DTIC FILE COPY

(2)

AD-A205 681

BROWN UNIVERSITY

Technical Report

PRELIMINARY TO A NEURAL NETWORK MODEL OF SONAR-BASED TARGET DISCRIMINATION IN THE ECHOLOCATING BAT

Cameron Brennan

Center for Neural Science

Lynda Chen

Walter S. Hunter Laboratory of Psychology
Brown University
Providence, RI 02912

Technical Report No. ONR-88-2

1 May 1988

This research was supported by the
Perceptual Sciences Program
ONR Contract No. N00014-86-K-0401

Approved for Public Release
Distribution Unlimited

89 3 27 085

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGF

Form Approved OMB No 0704-0188

1a REPORT SECURITY CLASSIFICATION Unclassified	1b RESTRICTIVE MARKINGS N. A
2a SECURITY CLASSIFICATION AUTHORITY N. A. 2b. DECLASSIFICATION / DOWNGRADING SCHEDULE N. A.	3 DISTRIBUTION / AVAILABILITY OF REPORT Approved for public release; distribution unlimited.
4. PERFORMING ORGANIZATION REPORT NUMBER(S) ONR-88-2	5 MONITORING ORGANIZATION REPORT NUMBER(S) Same
6a. NAME OF PERFORMING ORGANIZATION Dept. of Psychology Brown University 6b OFFICE SYMBOL (If applicable)	7a. NAME OF MONITORING ORGANIZATION Office of Naval Research
6c. ADDRESS (City, State, and ZIP Code) Providence, RI 02912	7b ADDRESS (City, State, and ZIP Code) 800 N. Quincy Street Arlington, VA 22217-5000
8a. NAME OF FUNDING / SPONSORING ORGANIZATION Office of Naval Research 8b OFFICE SYMBOL (If applicable) Code 1142PS	9 PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER N00014-86-K-0401
8c. ADDRESS (City, State, and ZIP Code) 800 N. Quincy Street Arlington, VA 22217-5000	10 SOURCE OF FUNDING NUMBERS PROGRAM ELEMENT NO: 61153N 42 PROJECT NO: RR 04209 TASK NO: 0420901 WORK UNIT ACCESSION NO: 4424202

11. TITLE (Include Security Classification)
(U) Preliminary to a neural network model of sonar-based target discrimination in the echolocating bat.

12 PERSONAL AUTHOR(S)
Brennan, C., & Chen, L.

13a. TYPE OF REPORT	13b TIME COVERED	14. DATE OF REPORT (Year, Month, Day)	15 PAGE COUNT
Technical	FROM 86/07/01 TO 89/06/30	1988, May 1	36

16 SUPPLEMENTARY NOTATION
N.A.

17 COSATI CODES			18 SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB-GROUP	Biosonar, target classification, echo delay, echo spectrum, neural network

19 ABSTRACT (Continue on reverse if necessary and identify by block number)

It's known from previous research (Griffin, 1958) that echolocating bats such as the big brown bat Eptesicus fuscus, can discriminate between edible and inedible airborne targets using information carried in the echo returns. The goal of this project is to build a neural network model which can perform a rudimentary discrimination task using the same sonar targets. The model is to serve as a preliminary test of the network's ability to discriminate, categorize, and generalize from a limited data base. We plan to use this model in a later study comparing the performance of our network with the behavioral data collected by Griffin, trying to duplicate as closely as possible the parameters of the task originally presented to bats.

20 DISTRIBUTION / AVAILABILITY OF ABSTRACT ☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT ☐ DTIC USERS	21 ABSTRACT SECURITY CLASSIFICATION Unclassified	
22a NAME OF RESPONSIBLE INDIVIDUAL John J. O'Hare	22b TELEPHONE (Include Area Code) (202) 696-4502	22c OFFICE SYMBOL Code 1142PS

DD Form 1473, JUN 86 Previous editions are obsolete SECURITY CLASSIFICATION OF THIS PAGE

S/N 0102-LF-014-6603

ABSTRACT

It's known from previous research (Griffin, 1958) that echolocating bats such as the big brown bat *Eptesicus fuscus*, can discriminate between edible and inedible airborne targets using information carried in the echo returns. The goal of this project is to build a neural network model which can perform a rudimentary discrimination task using the same sonar targets. The model is to serve as a preliminary test of the network's ability to discriminate, categorize, and generalize from a limited data base. We plan to use this model in a later study comparing the performance of our network with the behavioral data collected by Griffin, trying to duplicate as closely as possible the parameters of the task originally presented to bats.

INTRODUCTION

Bats (*Chiroptera*) orient themselves with a biological sonar system, called echolocation by its discover Donald Griffin (Griffin, 1958). The echolocating sounds are typically ultrasonic, with one or both of the frequency-modulated (FM) and constant-frequency (CF) components. They are used to as sonar signals to detect, locate, identify and capture airborne preys. The pursuit sequence is the pattern of behavioral activities associated with the interception of prey, stereotypical of all families of bats studied. Figure 1 illustrates different stages of the sequence. Upon detection of an object of interest, the bat approaches it, and enters the tracking stage of pursuit. It aims its head directly at the target, generally with an accuracy of better than 5 degrees horizontally and vertically (Simmons & Kick, 1984). At the same time it increases its emission rate of sonar signals from 5-20 sounds/sec to 20-40 sounds/sec. Tracking usually begins at a distance of approximately 1.5 meters and ends at about 50 cm away from the target. During the terminal stage, the bat abruptly increases emission rate up to 200 sounds/sec, at which point it seizes the target. Identification of targets is accomplished during tracking and decision-making occurs before a bat enters the terminal stage. When a bat decides to continue on with the terminal pursuit, it's committed to completing the sequence, i.e. physical contact with the target; otherwise, it breaks off from tracking and takes off in search of other objects of interest. The entire pursuit maneuver takes less than a second, over a distance of about 2 meters.

In the original behavioral experiment conducted by Griffin and associates (Griffin, 1958), target stimuli were projected into the flight path of a bat, either one at a time, or several together, in complete darkness. Bats of the family *Eptesicus Fuscus* quickly learned to intercept up to 98% of edible mealworms and avoid over 80% of inedible nylon spheres and disks of various diameters. This high level of performance suggested a very great selectivity; the results were far from random. Griffin's group attempted to measure the echo returns of those different targets in hope of finding the "mealwormness" or "diskness" that must be buried within the echoes which would account for shape recognition. They studied the spectral content of the returned echoes and could find little significant difference such as would be used to perform discrimination-- the echoes looked too similar in spectral content. Almost two decades later, with the advance of computer technology, we can now get high resolution snap shots of the time-varying transients in weak echo returns. The procedures involved acquiring the data are described in the following section.

ECHO MEASUREMENTS

An impulse signal is electronically synthesized containing spectral energy in the ultrasonic up to about 100KHz. It is delivered to a sound transducer aimed at a target which has been mounted on a tripod. The echo is picked up by a microphone, and sent to the A/D port of a signal processing hardware/software package by RC Electronic, Inc. The sampling rate is 500KHz. The RC Electronics software allows real-time averaging of the echoes; averaging brings out the echoes of interest which are otherwise masked by relatively loud electrical noises. The final averaged impulse response is then stored in a RC data file.

Measurement of a target is taken at different angular orientations (figure 2), since an airborne object always tumbles and turns, thus presenting different reflective surfaces to incoming sonar signals. For the purpose of this preliminary study, we chose to record only a small subset of orientations of the mealworm. We looked specifically for orientations which yielded echo returns that were similar to disk echoes. For instance, disks presented at an angle will characteristically produce two main echoes, one from the reflection off the leading edge and another from the trailing edge. Many orientations of the mealworm will actually produce multiple echoes, and we avoided sampling these returns in order to limit the discrimination task to the most challenging input vector set. The most ambiguous returns (two echoes) were obtained by orienting the mealworm at rotations about a vertical axis which bisects the length of the worm, and which lies in the plane defined by its natural curl (see figure 2). Samples were taken at 18 orientations corresponding to 10 degree increments from -90 to 90 degrees of rotation (see figure 2). The same was done for the 12 millimeter disk. Single samples were also taken of spheres of various sizes.

DATA PROCESSING

The RC data files were converted into binary files compatible with ILS (Signal Technology, Inc). An example of the raw data is given in figure 3 for eighteen orientations of mealworm. Given that the sampling rate is 500Khz, that the speed of sound is approximately 300m/s, and that the largest target diameter is less than 1.5 inches we allowed for 64 data points to represent the echo return. The files were edited down to 64 points per orientation. Using ILS, the data were normalized in amplitude so that all echo return are of the same apparent strength. The normalization constant was recorded at the end of the file to serve as an indicator of the original amplitude of the echo return.

THE NEURAL MODEL:

BUILDING THE INPUT STATE VECTOR

We chose to model neurons sensitive to amplitude at set times of arrival rather than to amplitude at set frequencies. The input is time-varying rather than spectral. This presented a problem, however, in accounting for the arrival time of the first (or leading) echo.

There is a sinusoidal shift in arrival time of the return echoes over changing angular orientation of mealworm and disk targets. This shift of arrival time is due to the difference in range to the leading surface of the target at different orientations. The bat performs target discrimination during a tracking phase which lasts approximately 500 msec. During a 500 msec interval, it's estimated empirically that an airborne object thrown by hand would change its aspect angle anywhere from zero to about 60 degrees while tumbling in space. As orientation changes, there will be a shift in the arrival time of the leading echo superimposed on the shift caused by the relative velocity of the bat to the target. It is possible that the bat uses this information to help characterize the target, but behavioral experiments have shown that size and shape discrimination of targets is possible even where the objects are stationary and do not reveal multiple orientations and corresponding shifts in the leading echo (Simmons & Vernon, 1971). We chose to eliminate this time shift by reframing the 64 data points so that the leading echo peaks occur effectively at the same time.

The overall amplitude of an echo is a parameter determined by target size, shape, and composition as well as range. As the bat approaches a target, strength of the returning echoes increases. However, the effect of this decreasing atmospheric attenuation is compensated biologically a system of automatic gain control in the bat's middle ear. The middle ear muscle contracts and reduces the bat's hearing sensitivity to returning echoes by about the same amount as the amplitude of echoes is increased over a range of 17 cm to 1.7 meter (Kick & Simmons, 1984) With this gain control, echoes are heard at a constant sensation level so all changes in amplitude among returning echoes would reflect changes in reflective surface of a target, thus useful in target discrimination. For this reason we included the amplitude information in our model. We chose to represent amplitude in our input vector by 32 units sensitive to a range of peak pressure change. There is an amount of overlap in the sensitivity ranges of these units such that for any peak pressure, four of the 32 units will respond.

The 32 amplitude units plus 64 temporal units set our input state vector at dimensionality 96. The full set of vectors can be seen in figures 4,5, & 6, and on the acetates which can be overlapped for visual comparison. The "relative weights", or contributions to length, of the amplitude and temporal portions of the input vector were adjusted so that the ratio of length in the dimension of the temporal units to that of the amplitude units was 2:1.

There are a total of 43 vectors in the input vector set. Items numbered one to eighteen correspond to the 18 orientations of the mealworm; nineteen to 36 are the eighteen orientations of the 12mm disk; 37 to 43 are the seven spheres ranging from 1 inch to 1/8 inch in diameter, respectively.

THE NETWORK MODEL

An Auto-hetero 96 x 96 matrix was constructed by associating the state vector of a target at a particular orientation to an output vector derived from an orthogonal set of walsh vectors. Learning is performed by the program LEARN.FOR (listing in appendix), which associates the input vector to the appropriate output vector, and then associates the output vector to itself. All learning is done with Widrow-Hoff error correction.

The output vector is composed of 96 elements divided into two fields. The first field contains a 32 dimensional walsh vector that represents the category of echo return. There are nine different possible walsh vectors in this field corresponding to the nine different objects: mealworm, disk, and seven spheres of different diameters. The second field contains a 64 dimensional walsh vector which is unique for every object and every orientation (a total of 43 different walsh vectors can appear in this field). The clustered nature of the activation in the input vector sets shown in figure 7a proved to be seriously non-orthogonal with the walsh vectors in the output set. This is because many of the walsh vectors had large regions of positive or negative activation which overlapped similarly contiguous regions in the input vectors. We overcame this problem by randomly shuffling the element order in the entire input vector set according to a fixed schedule, thereby distributing activation evenly over the length of the vector element list.
The entire input vector set both before and after shuffling appears in figure 7. The output vector set appears in figure 8.

Although all 43 vectors are shown in both the input and output set, *only the odd numbered orientations of both the disks and the mealworms were learned by LEARN.FOR*. LEARN associates individual input/output pairs on a random basis according to the following frequencies:

1 of 9 odd mealworm orientations 40%
1 of 9 odd disk orientations 40%
1 of 7 spheres 20%

A total of 5000 presentations were made to build the matrix used in this study. The presentation record is given in Table 1.

SYSTEM PERFORMANCE

Discrimination performance was tested with the program DISCRIM.FOR (listing in appendix). DISCRIM uses the Brain-State in-a-Box model (BSB, Anderson) of saturation disambiguation, and it features flexible menu-driven control of the iteration parameters which allowed us to "fine tune" the system for the best recall performance. The output is a three-dimensional plot which shows a partial decomposition of the input vector at certain points during the iterations. The decomposition indicates the relative strength of the 43 output vectors as components of the input vector. This is measured by normalizing and taking the dot product of the iterating vector with each of the output vectors to yield the cosine of the angle between the two. Figure 9 shows an example of the output for item #17 (mealworm at orientation 80 deg). The two-dimensional plot at the top shows a cross section of the three-dimensional surface at item #17 which is simply the cosine of the angle between the iterating vector and output vector #17. The first line of the 3-D plot at the bottom is a decomposition of the "first-pass" vector-- the normalized output vector resulting from the very first matrix multiplication. The first-pass vector represents the system's initial recognition using only the hetero (input to output) associations. This vector is then added to the original input vector and iterated according to the formula:

$$G_i = A(F_i^T)$$
$$F_{i+1} = \text{Thresh}(a*F_i + b*G_i + c*F_0)$$

where F_0 = the original input vector
A = the association matrix
a,b,c are scalar constants
For this study we found the following values to yield the best performance: a=0.99, b=0.05, c=0.0
a=0.99 corresponds to a decay of the input vector information to 50% its original length by 75 iterations.
Thresh is a thresholding function which limits the absolute value of every element in the vector to a maximum value. For this study, elements saturated at a value of $^+/_-$ 1.4.

For the case of the initial first-pass vector, i=0, and so

$$F_1 = a*F_0 + b*A(F_0^T) + c*F_0$$

F_1 is displayed on the seventh line of the plot, and subsequent lines show F_i at later iterations (in figure 9, a line is displayed after 5 iterations). It is important to note that while the vector F_i is always normalized before decomposition and display, F_i is never normalized during the actual iterations. It is allowed to grow through the excitatory auto (output to output) connections until either all the elements saturate at threshold or the system becomes unstable. If recognition is perfect the output plot will show a single peak for spheres, or a plateau and a peak for mealworms and disks. The plateau is caused by cross-excitation with other output vectors having the same 32-element classification field. Each sphere has a different classification vector so there is no cross-excitation.

The first line of the surface plot in figure 9 shows that recognition of item #17 is nearly perfect after first-pass. It appears that recognition falls off immediately after first-pass, but this is just an illusion caused by the slight non-orthogonality between the input and output vector sets. The first-pass line shows the decomposition of AF_0^T, whereas the next line (line number seven on the plot) is the decomposition of F_1

which is $0.99*F_0 + 0.05*AF_0^T$-- mostly sonar signal input and only less then 5% output. The projected length of vector F_1 in the direction of any the output vectors is very small; the surface plot does not indicate this because it is a plot of cosine. The plot shows that a perfect corner has been reached by about 50 iterations and that BSB has performed slight disambiguation on the output (the line at 50 iterations is a little smoother than the line representing first-pass).

RESULTS

For all the learned vectors (the odd orientations) first-pass gave excellent recall and there was no need for BSB disambiguation to isolate the top hypothesis. With one exception, the BSB iterations always drew the output vector to a virtually perfect corner-- making the top hypothesis absolutely clear. The exception is with item number 11 (Figure 10-f) This figure shows that the first-pass recall of item 11 is poor relative to the other odd orientations of the mealworm. There is a significant amount of excitation of item number 37 (large sphere) in Figure 10-r. The subsequent iterations draw the vector away from item 11, and after 250 iterations an equilibrium is established mostly between the mealworm category and various sphere categories. Even after 250 iterations, no stable corner has been reached. The confusion between items 11 and 37 may be partly explained by visually comparing the two input vectors-- they show exactly the same excitation of amplitude units. The subtractive error term in the Widrow-Hoff procedure causes learning to be poor when the different input vectors are very similar, or have substantial length in the same direction.

Presenting the system with the unlearned even orientations of the disk and mealworm gave us an indication of the system's ability to generalize from a limited data base. We found that first-pass categorize correctly about half the time, and that BSB usually did not serve to strengthen the top hypothesis-- it tended to distribute excitation over other categories (see figure 11). Generalized recognition of the mealworm vectors was best for orientations near +90, -90, and 0 degrees. The best recognized even-numbered disk orientations were near +90 and -90 degrees.

We wanted to get some indication of the amount of "confusion" that the system had between all the mealworm orientations and the various sphere sizes. Specifically, we wanted to know which of the sphere output vectors showed the greatest average excitation when the system was presented with every mealworm orientation. The Widrow-Hoff learning algorithm suppresses this kind of cross-excitation by subtractive associations. Because all the input vectors share similar features, the error term learned by one orientation will effect the cross-correlations of other vectors. Therefore, there is no simple way to determine which pairs of vectors were the hardest to learn to discriminate. If a particular sphere echo is similar to a number of mealworm orientations, then depending on the exact order of learning you might find a positive or negative residual correlation between the input vector of any one mealworm orientation and the output vector of the sphere. As learning progresses the cross-correlation between items will hover around zero but will not be exactly zero; it will vary positively and negatively over the learning trials depending upon which other vectors are presently being learned. Thus, there is no simple way to determine the "confusion" between the mealworm and sphere categories simply on the basis of their cross-correlation strengths. We got an estimate, however, on the assumption that large negative cross-correlations were just as bad as large positive ones. This is counter-intuitive, since we tend to think that knowing what an item is *not* tells you something about what it *is*. For example, one might be tempted to see figure 11-b and say that when the system is presented with item #4, it is confused most strongly by items #39, #40, and #41 (these show strong positive excitation), and least strongly by items #37 and #42 (these show strong negative excitation). However, the fact that #37 and #42 show excitation at all, *regardless of the sign*, indicates that the system was not able to distinguish the "mealwormness" of item #4 from the "non-sphereness" it shares with those two items. We made our calculations of system confusion by taking the average *magnitude* of the excitation of a particular sphere output vector with presentation of every mealworm input vector. We did separate calculations for odd and even orientations; these are presented in figures 12 & 13.

CONCLUSION

It is tempting to compare the performance of this system with data from actual behavioral experiments, but this preliminary system bears very little resemblance to the physical parameters of the original discrimination task. We have only used echo returns from a very limited range of orientations of mealworm, and we have only used data collected from one mealworm. We cannot assume that the bat has the necessary neural hardware to effect a Widrow-Hoff learning scheme, and we know for a fact that the bat learned to discriminate between inedible and edible targets through fewer then 5000 presentations. The discrimination tasks themselves are also incomparable, since the bat is only concerned with edible v.s. inedible, while the network was constructed to discriminate between individual orientations of mealworms and disks, and even to identify the size of the spheres. The next step will be to build a system that holds an input vector set containing echo returns from all angles of a mealworm along with sphere and disk vectors, and which associates these to only two output vectors--'mealworm' and 'other'.

We have shown that there is enough distinguishing information in even the most seemingly ambiguous echo returns to allow a rather idealized neural network to discriminate well beyond the categories of inedible and edible. Most importantly, we have shown that a model such as ours can be used to make predictions about which items will be the most difficult for a bat to discriminate between, and this suggests a series of behavioral experiments to confirm the model. Once we have collected data from the remainder of the mealworm orientations we can construct a system that is more accurate to a true discrimination task, and which will yield more relevant predictions about the limits of target discrimination in the bat.

REFERENCES

Griffin, D. R. 1958. *Listening in the Dark.* New Haven, CT: Yale University Press. 413 pp.

Simmons, J.A., Vernon, J.A. Echolocation: Discrimination of targets by the Bat, *Eptesicus fuscus* J. Exp. Zoology, Vol. 176, No.3 March 1971

Kick, S. A., Simmons, J.A. Automatic Gain Control in the Bat's Sonar Receiver and the Neuroethology of Echolocation J. Neuroscience, Vol. 4, No. 11, November 1984

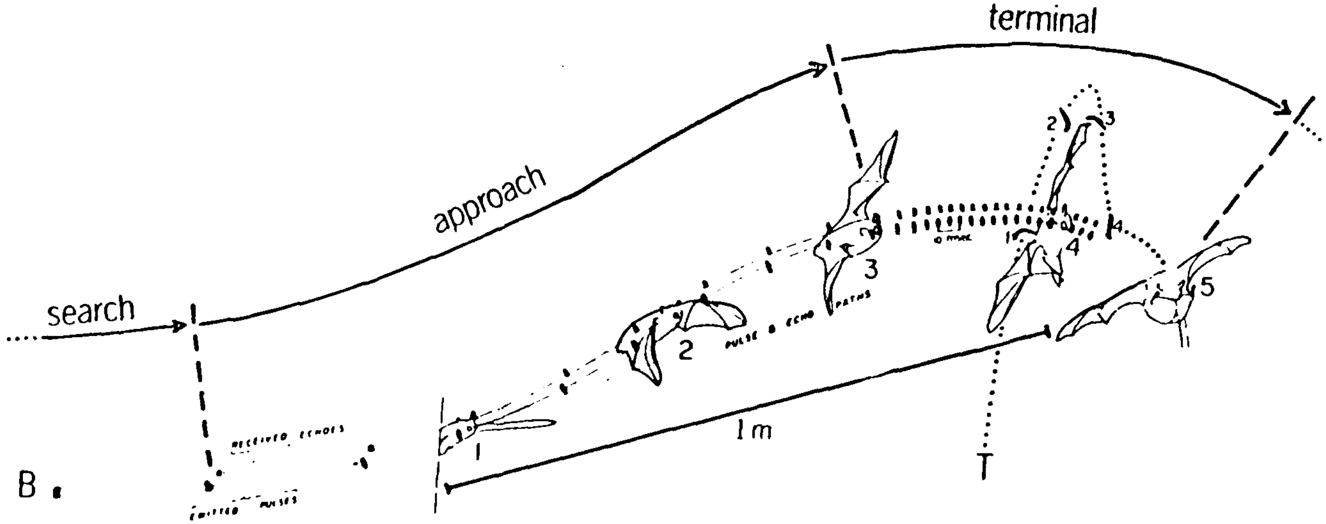
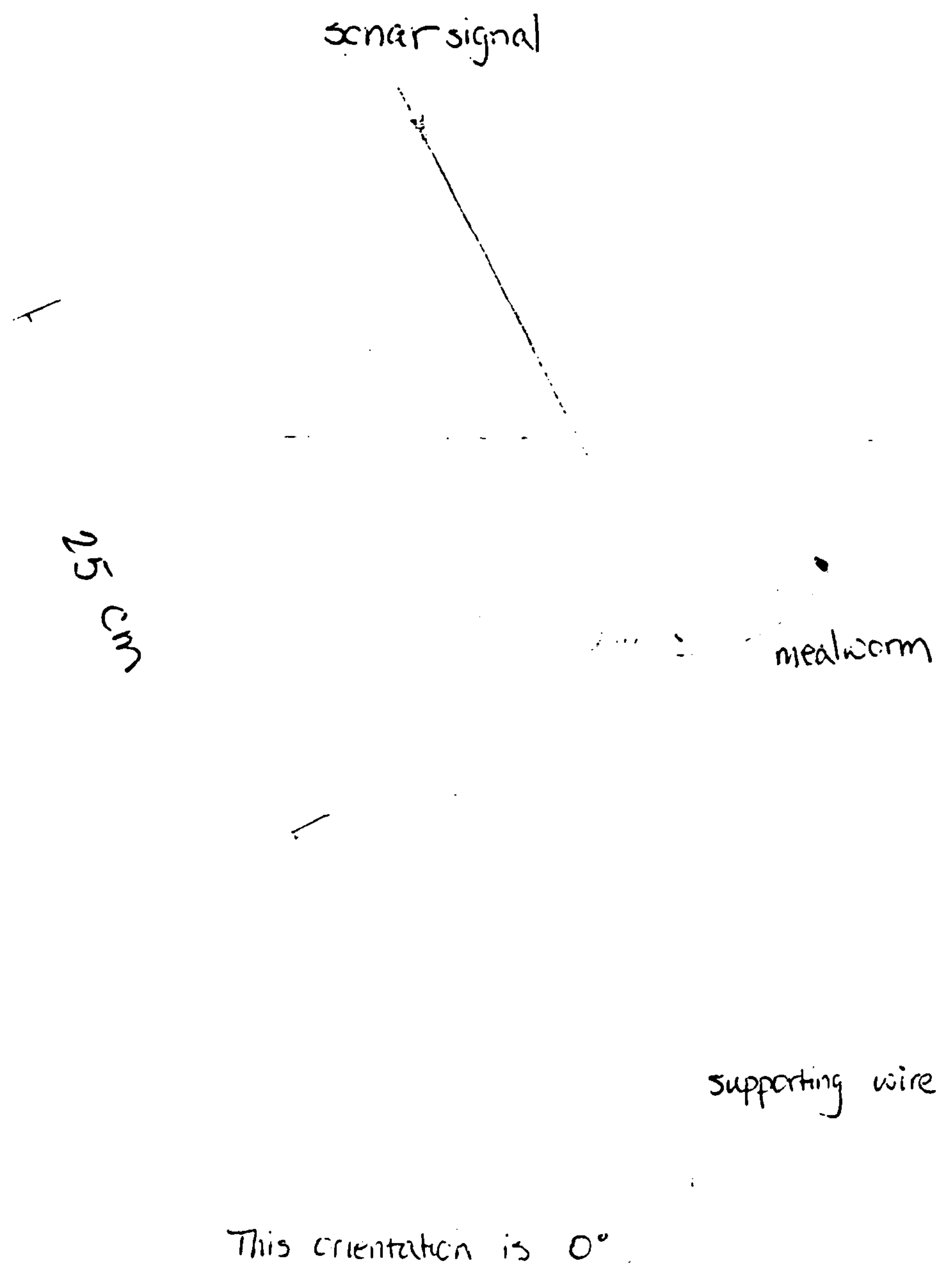


Figure 1 A drawing of the three stages in the interception of an airborne target by the little brown bat, *Myotis lucifugus*. Five numbered positions of the bat (B) on its approach and the target (T, a mealworm) on its trajectory are shown, based upon multiple-exposure stroboscopic photographs of the maneuver, which lasts less than a second (37). The track of the bat illustrates the timing of successive sonar emissions and returning echoes with tic-marks. Note that the bat increases the repetition rate of its emissions during the approach stage and again, and more abruptly, increases the repetition rate in the terminal stage. The capture occurs in position No. 5 with the bat seizing the mealworm in its tail membrane. The interception process is sufficiently stereotyped in different species that this diagram can be considered typical of airborne captures by bats. Different species do, however, emit different sonar signals throughout the maneuver, indicating that their acoustical strategies for finding prey may be different (27). (Drawing by F. A. Webster; reproduced with permission.)

Figure 1



sonar signal
25 cm
mealworm
supporting wire
This orientation is 0°

Figure 2

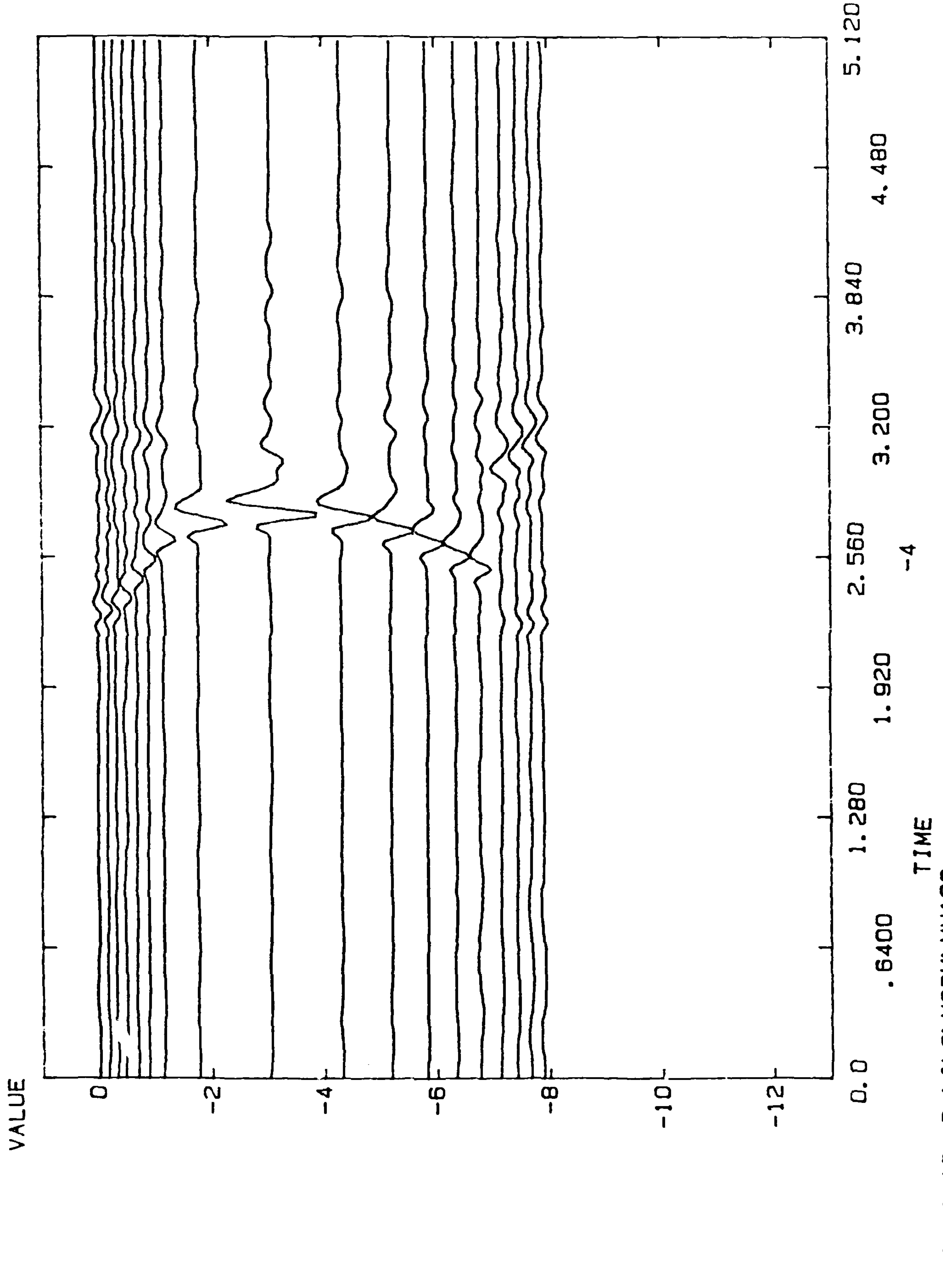

VALUE
0
-2
-4
-6
-8
-10
-12
0.0
.6400
1.280
1.920
2.560
3.200
3.840
4.480
5.120
TIME
-4
DRE 1.18 D:\ILS\WORK\MW100.

Figure 3

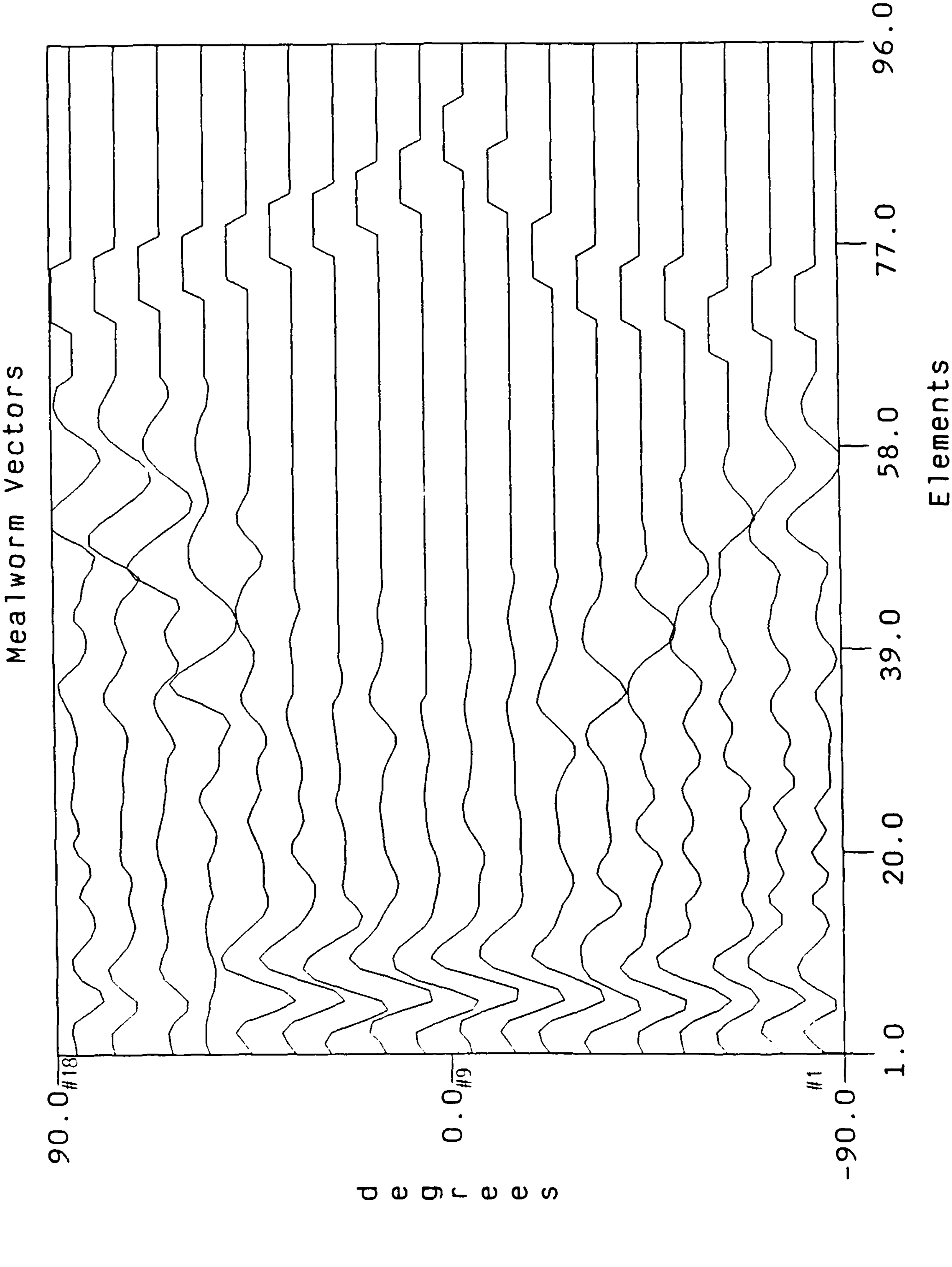

Mealworm Vectors
degrees
90.0
0.0
-90.0
#18
#9
#1
1.0
20.0
39.0
58.0
77.0
96.0
Elements

Figure 4

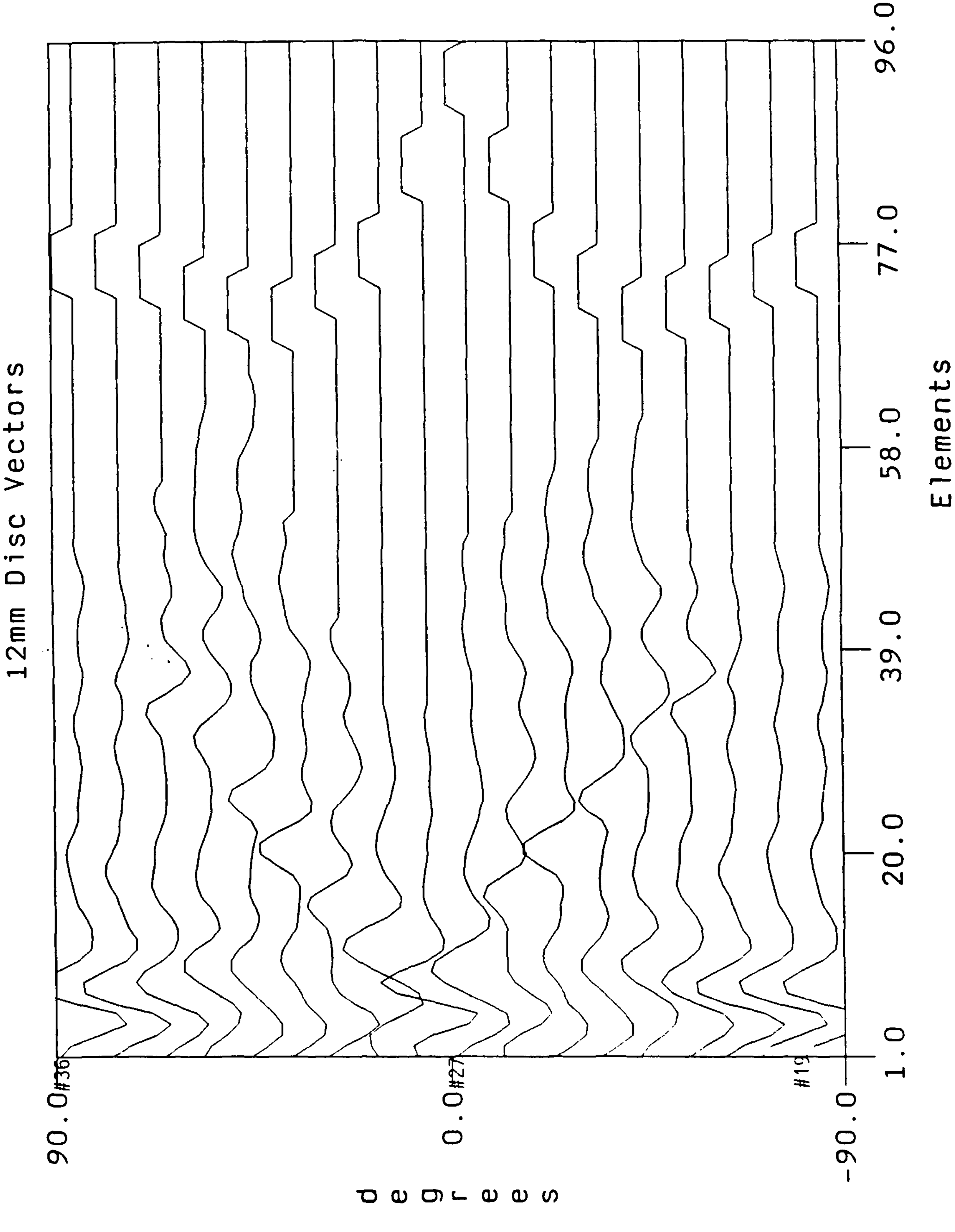

12mm Disc Vectors
degrees
90.0
#36
0.0
#27
#19
-90.0
1.0
20.0
39.0
58.0
77.0
96.0
Elements

Figure 5

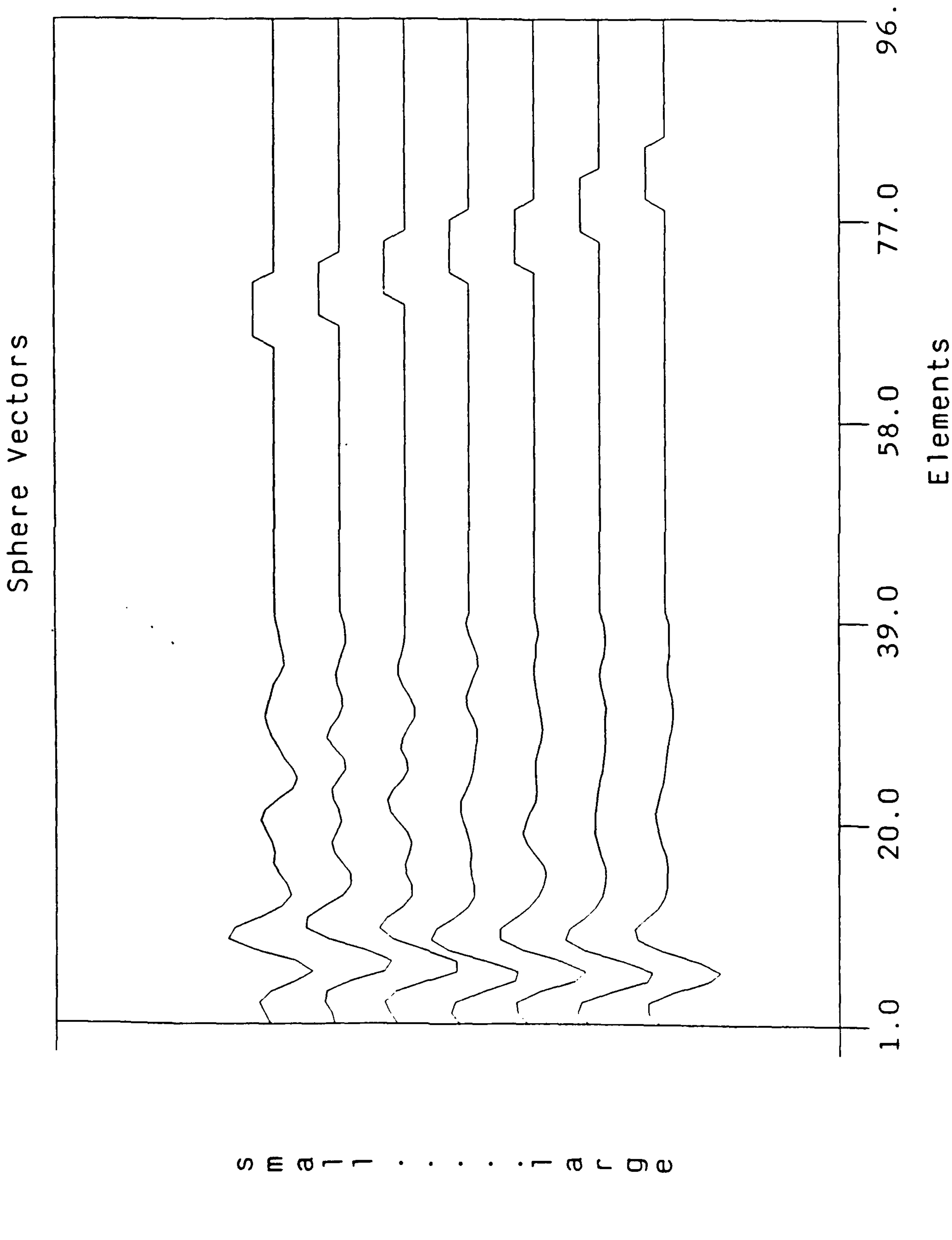


Figure 6

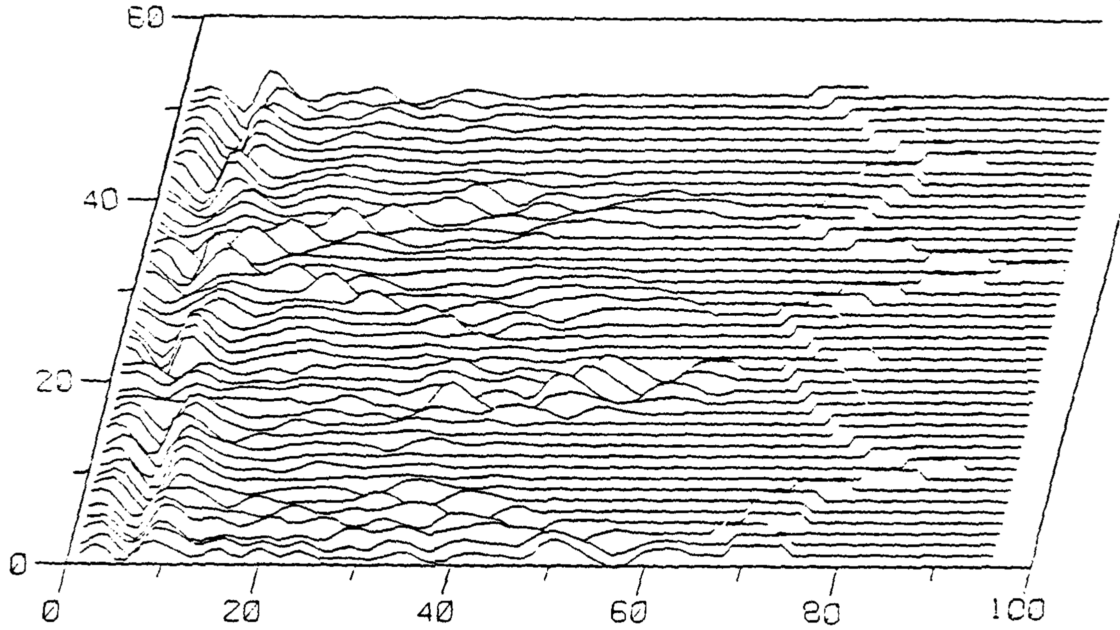


Fig 7a
Figure 7a

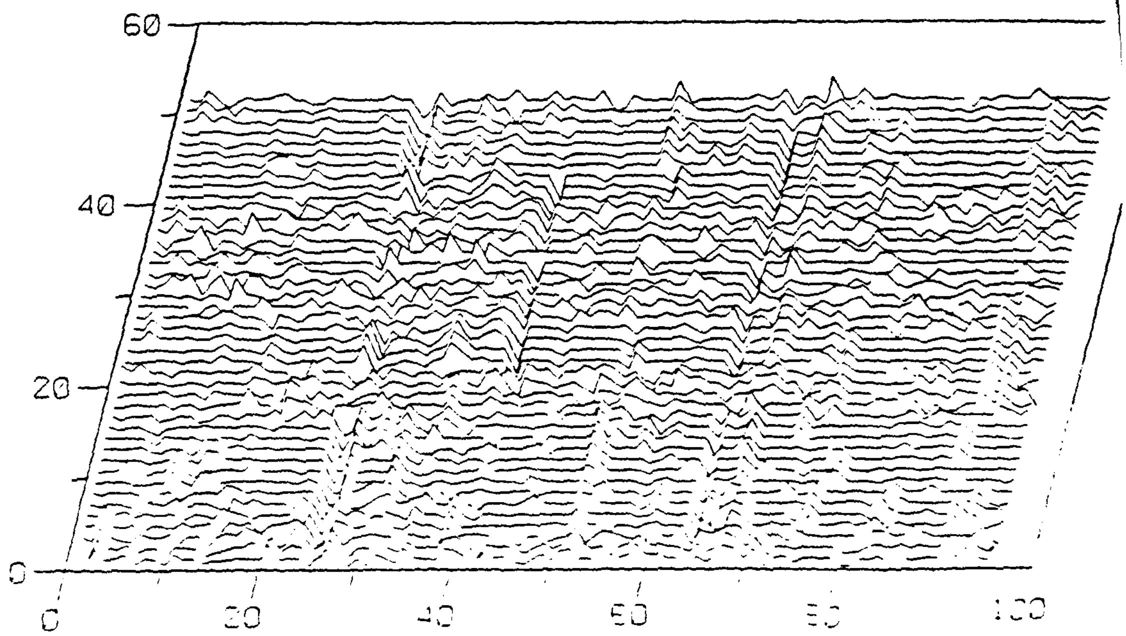


Figure 7b

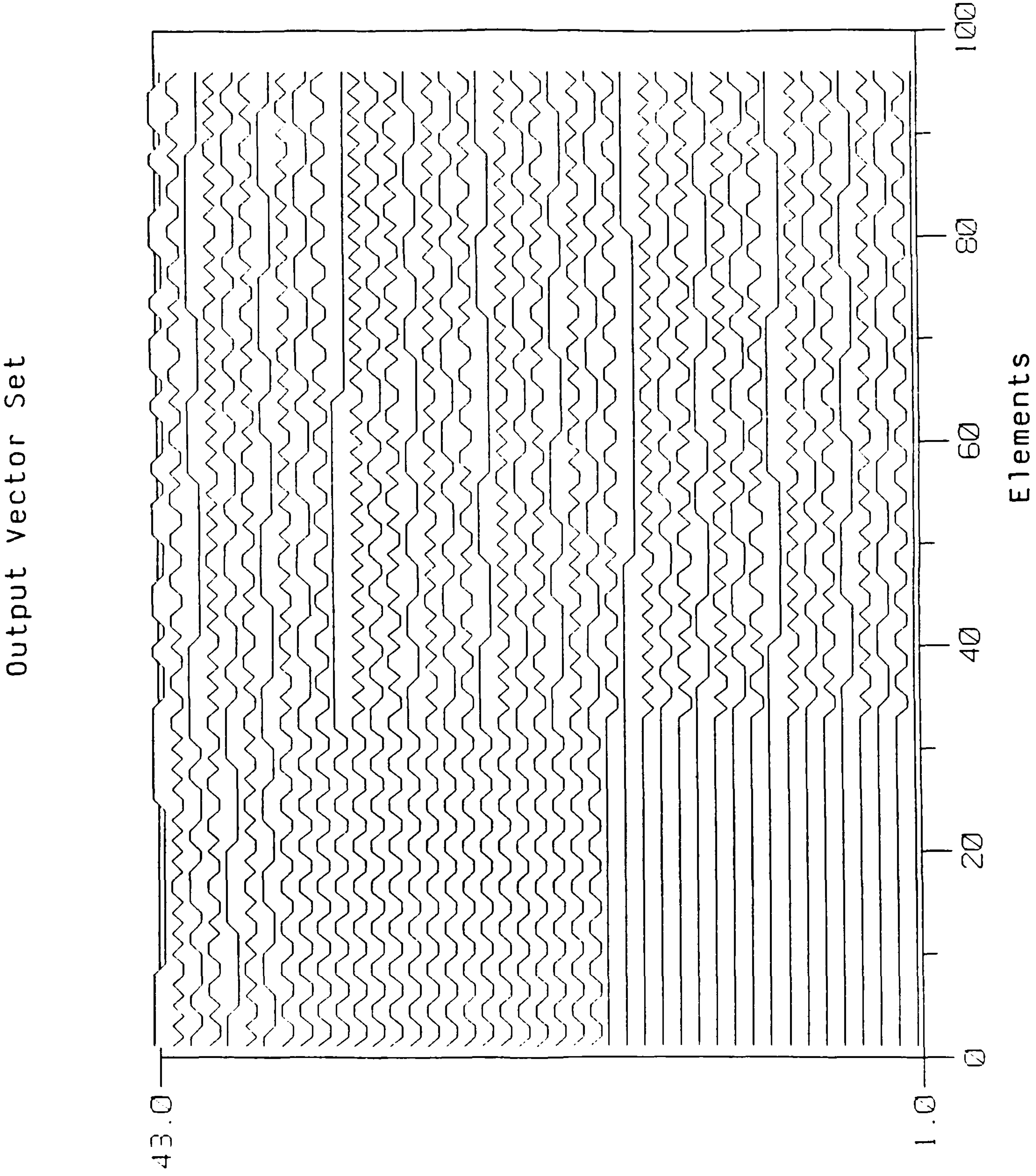


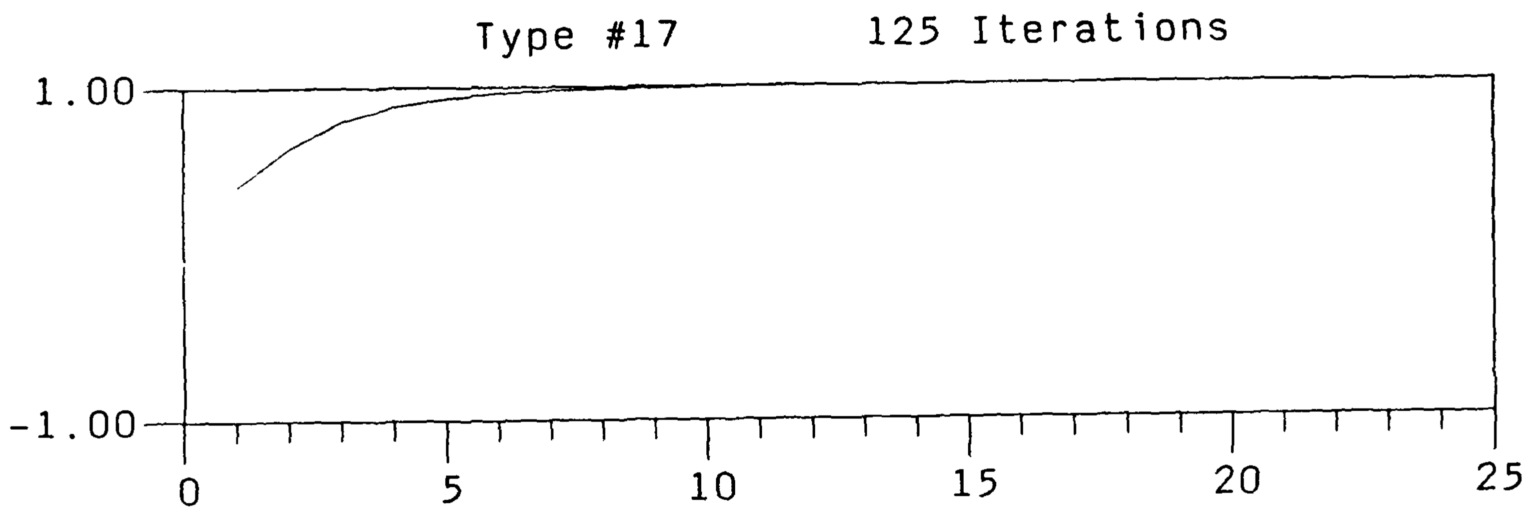
Figure 8

Data File Returned from LEARN.FOR

5000 TRIALS TOTAL
LCONST= 0.950000

VEC.#	Presentations	Cost Function
VEC: 1	LEARNED *242	0.999727
VEC: 2	LEARNED * 0	0.400495
VEC: 3	LEARNED *205	0.998903
VEC: 4	LEARNED * 0	0.093674
VEC: 5	LEARNED *240	0.993832
VEC: 6	LEARNED * 0	-0.061028
VEC: 7	LEARNED *223	0.987504
VEC: 8	LEARNED * 0	0.290609
VEC: 9	LEARNED *231	1.001384
VEC: 10	LEARNED * 0	0.370661
VEC: 11	LEARNED *200	0.919102
VEC: 12	LEARNED * 0	0.335900
VEC: 13	LEARNED *233	0.986664
VEC: 14	LEARNED * 0	0.234245
VEC: 15	LEARNED *200	1.005720
VEC: 16	LEARNED * 0	0.111282
VEC: 17	LEARNED *210	1.000441
VEC: 18	LEARNED * 0	0.302537
VEC: 19	LEARNED *203	0.985734
VEC: 20	LEARNED * 0	0.266765
VEC: 21	LEARNED *212	0.998938
VEC: 22	LEARNED * 0	0.276436
VEC: 23	LEARNED *231	0.970380
VEC: 24	LEARNED * 0	0.151559
VEC: 25	LEARNED *242	0.983373
VEC: 26	LEARNED * 0	0.128402
VEC: 27	LEARNED *241	0.992709
VEC: 28	LEARNED * 0	0.134533
VEC: 29	LEARNED *232	0.990835
VEC: 30	LEARNED * 0	0.502620
VEC: 31	LEARNED *217	0.942663
VEC: 32	LEARNED * 0	0.181624
VEC: 33	LEARNED *231	0.850625
VEC: 34	LEARNED * 0	0.177523
VEC: 35	LEARNED *224	0.803234
VEC: 36	LEARNED * 0	0.334827
VEC: 37	LEARNED *158	0.942247
VEC: 38	LEARNED *143	0.853462
VEC: 39	LEARNED *133	0.878983
VEC: 40	LEARNED *141	0.964195
VEC: 41	LEARNED *131	0.908487
VEC: 42	LEARNED *136	0.900051
VEC: 43	LEARNED *141	0.998985

Table 1


Type #17
125 Iterations
1.00
-1.00
0
5
10
15
20
25

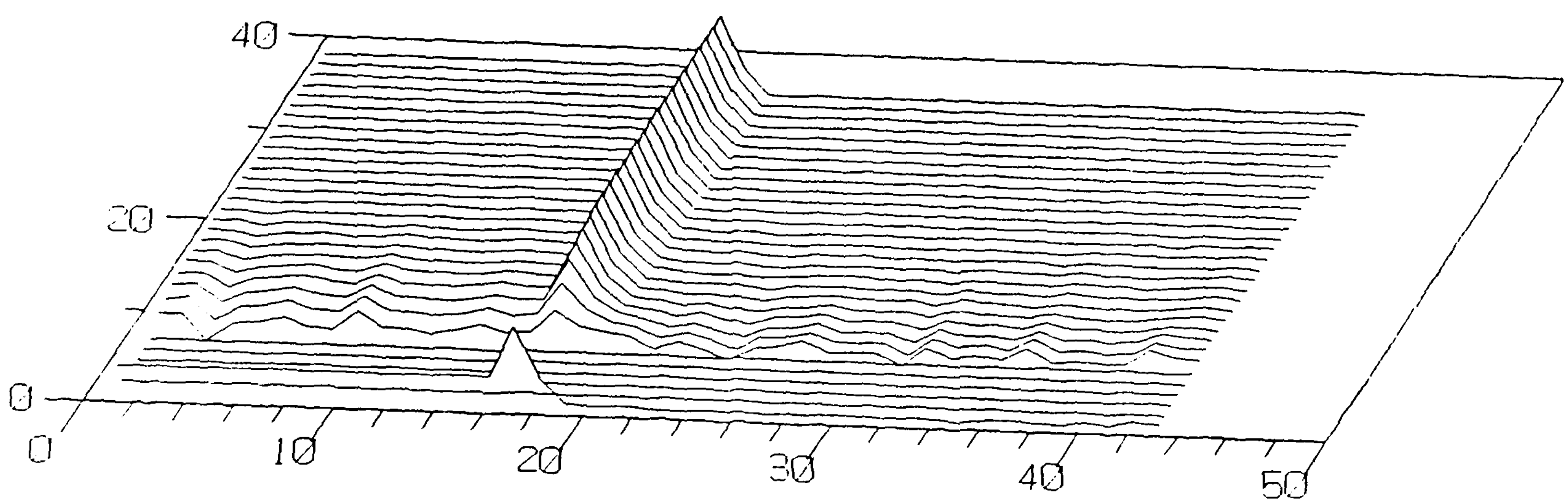

40
20
0
0
10
20
30
40
50

Figure 9

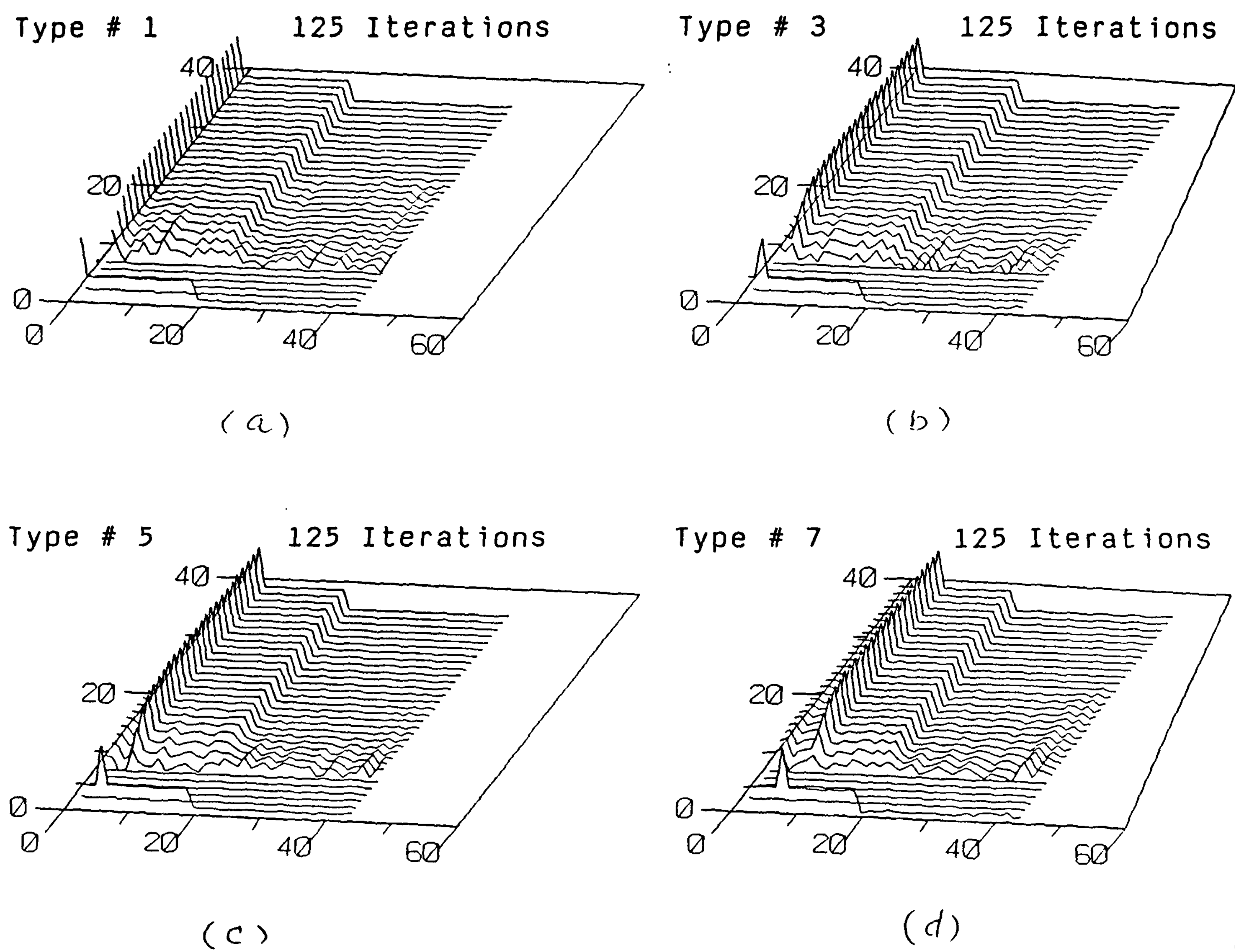
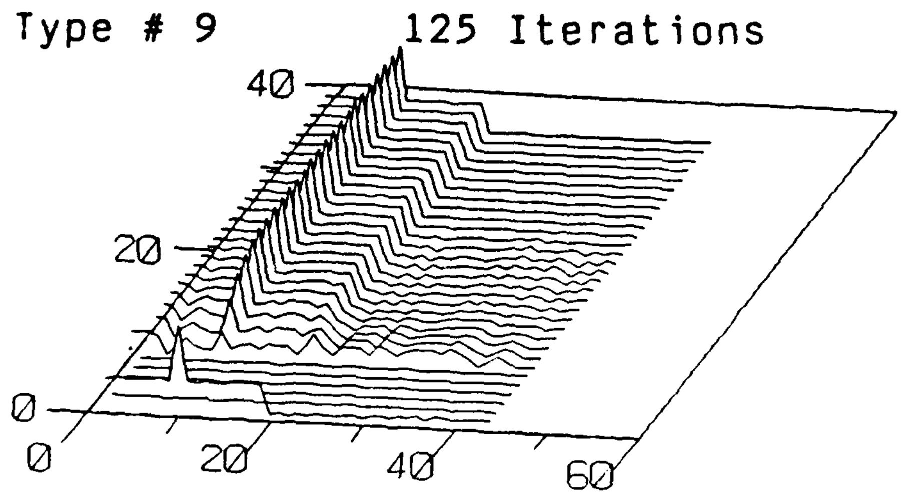
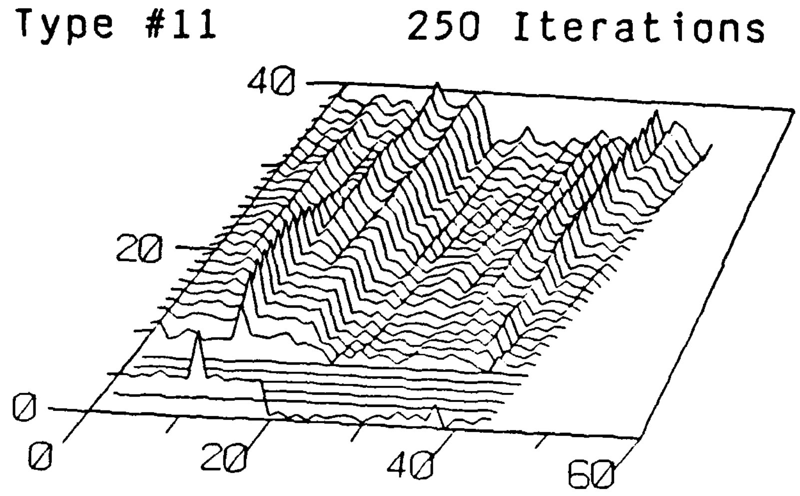

Type # 1
125 Iterations
0
20
40
0
20
40
60
(a)
Type # 3
125 Iterations
0
20
40
0
20
40
60
(b)
Type # 5
125 Iterations
0
20
40
0
20
40
60
(c)
Type # 7
125 Iterations
0
20
40
0
20
40
60
(d)

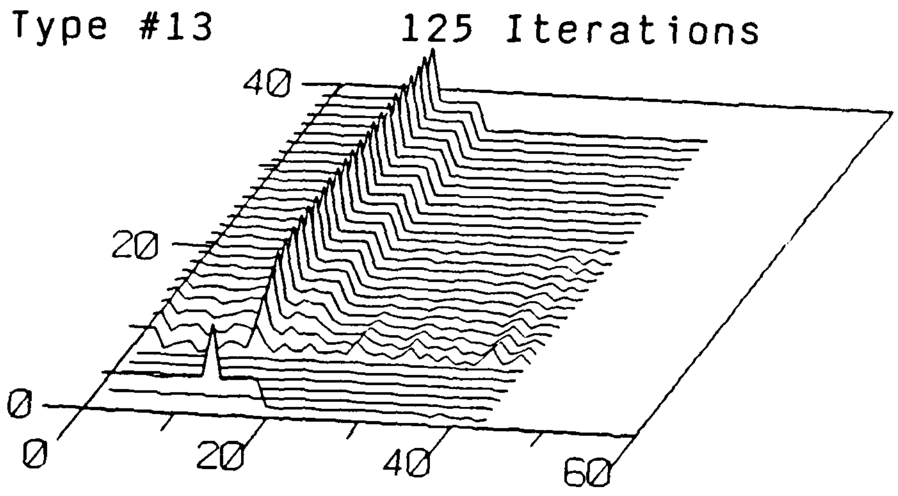
Figure 10


Type # 9
125 Iterations
40
20
0
0
20
40
60

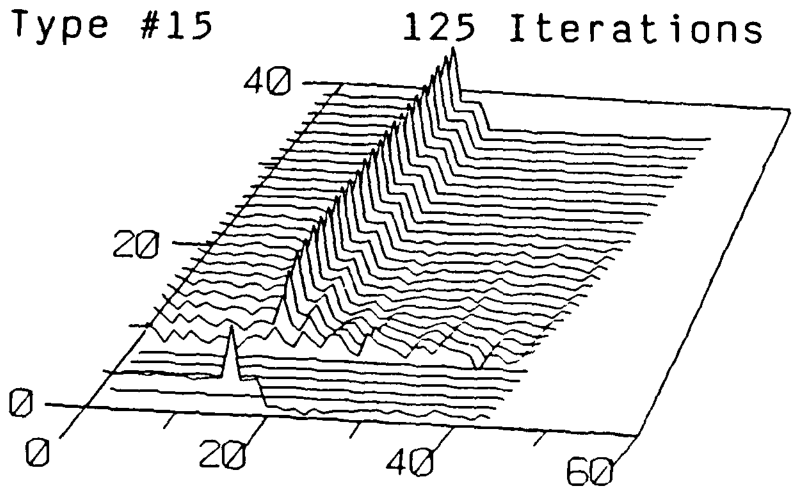
(e)


Type #11
250 Iterations
40
20
0
0
20
40
60

(f)


Type #13
125 Iterations
40
20
0
0
20
40
60

(g)


Type #15
125 Iterations
40
20
0
0
20
40
60

(h)

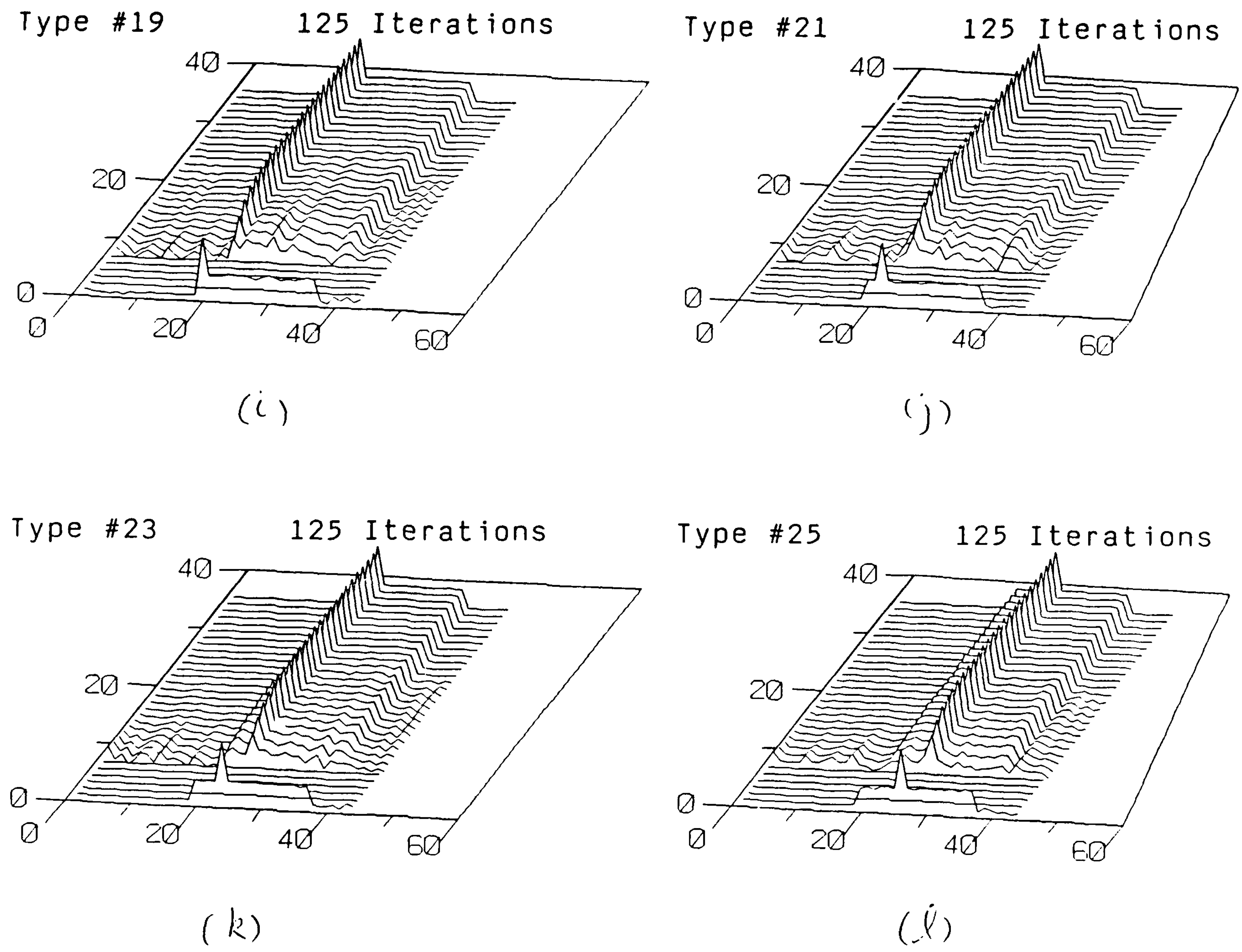

Type #19
125 Iterations
(i)
Type #21
125 Iterations
(j)
Type #23
125 Iterations
(k)
Type #25
125 Iterations
(l)

Figure 10

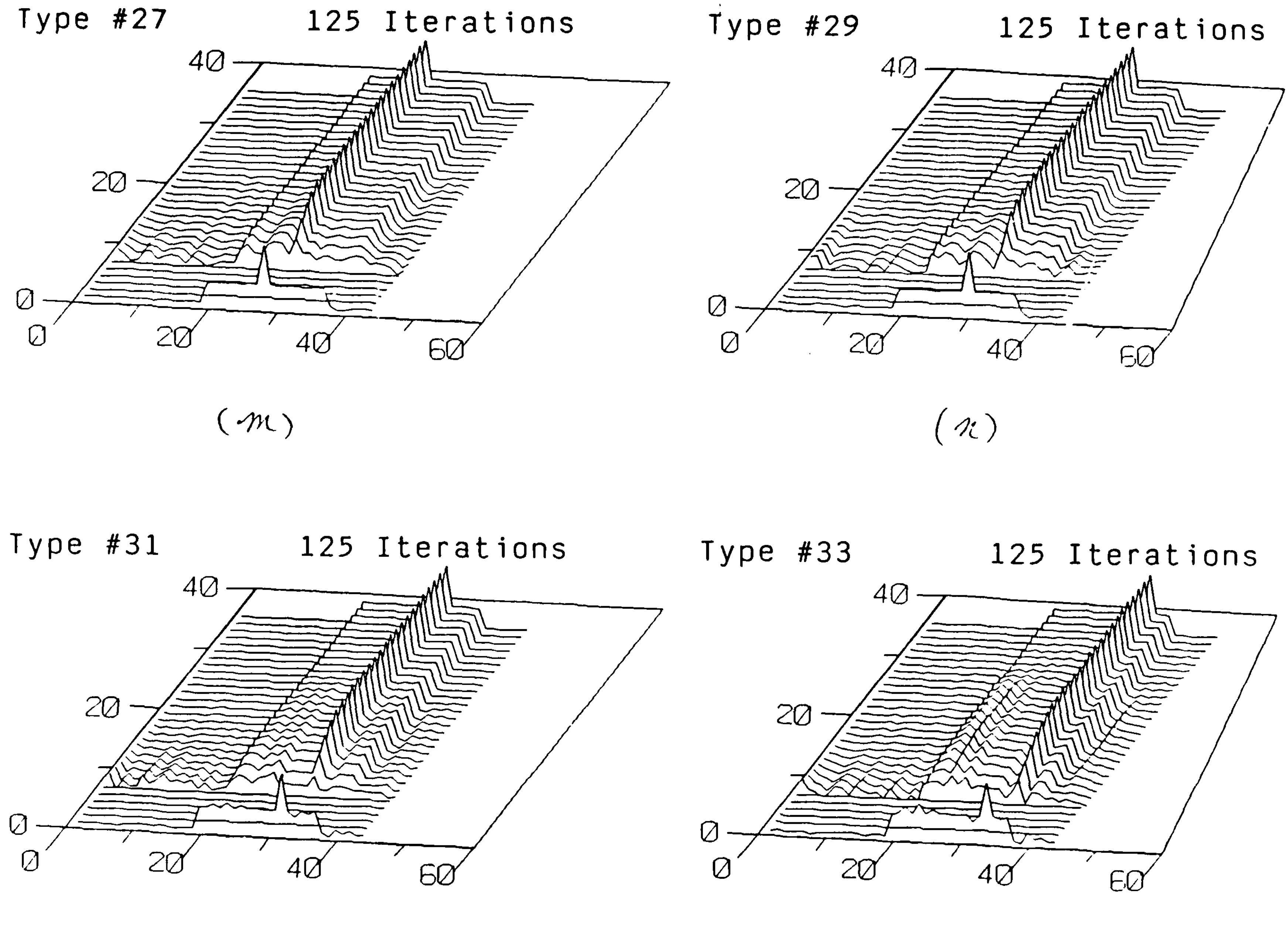
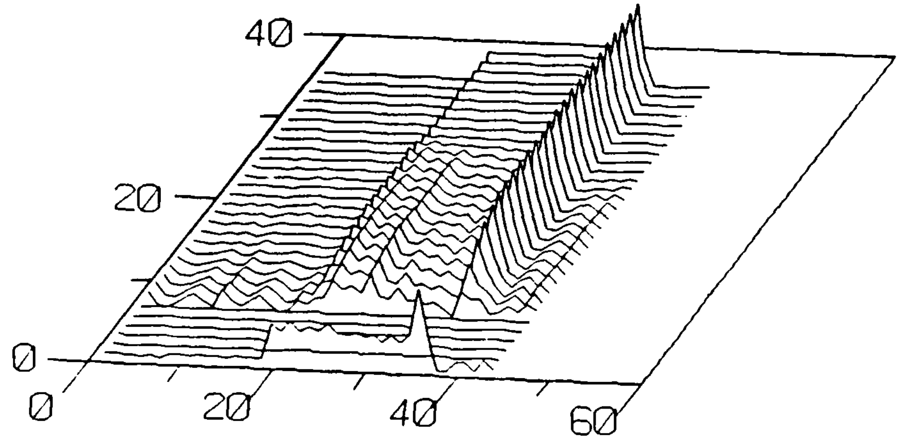

Type #27
125 Iterations
40
20
0
0
20
40
60
Type #29
125 Iterations
40
20
0
0
20
40
60
Type #31
125 Iterations
40
20
0
0
20
40
60
Type #33
125 Iterations
40
20
0
0
20
40
60
(m)
(n)
(o)
(p)

Figure 10

Type #35 125 Iterations



(g)

Figure 10

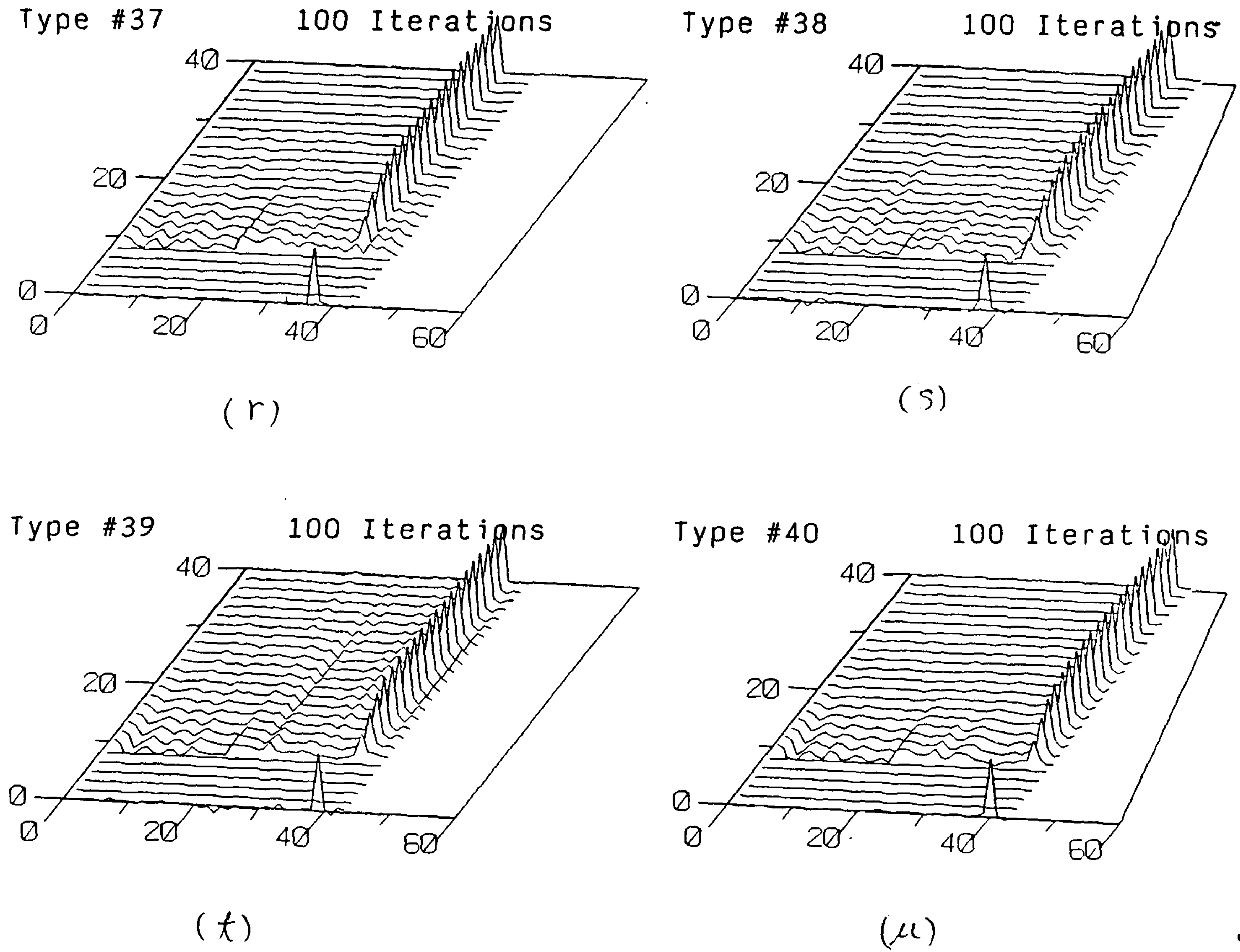
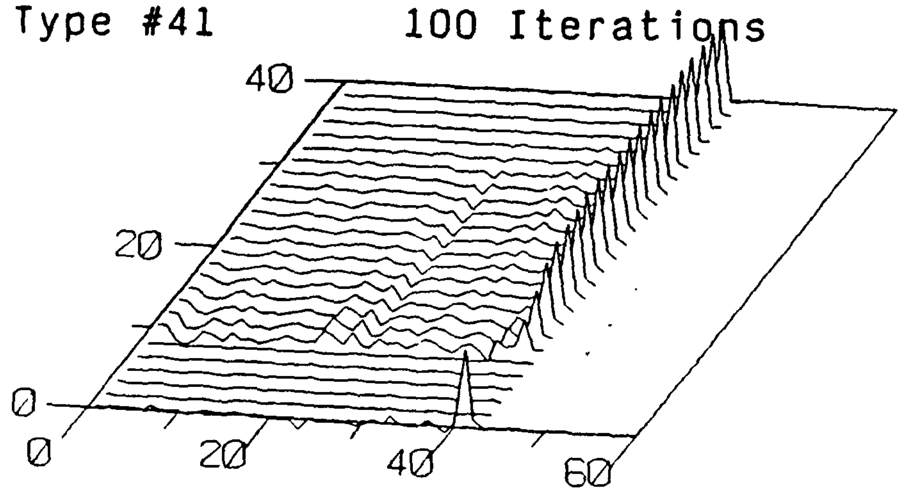
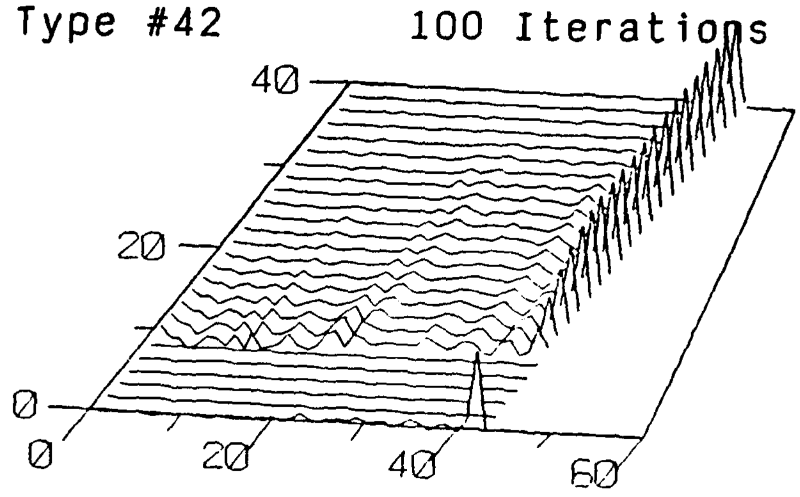

Type #37
100 Iterations
40
20
0
0
20
40
60
(r)
Type #38
100 Iterations
40
20
0
0
20
40
60
(s)
Type #39
100 Iterations
40
20
0
0
20
40
60
(t)
Type #40
100 Iterations
40
20
0
0
20
40
60
(u)

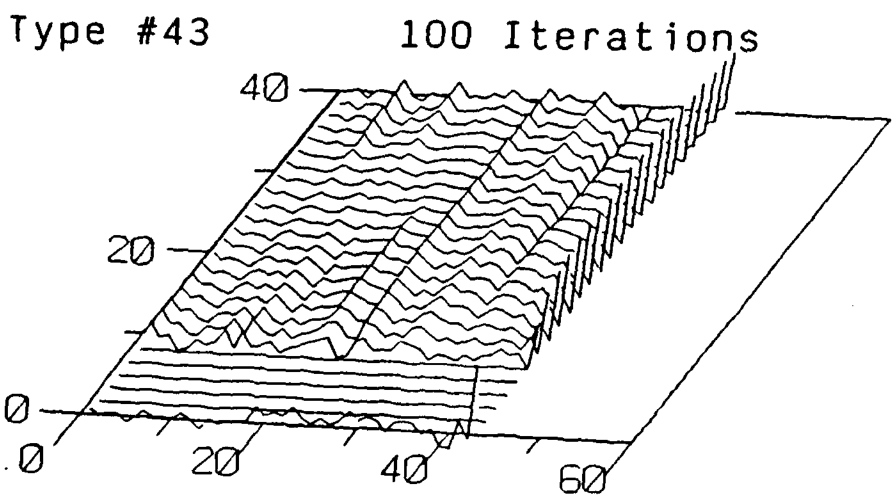
Figure 10


Type #41
100 Iterations
40
20
0
0
20
40
60

(υ)


Type #42
100 Iterations
40
20
0
0
20
40
60

(ω)


Type #43
100 Iterations
40
20
0
0
20
40
60

(χ)

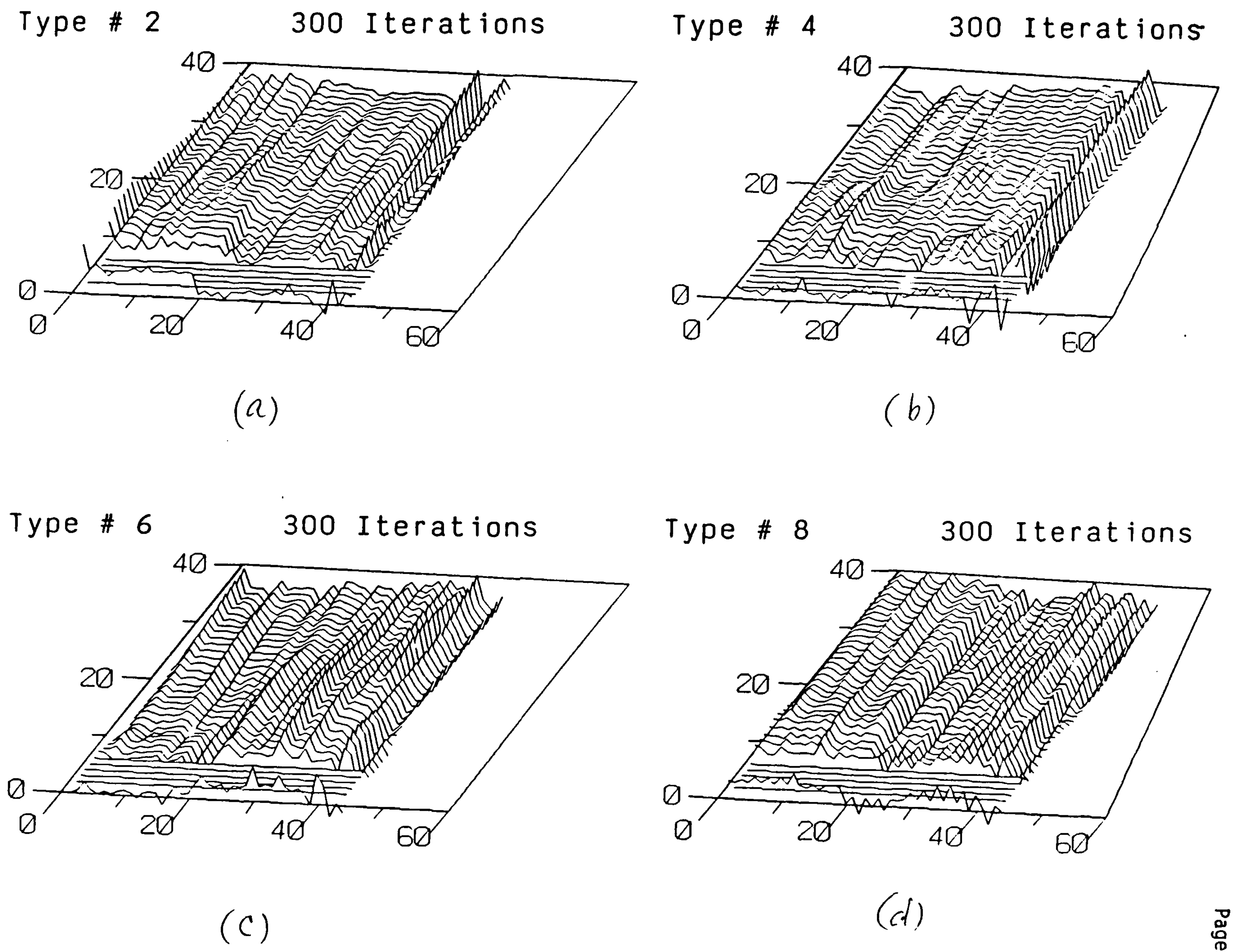
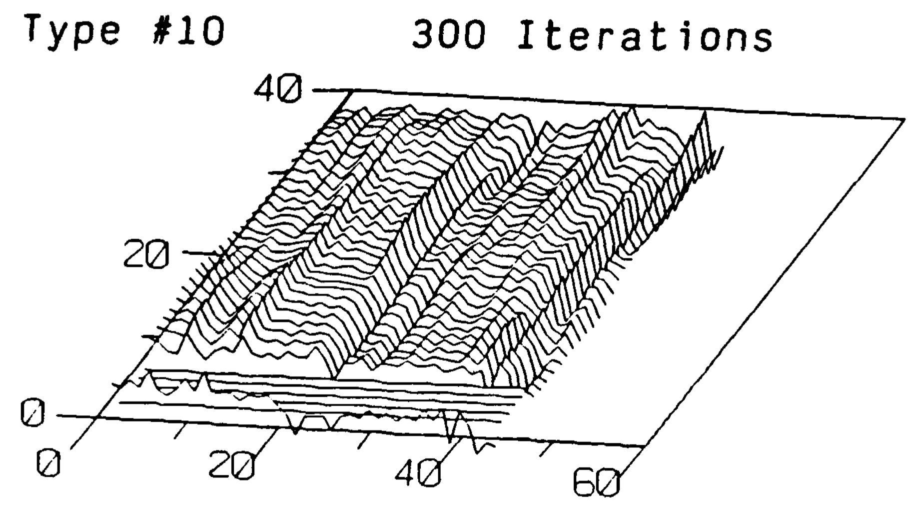
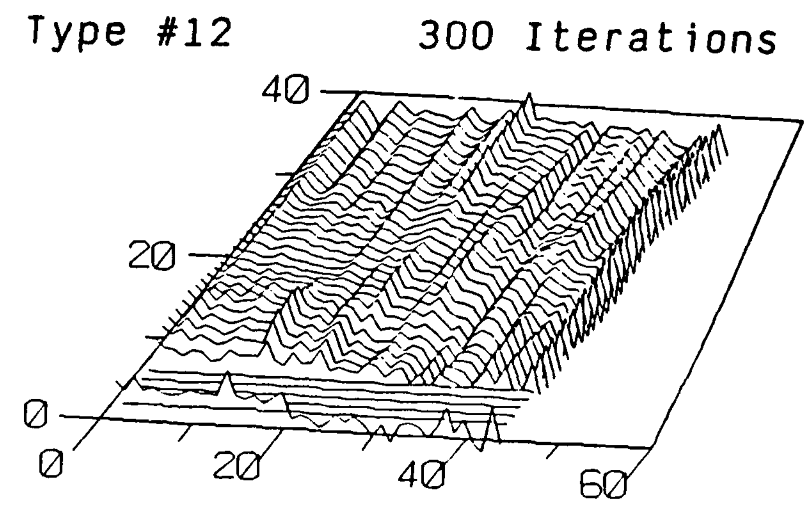

Type # 2
300 Iterations
(a)
Type # 4
300 Iterations
(b)
Type # 6
300 Iterations
(c)
Type # 8
300 Iterations
(d)

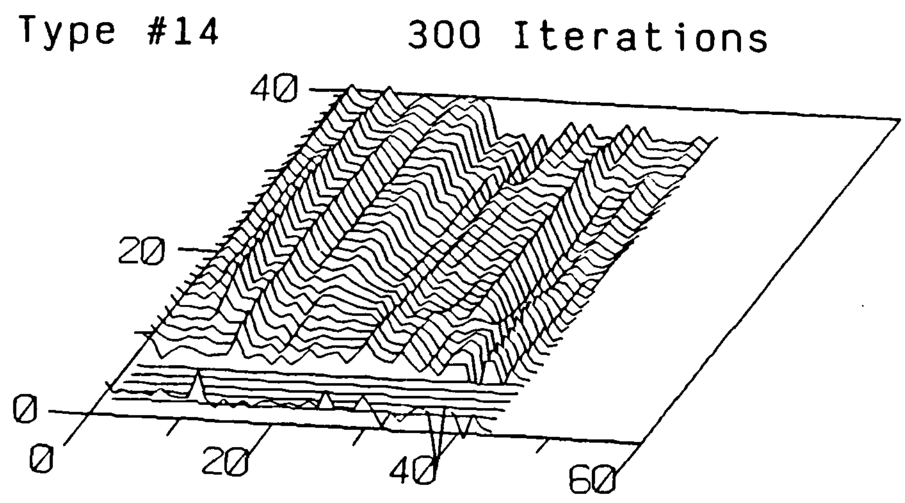
Figure 11


Type #10
300 Iterations
40
20
0
0
20
40
60

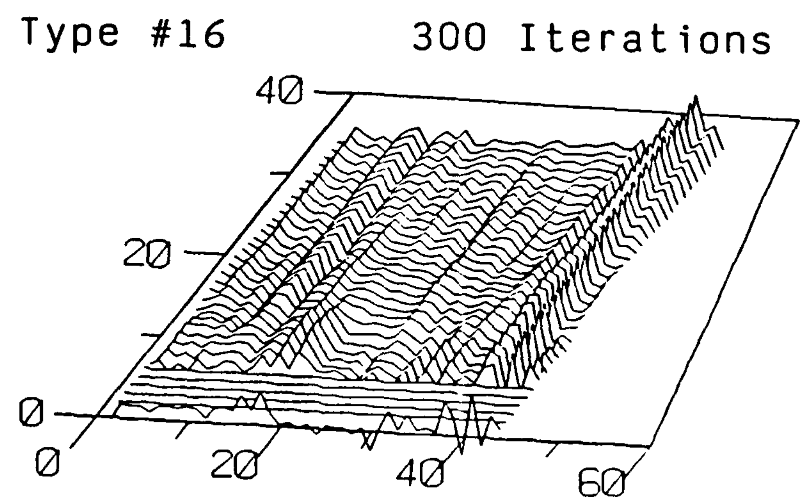
(e)


Type #12
300 Iterations
40
20
0
0
20
40
60

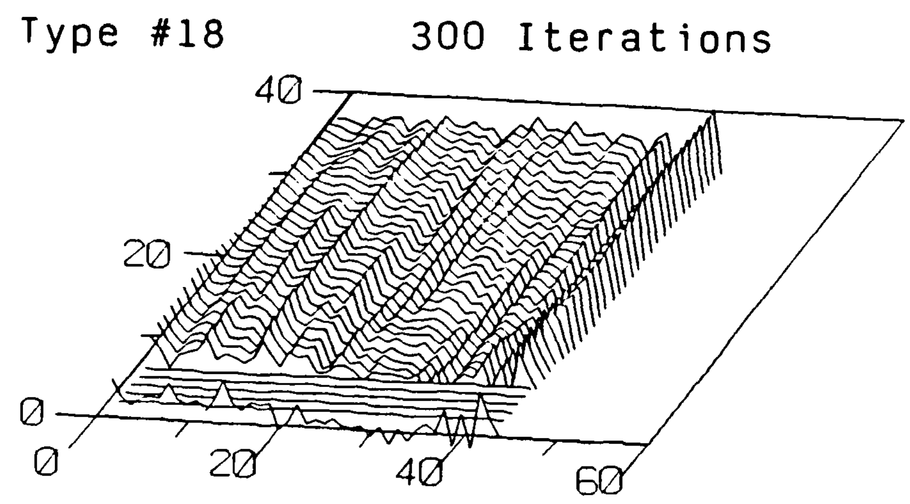
(f)


Type #14
300 Iterations
40
20
0
0
20
40
60

(g)


Type #16
300 Iterations
40
20
0
0
20
40
60

(h)



(i)

Figure 11

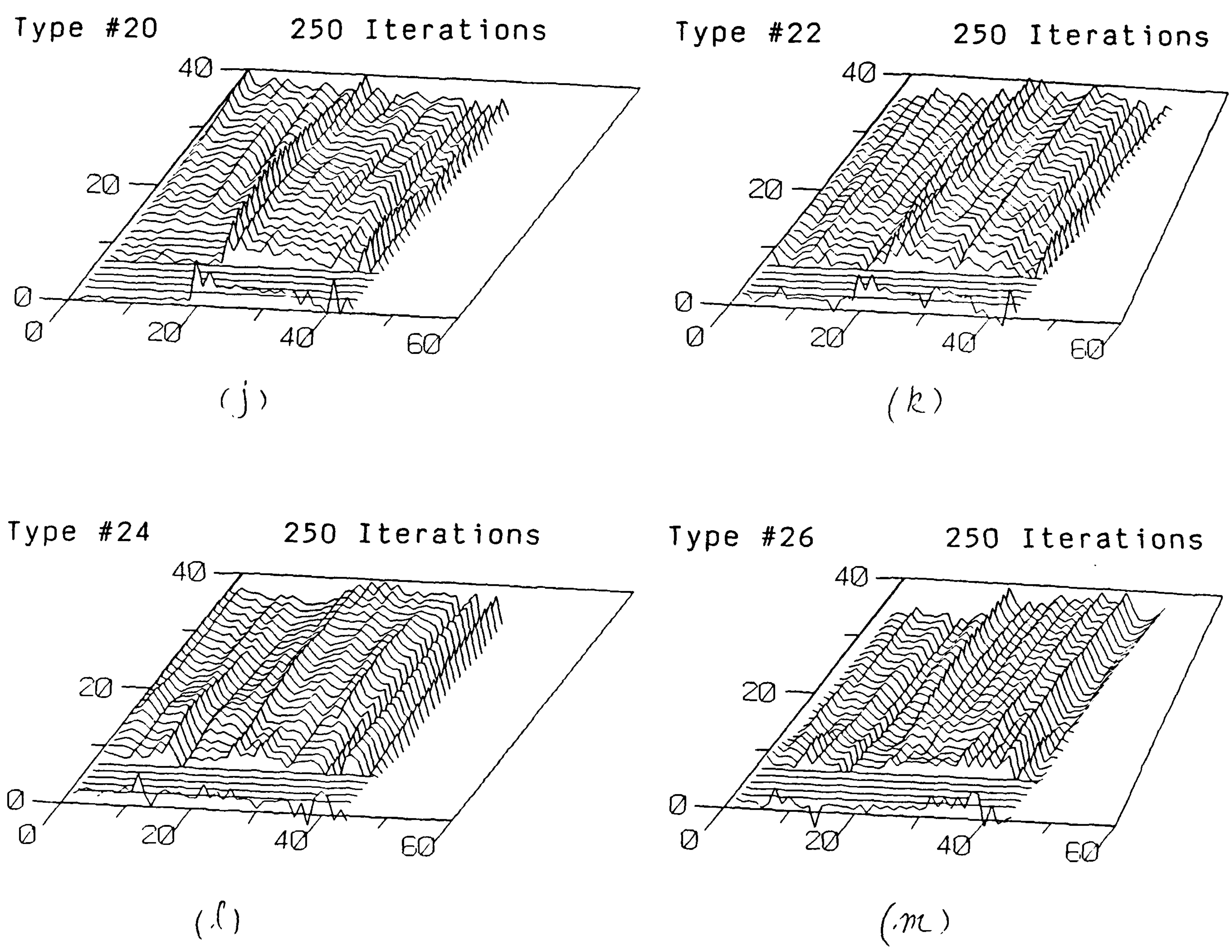

Type #20
250 Iterations
40
20
0
0
20
40
60
(j)
Type #22
250 Iterations
40
20
0
0
20
40
60
(k)
Type #24
250 Iterations
40
20
0
0
20
40
60
(l)
Type #26
250 Iterations
40
20
0
0
20
40
60
(m)

Figure 11

Data File Returned from LEARN.FOR

5000 TRIALS TOTAL
LCONST= 0.950000

VEC.#	Presentations	Cost Function
VEC: 1	LEARNED *242	0.999727
VEC: 2	LEARNED * 0	0.400495
VEC: 3	LEARNED *205	0.998903
VEC: 4	LEARNED * 0	0.093674
VEC: 5	LEARNED *240	0.993832
VEC: 6	LEARNED * 0	-0.061028
VEC: 7	LEARNED *223	0.987504
VEC: 8	LEARNED * 0	0.290609
VEC: 9	LEARNED *231	1.001384
VEC: 10	LEARNED * 0	0.370661
VEC: 11	LEARNED *200	0.919102
VEC: 12	LEARNED * 0	0.335900
VEC: 13	LEARNED *233	0.986664
VEC: 14	LEARNED * 0	0.234245
VEC: 15	LEARNED *200	1.005720
VEC: 16	LEARNED * 0	0.111282
VEC: 17	LEARNED *210	1.000441
VEC: 18	LEARNED * 0	0.302537
VEC: 19	LEARNED *203	0.985734
VEC: 20	LEARNED * 0	0.266765
VEC: 21	LEARNED *212	0.998938
VEC: 22	LEARNED * 0	0.276436
VEC: 23	LEARNED *231	0.970380
VEC: 24	LEARNED * 0	0.151559
VEC: 25	LEARNED *242	0.983373
VEC: 26	LEARNED * 0	0.128402
VEC: 27	LEARNED *241	0.992709
VEC: 28	LEARNED * 0	0.134533
VEC: 29	LEARNED *232	0.990835
VEC: 30	LEARNED * 0	0.502620
VEC: 31	LEARNED *217	0.942663
VEC: 32	LEARNED * 0	0.181624
VEC: 33	LEARNED *231	0.850625
VEC: 34	LEARNED * 0	0.177523
VEC: 35	LEARNED *224	0.803234
VEC: 36	LEARNED * 0	0.334827
VEC: 37	LEARNED *158	0.942247
VEC: 38	LEARNED *143	0.853462
VEC: 39	LEARNED *133	0.878983
VEC: 40	LEARNED *141	0.964195
VEC: 41	LEARNED *131	0.908487
VEC: 42	LEARNED *136	0.900051
VEC: 43	LEARNED *141	0.998985

Table 1

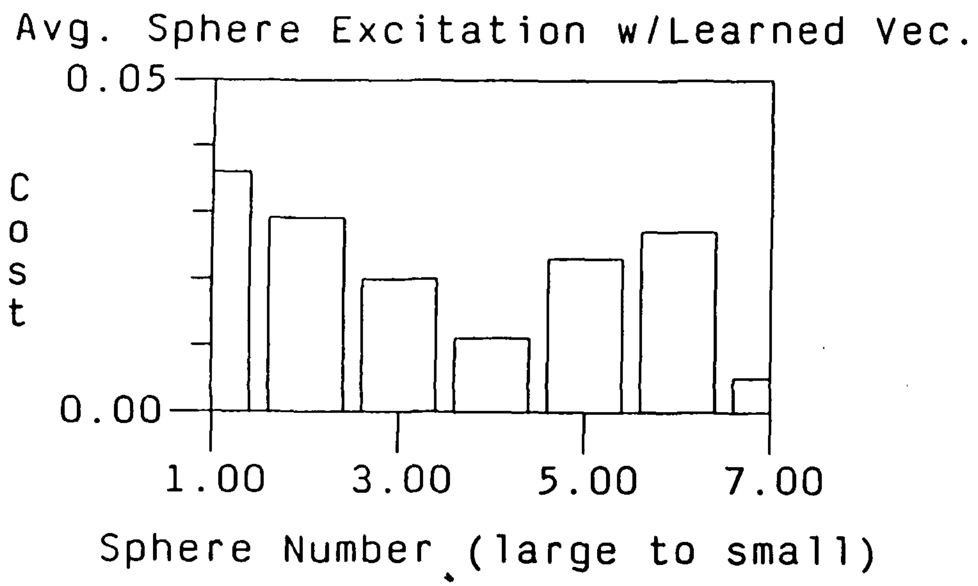

Avg. Sphere Excitation w/Learned Vec.
0.05
0.00
Cost
1.00
3.00
5.00
7.00
Sphere Number (large to small)

Figure 12

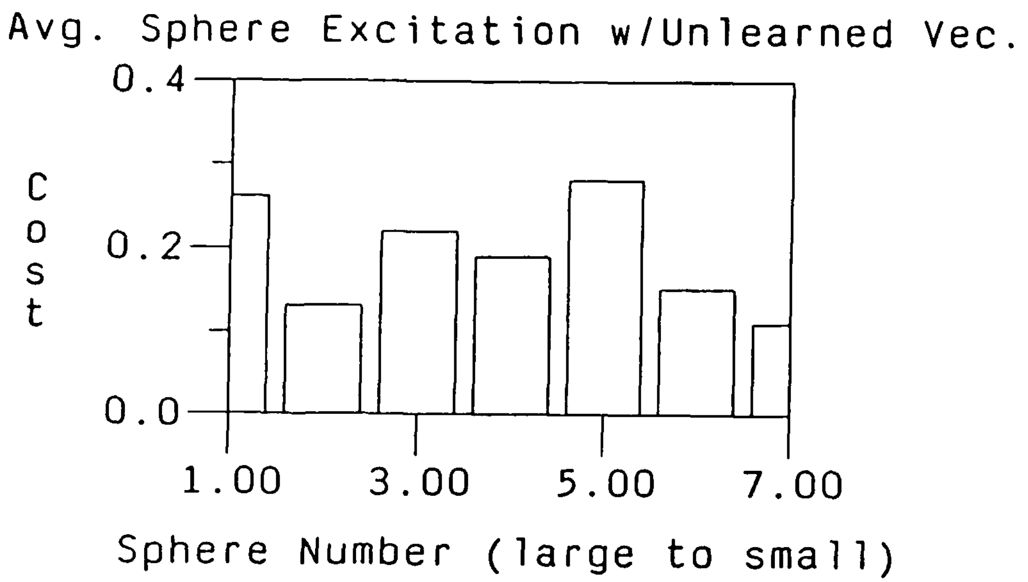

Avg. Sphere Excitation w/Unlearned Vec.
Cost
0.4
0.2
0.0
1.00
3.00
5.00
7.00
Sphere Number (large to small)

Figure 13

```
      PROGRAM LEARN
      CHARACTER DUMMYCHAR*30
      INTEGER INDUM,SEED, SHIFT(18)
      REAL DVEC(96,43)
      REAL INVEC(96,43),RRVAL(43),AMP(43)
      INTEGER TYPCNT(43),ISHARRAY(96)
      REAL OUTTYP(32,9), OUTVEC(96,43)
      REAL MATRIX(96,96)
      REAL LCONST,TVEC(96),TVEC2(96)

      WRITE(*,*) 'THIS PROGRAM LEARNS ONLY ODD ORIENTATIONS OF MEALWM'
      WRITE(*,*) '      AND DISKS'
      WRITE(*,*) 'INPUT NUMBER OF TRIALS TOTAL:'
      READ(*,900) NTRIALS
900   FORMAT(I5)
      WRITE(*,*) 'INPUT LEARNING CONSTANT (NEAR 0.80):'
      READ(*,901) LCONST
901   FORMAT(F10.6)
      SEED = 1678891
      RSUM=RAN(SEED)
      OPEN (10,FILE='MEALWM.VEC',STATUS='OLD')
      DO 1 I=1,18
      READ (10,'(A20)')DUMMYCHAR
      DO 2 J=1,80
      READ (10,'(I8)') INDUM
      DVEC(J,I)=(INDUM-2048)/2048.0
2     CONTINUE
      READ (10,'(I4)') SHIFT(I)
1     CONTINUE
      CLOSE( 10)
      DO 3 I=1,18
      CALL NORMALIZE(DVEC(1,I),64)
      DO 3 J=1,64
3     INVEC(J,I)=DVEC(J,I)

      OPEN (10,FILE='DISC12.VEC',STATUS='OLD')
      DO 4 I=1,18
      READ (10,'(A20)')DUMMYCHAR
      DO 5 J=1,80
      READ (10,'(I8)') INDUM
      DVEC(J,I)=(INDUM-2048)/2048.0
5     CONTINUE
      READ (10,'(I4)') SHIFT(I)
4     CONTINUE
      CLOSE ( 10)
      DO 6 I=1,18
      CALL NORMALIZE(DVEC(1,I),64)
      DO 6 J=1,64
6     INVEC(J,I+18)=DVEC(J,I)

      OPEN (10,FILE='SPHERES.VEC',STATUS='OLD')
      DO 7 I=1,7
      READ (10,'(A20)')DUMMYCHAR
      DO 8 J=1,80
      READ (10,'(I8)') INDUM
      DVEC(J,I)=(INDUM-2048)/2048.0
8     CONTINUE
      READ (10,'(I4)') SHIFT(I)
7     CONTINUE
      DO 9 I=1,7
```

```
      CALL NORMALIZE(DVEC(1,I),64)
      DO 9 J=1,64
9     INVEC(J,I+36)=DVEC(J,I)
      CLOSE(10)

      OPEN(10,FILE='AMPLITUD.DAT',STATUS='OLD')
      READ(10,'(F15.7)') (AMP(I),I=1,43)
      CLOSE(10)
C     AMPLITUDES RANGE FROM ABOUT 0.1--3.6  OR 0--4
      DO 10 ITYP=1,43
      ICURS=INT((LOG(AMP(ITYP))+2.3)*26.0/3.65+0.5555)+1
      DO 12 J= 1,32
      DVEC(J,ITYP)=0.0
12    IF ((J.GE.ICURS).AND.(J.LE.ICURS+5)) DVEC(J,ITYP)=1.0
      CALL NORMALIZE (DVEC(1,ITYP),32)
      DO 14 J=1,32
14    INVEC(J+64,ITYP)=0.5*DVEC(J,ITYP)
10    CONTINUE
      CLOSE(10)

      OPEN(10,FILE='SHUFFLE.DAT',STATUS='OLD')
      READ(10,'(I3)') (ISHARRAY(I),I=1,96)

      OPEN (11,FILE='WALSH64.DAT',STATUS='OLD')
      DO 20 I=1,43
      DO 20 J=1,64
20    READ(11,200) OUTVEC(J+32,I)
200   FORMAT(F3.1)
      CLOSE ( 11)

C     *** Read in 32dim walsh codes for type field of vector
C     *** we'll need 1+1+7=9 of them for 1*mealwm,1*disc12, and 7*sphere
      OPEN (11,FILE='WALSH32.DAT',STATUS='OLD')
      DO 30 I=1,9
      DO 30 J=1,32
30    READ (11,200) OUTTYP(J,I)
C     *** Assign type 1 to mealworm vectors
      DO 32 I=1,18
      DO 32 J=1,32
32    OUTVEC(J,I)=OUTTYP(J,1)
C     *** Assign type 2 to disc12 vectors
      DO 34 I=1,18
      DO 34 J=1,32
34    OUTVEC(J,I+18)=OUTTYP(J,2)
C     *** Assign types 3-9 to spheres
      DO 36 I=1,7
      DO 36 J=1,32
36    OUTVEC(J,I+36)=OUTTYP(J,I+2)
      CLOSE(11)

      OPEN(11,FILE='VECTORS.GPH',STATUS='NEW')
      WRITE(11,'(A20)') 'VECTORS'
      WRITE(11,'(A20)') 'X-AXIS'
      WRITE(11,'(A20)') 'Y-AXIS'
      WRITE(11,'(A20)') 'Z-AXIS'
      WRITE(11,'(I5)') 43*96
      DO 40 I=1,43
      CALL NORMALIZE(OUTVEC(1,I),96)
      CALL NORMALIZE(INVEC(1,I),96)
      DO 45 J=1,96
```

```
45        DVEC(J,I)=INVEC(J,I)
          DO 46 J=1,96
46        INVEC(J,I)=DVEC(ISHARRAY(J),I)
          WRITE(11,'(F7.4)') (INVEC(U,I),U=1,96)
40        TYPCNT(I)=0


          DO 50 I=1,96
          DO 50 J=1,96
50        MATRIX(J,I)=0.0
          WRITE(*,*)
          WRITE(*,*)
          WRITE(*,*)

          DO 75 ITVAL=1,NTRIALS
          ISUM=INT(RAN(SEED)*100)+1
          IF ((ISUM.GT.0).AND.(ISUM.LE.40)) IVEC=INT(RAN(SEED)*9)*2+1
          IF ((ISUM.GT.40).AND.(ISUM.LE.80)) IVEC=INT(RAN(SEED)*9)*2+19
          IF (ISUM.GT.80) IVEC=INT(RAN(SEED)*7)+37
          TYPCNT(IVEC)=TYPCNT(IVEC)+1
C         WRITE(*,*) 'TRIAL:',ITVAL,'  TYPE:',IVEC
C         *****MATRIX TIMES VECTOR
          DO 60 J=1,96
          TVEC2(J)=0.0
          TVEC(J)=0.0
          DO 60 R=1,96
          TVEC2(J)=TVEC2(J)+MATRIX(R,J)*OUTVEC(R,IVEC)
60        TVEC(J)=TVEC(J)+MATRIX(R,J)*INVEC(R,IVEC)
C         *****SUBTRACT VECTORS & SCALAR MULTIPLY
          DO 65 J=1,96
          TVEC2(J)=(OUTVEC(J,IVEC)-TVEC2(J))*LCONST
65        TVEC(J)=(OUTVEC(J,IVEC)-TVEC(J))*LCONST
C         *****OUTER PRODUCT & MATRIX ADD
          DO 70 J=1,96
          DO 70 R=1,96
          MATRIX(R,J)=MATRIX(R,J)+TVEC2(J)*OUTVEC(R,IVEC)
70        MATRIX(R,J)=MATRIX(R,J)+TVEC(J)*INVEC(R,IVEC)
75        CONTINUE

          OPEN(12,FILE='MATRIX.DAT',STATUS='NEW')
          WRITE(12,*)
          WRITE(12,499) NTRIALS
499       FORMAT(' NUMBER OF TRIALS TOTAL= ',I7)
          WRITE(12,498) LCONST
498       FORMAT(' LEARNING CONSTANT= ',F10.6)
          DO 82 I=1,96
          DO 82 J=1,96
82        WRITE(12,221) MATRIX(J,I)
221       FORMAT(F10.6)
          CLOSE (12)

          OPEN(12,FILE='LEARN.DAT',STATUS='NEW')
220       FORMAT(' VECTOR NUMBER, NUMBER OF PRESENTATIONS, COST FUNCTION')
          WRITE(12,222) NTRIALS
222       FORMAT(' ',I5,' TRIALS TOTAL')
          WRITE(12,223) LCONST
223       FORMAT(' LCONST= ',F10.6)
          WRITE(12,220)
          RMEAN=0.0
          DO 100 I=1,43
```

```
      PROGRAM DISCRIM
      CHARACTER DUMMYCHAR*40,CHDUM*1
      CHARACTER*20 MATFIL*20
      INTEGER INDUM,SEED, SHIFT(18)
      REAL DVEC(96,43)
      REAL INVEC(96,43)
      INTEGER TYPCNT(43),ISHARRAY(96)
      REAL OUTTYP(32,9), OUTVEC(96,43)
      REAL MATRIX(96,96),THRESH
      REAL TVEC(96),GVEC(96),HEAT,AMP(43),BVEC(96),RANVEC(96)
      REAL RARRAY(50),RDEC,RFEED,MAXLEN
      DATA  MATFIL / 'MAT.DAT            '/

      SEED = -1678711
      THRESH= 1.40
C     SET RDECAY TO 75 ITERATION HALF LIFE
      RDEC=0.9908
      RNQUOT=0.0
      RFEED=0.05
      RBIAS=0.0
      IHDECAY=50
      IHSTART=50
C     RHDECAY=(0.5)**(1/IHDECAY)
      IINT=10
      HEAT=0.0
      INPFLG=1
      RSUM=RAN(SEED)
      WRITE(*,*) 'READING INPUT VECTORS'
      OPEN (10,FILE='MEALWM.VEC',STATUS='OLD')
      DO 1 I=1,18
      READ (10,'(A20)')DUMMYCHAR
      DO 2 J=1,80
      READ (10,'(I8)') INDUM
      DVEC(J,I)=(INDUM-2048)/2048.0
2     CONTINUE
      READ (10,'(I4)') SHIFT(I)
1     CONTINUE
      CLOSE( 10)
      DO 3 I=1,18
      CALL NORMALIZE(DVEC(1,I),64)
      DO 3 J=1,64
3     INVEC(J,I)=DVEC(J,I)

      OPEN (10,FILE='DISC12.VEC',STATUS='OLD')
      DO 4 I=1,18
      READ (10,'(A20)')DUMMYCHAR
      DO 5 J=1,80
      READ (10,'(I8)') INDUM
      DVEC(J,I)=(INDUM-2048)/2048.0
5     CONTINUE
      READ (10,'(I4)') SHIFT(I)
4     CONTINUE
      CLOSE ( 10)
      DO 6 I=1,18
      CALL NORMALIZE(DVEC(1,I),64)
      DO 6 J=1,64
6     INVEC(J,I+18)=DVEC(J,I)

      OPEN (10,FILE='SPHERES.VEC',STATUS='OLD')
      DO 7 I=1,7
```

```
      READ (10,'(A20)')DUMMYCHAR
      DO 8 J=1,80
      READ (10,'(I8)') INDUM
      DVEC(J,I)=(INDUM-2048)/2048.0
8     CONTINUE
      READ (10,'(I4)') SHIFT(I)
7     CONTINUE
      DO 9 I=1,7
      CALL NORMALIZE(DVEC(1,I),64)
      DO 9 J=1,64
9     INVEC(J,I+36)=DVEC(J,I)
      CLOSE(10)

      WRITE(*,*) 'READING SONAR AMPLITUDES'
      OPEN (10,FILE='AMPLITUD.DAT',STATUS='OLD')
      READ(10,'(F15.7)') (AMP(I),I=1,43)
      DO 10 ITYP=1,43
      ICURS=INT((LOG(AMP(ITYP))+2.3)*26.0/3.65+0.5555)+1
      DO 12 J=1,32
      DVEC(J,ITYP)=0.0
12    IF ((J.GE.ICURS).AND.(J.LE.ICURS+5)) DVEC(J,ITYP)=1.0
      CALL NORMALIZE(DVEC(1,ITYP),32)
      DO 14 J=1,32
14    INVEC(J+64,ITYP)=0.5*DVEC(J,ITYP)
10    CONTINUE
      CLOSE(10)

      OPEN(10,FILE='SHUFFLE.DAT',STATUS='OLD')
      READ(10,'(I3)') (ISHARRAY(I),I=1,96)


      WRITE(*,*) 'READING OUTPUT VECTORS'
      OPEN (11,FILE='WALSH64.DAT',STATUS='OLD')
      DO 20 I=1,43
      DO 20 J=1,64
20    READ(11,97) OUTVEC(J+32,I)
97    FORMAT(F3.1)
      CLOSE ( 11)

C     *** Read in 32dim walsh codes for type field of vector
C     *** we'll need 1+1+7=9 of them for 1*mealwm,1*disc12, and 7*sphere
      OPEN (11,FILE='WALSH32.DAT',STATUS='OLD')
      DO 30 I=1,9
      DO 30 J=1,32
30    READ (11,97) OUTTYP(J,I)
C     *** Assign type 1 to mealworm vectors
      DO 32 I=1,18
      DO 32 J=1,32
32    OUTVEC(J,I)=OUTTYP(J,1)
C     *** Assign type 2 to disc12 vectors
      DO 34 I=1,18
      DO 34 J=1,32
34    OUTVEC(J,I+18)=OUTTYP(J,2)
C     *** Assign types 3-9 to spheres
      DO 36 I=1,7
      DO 36 J=1,32
36    OUTVEC(J,I+36)=OUTTYP(J,I+2)

      DO 40 I=1,43
      CALL NORMALIZE(OUTVEC(1,I),96)
```

```
      CALL NORMALIZE(INVEC(I,I),96)
      DO 45 J=1,96
45    DVEC(J,I)=INVEC(J,I)
      DO 46 J=1,96
46    INVEC(J,I)=DVEC(ISHARRAY(J),I)
40    TYPCNT(I)=0

      DO 50 I=1,96
      DO 50 J=1,96
50    MATRIX(J,I)=0.0
      WRITE(*,*)
      WRITE(*,*)
      WRITE(*,*)

192   WRITE(*,*) 'READING MATRIX ',MATFIL
      OPEN(12,FILE=MATFIL,STATUS='OLD')
      READ(12,99) DUMMYCHAR
      READ(12,99) DUMMYCHAR
99    FORMAT( A40)
      READ(12,99) DUMMYCHAR
      DO 82 I=1,96
      DO 82 J=1,96
82    READ(12,98) MATRIX(J,I)
98    FORMAT(F10.6)
      CLOSE (12)


190   CONTINUE
199   WRITE(*,*) 'INPUT TYPE 1-43  (0 TO CHANGE PARAMTERS):'
      READ(*,200) ITYP
200   FORMAT(I7)
      IF (ITYP.NE.0) GOTO 196
191   WRITE(*,*)
      WRITE(*,*) 'T .......  CHANGE THRESHOLD'
      WRITE(*,*) 'I .......  RUN INPUT VECTORS'
      WRITE(*,*) 'O .......  RUN OUTPUT VECTORS'
      WRITE(*,*) 'D .......  CHANGE DECAY/FEEDBACK PARAMETERS'
      WRITE(*,*) 'N .......  CHANGE THE NOISE QUOTIENT'
      WRITE(*,*) 'F .......  CHANGE INTERVAL BETWEEN GRAPH FRAMES'
      WRITE(*,*) 'M .......  READ IN NEW MATRIX'
      READ (*,'(A1)') CHDUM
      IF (CHDUM.NE.'T') GOTO 181
      WRITE(*,*)
      WRITE(*,*) 'OLD THRESHOLD= +/-',THRESH
      WRITE(*,*) 'INPUT NEW THRESHOLD *R:'
      READ (*,'(F10.6)') THRESH
      WRITE(*,*) 'THRESHOLD CHANGED TO +/-',THRESH
      GOTO 191
181   IF (CHDUM.NE.'I') GOTO 182
      INPFLG=1
      WRITE(*,*)
      WRITE(*,*) 'NOW RUNNING INPUT VECTORS'
      GOTO 191
182   IF (CHDUM.NE.'O') GOTO 183
      INPFLG=0
      WRITE(*,*)
      WRITE(*,*) 'NOW RUNNING OUTPUT VECTORS'
      GOTO 191
183   IF (CHDUM.NE.'D') GOTO 184
      WRITE(*,*)
```

```
          WRITE(12,201) RSUM
 201      FORMAT(F10.6)
          IF (I.EQ.ITYP) RARRAY(T)=RSUM
 140      CONTINUE
 C        WRITE(*,*)
          WRITE(*,*)
 150      CONTINUE
          IVAL=ITERS
          WRITE(12,202) IVAL
 202      FORMAT(I4)
          DO 160 T=1,ITERS
 160      WRITE(12,201) RARRAY(T)
 197      CONTINUE
          CLOSE(12)
          GOTO 190


          END



          SUBROUTINE NORMALIZE(VECTOR,DIM)
          REAL VECTOR(100),RSUM
          INTEGER DIM
          RSUM=0.0
          DO 25 J=1,DIM
 25       RSUM=RSUM+VECTOR(J)*VECTOR(J)
          RSUM=SQRT(RSUM)
          IF (RSUM.LT.(0.0000001)) GOTO 50
          DO 30 J=1,DIM
 30       VECTOR(J)=VECTOR(J)/RSUM
          RETURN
 50       WRITE(*,*) 'NORMALIZE ZERO VECTOR !!!!!!!!!!!!!!!!!!!!!!'
          RETURN
          END

 C        SUBROUTINE BOIL(VECTOR,RHEAT,SEED,RHDECAY)
 C        REAL VECTOR(96),THRESH,KICK,RHDECAY,RHEAT,X1,X2,S
 C        INTEGER SEED
 C
 C        DO 10 I=1,96
 C        5        X1= RAN(SEED)*2.0-1.0
 C        X2= RAN(SEED)*2.0-1.0
 C        S=X1*X1+X2*X2
 C        IF (S.GE.(1.0)) GOTO 5
 C        X1=X1*SQRT(-2.0*LOG(S)/S)
 C        KICK=ABS(X1)*2.0*RHEAT/100.0
 C        IF (KICK.GT.2) KICK=2.0
 C        VECTOR(I)=VECTOR(I)- KICK*VECTOR(I)
 C        10       CONTINUE
 C        RHEAT=RHEAT*RHDECAY
 C        RETURN
 C        END


          SUBROUTINE BOIL(VECTOR,HEAT,SEED,RHDECAY)
          REAL VECTOR(96),THRESH,HEAT,KICK,RHDECAY,RHEAT
          INTEGER SEED,IHEAT
          IHEAT=INT(HEAT)
          DO 5 I=1,IHEAT
```

DISTRIBUTION LIST

Dr. Earl Alluisi
Office of the Deputy Under
Secretary of Defence
OUSDRE (E&LS)
Pentagon, Rm. 3D129
Washington, DC 20301

Areospace Psychology Department
Naval Aerospace Medical
Research Lab
Pensacola, FL 32508

Dean of Academic Departments
U.S. Naval Academy
Annapolis, MD 21402

Director, Technical Information
Division
Code 2627
Naval Research Laboratory
Washington, DC 20375-5000

Dr. Raymond M. Fitzgerald
Code 1125AO
Office of Naval Research
800 N. Quincy Street
Arlington, VA 22217-5000

Capt. William M. Houk
Commanding Officer
Naval Medical R & D Command
Bethesda, MD 20814-5055

CDR Thomas Jones
Code 125
Office of Naval Research
800 North Quincy Street
Arlington, VA 22217-5000

Perceptual Science Program
Office of Naval Research
Code 1142PS
800 North Quincy Street
Arlington, VA 22217-5000

Dr. Gary Poock
Operations Research Department
Naval Postgraduate School
Monterey, CA 93940

Dr. A. L. Slafkosky
Scientific Advisor
Commandant of the Marine Corps
Washington, DC 20380

Special Assistant for Marine
Corps Matters
Code OOMC
Ofice of Naval Research
800 North Quincy Street
Arlington, VA 22217-5000

Dr. Thomas Hanna
Auditory & Communication
Sciences Department
Naval Submarine Medical
Research Lab.
Naval Submarine Base
Groton, CT 06340-5900

Dr. Edgar M. Johnson
Technical Director
U.S. Army Research Institute
Alexandria, VA 22333-5600

Technical Director
U.S. Army Human Engineering
Laboratory
Aberdeen Proving Ground, MD
21005

Dr. A. D. Baddeley
Director, Applied Psychology
Unit
Medical Research Council
15 Chaucer Road
Cambridge CB2 2EF ENGLAND

Mr. Charles Bates, Director
Human Engineering Division
USAF AMRL/HES
Wright-Patterson AFB, OH 45433

Dr. Kenneth R. Boff
AF AMRL/HE
Wright-Patterson AFB, OH 45433

Dr. J. Tangney
Life Sciences Directorate, AFSOR
Bolling AFB, Bldg. 410
Washington, DC 20032-6448

Defense Technical Information
 Center
Cameron Station, Bldg. 5
Alexandria, VA 22314

Dr. Clinton Kelly
Defense Advanced Research
 Projects Agency
1400 Wilson Blvd.
Arlington, VA 22209

Dr. Alan Leshner
Division of Behavioral & Neural
 Sciences
National Science Foundation
1800 G. Street, N.W.
Washington, DC 20550

Dr. M.C. Montemerlo
Information Sciences & Human
 Factors Code RC
NASA HQS
Washington, DC 20546

Dr. James Ballas
George Mason University
4400 University Drive
Fairfax, VA 22030

Dr. James H. Howard, Jr.
Department of Psychology
Catholic University
Washington, DC 20064

Dr. H.P. Van Cott
NAS-National Research Council
 (COHF)
2101 Constitution Avenue, N.W.
Washington, DC 20418