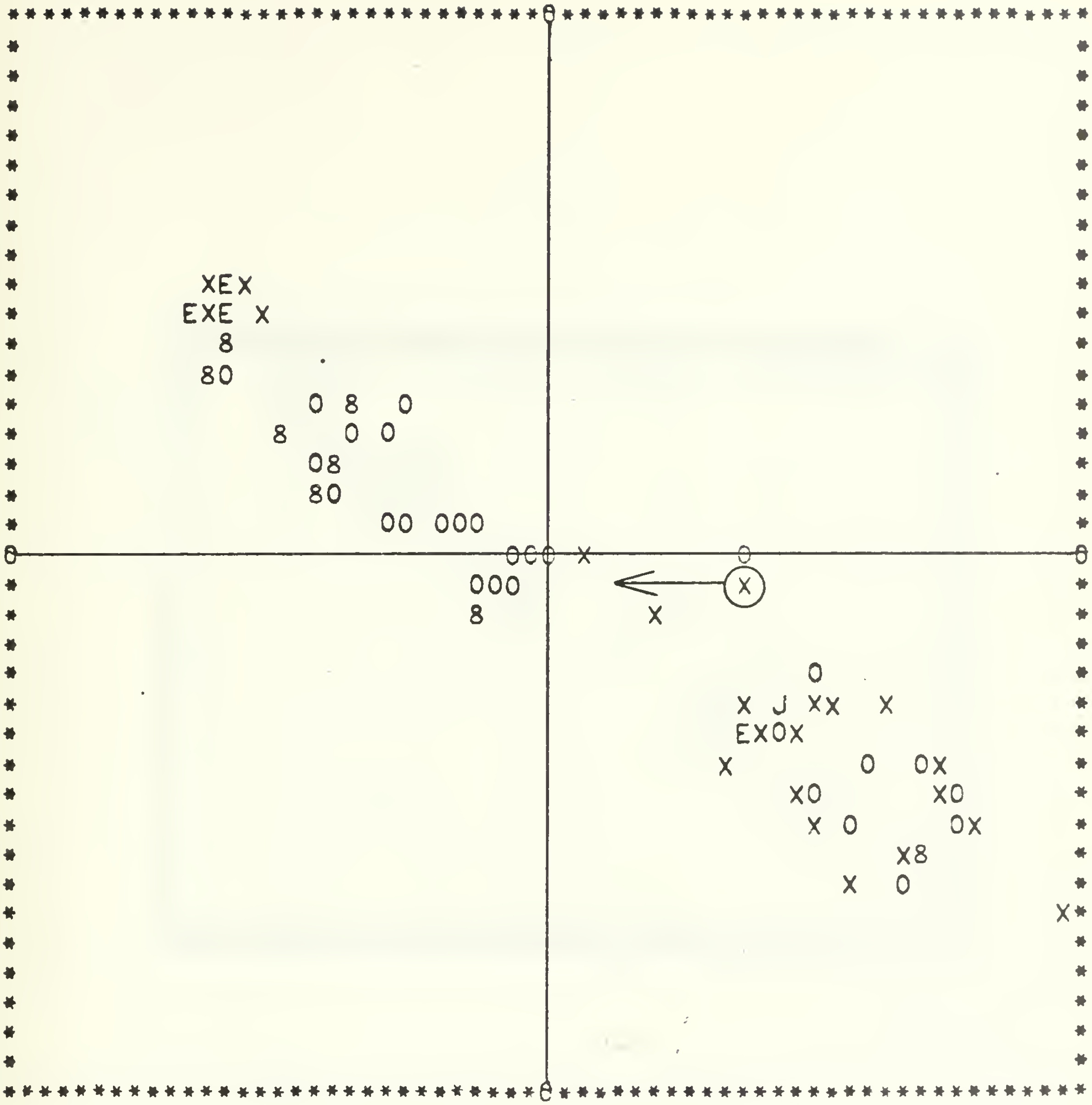

Figure IV-6-4. CASE 6, RUN 2, STEP 1

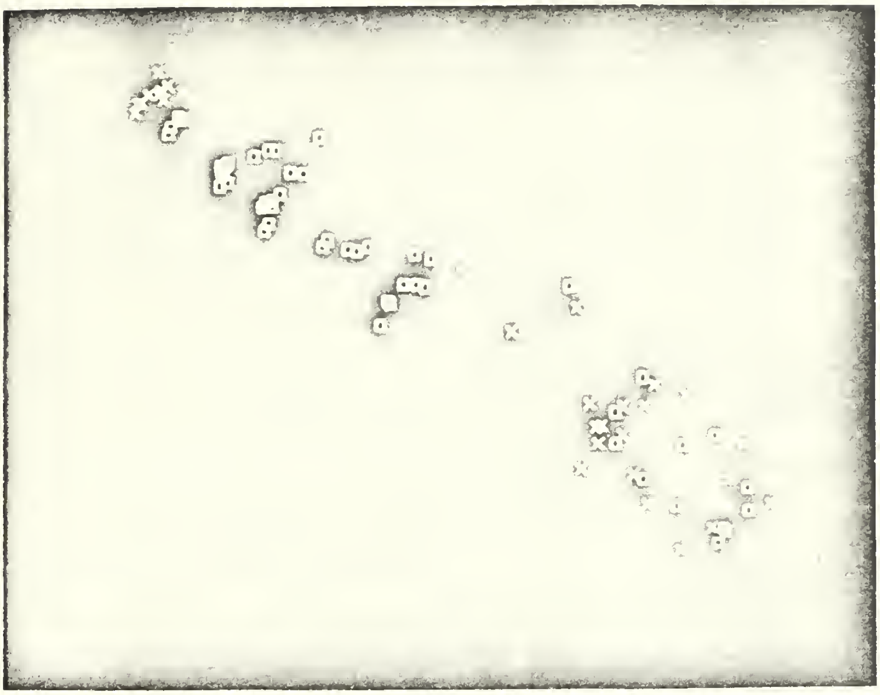

Figure IV-6-4P. CASE 6, RUN 2, STEP 1

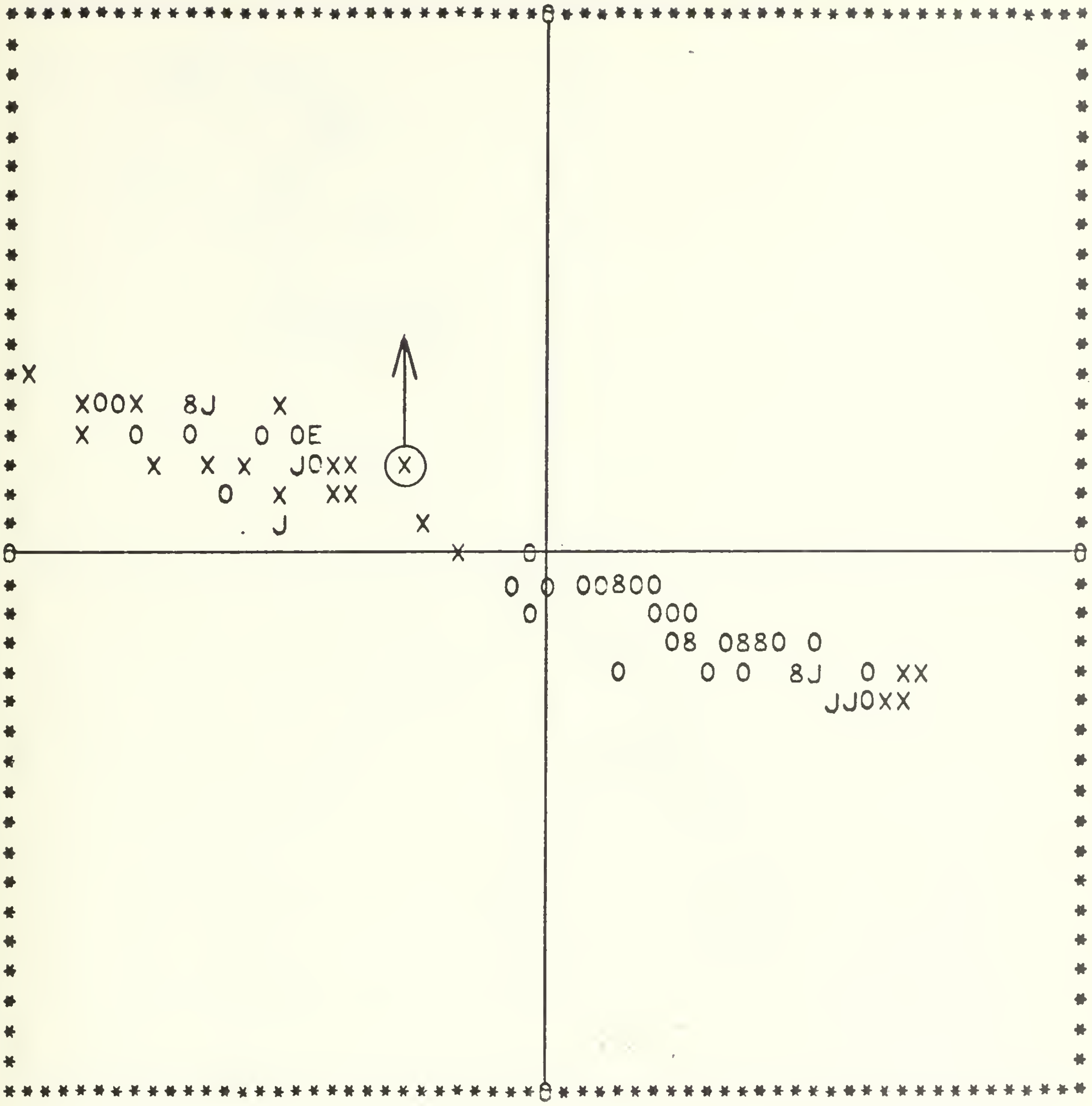

Figure IV-6-5. CASE 6, RUN 2, STEP 2

A(1) =-.8954	B(1) =-.3364
A(2) = .0121	B(2) = .0741
A(3) = .0268	B(3) =-.2296
A(4) = .0692	B(4) =-.0380
A(5) =-.2031	B(5) = .8498
A(6) =-.3890	B(6) = .3241

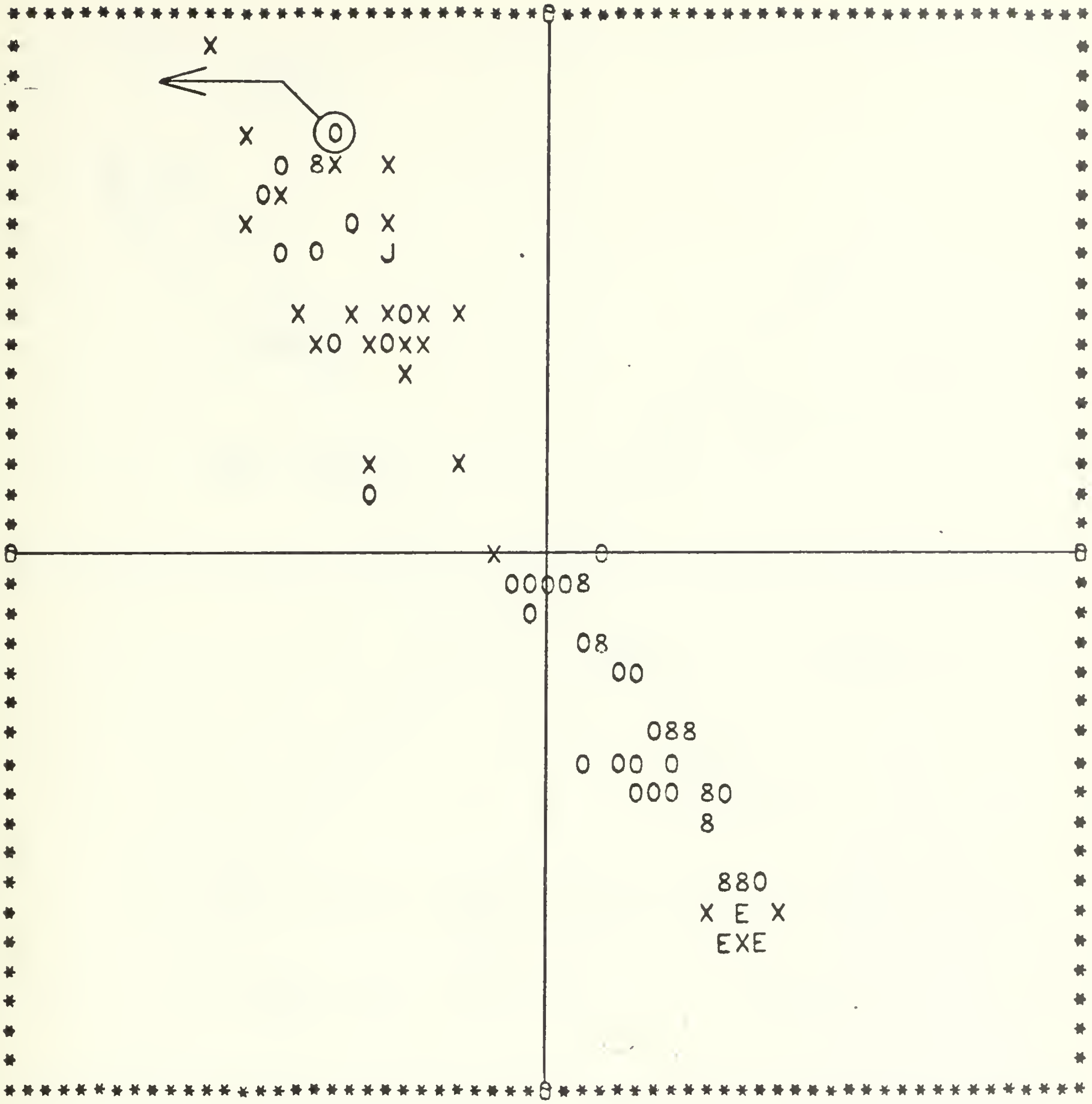

Figure IV-6-6. CASE 6, RUN 2, STEP 12

A(1) =-.9019	B(1) =-.3386
A(2) = .0130	B(2) = .0193
A(3) = .0246	B(3) = .0578
A(4) = .0703	B(4) = .0842
A(5) =-.1947	B(5) =-.1762
A(6) =-.3781	B(6) = .9184

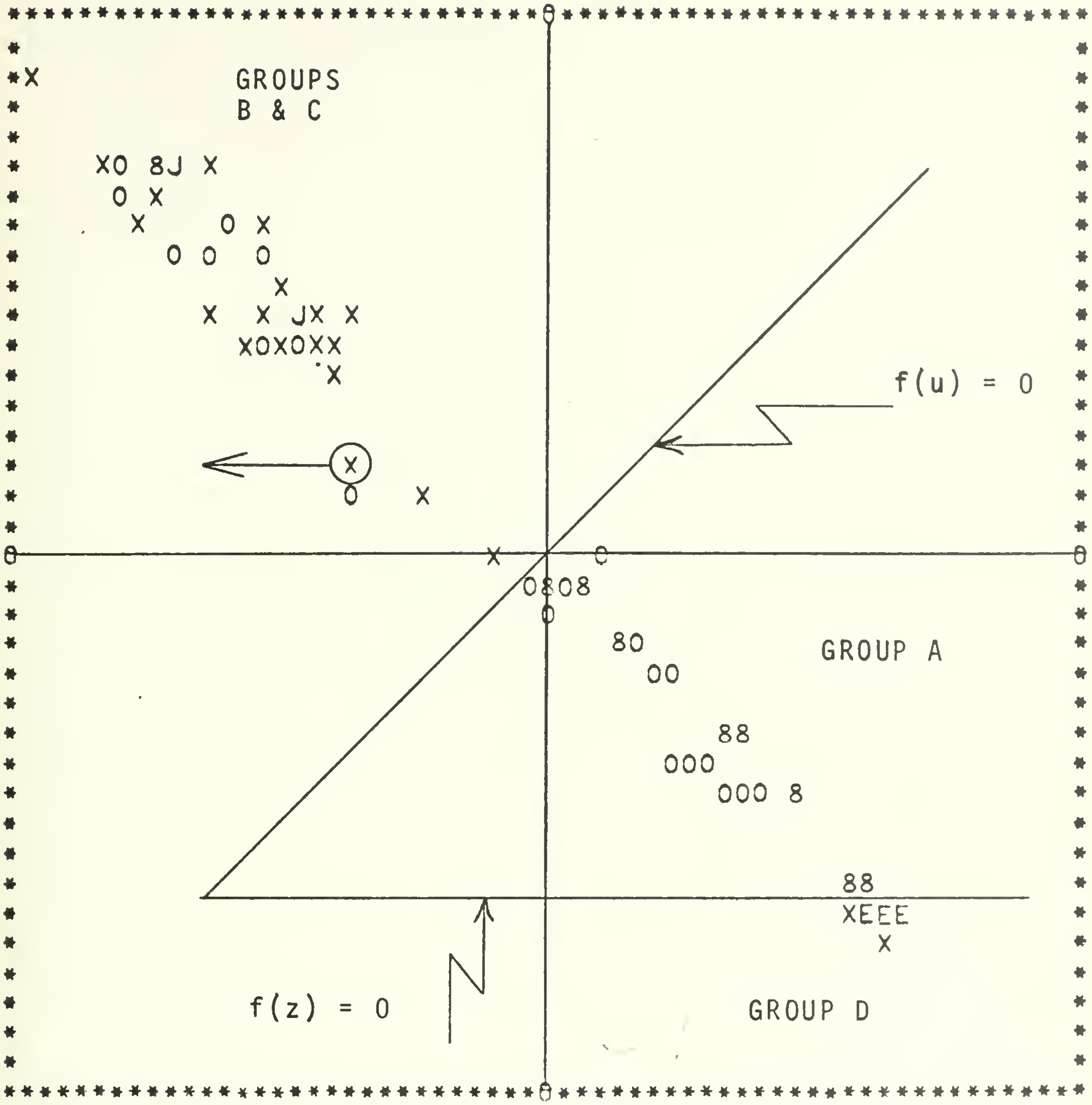

Figure IV-6-7. CASE 6, RUN 2, STEP 13

A(1) =-.6549	B(1) =-.3511
A(2) = .0030	B(2) = .0016
A(3) =-.0039	B(3) = .0134
A(4) = .0252	B(4) = .0384
A(5) =-.0953	B(5) =-.1795
A(6) =-.7492	B(6) = .9181

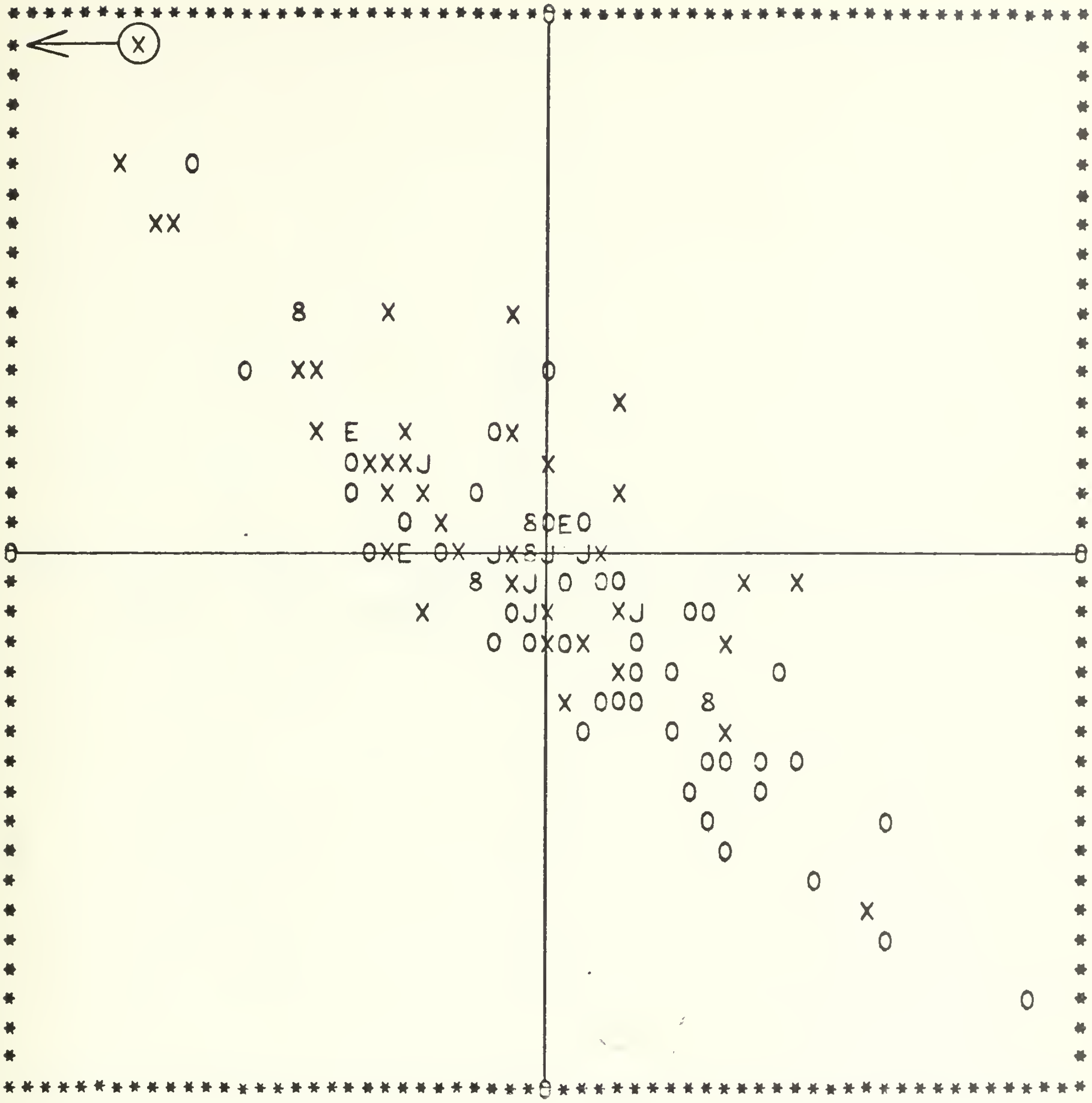

Figure IV-7-1. CASE 7, RUN 1, STEP 1

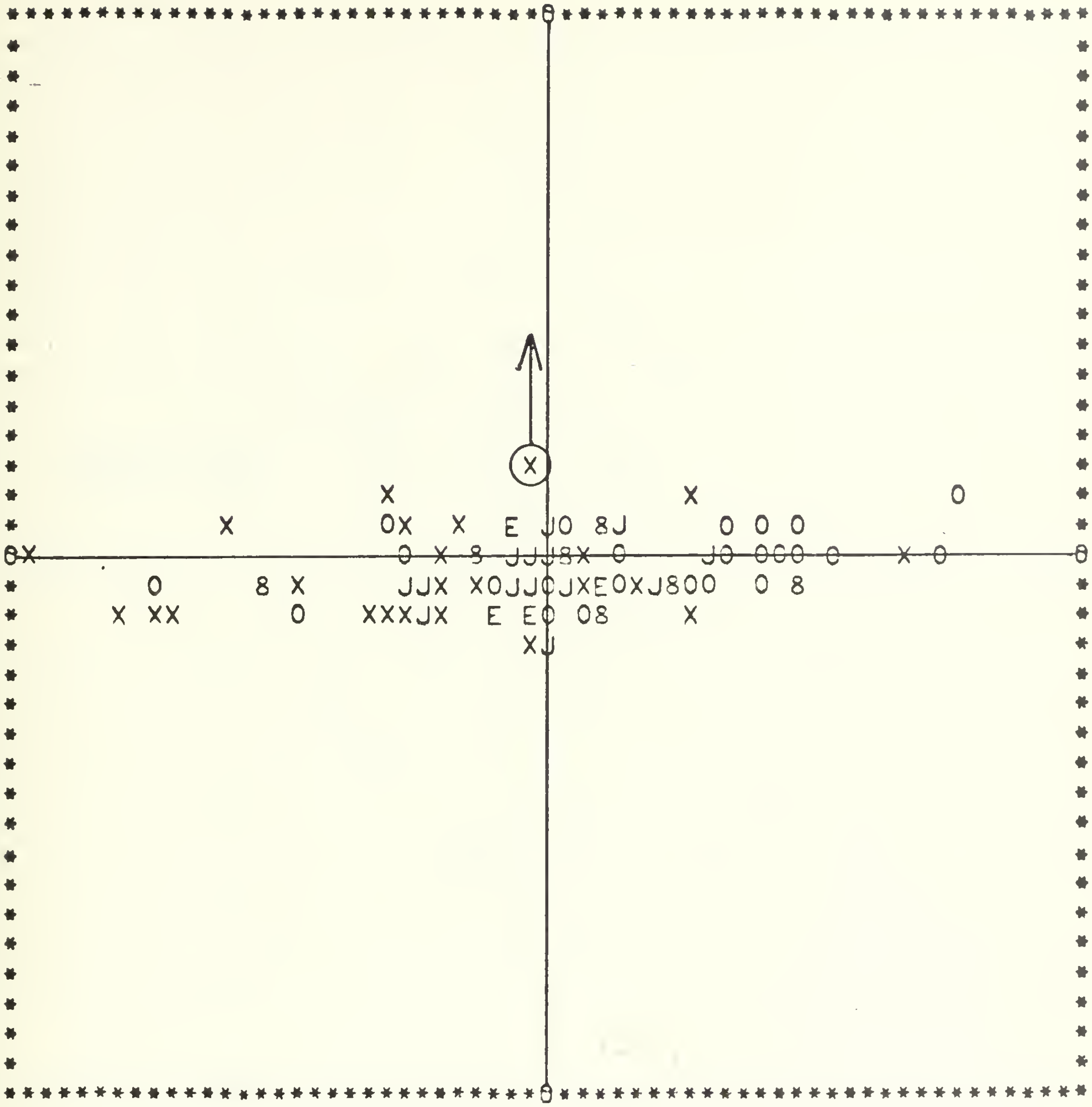

Figure IV-7-2. CASE 7, RUN 1, STEP 2

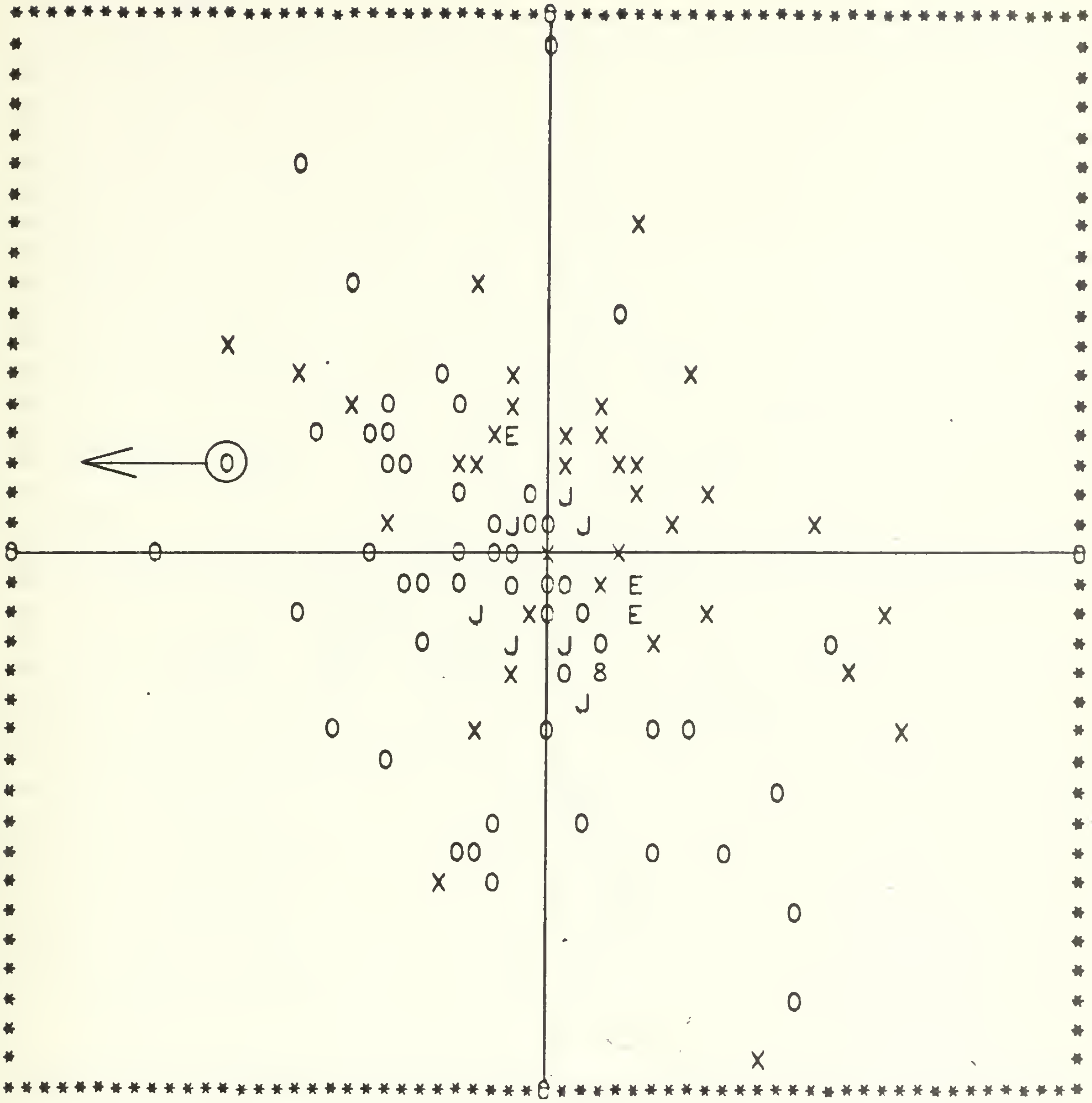

Figure IV-7-3. CASE 7, RUN 1, STEP 11

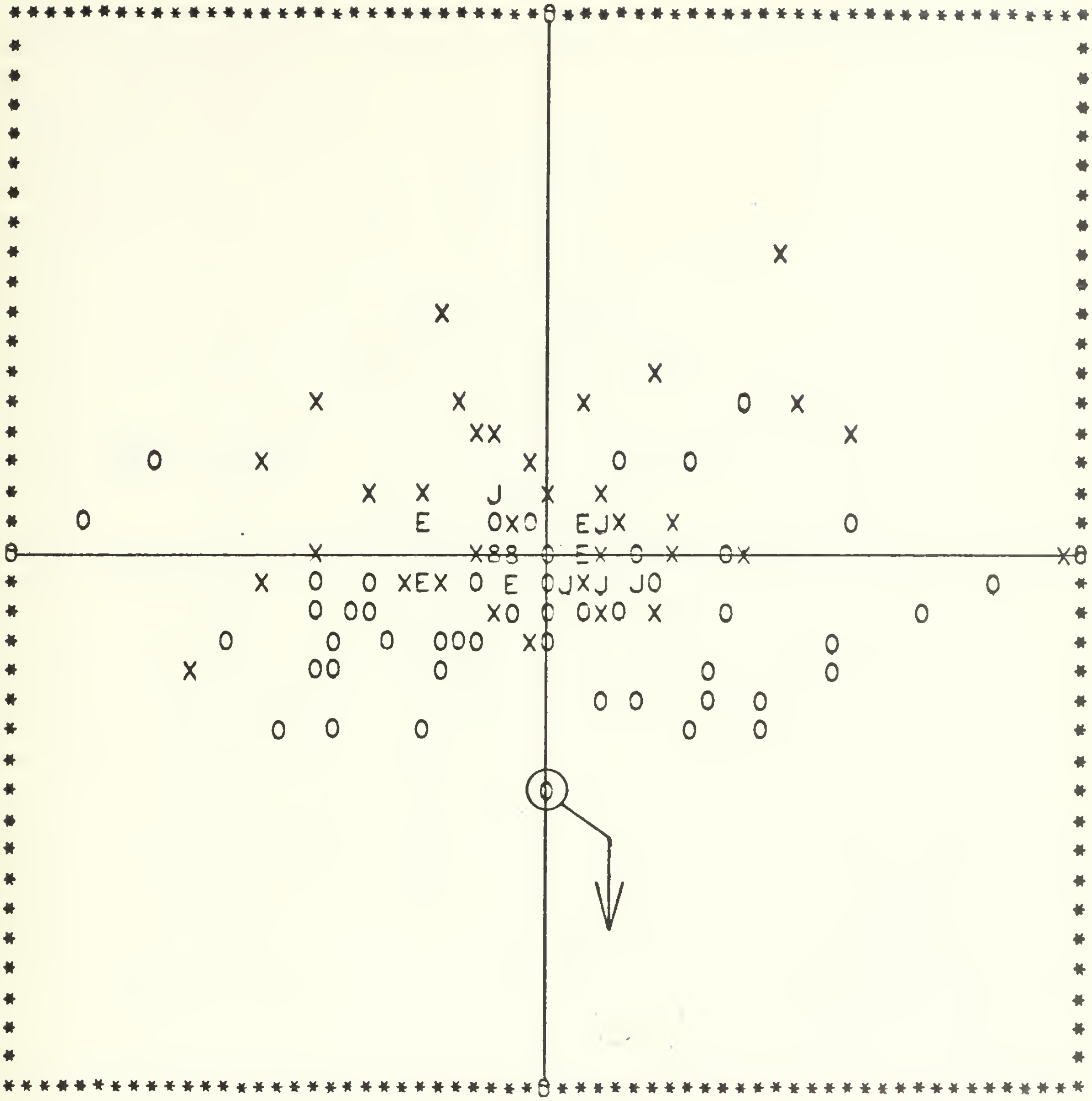

Figure IV-7-4. CASE 7, RUN 1, STEP 15

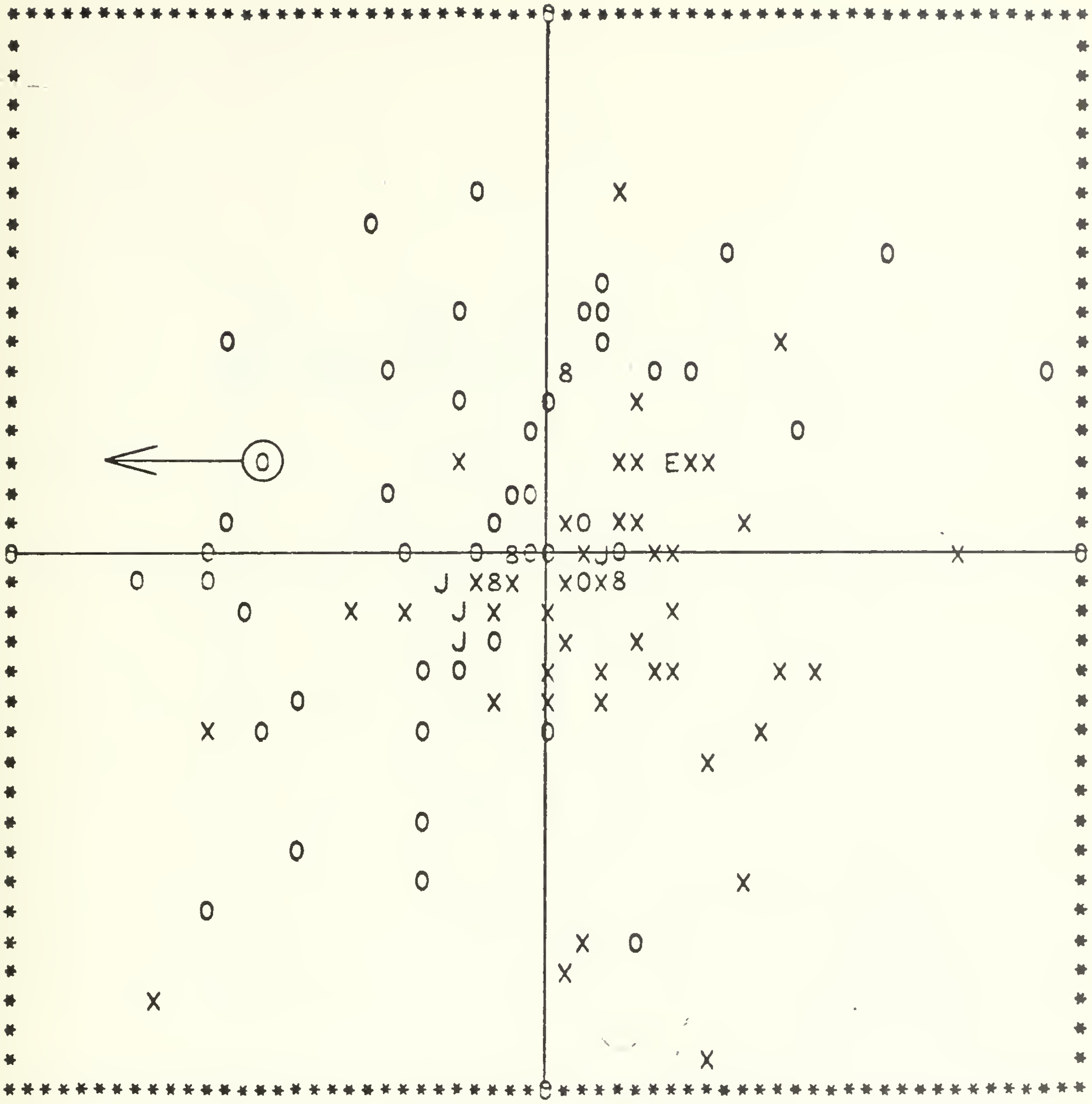

Figure IV-7-5. CASE 7, RUN 1, STEP 20

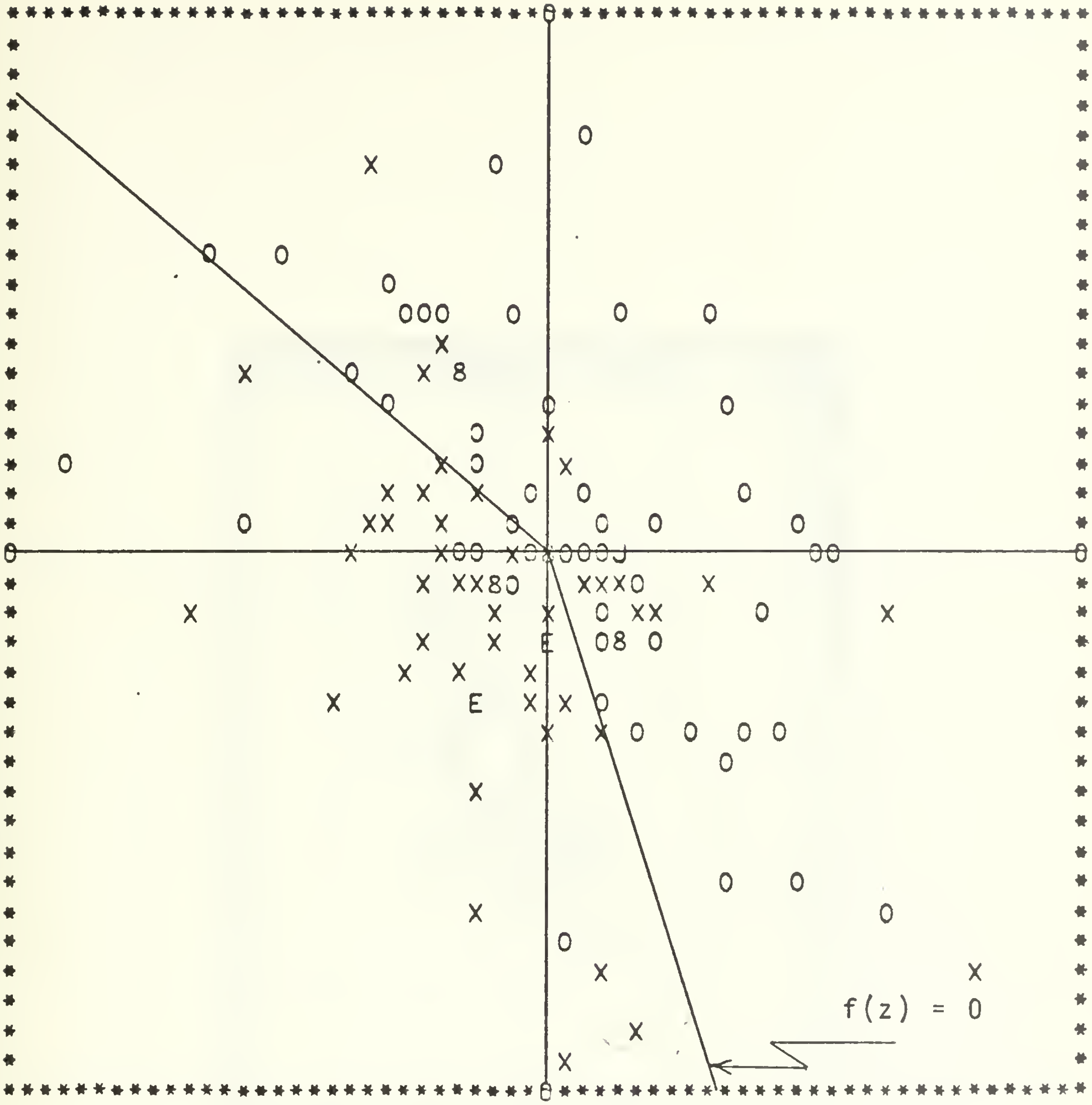

Figure IV-7-6. CASE 7, RUN 1, STEP 23

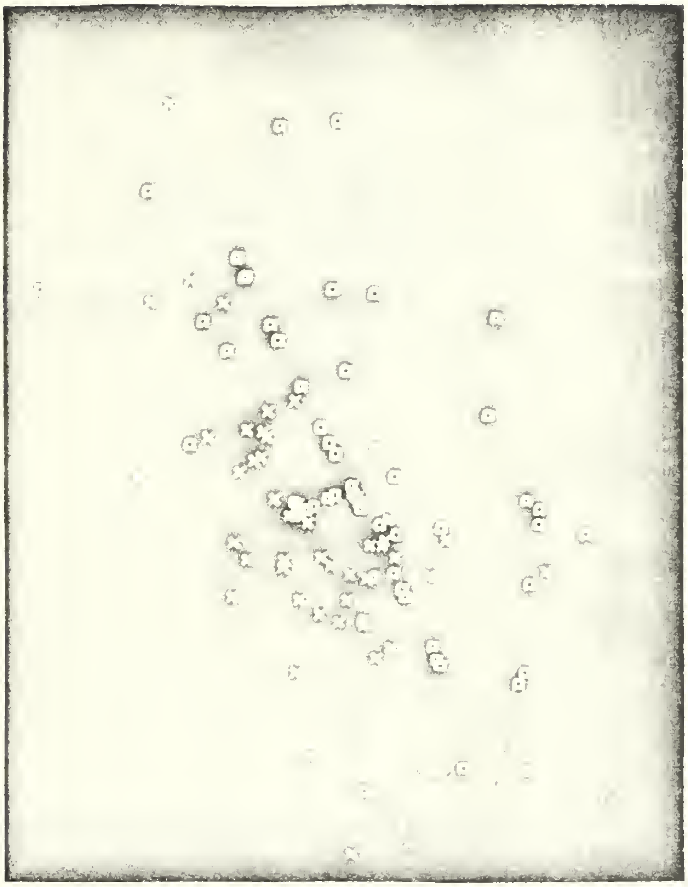

Figure IV-7-6P. CASE 7, RUN 1, STEP 23

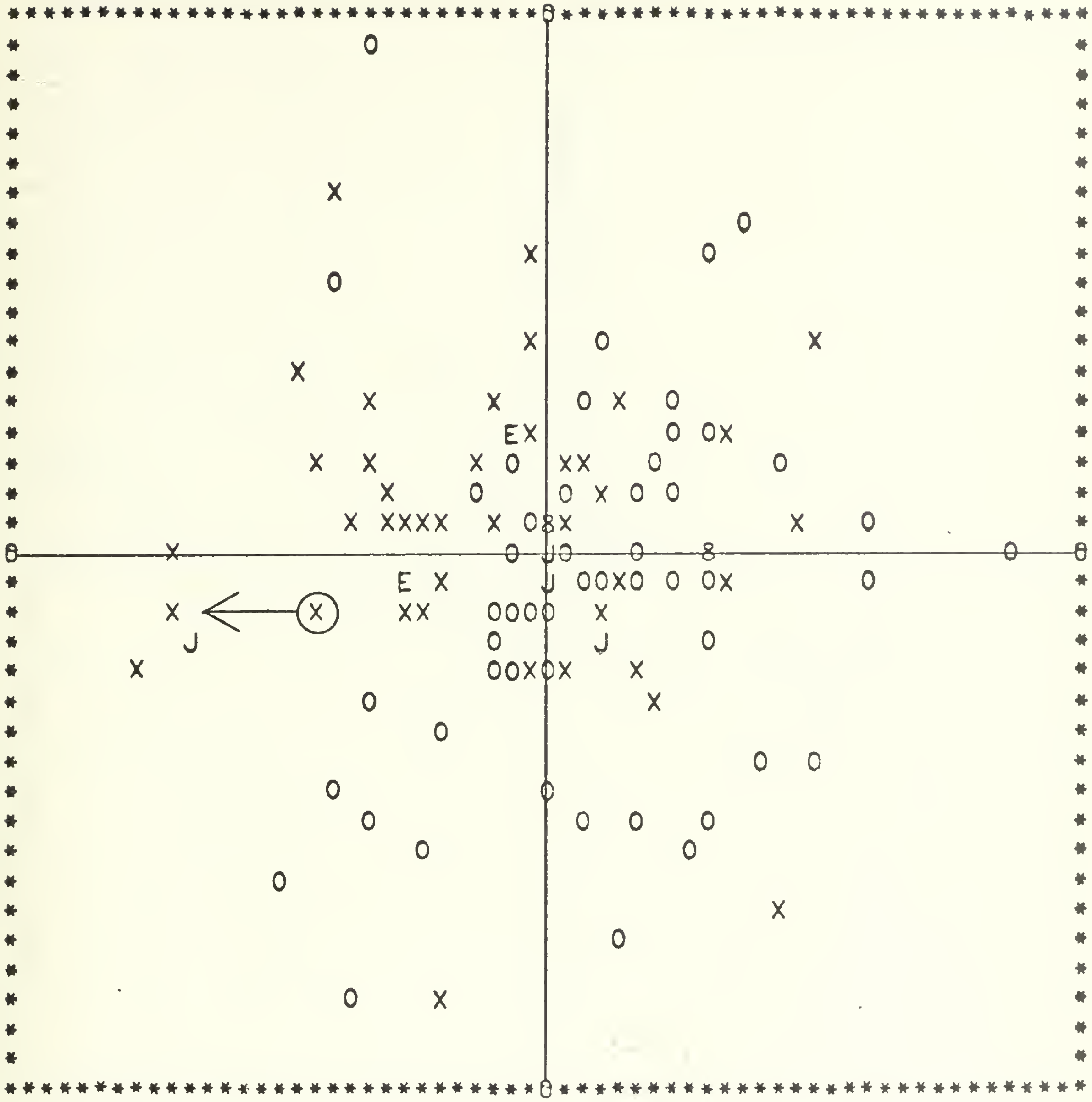

Figure IV-7-7. CASE 7, RUN 2, STEP 1

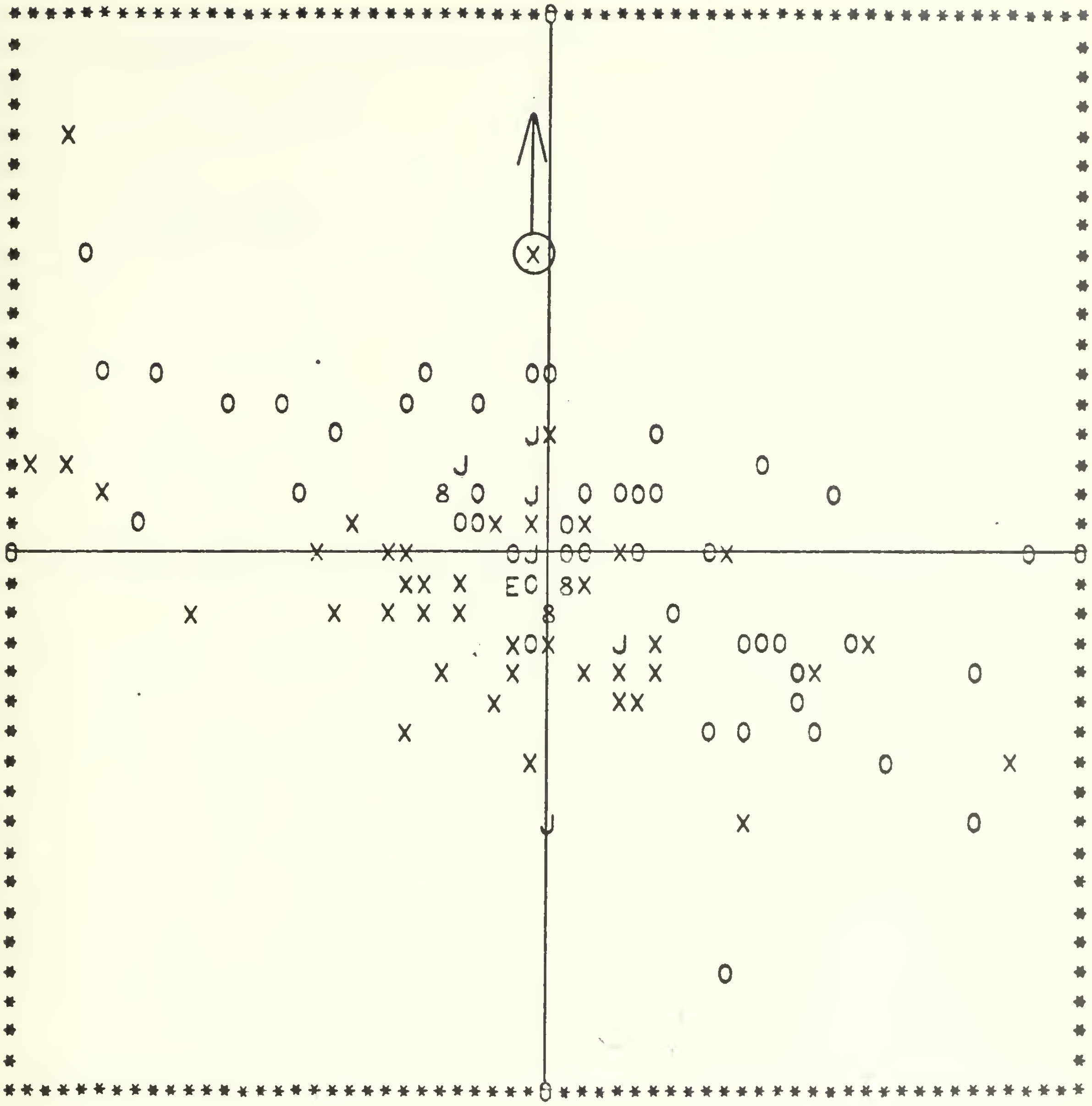

Figure IV-7-8. CASE 7, RUN 2, STEP 2

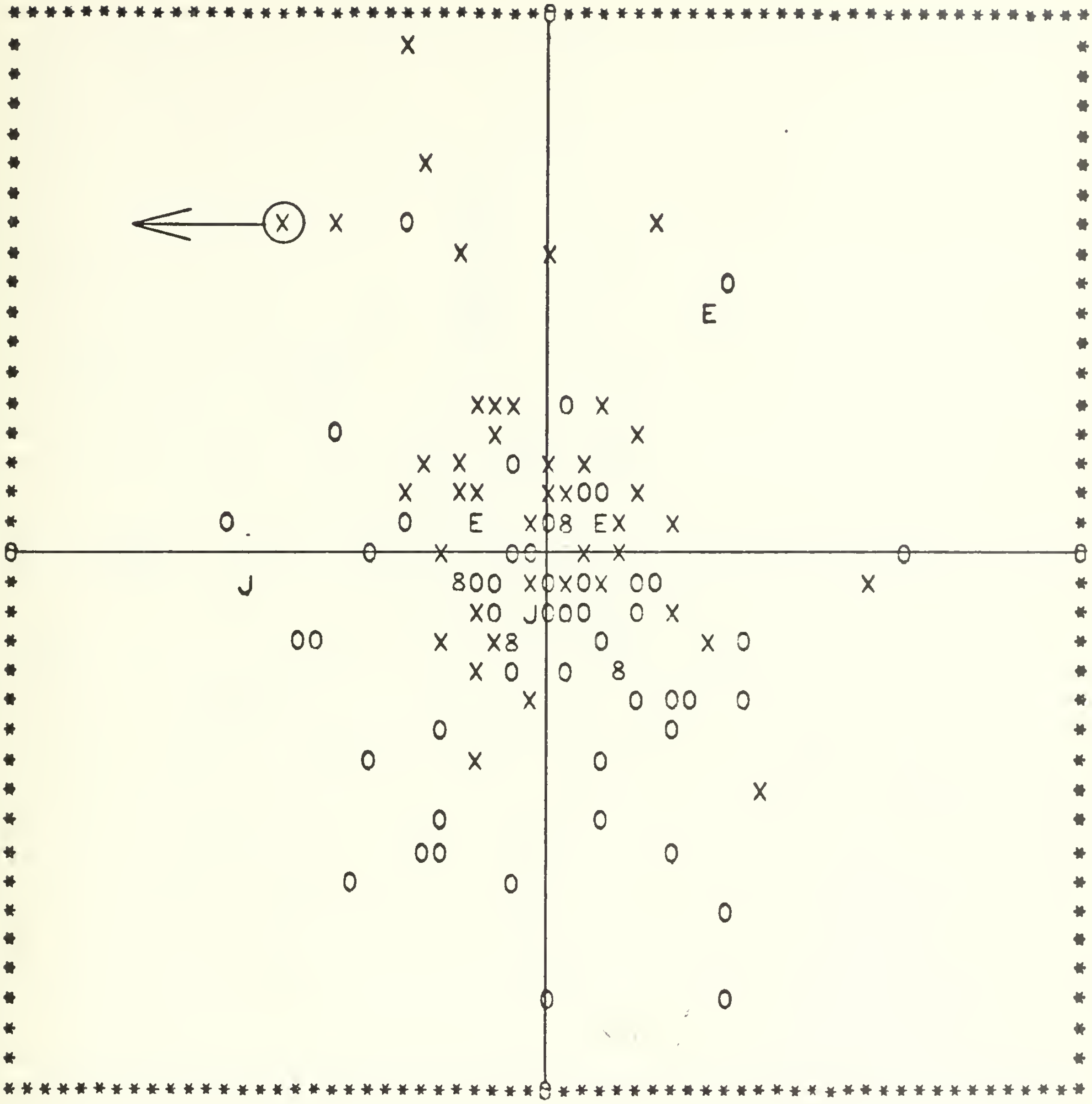

Figure IV-7-9. CASE 7, RUN 2, STEP 9

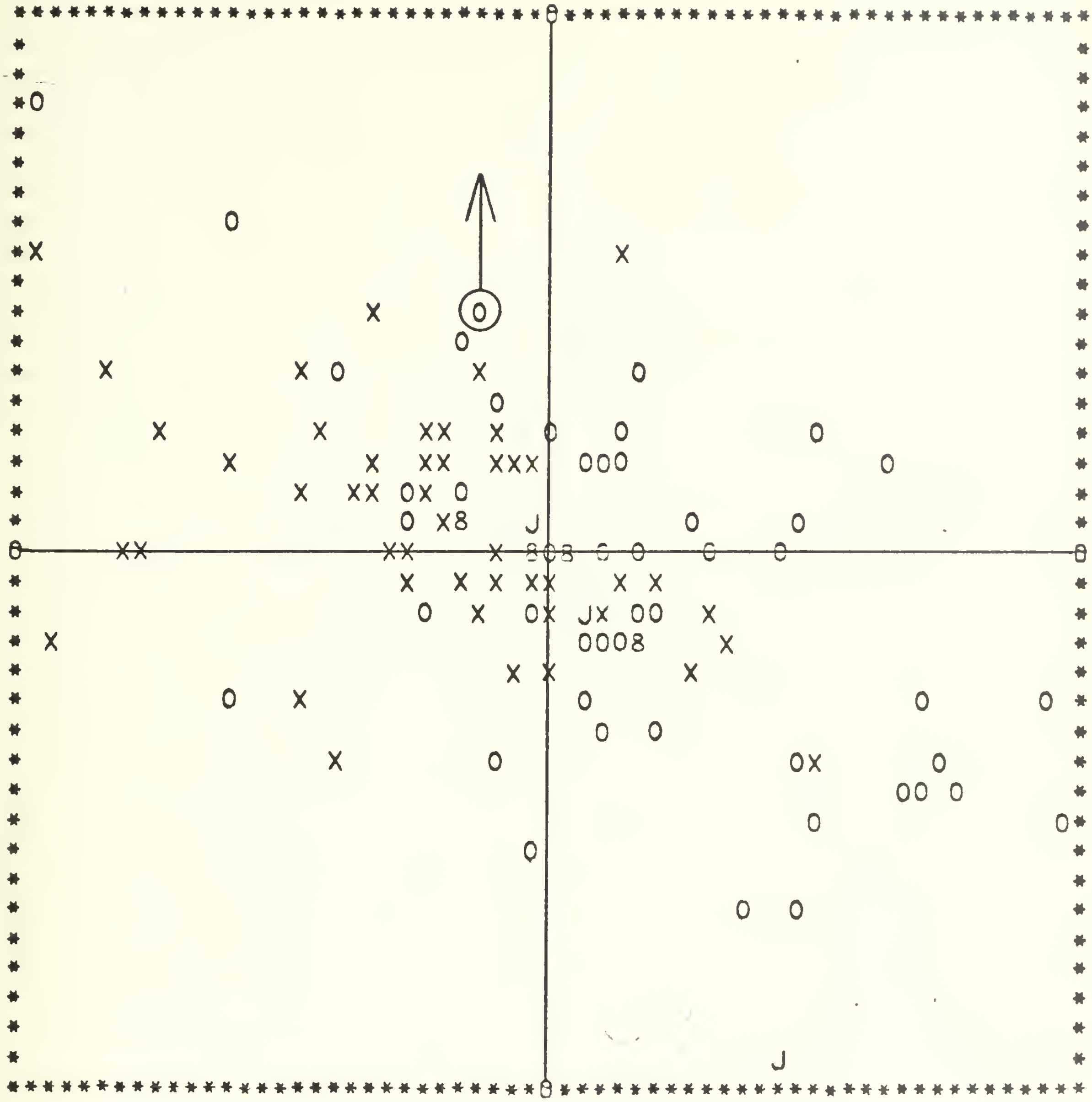

Figure IV-7-10. CASE 7, RUN 2, STEP 12

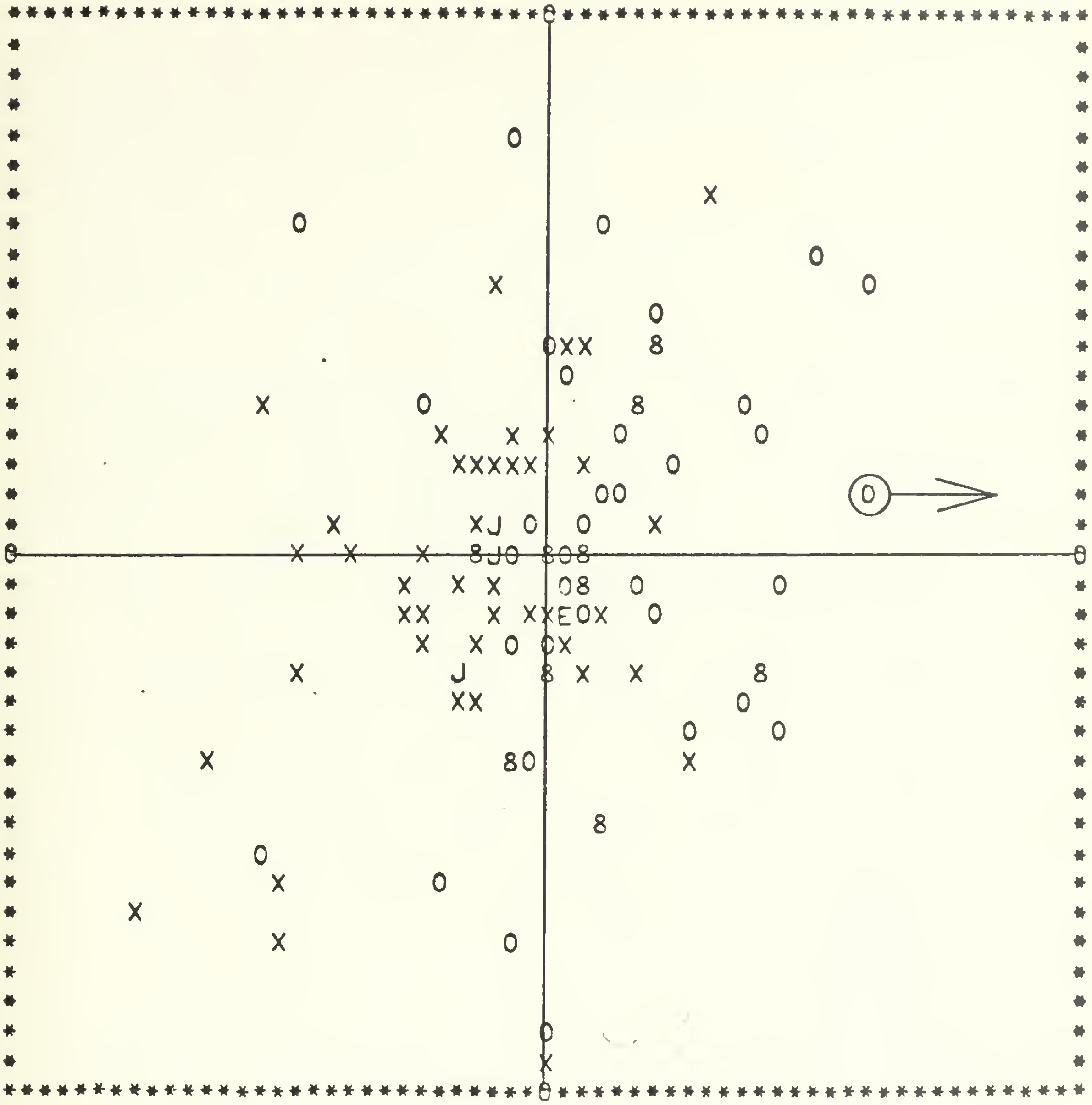

Figure IV-7-11. CASE 7, RUN 2, STEP 14

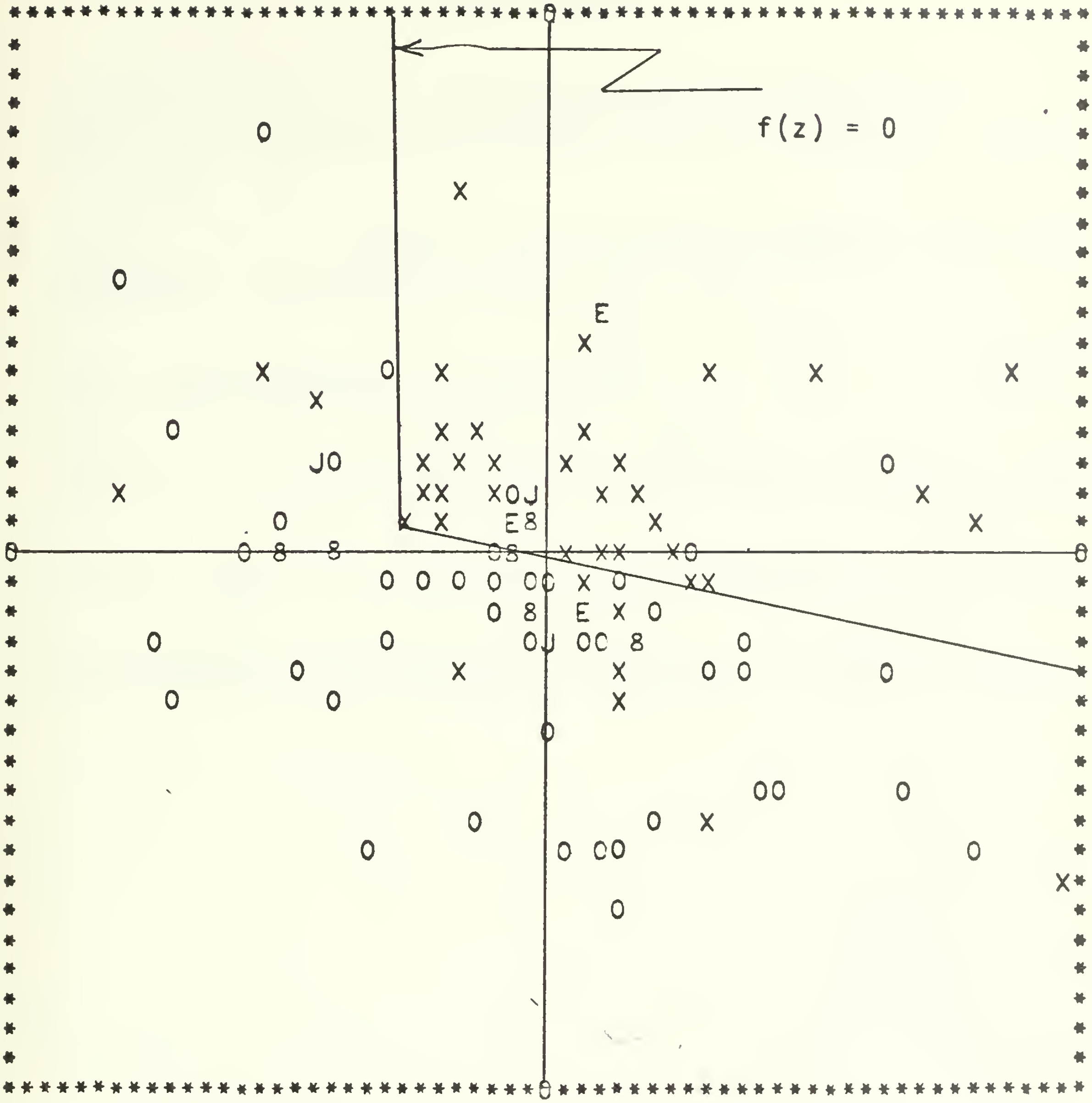

Figure IV-7-12. CASE 7, RUN 2, STEP 27

BIBLIOGRAPHY

1. Lauzon, Gilbert P., A Computer-Graphics Separation Algorithm for Pattern Classification and Cluster Analysis, M.S. Thesis, Naval Postgraduate School, Monterey, California, September 1973.

2. Delaura, R. D., Electrical Engineering Computer Laboratory, Manual for the Fortran User, Naval Postgraduate School, Monterey, California, 10 February 1972.

3. Duda, Richard O., and Hart, Peter E., Pattern Classification and Scene Analysis, Wiley, 1973.

4. Wright, R. M., and Switzer, P., "Numerical Classification Applied to Certain Jamaican Eocene Nummulitids," Mathematical Geology, Vol. 3, No. 3.

5. Fisher, Ronald A., Contributions to Mathematical Statistics, Wiley, 1950.

6. Feeman, George F., and Grabois, Neil R., Linear Algebra and Multivariable Calculus, McGraw-Hill, 1970.

INITIAL DISTRIBUTION LIST

No. Copies

~~1. Defense Documentation Center~~ ~~2~~
~~Cameron Station~~
~~Alexandria, VA 22314~~

2. Library, Code 0212 2
Naval Postgraduate School
Monterey, CA 93940

3. Asst. Professor B. O. Shubert, Code 55Sy 2
Department of Operations Research
and Administrative Sciences
Naval Postgraduate School
Monterey, CA 93940

4. Asst. Professor M. U. Thomas, Code 55To 1
Department of Operations Research
and Administrative Sciences
Naval Postgraduate School
Monterey, CA 93940

5. LT D. A. Rohrback, USN 1
2681 S.E. River Road
Hillsboro, Oregon 97123

6. Operations Research and Administrative 1
Sciences
Naval Postgraduate School
Monterey, CA 93940

~~7. Chief of Naval Personnel~~ ~~1~~
~~Pers 11b~~
~~Department of the Navy~~
~~Washington, D.C. 20370~~

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS
BEFORE COMPLETING FORM

1. REPORT NUMBER

2. GOVT ACCESSION NO.

3. RECIPIENT'S CATALOG NUMBER

4. TITLE (and Subtitle)
A Distribution Free Empirical Method for Classification or Clustering of Vector-Valued Samples

5. TYPE OF REPORT & PERIOD COVERED
Master's Thesis;
March 1974

6. PERFORMING ORG. REPORT NUMBER

7. AUTHOR(s)
Daniel Albert Rohrback

8. CONTRACT OR GRANT NUMBER(s)

9. PERFORMING ORGANIZATION NAME AND ADDRESS
Naval Postgraduate School
Monterey, California 93940

10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS

11. CONTROLLING OFFICE NAME AND ADDRESS
Naval Postgraduate School
Monterey, California 93940

12. REPORT DATE
March 1974

13. NUMBER OF PAGES
156

14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)
Naval Postgraduate School
Monterey, California 93940

15. SECURITY CLASS. (of this report)
UNCLASSIFIED

15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)
Approved for public release; Distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Pattern classification
Cluster analysis
Discriminant function
Computer graphics
Man-machine

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

An algorithm pertinent to pattern classification and cluster analysis of n-dimensional (n>2) data is presented. The algorithm simplifies the dimensionality of the sample set by a perpendicular projection of each point onto a plane which is determined by two orthonormal vectors. The operator views the plane on a computer graphics screen, selects a point on the screen with a lightpen,

and attempts to achieve separation of classes by choosing a direction of movement. Each iteration, a movement of a single point in a certain direction, is actually a reorientation of the plane into which the data points have been projected. These iterations continue until the operator decides successful separation has been accomplished. The method is neither restricted by statistical distribution assumption nor the dimensions of the sample set. From the investigation of seven case studies, this algorithm emerges as a tremendous assistance in the analysis of multidimensional data.

150351

Thesis
R6875 Rohrback
c.1 A distribution free empirical method for classification or clustering of vector-valued samples.