KERN, Frank John, 1931–
AN APPLICATION OF PATTERN RECOGNITION TECHNIQUES TO SOCIAL SYSTEMS MODELING.

The University of Oklahoma, Ph.D., 1966
Engineering, electrical

University Microfilms, Inc., Ann Arbor, Michigan

THE UNIVERSITY OF OKLAHOMA

GRADUATE COLLEGE

# AN APPLICATION OF PATTERN RECOGNITION TECHNIQUES TO SOCIAL SYSTEMS MODELING

A DISSERTATION

SUBMITTED TO THE GRADUATE FACULTY

in partial fulfillment of the requirements for the

degree of

DOCTOR OF PHILOSOPHY

BY

FRANK JOHN KERN

Norman, Oklahoma

1966

AN APPLICATION OF PATTERN RECOGNITION TECHNIQUES

TO SOCIAL SYSTEMS MODELING

APPROVED BY

DISSERTATION COMMITTEE

## ACKNOWLEDGEMENT

The author wishes to express his sincere gratitude to the many individuals from whom assistance has been obtained in the preparation of this thesis. In particular, the author wishes to thank Dr. R. A. Terry, School of Industrial Engineering of the University of Oklahoma, for his invaluable assistance in the Robbers Cave simulation, and to Dr. J. D. Palmer, Director of the School of Electrical Engineering at the University of Oklahoma, under whose patient guidance and council this program was performed.

The advice and suggestions of Dr. J. A. Nickel, Chairman, Mathematics Department, Oklahoma City University, are sincerely appreciated. Thanks are also extended to the various members of the Doctoral Advisory Committee for their interest in this work.

Finally, thanks are extended to Dr. Muzafer Sherif of the Institute of Group Relations of the University of Oklahoma for permission to utilize the results of his excellent Robbers Cave study.

## TABLE OF CONTENTS

## LIST OF TABLES

## LIST OF ILLUSTRATIONS

## LIST OF SYMBOLS AND EXPRESSIONS

$\lceil x \rceil$ The smallest integer greater than $x$.

$\begin{bmatrix} x_1 \\ \cdot \\ \cdot \\ \cdot \\ x_n \end{bmatrix}$ A vector.

$\left\{ \bar{x}_n \right\}$ Braces always denote a set.

$\bar{x}$ A bar over any quantity denotes that quantity as a vector.

$x_i$ The $i^{th}$ component of any vector is denoted by the subscript $i$.

$\bar{x}(t_j)$ Denotes a vector occurring at time $t_j$. The components of this vector would be expressed as $\left\{ x_i(t_j). \right\}$

$x$ Is always an input to a system.

$\bar{w}$ Is always a weight vector for a threshold logic unit (TLU).

$\bar{w}^k$ A superscript refers to a particular TLU.

$y^k$ The analog output of the $k^{th}$ TLU.

$z^k$ The binary output of the $k^{th}$ TLU.

$\bar{v}_i$ A vector corresponding to the $i^{th}$ output class.

Any combination of these symbols may be used. For instance

$$w_4^3(t_6)$$

denotes the fourth component of the weight vector for the third TLU which occurs at time increment six.

$\langle \bar{x}_1, \bar{x}_2 \rangle$ denotes the inner product of two vectors where

$$\bar{x}_1 = \begin{bmatrix} x_{11} \\ \cdot \\ \cdot \\ \cdot \\ x_{j1} \\ \cdot \\ \cdot \\ \cdot \\ x_{n1} \end{bmatrix} \qquad \bar{x}_2 = \begin{bmatrix} x_{12} \\ \cdot \\ \cdot \\ \cdot \\ x_{j2} \\ \cdot \\ \cdot \\ \cdot \\ x_{n2} \end{bmatrix}$$

and $\langle \bar{x}_1, \bar{x}_2 \rangle = \sum_{j=1}^{n} x_{j1}\, x_{j2}$ .

$\| \bar{x}_1 \|^2 = \langle \bar{x}_1, \bar{x}_1 \rangle$ = the square of the norm of $\bar{x}_1$ .

# AN APPLICATION OF PATTERN RECOGNITION TECHNIQUES TO SOCIAL SYSTEMS MODELING

## CHAPTER I

## INTRODUCTION

The concept of a system, at least in a formalized sense, is a relatively new one to the field of engineering. This broadened approach to the equipment design problems has been widely misinterpreted. As a result there is, as yet, no precise definition of just what system analysis or "the system approach" is. These terms rarely convey the same concept to any two hearers; therefore, no formal attempt will be made to define the rather hazy boundary between system and non-system. It will be more fruitful at this point to note certain characteristics which are present in many equipment systems. First, all elements of the system have some common purpose. It may not be rigidly controlled by a central controller but all parts of the system contribute in some manner to the production of a set of outputs from a specified set of inputs. The relationship between the variables of a system is seldom a simple one. A change in one variable generally will affect many of the others, usually in some non-linear fashion resulting in a richness of behavior which is difficult to describe with a high degree of precision. The same characterization may be noticed to hold for a large class of "non-hard ware" situations which occur in fields such as biology, economics, and the social sciences. Since there exists a large collection of techniques useful for the characterization, analysis, and synthesis of hardware systems, one is tempted to try (by analogy) to apply these same techniques, or modifications thereof, to non-hardware situations.

The applications of the "systems approach" to non-hardware systems generally have taken the form of some attempt at modeling the system in question. This paper specifically deals with a problem which occurs in the simulation of social systems, but it will be easily seen that the applications of data handling techniques described here are not necessarily restricted to sociological information.

In order to provide a basis for the introduction and discussion of these techniques, it will first be necessary to discuss a basic approach to system modeling and to examine some existing simulations of social systems. After observing a real world system in operation it may be argued that it is impossible to represent a social system (or any system for that matter) in all its complexity and variety. While this is certainly true in some situations, it is nevertheless desirable to attempt to model certain facets of the system behavior for the purpose of predicting its future performance. That is to say, certain aspects of system behavior may be describable to the extent that a model of the system may be built which under certain conditions will behave in a manner approximating the behavior of the original system.

As an example of a system consider an ordinary triode vacuum tube. In actuality it is certainly a very complex system. The behavior of the current through it is dependent on a population of electrons. Although it is possible to analytically treat the behavior of a single electron, it is not possible to describe the exact behavior of the entire population. Indeed, it is even impossible to obtain a census of this population at any given instant of time. In addition to this complication, the density and behavior of the population is affected by its environment, i.e., temperature, humidity, radiation, and supply voltage variations to name a few factors. These and other factors sometimes influence behavior in a rather vague and arbitrary manner. Yet, despite all of these complexities, it is possible under certain operating conditions to quite successfully represent this device by a single source and a resistor! In fact under these operating restrictions a generalized model of a vacuum tube may be drawn which can be said to represent the behavior of all triode vacuum tubes and may be used to test hypotheses (solve problems) involving these devices without reference to any particular tube type. To represent a particular vacuum tube some measurements are generally made in the laboratory and from these, parameters relating the theoretical model to the real world are obtained.

The foregoing discussion was not aimed at any particular justification of procedures regarding analysis of vacuum tube circuitry, but rather to illustrate a method of approach which is quite common in engineering. The employment of a "system approach" to a non-hardware problem consists largely of attempting to bring to bear the same viewpoint and techniques which have met with success in engineering problems.

In the investigation of behavior of social systems analysis and simulation could proceed based on one of two viewpoints. Studies could be undertaken purely for the purpose of testing hypotheses concerning the behavior of social systems in general. The basis for this type of investigation may either be a purely theoretical conjecture about the detailed structure of the system (internal model) or a set of empirical constructs derived from observations of the system's behavior (external model). In order to validate these modeling procedures, a second level of simulation must be undertaken aimed at relating the theoretical parameters to real world values, a procedure analogous to the laboratory measurement of vacuum tube parameters.

## CHAPTER II

## SELECTED EXAMPLES OF PREVIOUS WORK IN SOCIAL SYSTEMS MODELING

It will be instructive to look at some selected examples of previous work in the area of social systems modeling. The examples given are considered to be representative of classes of social system models which have been presented in the literature. The discussion of each model is not exhaustive in detail and is intended to present a brief view of the state-of-the-art of social system simulation. For a detailed analysis of each the reader is referred to the original article.

The first such model to be considered was developed by Gullahorn and Gullahorn (7). It is an attempt to simulate an elementary social situation using as a basis a set of propositions taken from Homans (8). These propositions are as follows:

<u>Proposition 1</u> - "If in the recent past the occurrence of a particular stimulus situation has been the occasion on which a man's activity has been rewarded, then the more similar the present stimulus situation is to the past one, the more likely he is to emit the activity, or some similar activity, now."

<u>Proposition 2</u> - "The more often within a given period of time a man's activity rewards the activity of another, the more

often the other will emit the activity."

<u>Proposition</u> <u>3</u> - "The more valuable to a man a unit of the activity another gives him, the more he will emit activity rewarded by the activity of the other."

<u>Proposition</u> <u>4</u> - "The more often a man has in the recent past received a rewarding activity from another, the less valuable any further unit of that activity becomes to him."

<u>Proposition</u> <u>5</u> - "The more to a man's disadvantage the rule of distributive justice fails of realization, the more likely he is to display the emotional behavior we call anger."

Using the programming language IPL-V a simulation has been written which represents the behavior of two hypothetical actors named Ted and George. The situation here is quite simple. Ted asks George for help (AR) and George must decide whether or not to honor the request (AE) using the previously stated propositions. A flow chart of the simulation is shown in Figure 1. If a box in the flow chart contains a question "+" indicates a "yes" answer and "-" a "no." Roman numerals indicate program segments and "P" indicates use of one of the propositions.

The simulation program is a sequence of operations on a collection of lists thus reducing a social process to one of symbol manipulation. The actors Ted and George are represented by highly specific lists of attributes including personal idiosyncracies and recent past histories. As a result the simulation emphasizes non-numerical processes and is essentially deterministic in nature. It further represents a person as a hypothesis testing information processing organism capable of

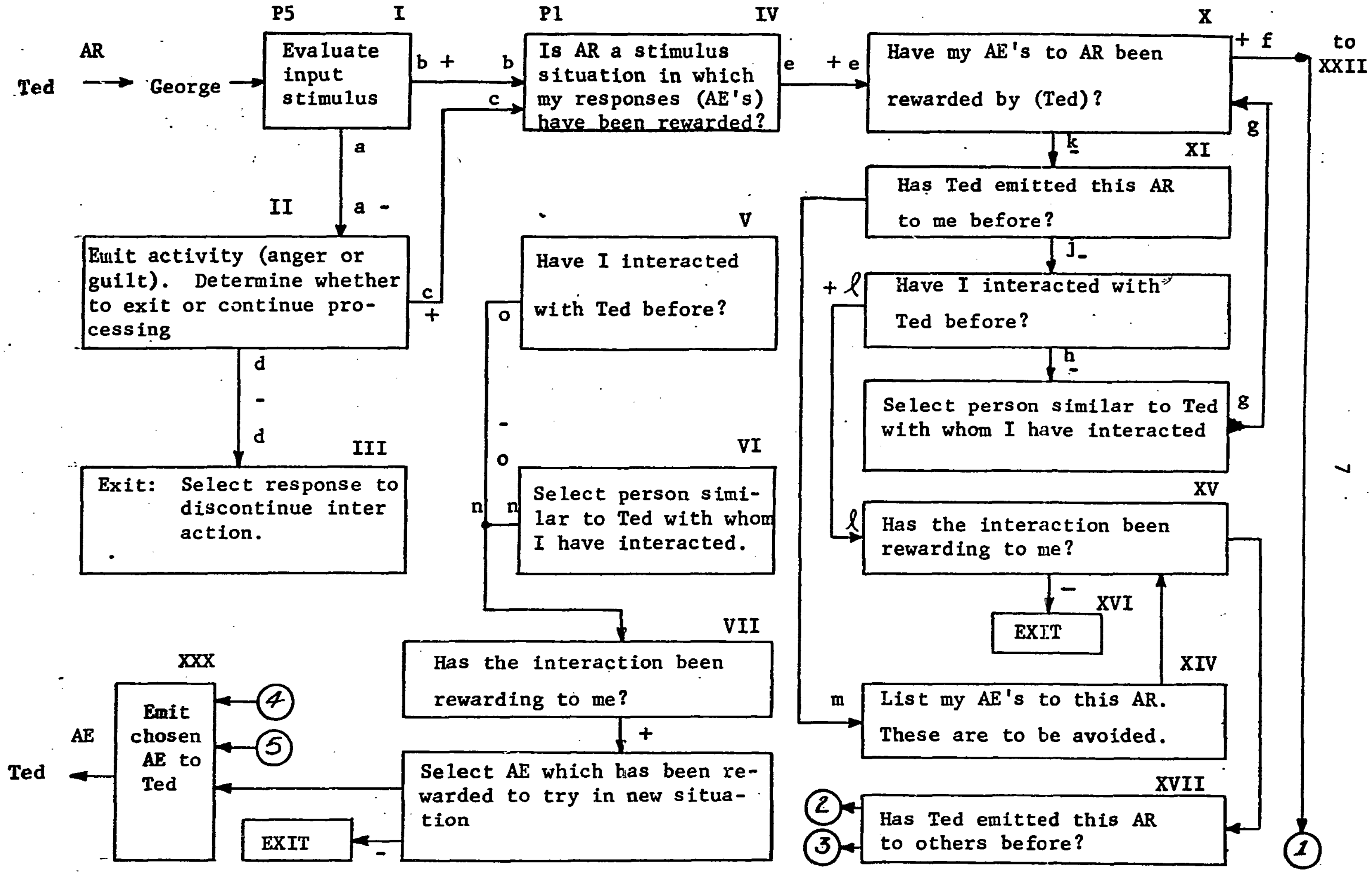



Figure 1 Flow Chart for Ted and George

Part 1

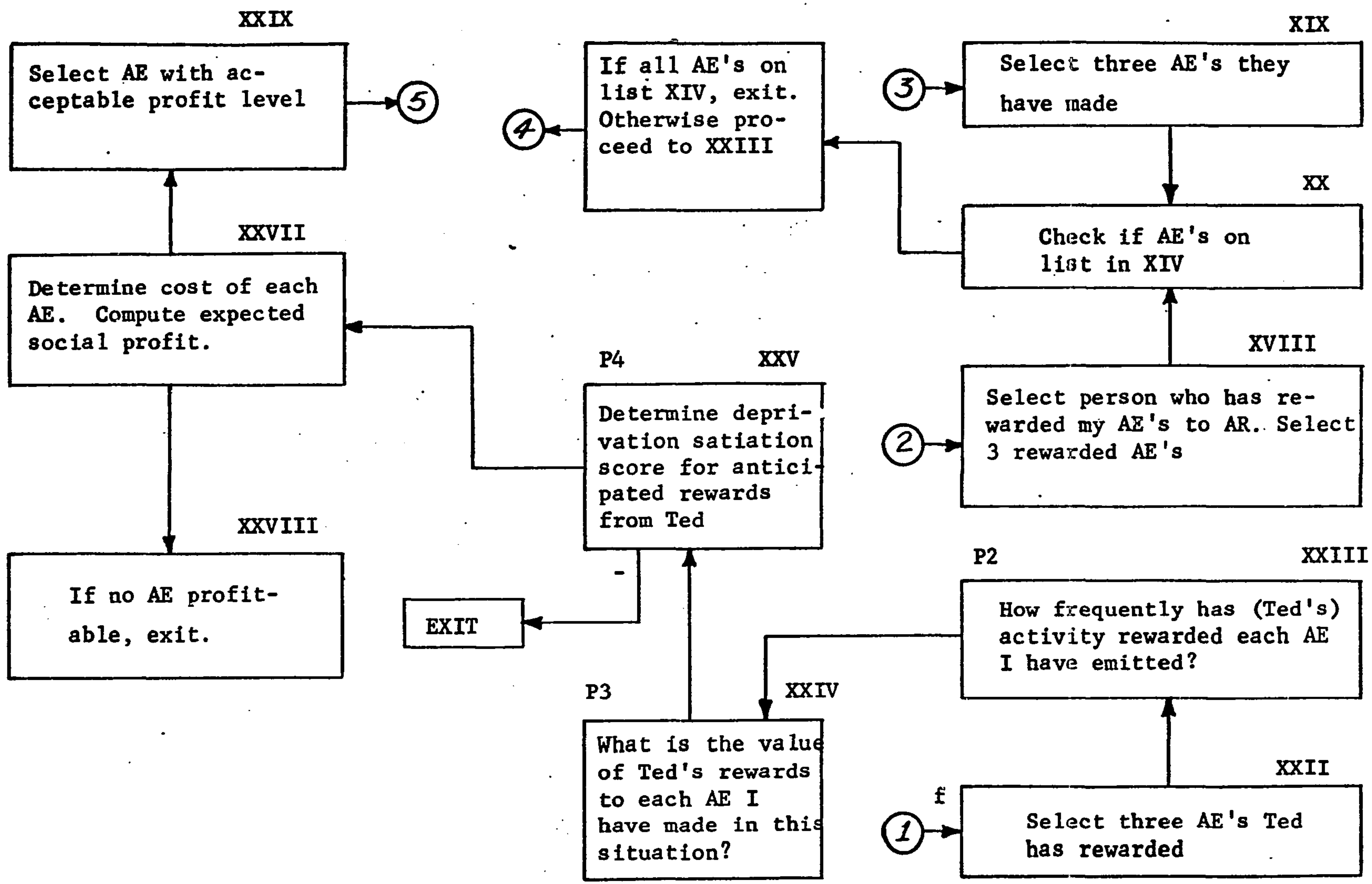
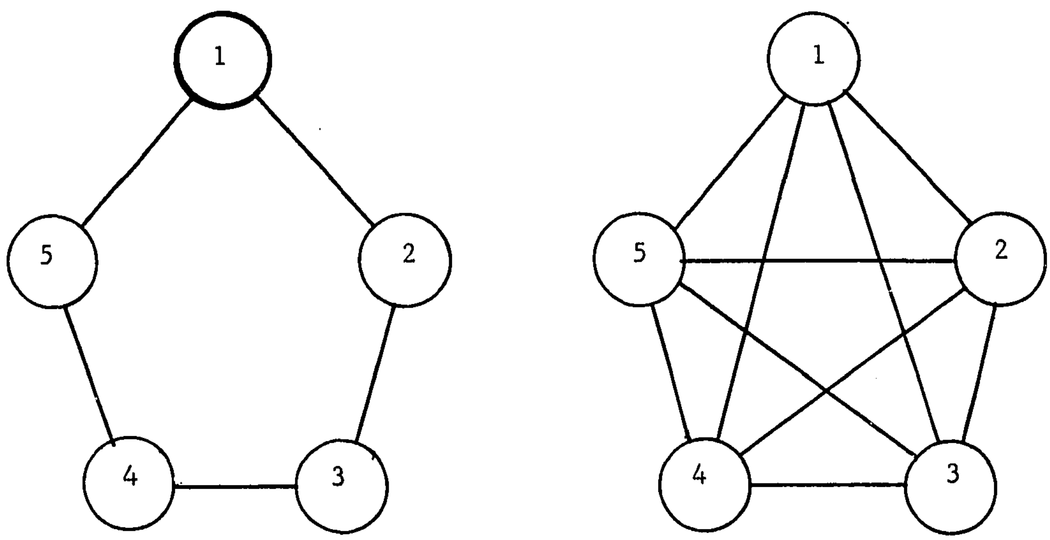



Figure 1 Flow Chart for Ted and George

Part 2

receiving, analyzing, reconstructing, and storing information. No attempt is made in this simulation to relate the hypothetical Ted and George to a real Ted and a real George, i.e., to construct personal data lists which actually represent a real person. The results of the program yielded sequences of events which in the opinion of the experimenter mapped onto real life situations. However, since this was a purely theoretical study, no attempt was made to model any real life situation.

A slightly different class of information flow and processing model has been investigated by McWhinney (11) in simulation of communication net experiments. In this set of experiments each member of the group is represented by a set of Boolean matrices. All information concerning the individual's knowledge of the structure of the net, information received and sent, recent past history, etc. was stored in some matrix in this set. To obtain initial data for these matrices, an examination of the records of previous small communication net studies was made in an effort to obtain values and relative weights of various variables. In addition some hypotheses were generated based on visual inspection of this same information. These hypotheses concerned local rationality, impatience and persistence. Local rationality implied that A will not communicate with B unless he has some information he does not know B to possess. Impatience dealt with the willingness of an individual to abandon a particular behavior if it seemed to conflict with group effectiveness during a particular trial while persistence dealt with the willingness of this same individual to try the same line of behavior in subsequent trials. Simulations of a type of experiment such as has been described were done using several different constraints on net geometry as shown in Figure 2. The results of the simulation using the circular net structure failed to yield an efficient group structure. However, comparison with empirical data showed a good correspondence to the behavior of groups which had similarly failed to develop an efficient structure. Simulated groups using the second structure failed to respond in any manner which could be related to empirical findings. The investigator feels that this lack of correspondence is due at least in part, to lack of proper information regarding values of program parameters. In the empirical data used these values could not be reliably determined using standard statistical measures.




Net for first part of simulation

Net for the second part of the simulation

Figure 2

A third example of computer simulation of a social process was developed by Coe (2) and concerns the interaction of two individuals named Alter and Ego. The simulation shown by the flow diagram in

Figure 3 is again a list processing one which attempts to simulate not only the interaction, but also the learning behavior of Alter and Ego. Both individuals are represented by lists of information such as previous contacts between the two individuals, tolerance for frustration, etc. Results of this simulation may be summarized by the following statement. The processing of these lists revolves around an algebraic scheme involving the calculation of utilities (costs and rewards) and an implementation of the frustration-aggression hypothesis. Learning is introduced through simple reinforcement. If both individuals have a high tolerance for frustration, no conflict will be generated between them. If both individuals have a low tolerance for frustration, the relationship between them will tend to be terminated at an early stage. If one individual has a high tolerance for frustration and the other a low tolerance for frustration, conflicts will tend to be solved in favor of the person who has a low frustration tolerance. The attractive feature of this study is its attempt to ask learning type questions of a learning type program. That is to say, it is conceivable that this type of simulation could be "trained" to represent a real situation by repeating the interaction and adjusting program parameters, e.g., tolerance levels, levels of perception, etc., so that previously obtained results about interactions between two people could be approximated.

It is the consensus of previous investigators in social system simulation that the role of computer simulation is primarily one of hypotheses testing. Assuming that a reasonable model of a given system can be developed, the main utility of such a model would be in the

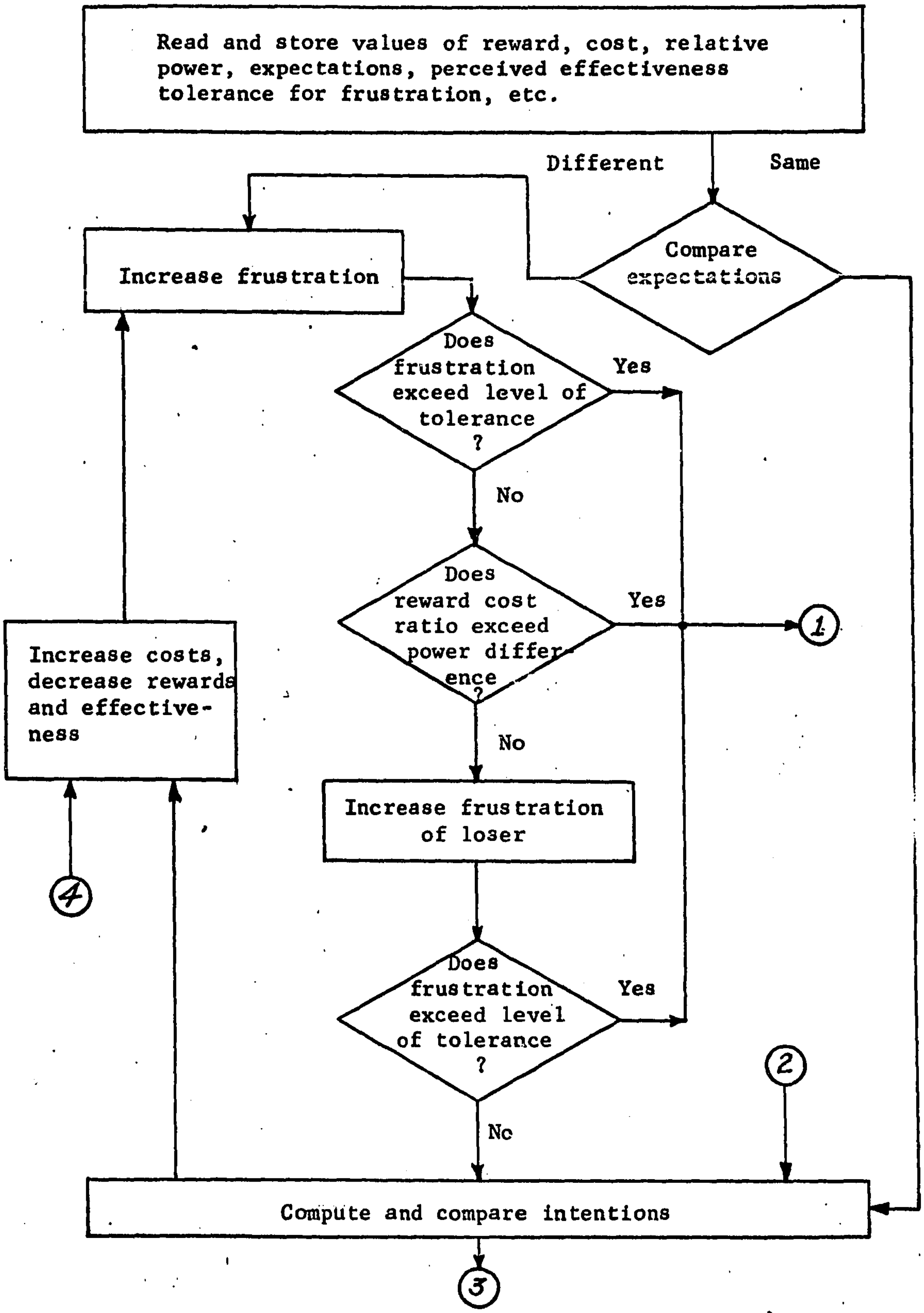



Figure 3 - Part 1

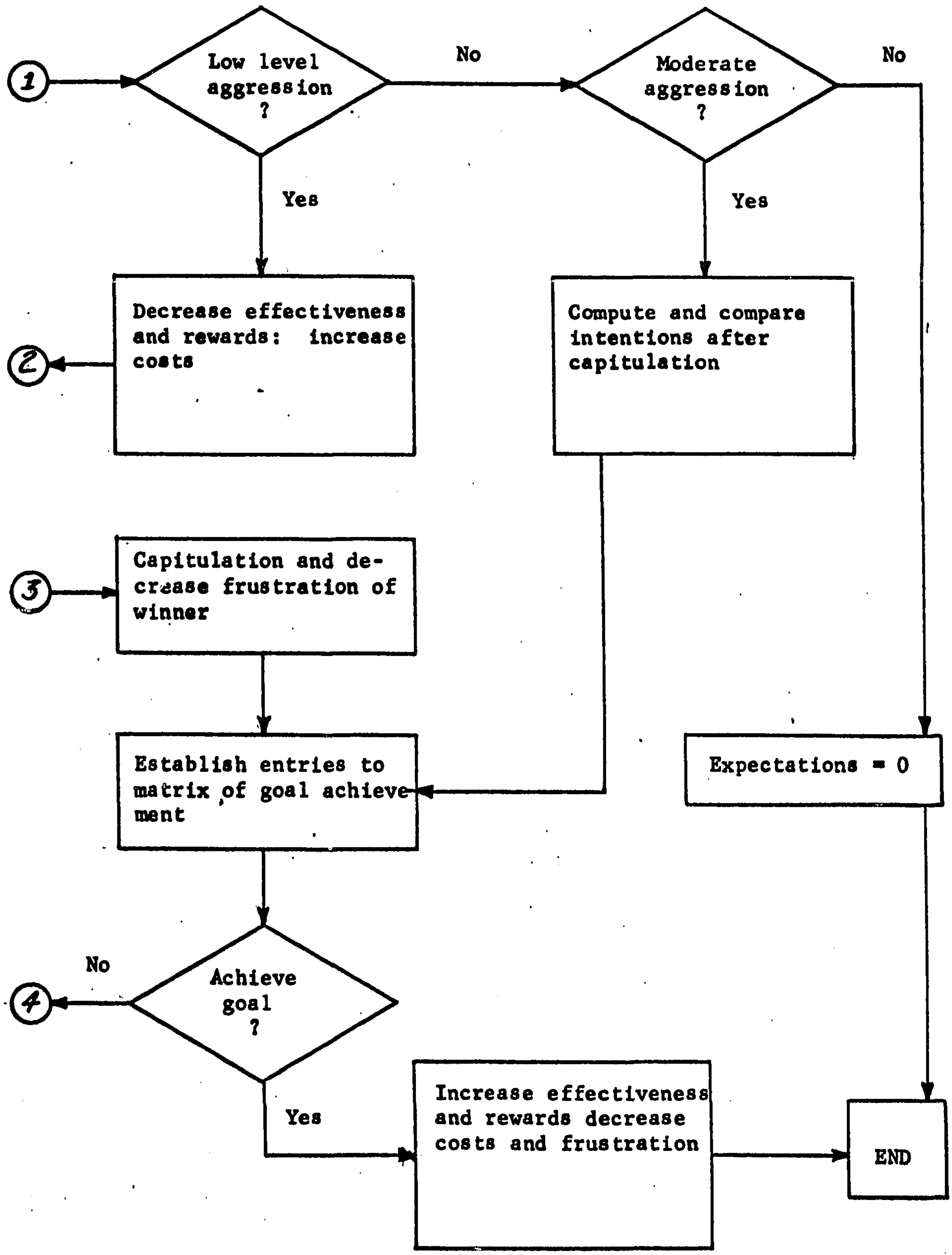



Figure 3 - Part 2

testing of hypotheses regarding behaviors of the system. In this manner several possible approaches to the solution of a particular problem could be tried with great rapidity and at much less cost when compared with the physical alternative of conducting an experiment or constructing a machine. A given system model should utilize "learning" type programs which produce new results based on past experience and evidence rather than problem solving type programs which involve solely the manipulation of arrays of numbers. For example, one could seek to determine common factors present in insurgency operations or investigate the possible impact of closing a military base in a certain area or determine the impact of a certain type of new technology on an underdeveloped country.

As yet, of course, no well developed body of social system models exists but with the increasing availability of advanced computer hardware such as associative and sketch pad memories and improved software for manipulation of pages of information in the storage hierarchy more and more sophisticated models are being developed. As was previously mentioned, there must be some means of relating these models to actual situations and vice versa if they are to have any utilitarian value. In the system model of the vacuum tube the proper parameter values may always be expressed in terms of values obtained by a rather simple measurement process. This enables one to use the model to good advantage to represent a particular vacuum tube. Unfortunately, no such simple general model of a social process exists. The relationship between variables and, in fact, the important variables themselves are, more often than not, obscured in masses of data which may or may not be relevant. Moreover, the data are usually not in quantitized form which is easily adaptable to manipulation by numeric methods. Information concerning social systems may be found in narrative form, as expressions of subjective or objective judgements, or as questionnaire type data as well as in numeric form. Because of this it is sometimes difficult to process all of the pertinent information regarding a given system by standard means such as analysis of variance, calculation of moments, or regression curve fitting techniques.

The task of data reduction in this situation becomes one of recognizing the patterns of dependency of the values of certain variables upon certain other variables. Supposing that a suitable numeric (not necessarily decimal) coding scheme can be devised, it is possible to perform operations analogous to curve fitting using "learning" machines. The use of pattern recognizing machines to discover regularities in sets of numbers has been found to yield surprising results in meteorology (9) and medicine (5). The remainder of this paper will be concerned with the adaptation of these techniques to the processing of information concerning social systems.

# CHAPTER III

## ENCODING OF INFORMATION

If one wishes to utilize a digital computer for data processing the information to be processed must be expressed in some symbolic form. For instance, the language IPL-V is a list processing language especially suited to the handling information in the form of English language statements. Similarly, the language FORTRAN is particularly well suited to the manipulation of arithmetic statements and to some extent logical statements. In choosing a scheme for processing a given class of information, the following questions could be asked:

1. In what form is the raw data?
2. What kind of operations are to be performed on this information?
3. What is the relationship between the important variables under investigation?

If, for instance, a chi-square test is to be performed on numeric information which is normally distributed, an algebraic scheme will do quite nicely. On the other hand, a list processing report generator might be used to compile questionnaire data.

The information concerning social processes appears in a variety of forms. An ordinary newspaper, for instance, contains a wealth of information about the social system of which we are a part. Reports of experiments and investigations in the social sciences are sometimes in narrative form, sometimes in the form of subjective judgements, sometimes as objective statements based on observations of the process under scrutiny, and sometimes in numeric form. In many psychological studies the number of variables is kept to a minimum so that rigid laboratory conditions may be maintained. Even a simple situation as, e.g., a communications net experiment may produce such a wealth of data and variety of behavior as to defy analysis and to produce only the most rudimentary conclusions.

In the main the results of the processing of data about a particular social system should yield at least partial answers to two questions. First, what variables are important in the production of a given line of behavior and secondly, once these variables are identified what is their relative importance, i.e., what "weight" does the process assign to each variable. Once these questions are decided then the precise functional relationship between the variables themselves may be investigated. Presently, the methods of analysis of variance, factor analysis, the calculation of correlation coefficients, and input-output analysis are employed in an effort to provide answers to the above questions. Unfortunately, these very convenient methods have limited application since existing algorithms for computing these quantities require the data to be expressed in decimal form. It is also assumed that all variables are linearly related thus ignoring all synergisms which occur. In contrast no such restrictions need be placed on data processed by a learning machine. Such a machine can be trained to classify data pertaining to a certain result by using the data as a basis for training. Once trained, the machine will then be able to classify further data pertaining to the same result. The machine "learns" to correctly classify information by assigning "weights" to input variables and thus will produce answers to the aforementioned two questions regarding critical variables. The numerical values of each of the variables need not be expressed in the same number base although for interpretive purposes it is desirable to do so. A priori statements concerning relationships between independent variables need not be made since these appear as a result of processing by the machine.

In most of the work which has been done previously the information was presented to the machine in terms of binary vectors. Accordingly, the following encoding schemes are given for encoding data into appropriate binary vectors. The information to be processed may be divided into three broad categories as follows:

Category 1: Finely quantized data. This category includes any data sets of more than 10 values, which contain a countable number of values. For instance, the set of real numbers $\left\{ r = 1 + \frac{1}{k} \,\middle|\, k = 1, 2, 3, \ldots \right\}$ would be finely quantized.

Category 2: Coarsely quantized data. Data sets which contain no more than 8 distinct values such as the set of numbers $\left\{ 0, 1, 3, 3, 6, 7, 1, 2, 4 \right\}$.

Category 3: Continuous data. All data sets which are not coarsely or finely quantized. The results of any experiment in which measurements are made at discrete intervals must necessarily be quantized. However, it is sometimes convenient to treat these results as being samplings of a set of numbers which may assume any value on some interval of the real line, e.g., sample readings taken of the voltage $e(t) = 10 \sin 25t$ at several different points in time.

For purposes of discussion a variable will be said to be continuous, coarse, or fine depending on whether the data concerning that variable is continuous, coarsely, or finely quantized.

In order to model any process the person making the model must decide which variables are independent (inputs and state variables), and which ones are dependent (outputs). If a learning machine is used to process data concerning these variables, no a priori statement need be made concerning which state variables and inputs influence which outputs. It may initially be assumed that all independent variables influence all dependent variables. Once the roles of the variables have been assigned, the block of data points must be subdivided according to time of occurrence. That is to say,

if $\bar{O}(t)$ is the output at time $t$

$\bar{S}(t)$ is the state of the process at time $t$

$\bar{I}(t)$ is the value of the inputs at time $t$

then the set

$$\sum(t_1) = \left\{\bar{O}(t_1),\ \bar{S}(t_1),\ \bar{I}(t_1)\right\}$$

is all the information concerning the system at time $t_1$ (or at about time $t_1$). After this grouping has been effected, the sets of numbers

$$\bar{X}(t_1) = \left\{\bar{S}(t_1),\ \bar{I}(t_1)\right\}$$
$$\bar{O}(t_1) = \bar{O}(t_1)$$

and encoded to form the binary vectors

$$\bar{X}(t_1) \rightarrow x(t_1) \overset{\Delta}{=} \text{the input at time } t_1$$

and $$\bar{O}(t_1) \rightarrow z(t_1) \overset{\Delta}{=} \text{the output at time } t_1$$

respectively.

One possible encoding procedure is given in Table 1 below which is self-explanatory.

Table 1. Binary Encoding Scheme

| Type of Variable | Code |
|---|---|
| Coarse<br>(Values $V_1$ through $V_8$) | Use the binary numbers zero through seven<br>$V_1 \rightarrow 000$, $V_2 \rightarrow 001$, ..., $V_8 \rightarrow 111$<br>If the variable does not assume eight values, select the binary numbers zero through $k - 1$ to encode $V_1$ through $V_k$. |
| Fine<br>(Values $V_1$ through $V_k$) | Change all values to binary coded decimal<br>$V_1 \rightarrow 00 \ldots 00$<br>$V_2 \rightarrow 00 \ldots 01$<br>$V_3 \rightarrow 00 \ldots 10$<br>etc.<br>If this results in too long a word length, e.g., $k = 10^9$, a reduction in a word length of approximately $1/p$ may be had by the encoding.<br>$(V_1, \ldots, V_p) \rightarrow 00 \ldots 00$<br>$(V_{p+1}, \ldots, V_{2p}) \rightarrow 00 \ldots 01$<br>$(V_{2p+1}, \ldots V_{3p}) \rightarrow 00 \ldots 10$<br>etc.<br>A variable may only be encoded into levels $(V_1 \ldots V_k)$. |

Table 1 (Continued)

| Type of Variable | |
|---|---|
| Continuous<br>Any value in range | Quantize into convenient steps and encode as for fine quantization. Round off values to closest quantization step. |

Once each number in the data set is encoded the input and output vectors for each pertinent instant of time may be formed. For example, suppose at time t, the input variables A, B, and C to a certain process are to be associated with an output W occurring at the same time. Further suppose that the class of each variable is as follows:

| Variable | Class | Range | Value at $t_1$ |
|---|---|---|---|
| A | Coarse | 0,1,2,3 | 2 |
| B | Fine | $\{k \mid k = 1, \dots 20\}$ | 12 |
| C | Continuous | $\{0 \text{ to } 4\}$ | 2.76 |
| W | Coarse | 0,1,2,3,4,5 | 4 |

Using the above encoding procedure as a guide, it is seen that A is encoded into 4 levels, B into 20 levels, C into 20 levels, and W into 7 levels. The binary representation for each variable becomes:

$$\bar{A}(t_1) = 011$$
$$\bar{B}(t_1) = 01101$$
$$\bar{C}(t_1) = 01110$$
$$\bar{W}(t_1) = 101$$

The value of $\bar{C}(t_1)$ is derived by dividing the interval $[0, 4]$ into 20 discrete levels namely, 0.1, 0.2, 0.5, etc. The value of $\bar{C}(t_1)$ is closest to 2.7 which is the fourteenth level. The vectors $\bar{X}(t_1)$

and $\overline{O}(t_1)$ are formed using $\overline{A}$, $\overline{B}$, $\overline{C}$, and $\overline{W}$.

$$\overline{X}(t_1) = \begin{bmatrix} 0 \\ 1 \\ 1 \\ 0 \\ 1 \\ 1 \\ 0 \\ 1 \\ 0 \\ 1 \\ 1 \\ 1 \\ 0 \end{bmatrix} \begin{matrix} \left.\vphantom{\begin{matrix}0\\1\\1\end{matrix}}\right\} A \\ \left.\vphantom{\begin{matrix}0\\1\\1\\0\\1\end{matrix}}\right\} B \\ \left.\vphantom{\begin{matrix}0\\1\\1\\1\\0\end{matrix}}\right\} C \end{matrix} \qquad \overline{O}(t_1) = \left.\begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix}\right\} W$$

This coding procedure results in a reasonably short code vector. It has the disadvantage that the distance (the number of digits in which they differ) between code words does not increase as the difference in their numerical values increases. This property of the code makes processing by use of learning machines more tedious.

A more straightforward binary assignment for which the distance between individual variables increases as their (decimal) difference increases may be given by:

$$\overline{x}(t_1) = \left\{a_1, a_2, a_3, \ldots a_i \ldots a_L\right\}$$

where $L$ is the number of values (levels) which $x(t_1)$ may assume and for a given level, say $k$,

$$\text{either} \quad a_i = \begin{cases} -1 & 0 \leq i \leq (L - k) \\ 1 & 1(L - k) < i \leq L \end{cases}$$

$$\text{or} \quad a_i = \begin{cases} 1 & 0 \leq i \leq k \\ -1 & k < i \leq L \end{cases}$$

This procedure results in the construction of fairly lengthy code words but will be preferred when the number of levels a variable can assume is small. An example of the application of this technique will be presented later in this paper.

The selection of a learning machine suitable for processing these code vectors proceeds in a straightforward manner. Basically, all learning machines are outgrowths from original work done on the brain modeling problem by McCulloch and Pitts (10). That is to say, all learning machines are networks of decision elements of varying degrees of complexity interconnected to form a logic network. The decision elements are considered to be analogs of neurons and vary in complexity from a simply majority element to very complicated circuits such as the Neurotron and MIND. If only relatively simple decision elements are used, it will be practical to simulate. In similar experiments simple weighting networks such as Rosenblatt's (13) "A" unit, the weighting units used by Farley (4) and those used by Chow (1) have been successfully employed as elements of learning machines. As a weighting unit the decision element known as a threshold logic unit (TLU) (15) represents a good compromise between flexibility and simplicity.

# CHAPTER IV

## CODE PROCESSING

The TLU may be thought of as the circuit which is shown in Figure 4.

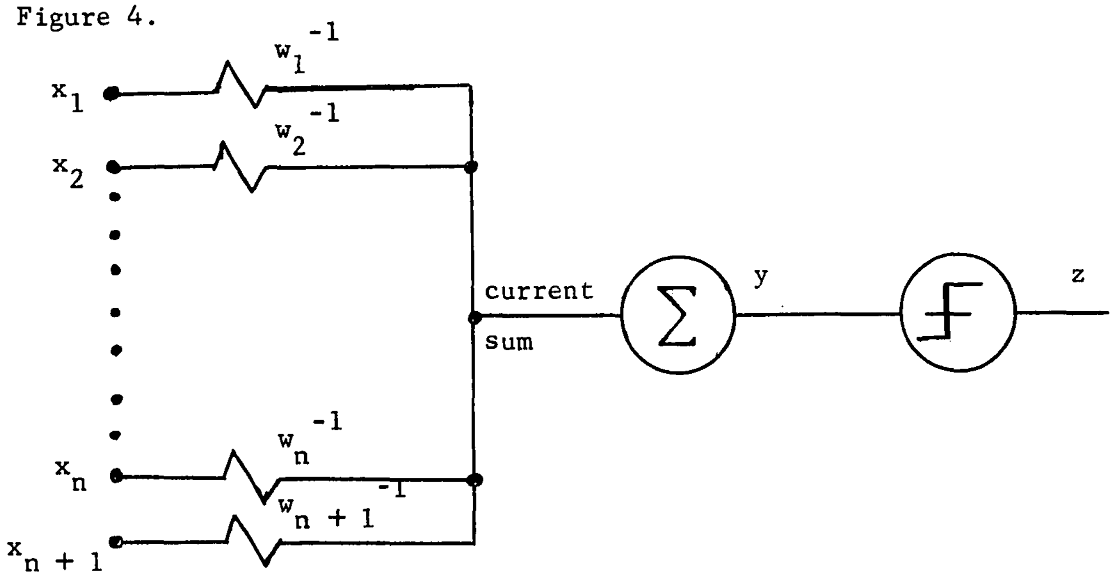



Figure 4. Threshold Logic Unit

In Figure 4 the input vector is represented by the set of numbers $\{x_1 \dots x_n\}$ and the weights $\{w_1 \dots w_{n+1}\}$ are a set of real numbers such that for the $k^{th}$ input vector in a sequence of $M$ vectors

$$\bar{x}_k = \begin{bmatrix} x_{1k} \\ x_{2k} \\ \vdots \\ x_{jk} \\ \vdots \\ x_{nk} \\ x_{n+1} \end{bmatrix} \qquad \begin{array}{l} \text{where } x_{jk} = x_j \quad j = 1, 2, \dots, n \\ \text{as above} \\ \text{and } x_{n+1} = +1 \end{array} \tag{1}$$

and the weight vector is

$$\bar{w} \overset{\Delta}{=} \begin{bmatrix} w_1 \\ \vdots \\ w_j \\ \vdots \\ w_{n+1} \end{bmatrix} \tag{2}$$

The analog output, $y_k$, of the summer is given by

$$y_k \overset{\Delta}{=} \bar{x}_k^{\,t}\, \bar{w} = \langle \bar{x}_k, \bar{w} \rangle = \sum_{j=1}^{n+1} x_{jk}\, w_j \tag{3}$$

and the binary output, $z_k$, defined as

$$z_k = \begin{cases} +1 & y_k \geq 0 \\ -1 & y_k < 0 \end{cases} \tag{4}$$

The network may be trained, that is, it can "learn" to recognize a class of vectors by the simple process of presenting a sequence of these vectors $\bar{x}_k(t_i)$, $k = (0, 1, 2, \ldots) \bmod M$ which may be thought of as occurring at discrete time intervals $i = 0, 1, 2, 3 \ldots$ . This sequence of vectors may be repeated as many times as is necessary. As each $\bar{x}_k(t_i)$ is presented to the network the actual binary output, $z_k(t_i)$ is compared with the desired binary output, $z_k^*(t_i)$. If these outputs coincide, no parameter values are changed in the TLU. If, however, there is disagreement between the desired and actual binary outputs, then the weights are changed according to some fixed scheme which is called a training rule. That is to say,

$$\bar{w}(t_{i+1}) = \begin{cases} \bar{w}(t_i) + a_k(t_i)\, \bar{x}_k(t_i) & z_k(t_i) z_k^*(t_i) < 0 \\ \bar{w}(t_i) & z_k(t_i) z_k^*(t_i) > 0 \end{cases} \tag{5}$$

where $a_k(t_i)$ is a scalar computed at time $(t_i)$ which indicates the magnitude of the correction to be done. The number $a_k(t_i)$ is dependent on the choice of a training algorithm. Several reliable training algorithms are available and all display essentially the same properties; that is, all can be shown to converge to a solution in a finite number of training steps provided the set of vectors, $\{\bar{x}_k,\}$ is linearly separable (15). A set of binary vectors is linearly separable if there exists a number T, a Boolean function f, called a threshold function, and a set of real numbers $\{w_i\}$ such that

$$
\begin{aligned}
f(\bar{x}_k) &= 1 \quad \text{if} \quad \sum_{i=1}^{n} x_i w_i \geq T \\
&= 0 \quad \text{if} \quad \sum_{i=1}^{n} x_i w_i < T
\end{aligned}
\tag{6}
$$

Some training algorithms which may be employed by a TLU follow.

Relaxation Rule

$$
a_{jk} = - \lambda \frac{\langle \bar{x}_k, \bar{w} \rangle}{\langle \bar{x}_k, \bar{x}_k \rangle} = - \lambda \frac{\langle \bar{x}_k, \bar{w} \rangle}{\|\bar{x}_k\|^2} \tag{7}
$$

$$0 < \lambda \leq 2$$

$$\lambda \text{ arbitrary}$$

This algorithm tends to decrease large weights and may lead to a solution $\langle \bar{x}_k, \bar{w} \rangle = 0$ for some or all $\bar{x}_k$. As long as $z = +1$ for $\langle \bar{x}_k, \bar{w} \rangle \geq 0$ this solution will not be in error. Nevertheless, on some computing machines implementation of this algorithm would be more cumbersome than either of the following ones.

Fixed Increment Rule.

$$a_k(t_i) = - z_k(t_i) \tag{7}$$

This algorithm is very simple to implement. The solution may tend to converge slowly if the initial error is large.

Error Correction Rule.

$$a_k(t_i) = - n_k(t_i)\ z_k(t_i) \tag{8}$$

$$n_k(t_i) = \left\lceil \frac{|\langle \bar{x}_k(t_i),\ \bar{w}(t_i)\rangle| + \gamma}{\langle \bar{x}_k(t_i),\ \bar{x}_k(t_i)\rangle} \right\rceil \tag{9}$$

The first two rules may be combined in concept and the error correction rule is the result. $n_k(t_i)$ is chosen as the smallest integer which will satisfy (9). Thus, if $\bar{x}_k$ is used as an input at time $t_i$ and also at time $t_{(i+1)}$ and letting

$$n_k(t_i) = \frac{|\langle \bar{x}_k(t_i),\ \bar{w}(t_i)\rangle| + \gamma}{\langle \bar{x}_k(t_i),\ \bar{x}_k(t_i)\rangle} + \varepsilon \tag{10}$$

$$y_k(t_{i+1}) = \langle \bar{x}_k(t_{i+1}),\ \bar{w}(t_{i+1})\rangle \tag{11}$$

$$= \sum_{j=1}^{n} x_{jk}(t_{i+1}) \left[ w(t_i) - n_k(t_i) z_k(t_i) x_{jk}(t_i) \right] \tag{12}$$

$$= \langle \bar{x}_k(t_i),\ \bar{w}(t_i)\rangle - n_k(t_i) z_k(t_i)\ \langle \bar{x}_k(t_i), \bar{x}_k(t_i)\rangle \tag{13}$$

$$= y_k(t_i) - y_k(t_i) - \left[ \gamma + \varepsilon \langle \bar{x}_k(t_i),\ \bar{x}_k(t_i)\rangle \right] z_k(t_i) \tag{14}$$

and it is seen that $n_k(t_i)$ is smallest integer which, when substituted into equations (3) and (8) and employed in the training rule, will yield new weights $\bar{w}(t_{i+1})$ of the magnitude and sign necessary to reverse the sign of the analog output of the TLU and produce a magnitude of that output at least equal to $\gamma$. The constant, $\gamma$, is sometimes called the training factor.

Using one of the above training algorithms it is possible to find a set of weights such that each $\bar{x}_k$ yields the corresponding $z_k$ when $\bar{x}_k$ is presented at the TLU terminals, provided the set $\{\bar{x}_k\}$ of vectors defined is separable.

Suppose one wishes to classify a set of binary vectors $\{\bar{x}_k\}$ into L distinct categories. If these binary vectors are separable then a parallel connection of several TLU as shown in Figure 5 will be sufficient to classify the set of vectors $\{\bar{x}_k\}$ into the set of categories $\{\bar{z}_i\}$.

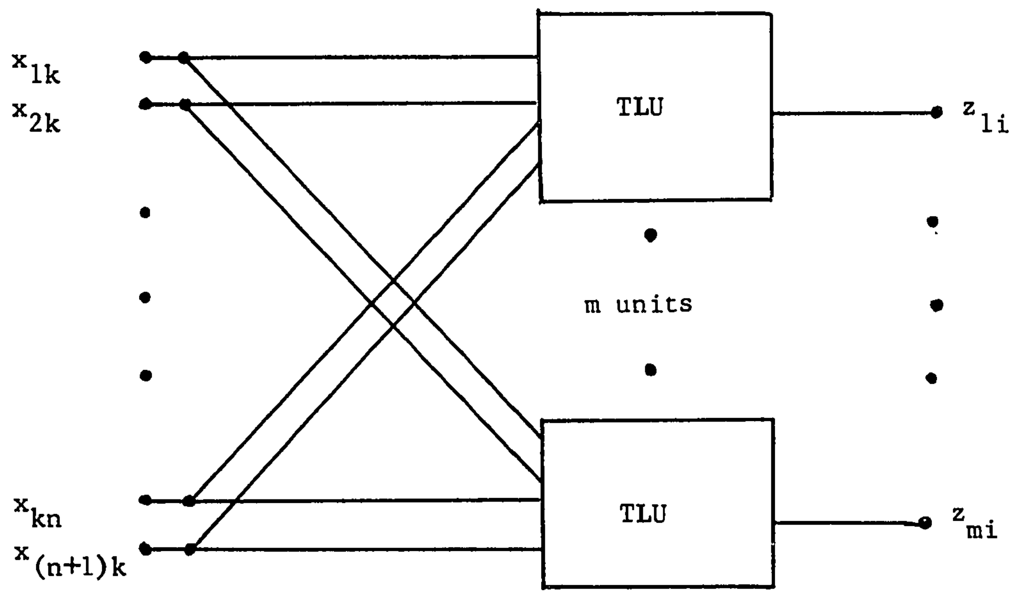



Figure 5

The output vectors $\{\bar{z}_i\}$, i = 1,2, ... L need contain only enough digits to represent L distinct binary vectors, e.g., for three categories only two digits would be required and any three of the four possible output vectors could be utilized to represent the desired categories. Using this machine structure any process or system, about which essentially noise free information is available, is representable to a degree of accuracy limited only by quantization error (3, 16). This error will approach, as a lower bound, the quantization error inherent in the input vector as the number of TLU employed in the representation becomes arbitrarily large (15). The training scheme employed in the computation of weights for a logic machine of this type would remain the same regardless of the number of TLU involved and would be the same as the one for a single TLU.

Since there is no guarantee that the vectors encoded from data concerning a social process will be separable, a learning machine which will classify non-separable vectors must be built using TLU as basic building blocks (16). One such machine is the majority logic machine shown in Figure 6.

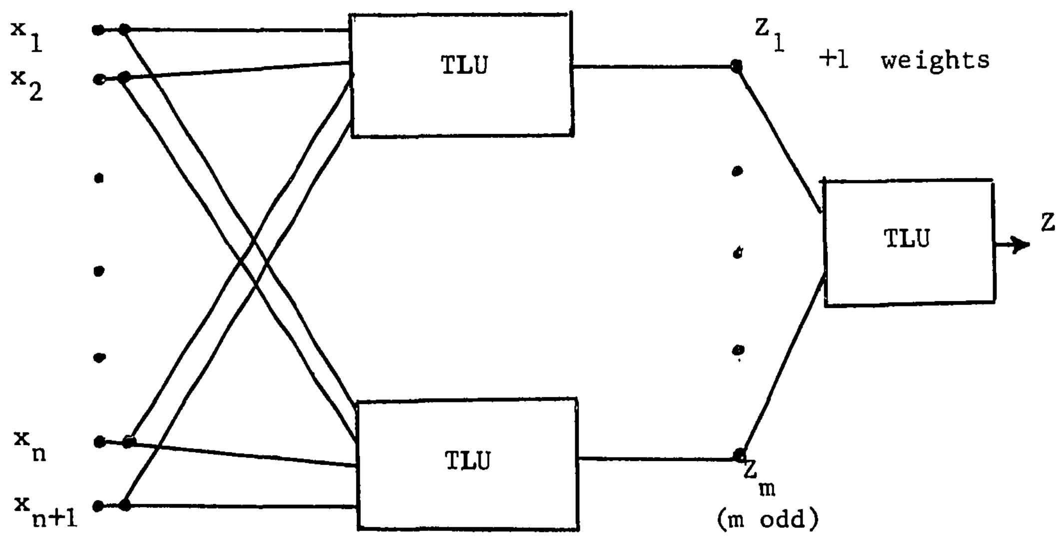



Figure 6. Majority Logic Machine

A slightly more general form of majority logic machine may be used to classify binary vectors as belonging to one of $L$ categories. In this machine a parallel connection of TLUs

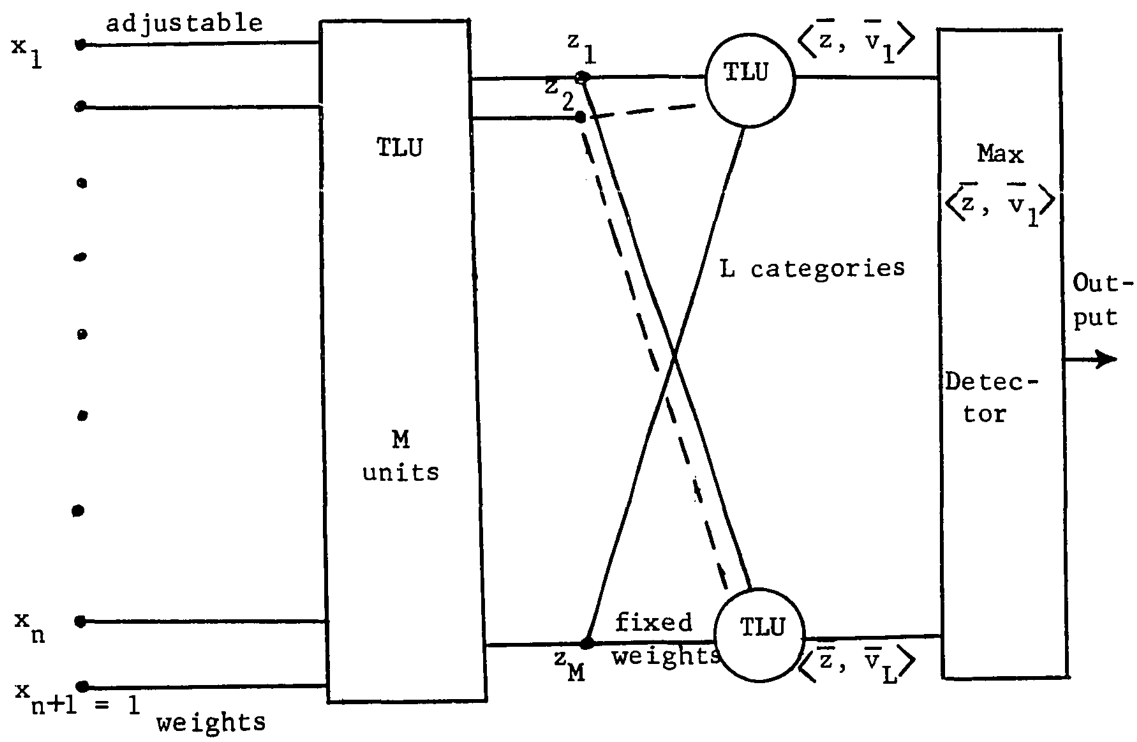



Figure 7. Matched Filter Logic Machine

is employed as a basic processor where each TLU operates on the entire code word as in Figure 5. As is shown in Figure 7, the output of this arrangement will be an $M$ dimensional binary vector

$$\bar{z} = \begin{bmatrix} z_1 \\ \vdots \\ z_M \end{bmatrix} \tag{15}$$

An output code which will classify $\bar{z}$ into one or more of the available $L$ classes is selected from a set of $L$ vectors $\{v_1, \ldots, v_i, \ldots v_L\}$ to represent the $L$ classes or levels (3).

For each vector $\bar{v}_i$, $\langle \bar{z}, \bar{v}_i \rangle$ is computed and the classification is determined by the relation

$$L = \left\{ i \,\middle|\, \langle \bar{z}, \bar{v}_i \rangle \text{ is a maximum; } i = 1, 2, \ldots, L. \right\} \qquad (16)$$

If there is no unique maximum $L$ is chosen as the smallest value of $i$ which makes $\langle \bar{z}, \bar{v}_i \rangle$ a maximum. Since the first layer output is matched against a class vector this machine is often called a "template matching" or Matched Filter Logic Machine.

If the number of TLU is chosen so that $M > L$ then some tolerance is allowed in that $\bar{z}$ need not exactly match $\bar{v}_i$ to produce a classification into level $i$ but must only be closer to $\bar{v}_i$ than to any of the other output code vectors. Since both $\bar{z}$ and $\bar{v}_i$ are binary vectors, max $\langle \bar{z}, \bar{v}_i \rangle$ will occur for the code vector $\bar{v}_i$ which differs from $\bar{z}$ by the least number of digits. Depending on the weight one wishes to place on miss distance these output codes may be chosen in a number of ways. For example, if no distinction is made between misclassifying a vector by one level and misclassifying it by, say, ten levels an output code may be used where the code vector for a certain level is equidistant from the code vectors for all other levels. For this purpose a code such as a Hamming error correcting code could be used (12). If, on the other hand, one wishes to make some distinction between misclassifying a vector by one level as opposed to, say, six levels an "x + y" code could be devised where adjacent classes were represented by code words differing by $x$ digits, those once removed from each other by $x + y$ digits, those twice removed by $x + 2y$ digits, etc., for all code words. Table 3 shows a representative sample of these codes suitable for representing three levels. For clarity -1 is replaced by zero.

Table 3. Typical Output Codes $\bar{v}_i$

| Level | Equidistant code | 3 + 3 code | 5 + 1 code |
|---|---|---|---|
| 1 | 0 1 1 1 1 | 0 0 0 0 0 0 | 0 0 0 0 0 0 0 0 |
| 2 | 1 1 1 0 0 | 1 1 1 0 0 0 | 1 1 1 1 1 0 0 0 |
| 3 | 1 0 0 1 1 | 1 1 1 1 1 1 | 1 1 1 0 0 1 1 1 |

Using the selected code the weights of the second layer of TLUs are fixed so that the weights for the $i^{th}$ TLU are $\bar{v}_i$ (the zeros in the table being replaced by -1). Since the input to the second layer is $\bar{z}$ the output vector will be the set of integers $\{\langle \bar{z}, \bar{v}_i \rangle\}$ and the classification of the input vector may be made by selecting the value of i corresponding to the largest number in this set. Ties are resolved as previously described.

With the MFL machine shown in Figure 7, the following training procedure is used. The adjustable weights are initially set equal to some constant, say +0 or +1, and the first input vector is presented. If the classification is correct, no changes are made and the next input vector is presented. Suppose, however, that the correct category is category i and that the classification of the input vector is incorrect. That is to say, the machine classifies this particular input vector as category j. Then some of the components of $\bar{z}$ differ from the corresponding components of $\bar{v}_i$. The number that differ being the distance, d, between $\bar{z}$ and $\bar{v}_i$. The responses of some of the corresponding d threshold logic units must then be reversed to bring $\bar{z}$ closer to $\bar{v}_i$. One training rule which may be used is that the number of units to be corrected, $d'$, is taken to be the smallest integer greater than or equal to $d\rho$

$$d' = \lceil d\rho \rceil \qquad 0 < \rho < 1 \tag{17}$$

where the number $\rho$ is an experimentally determined fraction. If, for example, $\rho = \frac{1}{2}$, then roughly $\frac{1}{2}$ of the incorrectly responding first layer threshold logic units would be corrected. The particular $d'$ units that are to be corrected are those incorrectly responding units whose analog sums are closest to 0. If all analog sums are equal no preference will be shown and the units will be corrected as they are found to be incorrect until the proper numbers have been corrected. Another training rule which may be used is

$$d' = \begin{cases} d - D & \\ 0 & \text{if } (d - D) < 0 \end{cases} \tag{18}$$

where $D$ is an experimentally determined integer, usually a small fraction of $M$. Once again the particular $d'$ units that are corrected are those incorrectly responding first layer threshold logic units whose analog sums are closest to zero. In practice, it is wise to allow the constant $D$ or $\rho$ to decrease in value as training progresses so that overcorrection does not take place and some correct responses are erased. It is generally a good idea to keep a running tally of the average number of misses so that these constants may be changed if necessary. In correcting each individual TLU, any of the previously mentioned training algorithms may be used. In the experiments described later in this paper, the error correction rule was used to adjust weights of each threshold logic unit. This rule is given by equation

(9) which, for binary vectors, equation reduces to

$$n_k(t_i) = \left\lceil \frac{|y_k(t_i)| + \gamma}{n + 1} \right\rceil \tag{19}$$

where $y_k$ is the analog sum of the TLU in question, and $n$ is the number of digits in the binary vector. This form of the error correction rule is the form that was used to correct each TLU in the experiments described later in this paper.

CHAPTER V

# A DECODING EXAMPLE

When code words are classified by the use of a MFL machine some insight as to the relative importance or relevance of each digit in the input code may be gained by examination of the weights corresponding to each digit by the MFL operator.

This remark may perhaps be best illustrated with a small example. Suppose that a set of seven vectors are to be used to train a machine to decode a larger set of 32 vectors into one of three classes.

The seven vectors to be used are:

(replacing -1 by zero for clarity)

$$\bar{x}_1 = \begin{bmatrix} 1 \\ -1 \\ 1 \\ -1 \\ -1 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} \quad \bar{x}_2 = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} \quad \bar{x}_3 = \begin{bmatrix} -1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} \quad \bar{x}_4 = \begin{bmatrix} +1 \\ -1 \\ -1 \\ -1 \\ -1 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

$$\bar{x}_5 = \begin{bmatrix} -1 \\ -1 \\ -1 \\ -1 \\ -1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad \bar{x}_6 = \begin{bmatrix} -1 \\ 1 \\ -1 \\ 1 \\ -1 \end{bmatrix} \quad \bar{x}_7 = \begin{bmatrix} 1 \\ -1 \\ -1 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \\ 1 \end{bmatrix} .$$

The output class code vectors for this set are the 1 + 1 code specified by

$$(1) = \bar{v}_1 = \begin{bmatrix} 1 \\ \\ 1 \end{bmatrix} \qquad (2) = \bar{v}_2 = \begin{bmatrix} -1 \\ \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ \\ 1 \end{bmatrix} \qquad (3) = \bar{v}_3 = \begin{bmatrix} -1 \\ \\ -1 \end{bmatrix} = \begin{bmatrix} 0 \\ \\ 0 \end{bmatrix}$$

The seven training vectors are known to correspond to output level 1, 2, or 3 as listed in Table 4.

Table 4. Code Classes for Example

| Code Word | Class |
|---|---|
| $\bar{x}_1$ | 1 |
| $\bar{x}_2$ | 1 |
| $\bar{x}_3$ | 2 |
| $\bar{x}_4$ | 2 |
| $\bar{x}_5$ | 3 |
| $\bar{x}_6$ | 3 |
| $\bar{x}_7$ | 2 |

There is no special significance attached here (in the larger sense) to either the input or output code book. The choice of both input and output vectors was made for purely practical reasons, i.e., they are short.

The structure of the MFL machine to be used for classification is then as shown below,

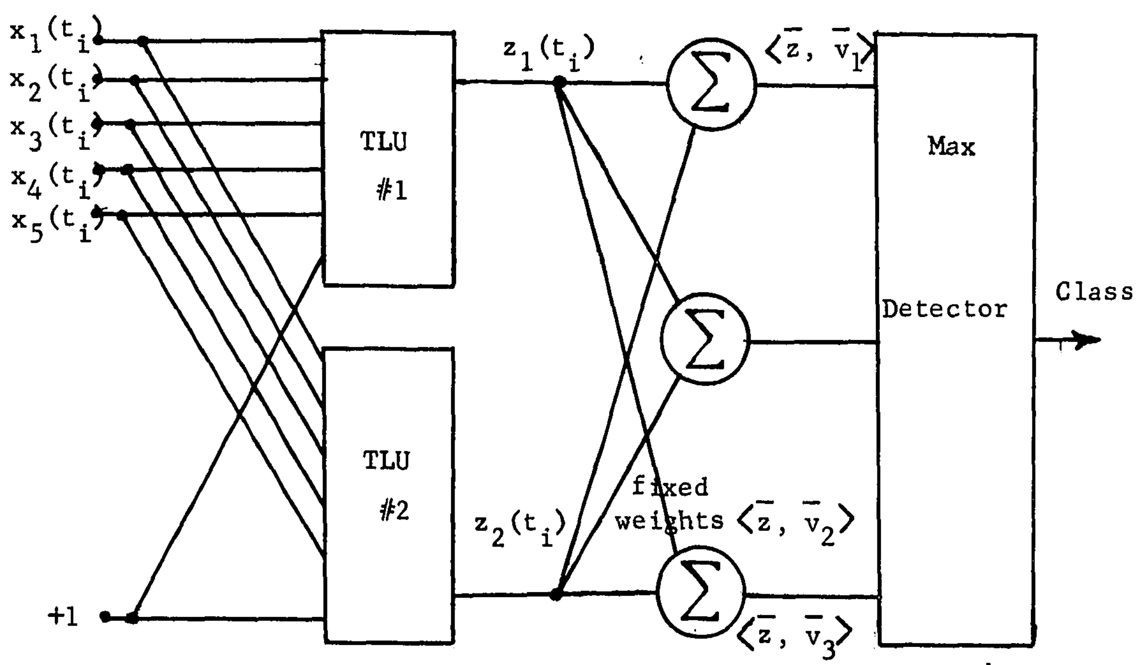



Figure 8. MFL Machine for Example 1

where the fixed weights of the second layer TLUs are $v_1$, $v_2$ and $v_3$ respectively. For this particular example let $\rho = 0.5$, $\gamma = 5$ and all initial weights be +1. The input vector occurring at time $t_i$ will be denoted by

$$\bar{x}(t_i) = \begin{bmatrix} x_1(t_i) \\ x_2(t_i) \\ x_3(t_i) \\ x_4(t_i) \\ x_5(t_i) \\ +1 \end{bmatrix}$$

and the weights at time $t_i$ will be

$$\bar{w}^1(t_i) = \begin{bmatrix} w_1^1(t_i) \\ w_2^1(t_i) \\ w_3^1(t_i) \\ w_4^1(t_i) \\ w_5^1(t_i) \\ w_6^1(t_i) \end{bmatrix} \quad \text{for TLU \#1}$$

and similarly using a superscript 2 for TLU #2. The superscript and subscript notations will also be consistently carried out to denote analog sums and components of the z vector relating to each TLU. $\bar{x}(t_1) = \bar{x}_1$; $\bar{x}(t_2) = \bar{x}_2$ and if more than seven words are needed in training $\bar{x}(t_8) = \bar{x}_1$; $\bar{x}(t_9) = \bar{x}_2$, etc. so that the code book will be cyclic with period 7, and will initially start with the code word previously denoted as $x_1$ with the addition of +1. Throughout -1 will be replaced by zero for the sake of clarity.

The results of each decoding and training operation will now be shown.

This first case should be

$$\bar{x}(t_i) = \bar{x}_1 = \begin{bmatrix}1\\0\\1\\0\\0\\1\end{bmatrix} \qquad \bar{w}^1(t_i) = \text{the weight vector for TLU \#1 for time } (t_i) = \begin{bmatrix}1\\1\\1\\1\\1\\1\end{bmatrix}$$

$$w^2(t_1) = \text{the weight vector for TLU \#2 for time } (t_i) = \begin{bmatrix}1\\1\\1\\1\\1\\1\end{bmatrix}$$

$$y^1(t_1) = \text{the analog output of TLU \#1 at time } t_1 = \langle \bar{x}(t_1), \bar{w}^1(t_i) \rangle = 0$$

$$y^2(t_1) = \text{the analog output of TLU \#2 at time } t_1 = \langle \bar{x}(t_1), \bar{w}^2(t_1) \rangle = 0$$

$$z^1(t_1) = \text{the binary output of TLU \#1 at time } t_1 = +1$$

$$z^2(t_1) = \text{the binary output of TLU \#2 at time } t_1 = +1$$

$$\bar{z} = \begin{bmatrix}z^1\\z^2\end{bmatrix} = \begin{bmatrix}1\\1\end{bmatrix} \quad \text{and} \quad \langle \bar{z}, \bar{v}_i \rangle \begin{array}{ll} = +2 & \text{for } i = 1 \\ = 0 & \text{for } i = 2 \\ = -2 & \text{for } i = 3 \end{array}$$

$\langle \bar{z}, \bar{v}_i \rangle$ is maximum for $i = 1$

so class is 1. Since, from Table 4, 1 is the desired class no correction is necessary.

Similarly,

$$\bar{x}(t_2) = \bar{x}_2 = \begin{bmatrix}1\\1\\1\\1\\1\\1\end{bmatrix} \quad \bar{w}^1(t_2) = \begin{bmatrix}1\\1\\1\\1\\1\\1\end{bmatrix} \quad \bar{w}^2(t_2) = \begin{bmatrix}1\\1\\1\\1\\1\\1\end{bmatrix} \quad \begin{array}{ll} y^1(t_2) = +6 & z^1(t_2) = +1 \\ y^2(t_2) = +6 & z^2(t_2) = +1 \end{array}$$

$\langle \bar{z}(t_2), \bar{v}_i \rangle$ is max for $i = 1$. Therefore class is correct and no correction is necessary.

$$x(t_3) = \bar{x}_3 = \begin{bmatrix} 0 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} \quad \bar{w}^1(t_3) = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} \quad \bar{w}^2(t_3) = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} \quad \begin{array}{ll} y^1(t_3) = +4 & z^1(t_3) = +1 \\ y^2(t_3) = +4 & z^2(t_3) = +1 \end{array}$$

$\langle \bar{z}(t_3), \bar{v}_i \rangle$ is max for $i = 1$ which is wrong.

Some correction must now take place. There is one incorrect TLU and it must be corrected since $\rho \times 1 = 0.5$ and the smallest integral number of TLU greater than this must be corrected. Therefore, for TLU #1

$$n^1(t_3) = \text{correction factor for TLU \#1 at time } t_3 \text{ (eq.19)} = \left\lceil \frac{|\, y^1(t_3) \,| + \gamma}{M + 1} \right\rceil$$

$$= \left\lceil \frac{4 + 5}{6} \right\rceil = 2$$

Remembering eq. (5), $\bar{w}^1(t_4) = \bar{w}^1(t_3) - z^1(t_3) n^1(t_3) \bar{x}(t_3)$

$$w_1^1(t_4) = w_1^1(t_3) - z^1(t_3)\, n^1(t_3)\, x_1(t_3)$$

$$\vdots$$

$$w_M^1(t_4) = w_M(t_3) - z^1(t_3)\, n^1(t_3)\, x_M(t_3)$$

$$w_{M+1}^1(t_4) = w_{M+1}(t_3) - z^1(t_3)\, n^1(t_3)\, (+1)$$

the new weights are

$$w_1^1(t_4) = 1 - (1) \times 2 \times (-1) = 3$$

$$w_2^1(t_4) = w_3^1(t_4) = w_4^1(t_4) = w_5^1(t_4) = 1 - (1) \times (2) \times (1) = -1$$

$$w_6^1(t_4) = 1 - 1 \times 2 \times (+1) = -1$$

therefore the resulting new weight vector for TLU #1 is

$$\bar{w}^1(t_4) = \begin{bmatrix} w_1^1(t_4) \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ w_{M+1}^1(t_4) \end{bmatrix} = \begin{bmatrix} 3 \\ -1 \\ -1 \\ -1 \\ -1 \\ -1 \end{bmatrix}$$

Hereafter the results will be more briefly summarized and the result of each correction will appear as the new weight vector for the succeeding code word.

The next code word is now presented to the MFL machine and

$$x(t_4) = \bar{x}_4 = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \quad \bar{w}^1(t_4) = \begin{bmatrix} 3 \\ -1 \\ -1 \\ -1 \\ -1 \\ -1 \end{bmatrix} \quad \bar{w}^2(t_4) = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} \begin{matrix} y^1(t_4) = +6 & z^1(t_4) = +1 \\ \\ y^2(t_4) = -2 & z^2(t_4) = -1 \end{matrix}$$

$\langle \bar{z}(t_4), \bar{v}_i \rangle$ is max for both $i = 1$ and $i = 3$ both of which are incorrect classes. Both TLU are wrong but since $\rho \times 2 = 1$ only TLU #2 will be corrected (since $y^2(t_4)$ is closest to zero) with

$$2 = n(_4) = \left\lceil \frac{|-2|+5}{6} \right\rceil$$

$$x(t_5) = \bar{x}_5 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \quad \bar{w}^1(t_5) = \begin{bmatrix} 3 \\ -1 \\ -1 \\ -1 \\ -1 \\ -1 \end{bmatrix} \quad \bar{w}^2(t_5) = \begin{bmatrix} 3 \\ -1 \\ -1 \\ -1 \\ -1 \\ 3 \end{bmatrix} \begin{matrix} y^1(t_5) = 0 & z^1(t_5) = +1 \\ \\ y^2(t_5) = +4 & z^2(t_5) = +1 \end{matrix}$$

$\langle \bar{z}(t_5), \bar{v}_i \rangle$ is max for $i = 1$. Therefore TLU #1 is corrected with

$$1 = n^1(t_5) = \left\lceil \frac{0+5}{6} \right\rceil$$

$$\bar{x}(t_6) = \bar{x}_6 = \begin{bmatrix} 0\\1\\0\\1\\0\\1 \end{bmatrix} \quad \bar{w}^1(t_6) = \begin{bmatrix} 4\\0\\0\\0\\0\\2 \end{bmatrix} \quad \bar{w}^2(t_6) = \begin{bmatrix} 3\\-1\\-1\\-1\\-1\\3 \end{bmatrix} \begin{matrix} y^1(t_6) = -2 & z^1(t_6) = -1 \\ \\ y^2(t_6) = 0 & z^2(t_6) = +1 \end{matrix}$$

$\langle \bar{z}(t_6), \bar{v}_i \rangle$ is max for either $i = 1$ or $i = 3$ and so is classified as class 1 which is an error. Since TLU #2 is in error it will be corrected with

$$1 = n^2(t_6) = \left\lceil \frac{5}{6} \right\rceil$$

$$\bar{x}(t_7) = \bar{x}_7 = \begin{bmatrix} 1\\0\\0\\1\\1\\1 \end{bmatrix} \quad \bar{w}^1(t_7) = \begin{bmatrix} 4\\0\\0\\0\\0\\2 \end{bmatrix} \quad \bar{w}^2(t_7) = \begin{bmatrix} 4\\0\\2\\0\\2\\3 \end{bmatrix} \begin{matrix} y^1(t_7) = +6 & z^1(t_7) = +1 \\ \\ y^2(t_7) = +7 & z^2(t_7) = +1 \end{matrix}$$

$\langle \bar{z}(t_7), \bar{v}_i \rangle$ is max for $\bar{v}_i = \bar{v}_1$. TLU #1 will be corrected with

$$2 = n^1(t_7) = \left\lceil \frac{6 + 5}{6} \right\rceil$$

$$\bar{x}(t_8) = \bar{x}_1 = \begin{bmatrix} 1\\0\\1\\0\\0\\1 \end{bmatrix} \quad \bar{w}^1(t_8) = \begin{bmatrix} 2\\2\\2\\-2\\-2\\0 \end{bmatrix} \quad \bar{w}^2(t_8) = \begin{bmatrix} 4\\0\\2\\0\\2\\3 \end{bmatrix} \begin{matrix} y^1(t_8) = +6 & z^1(t_8) = +1 \\ \\ y^2(t_8) = +7 & z^2(t_8) = +1 \end{matrix}$$

$\langle \bar{z}(t_8), \bar{v}_i \rangle$ is max for $i = 1$. Therefore no correction is necessary.

$$x(t_9) = \bar{x}_2 = \begin{bmatrix} 1\\1\\1\\1\\1\\1 \end{bmatrix} \quad \bar{w}^1(t_9) = \begin{bmatrix} 2\\2\\2\\-2\\-2\\0 \end{bmatrix} \quad \bar{w}^2(t_9) = \begin{bmatrix} 4\\0\\2\\0\\2\\3 \end{bmatrix} \begin{matrix} y^1(t_9) = +2 & z^1(t_9) = +1 \\ \\ y^2(t_9) = +9 & z^2(t_9) = +1 \end{matrix}$$

$\langle \bar{z}(t_9), \bar{v}_i \rangle$ is max for $i = 1$ so no correction is necessary.

$$\bar{x}(t_{10}) = \bar{x}_3 = \begin{bmatrix} 0\\1\\1\\1\\1\\1 \end{bmatrix} \bar{w}^1(t_{10}) = \begin{bmatrix} 2\\2\\2\\-2\\-2\\0 \end{bmatrix} \bar{w}^2(t_{10}) = \begin{bmatrix} 4\\0\\2\\0\\2\\3 \end{bmatrix} \begin{matrix} y^1(t_{10}) = -2 & z^1(t_{10}) = -1 \\ \\ y^2(t_{10}) = +3 & z^2(t_{10}) = +1 \end{matrix}$$

$\langle \bar{z}(t_{10}), \bar{v}_i \rangle$ is max for $i = 2$ so no correction is necessary.

$$x(t_{11}) = \bar{x}_4 = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \bar{w}^1(t_{11}) = \begin{bmatrix} 2 \\ 2 \\ 2 \\ -2 \\ -2 \\ 0 \end{bmatrix} \bar{w}^2(t_{11}) = \begin{bmatrix} 4 \\ 0 \\ 2 \\ 0 \\ 2 \\ 3 \end{bmatrix} \begin{matrix} y^1(t_{11}) = +2 & z^1(t_{11}) = +1 \\ y^2(t_{11}) = +3 & z^2(t_{11}) = +1 \end{matrix}$$

$\langle \bar{z}(t_{11}), \bar{v}_i \rangle$ is max for $i = 1$ so TLU #1 is corrected with

$$2 = n^1(t_{11}) = \left\lceil \frac{2 + 5}{6} \right\rceil$$

$$\bar{x}(t_{12}) = \bar{x}_5 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \bar{w}^1(t_{12}) = \begin{bmatrix} 0 \\ 4 \\ 4 \\ 0 \\ 0 \\ -2 \end{bmatrix} \bar{w}^2(t_{12}) = \begin{bmatrix} 4 \\ 0 \\ 2 \\ 0 \\ 2 \\ 3 \end{bmatrix} \begin{matrix} y^1(t_{12}) = -6 & z^1(t_{12}) = -1 \\ y^2(t_{12}) = -5 & z^2(t_{12}) = -1 \end{matrix}$$

$\langle \bar{z}(t_{12}), \bar{v}_i \rangle$ is max for $i = 3$ so no correction is necessary.

$$\bar{x}(t_{13}) = \bar{x}_6 = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 1 \\ 0 \\ 1 \end{bmatrix} \bar{w}^1(t_{13}) = \begin{bmatrix} 0 \\ 4 \\ 4 \\ 0 \\ 0 \\ -2 \end{bmatrix} \bar{w}^2(t_{13}) = \begin{bmatrix} 4 \\ 0 \\ 2 \\ 0 \\ 2 \\ 3 \end{bmatrix} \begin{matrix} y^1(t_{13}) = -2 & z^1(t_{13}) = -1 \\ y^2(t_{13}) = -5 & z^2(t_{13}) = -1 \end{matrix}$$

$\langle \bar{z}(t_{13}), \bar{v}_i \rangle$ is max for $i = 3$ so no correction is necessary.

$$\bar{x}(t_{14}) = \bar{x}_7 = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \\ 1 \\ 1 \end{bmatrix} \bar{w}^1(t_{14}) = \begin{bmatrix} 0 \\ 4 \\ 4 \\ 0 \\ 0 \\ -2 \end{bmatrix} \bar{w}^2(t_{14}) = \begin{bmatrix} 4 \\ 0 \\ 2 \\ 0 \\ 2 \\ 3 \end{bmatrix} \begin{matrix} y^1(t_{14}) = -10 & z^1(t_{14}) = -1 \\ y^2(t_{14}) = +7 & z^2(t_{14}) = +1 \end{matrix}$$

$\langle \bar{z}(t_{14}), \bar{v}_i \rangle$ is max for $i = 2$ so no correction is necessary.

$$\bar{x}(t_{15}) = \bar{x}_1 = \begin{bmatrix} 1 \\ 0 \\ 1 \\ 0 \\ 0 \\ 1 \end{bmatrix} \bar{w}^1(t_{15}) = \begin{bmatrix} 0 \\ 4 \\ 4 \\ 0 \\ 0 \\ -2 \end{bmatrix} \bar{w}^2(t_{15}) = \begin{bmatrix} 4 \\ 0 \\ 2 \\ 0 \\ 2 \\ 3 \end{bmatrix} \begin{matrix} y^1(t_{15}) = +2 & z^1(t_{15}) = +1 \\ y^2(t_{15}) = +7 & z^2(t_{15}) = +1 \end{matrix}$$

$\langle \bar{z}(t_{15}), \bar{v}_i \rangle$ is max for $i = 1$ so no correction is necessary.

$$\bar{x}(t_{16}) = \bar{x}_2 = \begin{bmatrix}1\\1\\1\\1\\1\\1\end{bmatrix} \bar{w}^1(t_{16}) = \begin{bmatrix}0\\4\\4\\0\\0\\-2\end{bmatrix} \bar{w}^2(t_{16}) = \begin{bmatrix}4\\0\\2\\0\\2\\3\end{bmatrix} \begin{matrix} y^1(t_{16}) = +6 & z^1(t_{16}) = +1 \\ \\ y^2(t_{16}) = +11 & z^2(t_{16}) = +1 \end{matrix}$$

$\langle \bar{z}(t_{16}), \bar{v}_i \rangle$ is max for $i = 1$ so no correction is necessary.

$$\bar{x}(t_{17}) = \bar{x}_3 = \begin{bmatrix}1\\1\\1\\1\\1\\1\end{bmatrix} \bar{w}^1(t_{17}) = \begin{bmatrix}0\\4\\4\\0\\0\\-2\end{bmatrix} \bar{w}^2(t_{17}) = \begin{bmatrix}4\\0\\2\\0\\2\\3\end{bmatrix} \begin{matrix} y^1(t_{17}) = +6 & z^1(t_{17}) = +1 \\ \\ y^2(t_{17}) = +3 & z^2(t_{17}) = +1 \end{matrix}$$

$\langle \bar{z}(t_{17}), \bar{v}_i \rangle$ is max for $i = 1$ so TLU #1 must be corrected with

$$2 = n^1(t_{17}) = \left\lceil \frac{6+5}{6} \right\rceil$$

$$\bar{x}(t_{18}) = \bar{x}_4 = \begin{bmatrix}1\\0\\0\\0\\0\\1\end{bmatrix} \bar{w}^1(t_{18}) = \begin{bmatrix}2\\2\\2\\-2\\-2\\-4\end{bmatrix} \bar{w}^2(t_{18}) = \begin{bmatrix}4\\0\\2\\0\\2\\3\end{bmatrix} \begin{matrix} y^1(t_{18}) = -2 & z^1(t_{18}) = -1 \\ \\ y^2(t_{18}) = +3 & z^2(t_{18}) = +2 \end{matrix}$$

$\langle \bar{z}(t_{18}), \bar{v}_i \rangle$ is max for $i = 2$ so classification is correct.

$$\bar{x}(t_{19}) = \bar{x}_5 = \begin{bmatrix}0\\0\\0\\0\\0\\1\end{bmatrix} \bar{w}^1(t_{19}) = \begin{bmatrix}2\\2\\2\\-2\\-2\\-4\end{bmatrix} \bar{w}^2(t_{19}) = \begin{bmatrix}4\\0\\2\\0\\2\\3\end{bmatrix} \begin{matrix} y^1(t_{19}) = -6 & z^1(t_{19}) = -1 \\ \\ y^2(t_{19}) = -5 & z^2(t_{19}) = -1 \end{matrix}$$

$\langle \bar{z}(t_{19}), \bar{v}_i \rangle$ is max for $i = 3$ so classification is correct.

$$\bar{x}(t_{20}) = \bar{x}_6 = \begin{bmatrix}0\\1\\0\\1\\0\\1\end{bmatrix} \bar{w}^1(t_{20}) = \begin{bmatrix}2\\2\\2\\-2\\-2\\-4\end{bmatrix} \bar{w}^2(t_{20}) = \begin{bmatrix}4\\0\\2\\0\\2\\3\end{bmatrix} \begin{matrix} y^1(t_{20}) = -6 & z^1(t_{20}) = -1 \\ \\ y^2(t_{20}) = -5 & z^2(t_{20}) = -1 \end{matrix}$$

$\langle \bar{z}(t_{20}), \bar{v}_i \rangle$ is max for $i = 3$ so classification is correct.

$$x(t_{21}) = \bar{x}_7 = \begin{bmatrix}1\\0\\0\\1\\1\\1\end{bmatrix} \bar{w}^1(t_{21}) = \begin{bmatrix}2\\2\\2\\-2\\-2\\-4\end{bmatrix} \bar{w}^2(t_{21}) = \begin{bmatrix}4\\0\\2\\0\\2\\3\end{bmatrix} \begin{matrix} y^1(t_{21}) = -10 & z^1(t_{21}) = -1 \\ \\ y^2(t_{21}) = +7 & z^2(t_{21}) = +1 \end{matrix}$$

$\langle \bar{z}(t_{21}), \bar{v}_i \rangle$ is max for $i = 2$ so classification is correct.

$$x(t_{22}) = \bar{x}_1 = \begin{bmatrix}1\\0\\1\\0\\0\\1\end{bmatrix} \bar{w}^1(t_{22}) = \begin{bmatrix}2\\2\\2\\-2\\-2\\-4\end{bmatrix} \bar{w}^2(t_{22}) = \begin{bmatrix}4\\0\\2\\0\\2\\3\end{bmatrix} \begin{matrix} y^1(t_{22}) = +2 & z^1(t_{22}) = +1 \\ \\ y^2(t_{22}) = +7 & z^2(t_{22}) = +1 \end{matrix}$$

$\langle \bar{z}(t_{22}), \bar{v}_i \rangle$ is max for $i = 1$ so classification is correct.

$$x(t_{23}) = \bar{x}_2 = \begin{bmatrix}1\\1\\1\\1\\1\\1\end{bmatrix} \bar{w}^1(t_{23}) = \begin{bmatrix}2\\2\\2\\-2\\-2\\-4\end{bmatrix} \bar{w}^2(t_{23}) = \begin{bmatrix}4\\0\\2\\0\\2\\3\end{bmatrix} \begin{matrix} y^1(t_{23}) = -2 & z^1(t_{23}) = -1 \\ \\ y^2(t_{23}) = +11 & z^2(t_{23}) = +1 \end{matrix}$$

$\langle \bar{z}(t_{23}), \bar{v}_i \rangle$ is max for $i = 2$ so TLU #1 is corrected with

$$2 = n^1(t_{23}) = \left\lceil \frac{|-2| + 5}{6} \right\rceil$$

$$\bar{x}(t_{24}) = \bar{x}_3 = \begin{bmatrix}0\\1\\1\\1\\1\\1\end{bmatrix} \bar{w}^1(t_{24}) = \begin{bmatrix}4\\4\\4\\0\\0\\-2\end{bmatrix} \bar{w}^2(t_{24}) = \begin{bmatrix}4\\0\\2\\0\\2\\3\end{bmatrix} \begin{matrix} y^1(t_{24}) = +2 & z^1(t_{24}) = +1 \\ \\ y^2(t_{24}) = +3 & z^2(t_{24}) = +1 \end{matrix}$$

$\langle \bar{z}(t_{24}), \bar{v}_i \rangle$ is max for $i = 2$ so TLU #1 is corrected with

$$2 = n^1(t_{24}) = \left\lceil \frac{2 + 5}{6} \right\rceil$$

$$\bar{x}(t_{25}) = \bar{x}_4 = \begin{bmatrix}1\\0\\0\\0\\0\\1\end{bmatrix} \bar{w}^1(t_{25}) = \begin{bmatrix}6\\2\\2\\-2\\-2\\-4\end{bmatrix} \bar{w}^2(t_{25}) = \begin{bmatrix}4\\0\\2\\0\\2\\3\end{bmatrix} \begin{matrix} y^1(t_{25}) = +2 & z^1(t_{25}) = +1 \\ \\ y^2(t_{25}) = +3 & z^2(t_{25}) = +1 \end{matrix}$$

$\langle \bar{z}(t_{25}), \bar{v}_i \rangle$ is max for $i = 1$ so TLU #1 is corrected with

$$2 = n^1(t_{25}) = \left\lceil \frac{2 + 5}{6} \right\rceil$$

$$x(t_{26}) = \bar{x}_5 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \bar{w}^1(t_{26}) = \begin{bmatrix} 4 \\ 4 \\ 4 \\ 0 \\ 0 \\ -6 \end{bmatrix} \bar{w}^2(t_{26}) = \begin{bmatrix} 4 \\ 0 \\ 2 \\ 0 \\ 2 \\ 3 \end{bmatrix} \begin{matrix} y^1(t_{26}) = -18 & z^1(t_{26}) = -1 \\ y^2(t_{26}) = -5 & z^2(t_{26}) = -1 \end{matrix}$$

$\langle \bar{z}(t_{26}), \bar{v}_i \rangle$ is max for $i = 3$ so classification is correct.

$$x(t_{27}) = \bar{x}_6 = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 1 \\ 0 \\ 1 \end{bmatrix} \bar{w}^1(t_{27}) = \begin{bmatrix} 4 \\ 4 \\ 4 \\ 0 \\ 0 \\ -6 \end{bmatrix} \bar{w}^2(t_{27}) = \begin{bmatrix} 4 \\ 0 \\ 2 \\ 0 \\ 2 \\ 3 \end{bmatrix} \begin{matrix} y^1(t_{27}) = -10 & z^1(t_{27}) = -1 \\ y^2(t_{27}) = -5 & z^2(t_{27}) = -1 \end{matrix}$$

$\langle \bar{z}(t_{27}), \bar{v}_i \rangle$ is max for $i = 3$ so classification is correct.

$$x(t_{28}) = \bar{x}_7 = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \\ 1 \\ 1 \end{bmatrix} \bar{w}^1(t_{28}) = \begin{bmatrix} 4 \\ 4 \\ 4 \\ 0 \\ 0 \\ -6 \end{bmatrix} \bar{w}^2(t_{28}) = \begin{bmatrix} 4 \\ 0 \\ 2 \\ 0 \\ 2 \\ 3 \end{bmatrix} \begin{matrix} y^1(t_{28}) = -2 & z^1(t_{28}) = -1 \\ y^2(t_{28}) = +7 & z^2(t_{28}) = +1 \end{matrix}$$

$\langle \bar{z}(t_{28}), \bar{v}_i \rangle$ is max for $i = 3$ so classification is correct.

$$\bar{x}(t_{29}) = \bar{x}_1 = \begin{bmatrix} 1 \\ 0 \\ 1 \\ 0 \\ 0 \\ 1 \end{bmatrix} \bar{w}^1(t_{29}) = \begin{bmatrix} 4 \\ 4 \\ 4 \\ 0 \\ 0 \\ -6 \end{bmatrix} \bar{w}^2(t_{29}) = \begin{bmatrix} 4 \\ 0 \\ 2 \\ 0 \\ 2 \\ 3 \end{bmatrix} \begin{matrix} y^1(t_{29}) = -2 & z^1(t_{29}) = -1 \\ y^2(t_{29}) = +7 & z^2(t_{29}) = +1 \end{matrix}$$

$\langle \bar{z}(t_{29}), \bar{v}_i \rangle$ is max for $i = 3$ so TLU #1 is corrected with

$$2 = n^1(t_{29}) = \left\lceil \frac{|-2| + 5}{6} \right\rceil$$

$$\bar{x}(t_{30}) = \bar{x}_2 = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} \bar{w}^1(t_{30}) = \begin{bmatrix} 6 \\ 2 \\ 6 \\ -2 \\ -2 \\ -4 \end{bmatrix} \bar{w}^2(t_{30}) = \begin{bmatrix} 4 \\ 0 \\ 2 \\ 0 \\ 2 \\ 3 \end{bmatrix} \begin{matrix} y^1(t_{30}) = +6 & z^1(t_{30}) = +1 \\ y^2(t_{30}) = +11 & z^2(t_{30}) = +1 \end{matrix}$$

$\langle \bar{z}(t_{30}), \bar{v}_i \rangle$ is max for $i = 3$ so no correction is necessary.

$$\bar{x}(t_{31}) = \bar{x}_3 = \begin{bmatrix} 0 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} \bar{w}^1(t_{31}) = \begin{bmatrix} 6 \\ 2 \\ 6 \\ -2 \\ -2 \\ -4 \end{bmatrix} \bar{w}^2(t_{31}) = \begin{bmatrix} 4 \\ 0 \\ 2 \\ 0 \\ 2 \\ 3 \end{bmatrix} \begin{matrix} y^1(t_{31}) = -6 & z^1(t_{31}) = -1 \\ y^2(t_{31}) = +3 & z^2(t_{31}) = +1 \end{matrix}$$

$\langle \bar{z}(t_{31}), \bar{v}_i \rangle$ is max for $i = 2$ so no correction is necessary

$$\bar{x}(t_{32}) = \bar{x}_4 = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \bar{w}^1(t_{32}) = \begin{bmatrix} 6 \\ 2 \\ 6 \\ -2 \\ -2 \\ -4 \end{bmatrix} \bar{w}^2(t_{32}) = \begin{bmatrix} 4 \\ 0 \\ 2 \\ 0 \\ 2 \\ 3 \end{bmatrix} \begin{matrix} y^1(t_{32}) = -2 & z^1(t_{32}) = -1 \\ y^2(t_{32}) = +3 & z^2(t_{32}) = +1 \end{matrix}$$

$\langle \bar{z}(t_{32}), \bar{v}_i \rangle$ is max for $i = 2$ so no correction is necessary.

$$\bar{x}(t_{33}) = \bar{x}_5 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \bar{w}^1(t_{33}) = \begin{bmatrix} 6 \\ 2 \\ 6 \\ -2 \\ -2 \\ -4 \end{bmatrix} \bar{w}^2(t_{33}) = \begin{bmatrix} 4 \\ 0 \\ 2 \\ 0 \\ 2 \\ 3 \end{bmatrix} \begin{matrix} y^1(t_{33}) = -14 & z^1(t_{33}) = -1 \\ y^2(t_{33}) = -5 & z^2(t_{33}) = -1 \end{matrix}$$

$\langle \bar{z}(t_{33}), \bar{v}_i \rangle$ is max for $i = 3$ so no correction is necessary.

$$x(t_{34}) = \bar{x}_6 = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 1 \\ 0 \\ 1 \end{bmatrix} \bar{w}^1(t_{34}) = \begin{bmatrix} 6 \\ 2 \\ 6 \\ -2 \\ -2 \\ -4 \end{bmatrix} \bar{w}^2(t_{34}) = \begin{bmatrix} 4 \\ 0 \\ 2 \\ 0 \\ 2 \\ 3 \end{bmatrix} \begin{matrix} y^1(t_{34}) = -14 & z^1(t_{34}) = -1 \\ y^2(t_{34}) = -5 & z^2(t_{34}) = -1 \end{matrix}$$

$\langle \bar{z}(t_{34}), \bar{v}_i \rangle$ is max for $i = 3$ so no correction is necessary.

$$\bar{x}(t_{35}) = \bar{x}_7 = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \\ 1 \\ 1 \end{bmatrix} \bar{w}^1(t_{35}) = \begin{bmatrix} 6 \\ 2 \\ 6 \\ -2 \\ -2 \\ -4 \end{bmatrix} \bar{w}^2(t_{35}) = \begin{bmatrix} 4 \\ 0 \\ 2 \\ 0 \\ 2 \\ 3 \end{bmatrix} \begin{matrix} y^1(t_{35}) = -10 & z^1(t_{35}) = -1 \\ y^2(t_{35}) = +7 & z^2(t_{35}) = +1 \end{matrix}$$

$\langle \bar{z}(t_{35}), \bar{v}_i \rangle$ is max for $i = 3$ so no correction is necessary.

$$x(t_{36}) = \bar{x}_1 = \begin{bmatrix}1\\0\\1\\0\\0\\1\end{bmatrix} \quad \bar{w}^1(t_{36}) = \begin{bmatrix}6\\2\\6\\-2\\-2\\-4\end{bmatrix} \quad \bar{w}^2(t_{36}) = \begin{bmatrix}4\\0\\2\\0\\2\\3\end{bmatrix} \quad \begin{matrix} y^1(t_{36}) = +10 & z^1(t_{36}) = +1 \\ y^2(t_{36}) = +7 & z^2(t_{36}) = +1 \end{matrix}$$

$\langle \bar{z}(t_{36}), \bar{v}_i \rangle$ is max for $i = 1$ so no correction is necessary.

The entire code book has now been decoded without error and the final weight vectors are

$$\bar{w}^1 = \begin{bmatrix}6\\2\\6\\-2\\-2\\-4\end{bmatrix} \quad \bar{w}^2 = \begin{bmatrix}4\\0\\2\\0\\2\\3\end{bmatrix}$$

Some interesting remarks can be made about the relative importance of each code digit. If one ignores the last weight digit (which pertains to the +1 tag and not to the original code word proper) and investigates the relative importance placed on each code digit by $\bar{w}^1$ and $\bar{w}^2$ the following table is one possible result.

| | 1st digit | 2nd digit | 3rd digit | 4th digit | 5th digit |
|---|---|---|---|---|---|
| $\bar{w}^1$ | 6 | 2 | 6 | -2 | -2 |
| $\bar{w}^2$ | 4 | 0 | 2 | 0 | 2 |
| ave.wt. | 5 | 1 | 4 | -1 | 0 |

The reader will notice that if the numbers associated with the +1 tags are appended to the vector described by the "ave.wt." row in the table that the resulting weight vectors

$$\bar{w}^1 = \begin{bmatrix}5\\1\\4\\-1\\0\\-4\end{bmatrix} \quad \text{and} \quad \bar{w}^2 = \begin{bmatrix}5\\1\\4\\-1\\0\\+3\end{bmatrix}$$

also are a set of weight vectors which will properly decode all of the input code words. Although this property holds for this particular set of code words it may or may not hold for another more complex set of code words.

The reader has already, no doubt, noticed that the general property of the code book, i.e., that the classification of the code is according to the number of ones in the first and third digits of the code word, is indicated by the average relative weights. This observation may serve as an illustration of the use of MFL machines to determine the relative importance of each of the variables which go to make up the input code words.

# CHAPTER VI

## A SOCIAL SYSTEM MODEL: ROBBERS CAVE

As an example of the application of learning machines to the social system simulation problem, a modeling study of the classic "Robbers Cave Experiment" (14) has been conducted. The Robbers Cave Experiment was chosen for several reasons. First, it contained (as nearly as possible) all available information about a given social system over its complete history. In this instance the system is defined to consist of two groups of small boys along with their interactions. Second, great pains were taken to insure precision in measurement of relevant variables and third, the system size was small enough to be amenable to simulation with a reasonably small computing machine. The sections to follow will discuss in detail the technique used in the simulation study of this experiment.

The Robbers Cave Experiment was concerned with the interaction of two groups of young boys placed in various competitive and cooperative situations which "naturally" arose in a summer camp environment.

As a beginning step it was necessary to extract all of the pertinent information about the two groups to be simulated and arrange it in a form suitable for coding into a form recognizable by a digital computer. With this end in mind, the report of the experiment was analyzed in detail for content. The content analysis was examined for indication of input-output relationships pertinent to the experimental behavior of either group. The final analysis of the experiment is contained in Appendix A.

Once this list was reduced to its final form, the dependent and independent variables were selected and scale values were assigned on the basis of information contained in the original report. This selection of variables and assignment of values was done by the author in concert with Dr. J. D. Palmer and Dr. R. A. Terry. This was done to minimize any bias an individual might unconsciously introduce into variable selection and evaluation and thus to arrive at a reasonable estimate of what the important parameters in the experiment were. Table 5 gives a list of the dependent and independent variables.

Table 5. Variable Definitions

| Independent | | Dependent | |
|---|---|---|---|
| (1) | goal type | | |
| (2) | goal value to group | (1) | frustration level of group |
| (3) | number of times in succession some goal has occurred | (2) | attitude of group toward other group |
| (4) | was goal attained? | (3) | morale of group |
| (5) | was other group present? | (4) | solidity of group structure |

The variables "attitude", "morale", and "structure" were specifically mentioned as variables in the original experiment. The variable "frustration" was implied to exist in the experimental definition of the other variables and so was included in the list of dependent variables. At the outset frustration was assumed to depend on all of the independent variables and also on its own previous value. Attitude, morale, and structure were assumed to depend on all of the independent variables and a past average value of attitude, morale, and structure respectively. Since processing with a pattern recognition machine will yield relative weights, no a priori assumptions were made regarding the relative influence exerted on each dependent variable by each independent variable.

Based on the information available, the values of variables "frustration" and "attitude" were assigned one of five different levels and values of the variables "morale" and "structure" were assigned one of three different levels. Table 6, shown on following page, is a summary of all variable class and numeric value assignments.

The reader will notice that all of the variables listed as independent variables are quantities which were under the control of the experimenter at the time of the experiment. In some instances the value range which variables should cover was obvious from the context of the experiment, e.g., "presence". In most instances the range of values for each variable was derived from the "possible classes" information in Table 6.

Once the range of values for each variable was fixed each situation in the Appendix A was examined as a possible candidate for the formation of a situation vector. For each situation where a value could be determined for all of the previously designated independent variables, an attempt was made to deduce a value of all of the dependent variables. If a value was found for a least one of the dependent variables then the situation was given a situation number and included

Table 6. Class Assignments

| Variable | Possible Classes | Numeric Value |
|---|---|---|
| goal type (independent) | competitive | 3 |
| | neutral | 2 |
| | cooperative | 1 |
| goal value (independent) | none | 0 |
| | low | 1 |
| | medium | 2 |
| | high | 3 |
| no. of times in succession for same type goal (independent) | up to eleven times in succession | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11 |
| attainment (independent) | attained | 3 |
| | tie | 2 |
| | not attained | 1 |
| presence (independent) | present | 1 |
| | not present | 0 |
| frustration | very frustrated | 5 |
| | frustrated | 4 |
| | neutral | 3 |
| | unfrustrated | 2 |
| | very unfrustrated | 1 |
| attitude | very bad | 1 |
| | bad | 2 |
| | indifferent | 3 |
| | good | 4 |
| | very good | 5 |
| morale | low | 1 |
| | medium | 2 |
| | high | 3 |
| structure | low | 1 |
| | medium | 2 |
| | high | 3 |

in the situation list to be processed by the learning machine. Examination of Appendix A reveals that not every situation contained in the content analysis was assigned a situation number. The final situation list with the attendant values of all pertinent variables is summarized in Table 7 for the group called the "Rattlers", and in Table 8 for the "Eagles" group. Each situation is listed by number with the first nine entries in each column being inputs and the last four entries being the desired value of output for that situation. It is coincidental that there are forty-one situations in each list.

To convert these situations into the binary coded form, which must be used for ease of computer simulation, the coding procedure which was previously described in this paper was used. Application of this procedure results in the binary encoding shown in Table 9. In this table -1 has been replaced by zero for clarity. Using this encoding for individual pieces of each vector, each situation was encoded by forming combinations of the independent variables as previously indicated. That is to say,

Frustration = $f_1$(no. times, type, value, attainment, frustration last time, presence)

Attitude = $f_2$(no. times, type, value, attainment, frustration last time, average attitude last three times, presence)

Morale = $f_3$(no. times, type, value, attainment, frustration last time, average morale last three times, presence)

Structure = $f_4$(no. times, type, value, attainment, frustration last time, average structure last three times, presence) .

So that all levels would always be integers, the values of average attitude, morale, and structure were taken to be the majority of the three preceeding values, i.e., if the last three values of, say, morale

| | RATTLERS COMPOSITE TABLE 7 | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | PHASE I | | | | | | | | | | | PHASE II | | | | | | | | | |
| SITUATION NUMBER | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 |
| No.of times this goal type has occurred | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 1 | 2 | 3 | 4 | 1 | 2 | 1 | 1 | 1 | 2 |
| Goal type | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 1 | 1 | 1 | 1 | 2 | 2 | 1 | 2 | 1 | 1 |
| Goal value | 2 | 2 | 1 | 1 | 2 | 2 | 2 | 1 | 3 | 2 | 2 | 3 | 3 | 3 | 3 | 0 | 1 | 2 | 3 | 3 | 3 |
| Goal attainment | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 2 | 2 | 2 | 3 | 2 | 3 | 3 | 2 | 1 | 1 |
| Frustration last time | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 4 | 3 | 4 | 3 | 3 | 4 | 5 |
| Ave. attitude last 3 times | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 2 | 2 | 2 | 3 | 2 | 3 | 3 | 3 | 2 |
| Ave. structure last 3 times | 1 | 1 | 1 | 1 | 1 | 1 | 1 | - | - | 2 | 2 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 |
| Ave. morale last 3 times | 1 | - | - | 2 | - | - | 2 | - | - | - | 2 | 2 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 |
| Presence | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 |
| Frustration this time | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 4 | 3 | 4 | 3 | 3 | 4 | 5 | 5 |
| Attitude this time | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 2 | 2 | 3 | 2 | 3 | 2 | 3 | 3 | 2 | 2 | 1 |
| Structure this time | 1 | 1 | 1 | 1 | 2 | 2 | - | - | 2 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 |
| Morale this time | 2 | - | - | 2 | - | - | 2 | - | - | - | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 2 | 2 |

RATTLERS COMPOSITE TABLE 7 (Continued)

| | PHASE II | | | | | | | | | PHASE III | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| SITUATION NUMBER | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 | 33 | 34 | 35 | 36 | 37 | 38 | 39 | 40 | 41 |
| No. of times this goal type has occurred | 1 | 2 | 1 | 2 | 3 | 4 | 5 | 1 | 2 | 1 | 2 | 1 | 1 | 2 | 3 | 1 | 1 | 2 | 1 | 2 |
| Goal type | 2 | 2 | 1 | 1 | 1 | 1 | 1 | 2 | 2 | 3 | 3 | 2 | 3 | 3 | 3 | 2 | 3 | 3 | 2 | 2 |
| Goal value | 3 | 0 | 2 | 3 | 3 | 3 | 0 | 0 | 0 | 2 | 2 | 1 | 3 | 3 | 2 | 2 | 2 | 2 | 2 | 2 |
| Goal attainment | 3 | 1 | 3 | 1 | 1 | 3 | 1 | 3 | 1 | 3 | 3 | 3 | 3 | 3 | 3 | 2 | 3 | 2 | 2 | 2 |
| Frustration last time | 5 | 4 | 5 | 3 | 4 | 5 | 4 | 5 | 4 | 5 | 4 | 4 | 4 | 3 | 2 | 2 | 1 | 1 | 1 | 1 |
| Ave. attitude last 3 times | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 1 | 1 | 1 | 2 | 2 | 2 | 2 | 3 | 3 | 4 | 4 | 4 | 5 |
| Ave. structure last 3 times | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 |
| Ave. morale last 3 times | 2 | 2 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 |
| Presence | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| Frustration this time | 4 | 5 | 3 | 4 | 5 | 4 | 5 | 4 | 5 | 4 | 4 | 4 | 3 | 2 | 2 | 1 | 1 | 1 | 1 | 1 |
| Attitude this time | 2 | 2 | 2 | 2 | 2 | 1 | 1 | 2 | 2 | 2 | 2 | 2 | 3 | 3 | 4 | 4 | 4 | 5 | 5 | 5 |
| Structure this time | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 |
| Morale this time | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 |

| | EAGLES COMPOSITE TABLE 8 | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | PHASE I | | | | | | | | | | | PHASE II | | | | | | | | | |
| SITUATION NUMBER | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 |
| No. of times this goal type has occurred | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 1 | 2 | 3 | 4 | 1 | 2 | 1 | 1 | 1 | 2 |
| Goal type | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 1 | 1 | 1 | 1 | 2 | 2 | 1 | 2 | 1 | 1 |
| Goal value | 1 | 1 | 1 | 3 | 2 | 2 | 2 | 2 | 3 | 2 | 2 | 3 | 3 | 3 | 3 | 0 | 1 | 2 | 3 | 3 | 3 |
| Goal attainment | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 2 | 2 | 2 | 1 | 2 | 3 | 1 | 3 | 3 | 3 |
| Frustration last time | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 4 | 4 | 4 | 3 | 4 | 5 | 4 |
| Ave. attitude last 3 times | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 |
| Ave. structure last 3 times | 1 | 1 | 1 | 1 | 1 | 1 | 2 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 2 | 2 | 2 | 2 | 2 | 2 | 2 |
| Ave. morale last 3 times | - | - | - | 2 | - | - | 2 | - | - | - | - | 2 | 2 | 2 | 2 | 2 | 1 | 1 | 1 | 2 | 2 |
| Presence | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 |
| Frustration this time | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 4 | 4 | 4 | 3 | 4 | 5 | 4 | 3 |
| Attitude this time | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 2 | 2 | 3 | 2 | 2 | 2 | 3 | 2 | 2 | 3 | 3 |
| Structure this time | 1 | 1 | 2 | 1 | 1 | 2 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 2 | 2 | 2 | 2 | 3 | 3 | 3 | 3 |
| Morale this time | - | - | - | 2 | - | - | 2 | - | - | - | - | 2 | 3 | 2 | 1 | 1 | 2 | 1 | 2 | 3 | 3 |

EAGLES COMPOSITE TABLE 8 (Continued)

| | PHASE II | | | | | | | | | PHASE III | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| SITUATION NUMBER | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 | 33 | 34 | 35 | 36 | 37 | 38 | 39 | 40 | 41 |
| No. of times this goal type has occurred | 1 | 2 | 1 | 2 | 3 | 4 | 5 | 1 | 2 | 1 | 2 | 1 | 1 | 2 | 3 | 1 | 1 | 2 | 1 | 2 |
| Goal type | 2 | 2 | 1 | 1 | 1 | 1 | 1 | 2 | 2 | 3 | 3 | 2 | 3 | 3 | 3 | 2 | 3 | 3 | 2 | 2 |
| Goal value | 0 | 3 | 2 | 3 | 3 | 0 | 0 | 0 | 0 | 3 | 2 | 1 | 3 | 3 | 2 | 2 | 2 | 2 | 2 | 2 |
| Goal attainment | 1 | 3 | 1 | 3 | 1 | 1 | 1 | 1 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 2 | 3 | 2 | 2 |
| Frustration last time | 3 | 4 | 3 | 4 | 3 | 3 | 4 | 5 | 4 | 5 | 4 | 4 | 4 | 3 | 2 | 2 | 1 | 1 | 1 | 1 |
| Ave. attitude last 3 times | 3 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 1 | 2 | 2 | 2 | 2 | 2 | 3 | 4 | 4 | 5 | 5 |
| Ave. structure last 3 times | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 2 | 1 | 1 | 1 | 1 |
| Ave. morale last 3 times | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 |
| Presence | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| Frustration this time | 4 | 3 | 4 | 3 | 3 | 4 | 5 | 4 | 5 | 4 | 4 | 4 | 3 | 2 | 2 | 1 | 1 | 1 | 1 | 1 |
| Attitude this time | 1 | 2 | 2 | 2 | 2 | 2 | 1 | 2 | 1 | 2 | 2 | 2 | 3 | 3 | 4 | 4 | 4 | 5 | 5 | 5 |
| Structure this time | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 2 | 1 | 1 | 1 | 1 | 1 | 1 |
| Morale this time | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 2 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 |

Table 9. Binary Encoding

| Variable | Level | |
|---|---|---|
| no. this type | 1 | 0 0 0 0 0 0 0 0 0 0 1 |
| | 2 | 0 0 0 0 0 0 0 0 0 1 1 |
| | 3 | 0 0 0 0 0 0 0 0 1 1 1 |
| | 4 | 0 0 0 0 0 0 0 1 1 1 1 |
| | 5 | 0 0 0 0 0 0 1 1 1 1 1 |
| | 6 | 0 0 0 0 0 1 1 1 1 1 1 |
| | 7 | 0 0 0 0 1 1 1 1 1 1 1 |
| | 8 | 0 0 0 1 1 1 1 1 1 1 1 |
| | 9 | 0 0 1 1 1 1 1 1 1 1 1 |
| | 10 | 0 1 1 1 1 1 1 1 1 1 1 |
| | 11 | 1 1 1 1 1 1 1 1 1 1 1 |
| type | 1 | 1 0 0 0 |
| | 2 | 1 1 0 0 |
| | 3 | 1 1 1 0 |
| value | 0 | 0 0 0 0 |
| | 1 | 1 0 0 0 |
| | 2 | 1 1 0 0 |
| | 3 | 1 1 1 0 |
| attainment | 1 | 1 0 0 |
| | 2 | 1 1 0 |
| | 3 | 1 1 1 |
| frustration | 1 | 1 0 0 0 0 |
| | 2 | 1 1 0 0 0 |
| | 3 | 1 1 1 0 0 |
| | 4 | 1 1 1 1 0 |
| | 5 | 1 1 1 1 1 |
| attitude | 1 | 1 0 0 0 0 |
| | 2 | 1 1 0 0 0 |
| | 3 | 1 1 1 0 0 |
| | 4 | 1 1 1 1 0 |
| | 5 | 1 1 1 1 1 |
| morale | 1 | 1 0 0 |
| | 2 | 1 1 0 |
| | 3 | 1 1 1 |
| structure | 1 | 1 0 0 |
| | 2 | 1 1 0 |
| | 3 | 1 1 1 |
| presence | 1 | 1 1 |
| | 0 | 0 0 |

were 3, 3, and 2, the average morale was taken to be 3. For example, the input code vectors for situation 23, Table 7 were formed as follows:

$$
\begin{array}{l}\text{Attitude} \\ \text{input 21}\end{array} =
\left[\begin{matrix} -1\\-1\\-1\\-1\\-1\\-1\\-1\\-1\\-1\\+1\\+1\\+1\\-1\\-1\\-1\\+1\\+1\\+1\\-1\\+1\\-1\\-1\\+1\\+1\\+1\\+1\\+1\\+1\\+1\\-1\\-1\\-1\\+1\\+1\\+1 \end{matrix}\right]
\begin{array}{l}
\left.\begin{matrix} \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \end{matrix}\right\}\text{no. times} \\
\left.\begin{matrix} \\ \\ \\ \\ \end{matrix}\right\}\text{type} \\
\left.\begin{matrix} \\ \\ \\ \\ \end{matrix}\right\}\text{value} \\
\left.\begin{matrix} \\ \\ \\ \end{matrix}\right\}\text{attainment} \\
\left.\begin{matrix} \\ \\ \\ \\ \\ \end{matrix}\right\}\begin{array}{l}\text{frustration}\\ \text{last time}\end{array} \\
\left.\begin{matrix} \\ \\ \\ \\ \\ \end{matrix}\right\}\begin{array}{l}\text{ave.attitude}\\ \text{last 3 times}\end{array} \\
\left.\begin{matrix} \\ \\ \end{matrix}\right\}\text{presence} \\
\ \text{tag}
\end{array}
\qquad
\begin{array}{l}\text{Morale} \\ \text{input 21}\end{array} =
\left[\begin{matrix} -1\\-1\\-1\\-1\\-1\\-1\\-1\\-1\\-1\\+1\\+1\\+1\\-1\\-1\\-1\\+1\\+1\\+1\\-1\\+1\\-1\\-1\\+1\\+1\\+1\\+1\\+1\\+1\\+1\\+1\\+1\\+1\\+1 \end{matrix}\right]
\begin{array}{l}
\left.\begin{matrix} \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \end{matrix}\right\}\text{no. times} \\
\left.\begin{matrix} \\ \\ \\ \\ \end{matrix}\right\}\text{type} \\
\left.\begin{matrix} \\ \\ \\ \\ \end{matrix}\right\}\text{value} \\
\left.\begin{matrix} \\ \\ \\ \end{matrix}\right\}\text{attainment} \\
\left.\begin{matrix} \\ \\ \\ \\ \\ \end{matrix}\right\}\begin{array}{l}\text{frustration}\\ \text{last time}\end{array} \\
\left.\begin{matrix} \\ \\ \\ \end{matrix}\right\}\begin{array}{l}\text{ave.morale}\\ \text{last 3 times}\end{array} \\
\left.\begin{matrix} \\ \\ \end{matrix}\right\}\text{presence} \\
\ \text{tag}
\end{array}
$$

All situations were similarly encoded to provide input code books suitable for processing by an MFL machine.

In Table 9 some additional features are noticed. The encoding for the variable "type" contains an extra digit. This was inserted as a check to see if mixed coding schemes could be used in forming the code vector. In every training situation this digit was lightly weighted by the learning machine. An additional redundancy was added by coding the "presence" variable with the distance 2 code shown in the table. Comparing training cases using the distance one encoding (1, 0), and (0, 0) revealed the weight assigned to each digit for the distance 2 code is approximately one-half that assigned to the digit of the corresponding distance one code.

Several output code assignments were compared. Since all of the dependent variables are coarsely quantized into either three or five levels only two output codes, a three and five level, were required for any encoding scheme, a 3 + 1, 5 + 1, and 5 + 3 set of output codes were devised. Replacing -1 with 0 for clarity, the three code sets are shown in Table 10.

Table 10. Output Codes

| Type | Vector | Code |
|---|---|---|
| 3 + 1 3 level | $\bar{v}_1^t$ | 1 1 1 1 1 |
| | $\bar{v}_2^t$ | 0 0 0 1 1 |
| | $\bar{v}_3^t$ | 0 0 1 0 0 |
| 3 + 1 5 level | $\bar{v}_1^t$ | 1 1 1 1 1 1 1 1 1 |
| | $\bar{v}_2^t$ | 0 0 0 1 1 1 1 1 1 |
| | $\bar{v}_3^t$ | 0 0 1 0 0 1 1 1 1 |
| | $\bar{v}_4^t$ | 0 0 1 0 1 0 0 1 1 |
| | $\bar{v}_5^t$ | 0 0 1 0 1 0 1 0 0 |
| 5 + 1 3 level | $\bar{v}_1^t$ | 1 1 1 1 1 1 1 1 |
| | $\bar{v}_2^t$ | 0 0 0 0 0 1 1 1 |
| | $\bar{v}_3^t$ | 0 0 0 1 1 0 0 0 |

Table 10. Output Codes
(Continued)

| Type | | Vector | Code |
|---|---|---|---|
| 5 + 1 | 5 level | $\bar{v}_1^t$ | 1 1 1 1 1 1 1 1 1 1 1 1 1 1 |
| | | $\bar{v}_2^t$ | 0 0 0 0 0 1 1 1 1 1 1 1 1 1 |
| | | $\bar{v}_3^t$ | 0 0 0 1 1 0 0 0 1 1 1 1 1 1 |
| | | $\bar{v}_4^t$ | 0 0 0 1 1 0 1 1 0 0 0 1 1 1 |
| | | $\bar{v}_5^t$ | 0 0 0 1 1 0 1 1 0 1 1 0 0 0 |
| 5 + 3 | 3 level | $\bar{v}_1^t$ | 1 1 1 1 1 1 1 1 1 |
| | | $\bar{v}_2^t$ | 0 0 0 0 0 1 1 1 1 |
| | | $\bar{v}_3^t$ | 0 0 0 0 1 0 0 0 0 |
| 5 + 3 | 5 level | $\bar{v}_1^t$ | 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 |
| | | $\bar{v}_2^t$ | 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 |
| | | $\bar{v}_3^t$ | 0 0 0 0 1 0 0 0 0 1 1 1 1 1 1 1 1 |
| | | $\bar{v}_4^t$ | 0 0 0 0 1 0 0 0 1 0 0 0 0 1 1 1 1 |
| | | $\bar{v}_5^t$ | 0 0 0 0 1 0 0 0 1 0 0 0 1 0 0 0 0 |

A convenient algorithm for generating the above and similar codes may be stated as follows:

To form an L level $x + y$ code where $x \geq y$ and $x + y$ is even

$\bar{v}_1^t$ = a string of $x + (L-2)\left(\frac{x+y}{2}\right) = P$ ones.

$\bar{v}_2^t$ = x zeros followed by $P - x$ ones.

$\bar{v}_3^t$ = $x - \left(\frac{x-y}{2}\right)$ zeros, followed by $\left(\frac{x-y}{2}\right)$ ones.
followed by $\left(\frac{x+y}{2}\right)$ zeros followed by $P - \left(\frac{3x+y}{2}\right)$ ones.

$\bar{v}_j^{\,t} = x - \left(\frac{x - y}{2}\right)$ zeros followed by $\left(\frac{x - y}{2}\right)$ ones followed by $(j - 3)$ repetitions of the string (y zeros followed by $\left(\frac{x - y}{2}\right)$ ones) followed by $P - \left(\frac{3x + y}{2}\right) + (j - 3)\left(\frac{x + y}{2}\right)$ ones for $j = 4, 5, \ldots, L$

If this algorithm is used it is noticed that adjacent words differ by

$$\left(\frac{x + y}{2}\right) - y + \left(\frac{x + y}{2}\right) = x \text{ digits}$$

and that the $j^{th}$ word differs from the $k^{th}$ word by

$$\left(\frac{x + y}{2}\right) - (k - j + 1)y + \left(\frac{x + y}{2}\right) = x + (k - j)y$$

digits for $k > j > 3$. Also comparing $\bar{v}_1$ and $\bar{v}_2$, $\bar{v}_1$ and $\bar{v}_3$, and $\bar{v}_2$ and $\bar{v}_3$ it is noticed that these word pairs differ by $x$, $x + y$, and $x$ digits respectively. Thus, this algorithm defines an $x + y$ distance code.

Using the vectors described above a simulation study of the two groups was performed. This study consisted of two distinct phases. Phase I consisted of training a set of MFL machines of appropriate configurations to classify the eight sets of input vectors, four for each group, into the proper output classes denoted by the frustration this time, attitude this time, morale this time, and structure this time, rows in Tables 7 and 8. Phase II utilized the trained MFL machines as predictors in order to investigate possible behaviors of each group.

To perform the training a general program was written to simulate a Matched Filter Logic Machine of arbitrary size. This program, listed in Appendix B, was then used to process the eight sets of vectors which represented the information pertinent to the Rattlers and the Eagles. Initially, one set of inputs was processed three times using different output codes in order to determine the sensitivity of the process to output coding. The set of inputs corresponding to the variable "attitude" for the Eagles group was arbitrarily selected for this purpose and processed using the 3 + 1 code, the 5 + 1 code, and the 5 + 3 code. The results, which are summarized in Figure 8, indicated no decided preference for any output code. Since there was no decided preference the 5 + 3 output code was used for all training runs.

As the program flow chart in Appendix B indicates, each training run consisted of presenting 1000 code vectors to the machine, comparing the machines classification of each vector to the correct classification and correcting some of the incorrectly responding TLUs if the classification was incorrect. For the first 500 training vectors 3/4 of the incorrectly responding TLU s were changed and for the remaining 500, 1/4 of incorrect TLUs were corrected. This variation lessened the possibility of inadvertently changing correct weights to incorrect ones. The change at the end of 500 training cases was suggested by the results Singleton (15) obtained in a somewhat similar experiment. In order to produce the necessary number of test vectors for training purposes each code book of length K was presented cyclically so that the $i^{th}$ vector in the training sequence was the $i(\mathrm{mod}\ K)^{th}$ word of the input code book.

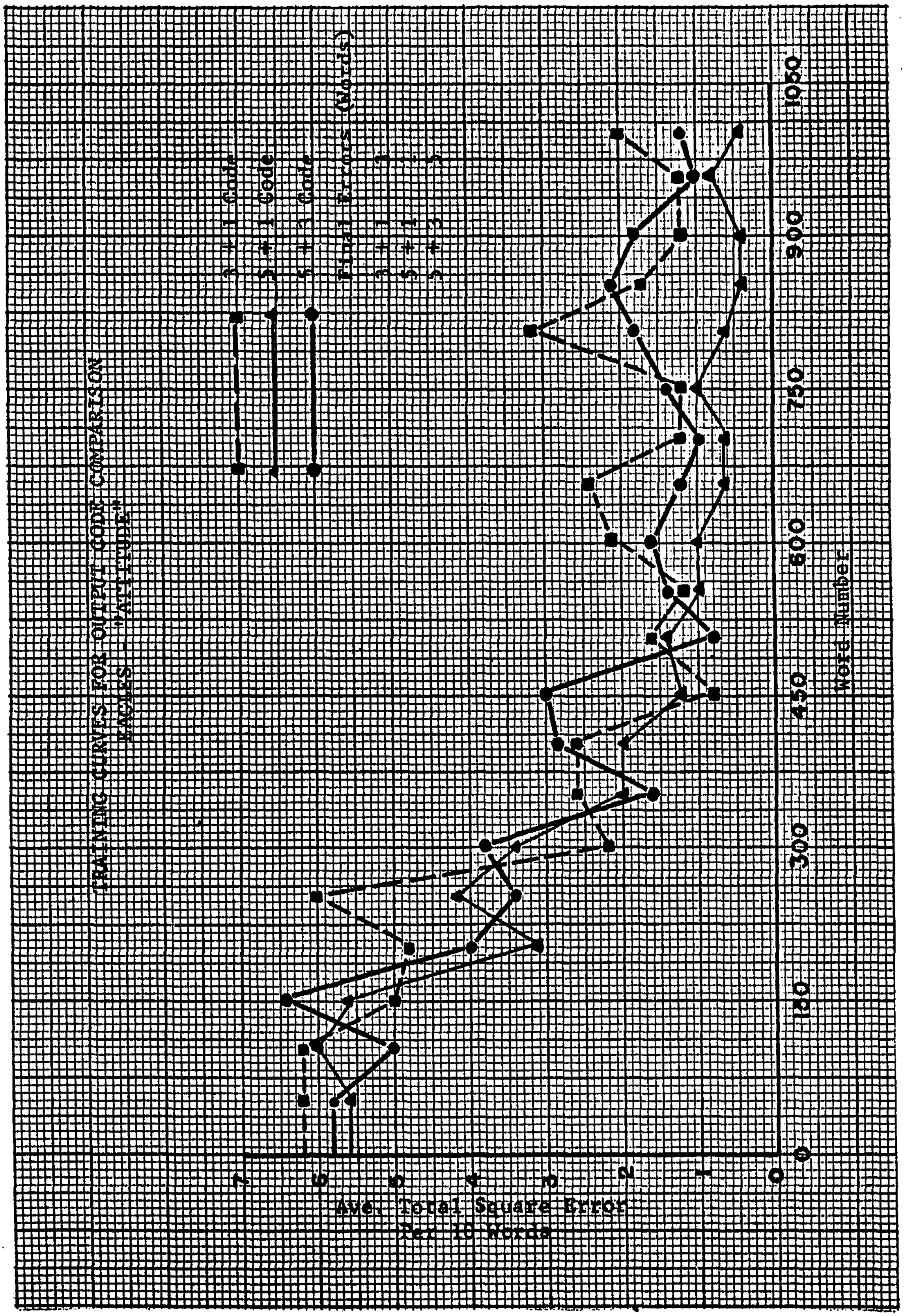

TRAINING CURVES FOR OUTPUT CODE COMPARISON
3 + 1 Code
5 + 1 Code
5 + 3 Code
Final Errors (Words)
Word Number
0
150
300
450
600
750
900
1050
Ave. Total Square Error
Per 10 Words
0
1
2
3
4
5
6
7

Figure 9

After each training sequence was complete the entire code book was decoded using the final weights. Table 11 summarizes the information pertaining to the code book for each variable and also lists the number of words in the entire code book which were classified incorrectly at the end of the training procedure.

Table 11

| Group | Variable | Errors | No.of Binary Digits in ea. Code Word | Total No.of Weights | No.of Code Words | No.of TLU in MFL Machine |
|---|---|---|---|---|---|---|
| Rattlers | Attitude | 4 | 35 | 595 | 41 | 17 |
| | Morale | 0 | 33 | 297 | 32 | 9 |
| | Structure | 0 | 33 | 297 | 39 | 9 |
| | Frustration | 2 | 27 | 459 | 41 | 17 |
| Eagles | Attitude | 5 | 35 | 595 | 41 | 17 |
| | Morale | 0 | 33 | 297 | 34 | 9 |
| | Structure | 0 | 33 | 297 | 38 | 9 |
| | Frustration | 0 | 27 | 459 | 41 | 17 |

As may be seen the maximum classification error for any code book is about 12% and for five out of the eight code books all words may be classified correctly by using the final values of the weights computed for the various TLUs in the machine.

The MFL machine program was designed to keep a current tally of the number of training cases which have been presented to the machine. It also calculated the difference and the square of the difference between the actual and desired output classification for each word in the training sequence and kept 10 word group totals for these quantities. Using the error values, a training or "learning" curve may be plotted for each code book. These curves are shown in Figures 10, 11,12, and 13.

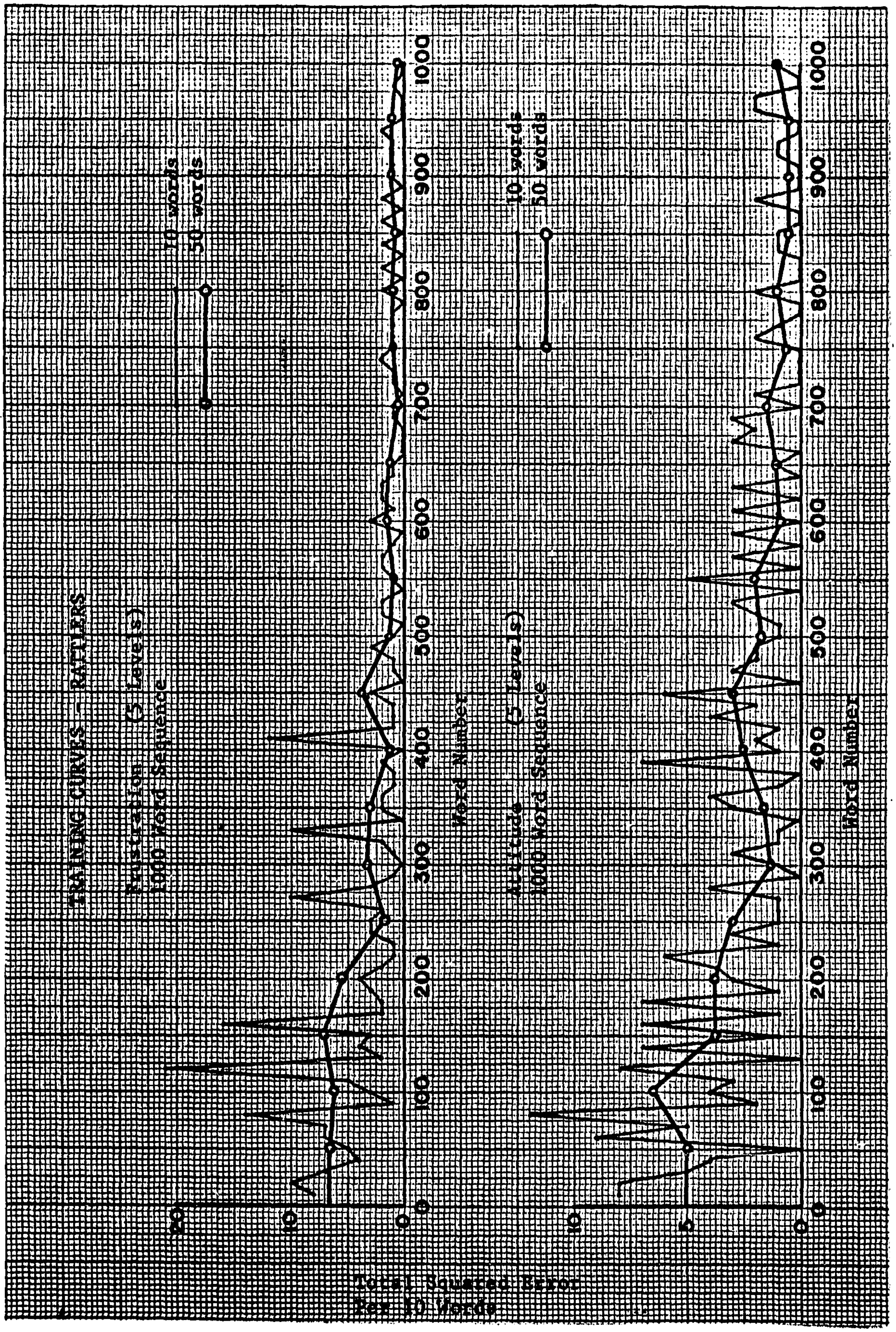

TRAINING CURVES - RATTLERS
Frustration (5 Levels)
1000 Word Sequence
10 words
50 words
Word Number
Attitude (5 Levels)
1000 Word Sequence
10 words
50 words
Word Number
Total Squared Error
Per 10 Words
0
100
200
300
400
500
600
700
800
900
1000
0
5
10
20

Figure 10

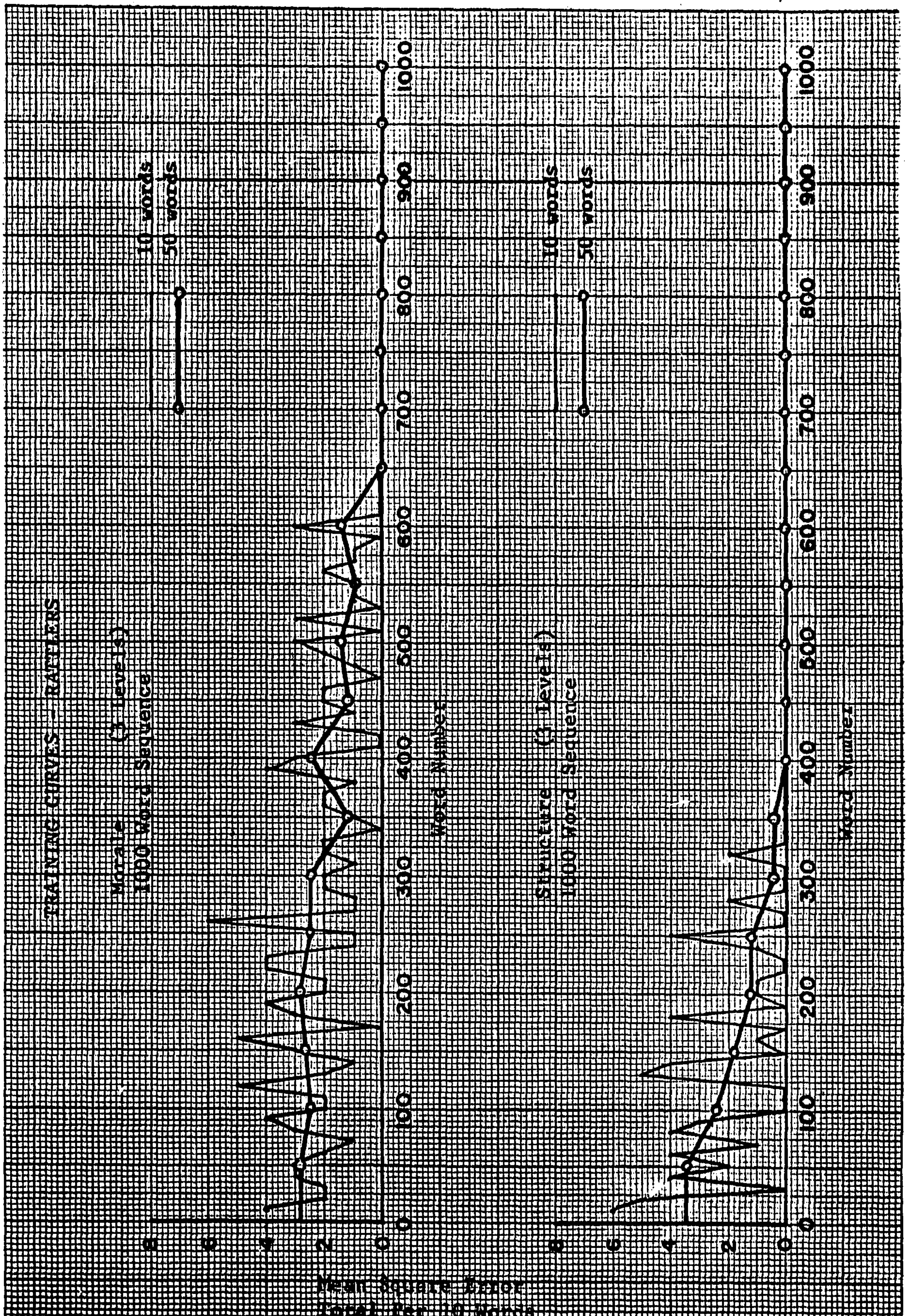
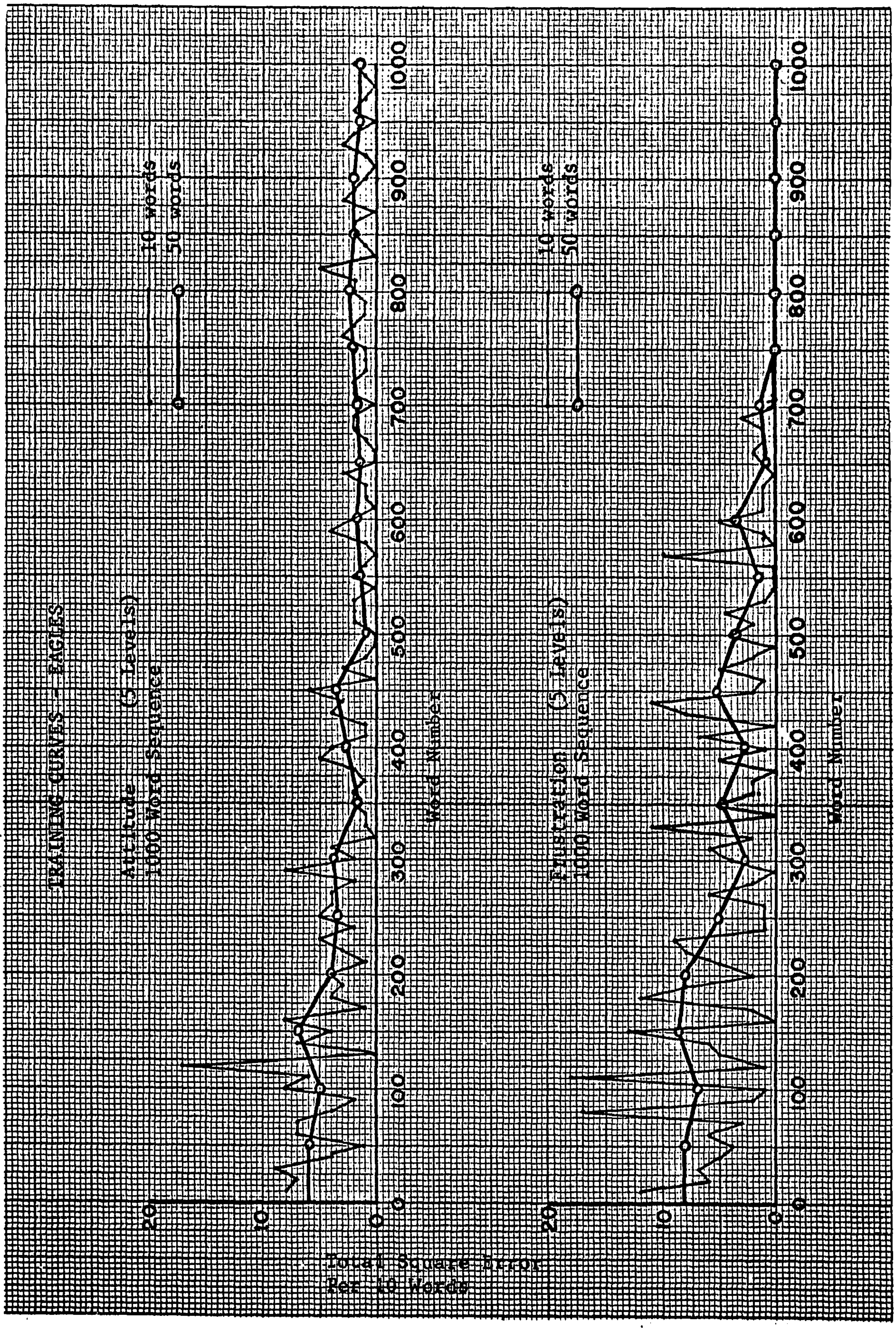

TRAINING CURVES - RATTLERS
Morale (3 levels)
1000 Word Sequence
10 words
50 words
Word Number
Structure (3 levels)
1000 Word Sequence
10 words
50 words
Word Number
Mean Square Error
Total Per 10 Words

**Figure 11**



TRAINING CURVES - EAGLES
Attitude (5 Levels)
1000 Word Sequence
10 words
50 words
Word Number
Frustration (5 Levels)
1000 Word Sequence
10 words
50 words
Word Number
Total Square Error
Per 10 Words
0 100 200 300 400 500 600 700 800 900 1000
0 10 20


**Figure 12**

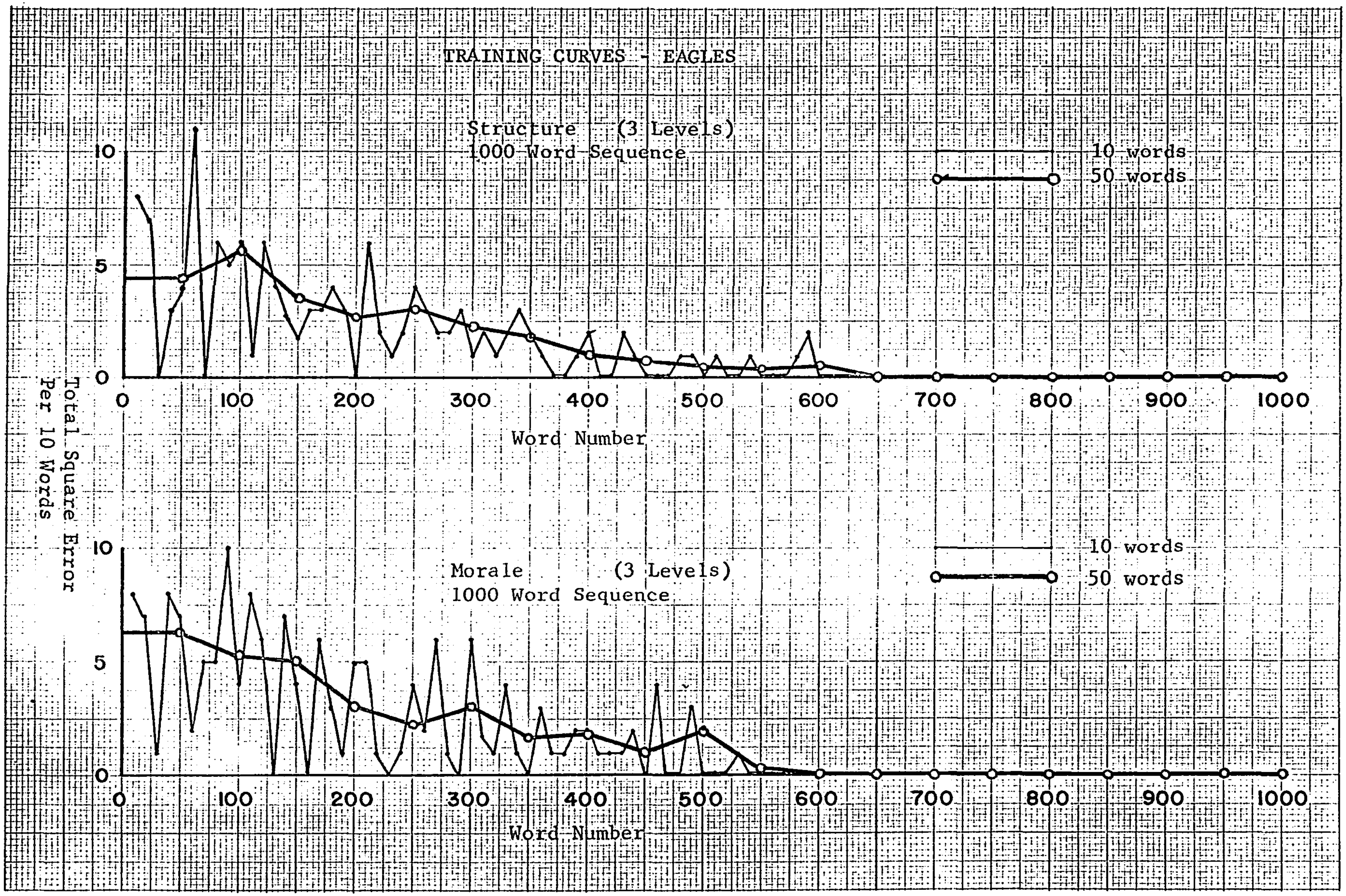

TRAINING CURVES - EAGLES
Structure (3 Levels)
1000 Word Sequence
10 words
50 words
Total Square Error
Per 10 Words
Word Number
Morale (3 Levels)
1000 Word Sequence
10 words
50 words
Word Number

Figure 13

On each learning curve the values of the total squared error for the preceeding group of 10 words is connected by a light line to show the local behavior of the training process. The heavy lines represent a value equal to one-fifth the total squared error for the preceeding group of 50 words. The heavy lined curves thus represent the average squared error per 10 words over the last 50 words and are more nearly indicative of the overall learning behavior of the machine.

Examination of Figures 10 and 12 shows that in general predictors (weights) which will lead to proper classification of the vectors representing the variables "attitude" and "frustration" are more difficult to determine than are those for the classification of the variables "morale" and "structure". Nevertheless, examination of the weights resulting from the Phase I calculations reveals that each group has decided on a well defined group of predictors. From the preceeding example presented in this paper a plot of the final average weights showed a strong preference for two portions of the input code as predictors of the desired output. A plot of these weights given in Figure 14, shows

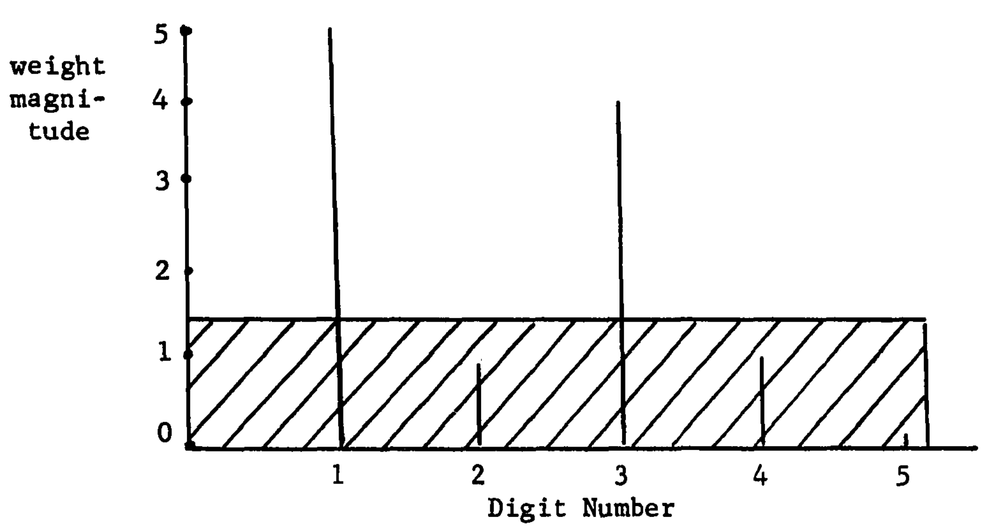



Figure 14. Average Weights for First Example

that although digits one and three are strong predictors, digits two and four also receive non-zero average weight. The strong predictors, then, must be ones whose value is larger than some "noise level" or arbitrary cutoff level indicated by the shaded area in Figure 14. If the weights for the Rattlers and Eagles are plotted in a similar manner some very strong predictors are indicated.

The weight profiles, consisting of Figures 15 through 18 for the Rattlers and Figures 19 through 22 for the Eagles, are very good graphic indicators of the predictors used by each group. These plots are derived from the raw data contained in Appendix C by adding the weights for the $i^{th}$ digits for each TLU to obtain a total weight for the $i^{th}$ digit. The marks indicate the dividing lines between the various variables in the code word and are interpreted as follows:

<u>Attitude</u>

| no. times | type | value | attainment | last frust. |
|---|---|---|---|---|
| eleven digits ▲ | four digits ▲ | four digits ▲ | three digits ▲ | five digits ▲ |

| ave.att. | presence | tag |
|---|---|---|
| five digits ▲ | two digits ▲ | one |

<u>Structure</u>

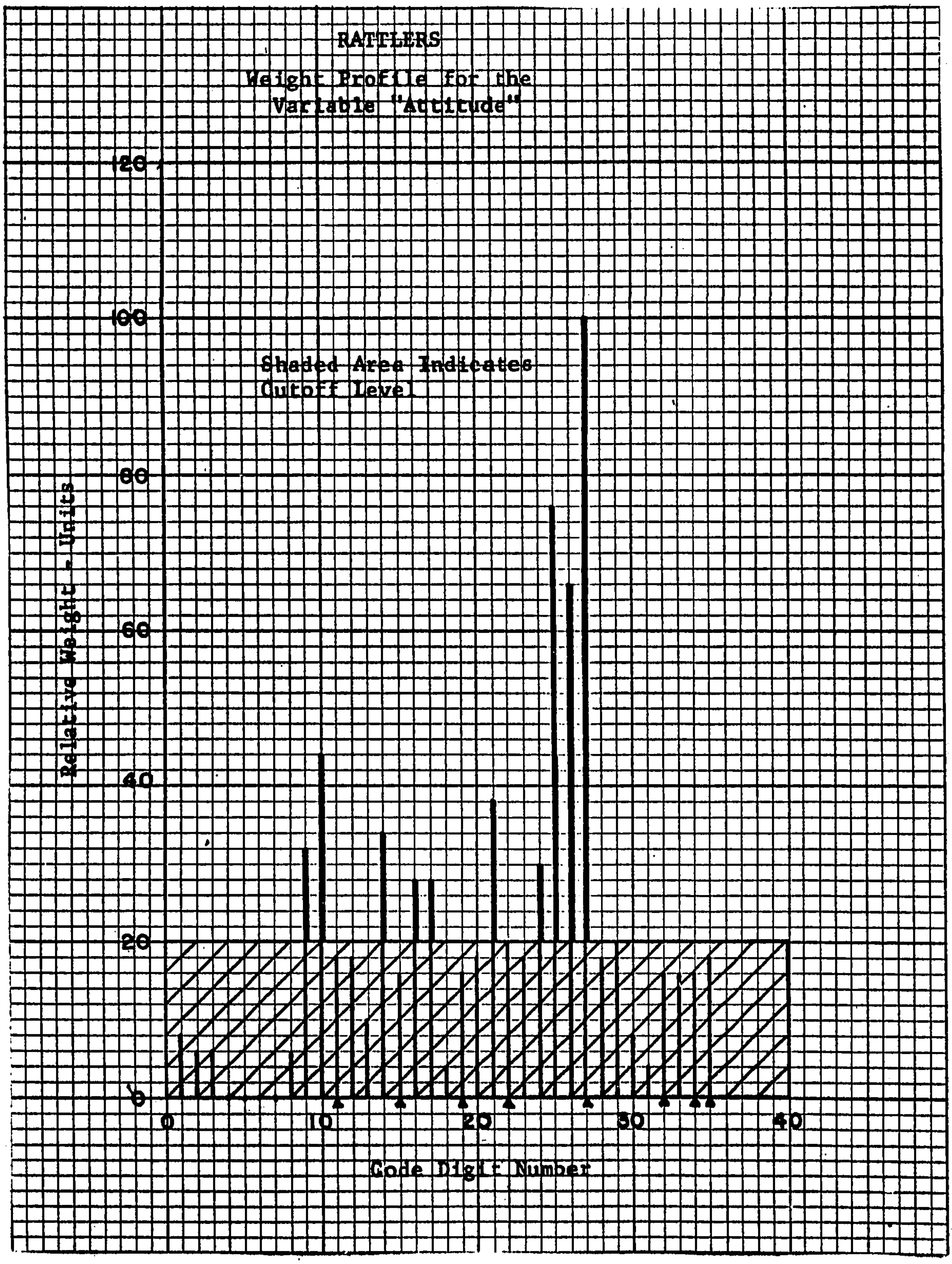
| no. times | type | value | attainment | last frust. |
|---|---|---|---|---|
| eleven digits ▲ | four digits ▲ | four digits ▲ | three digits ▲ | five digits ▲ |

| ave. struct. | presence | tag |
|---|---|---|
| three digits ▲ | two digits ▲ | one |

<u>Morale</u>

| no. times | type | value | attainment | last frust. |
|---|---|---|---|---|
| eleven digits ▲ | four digits ▲ | four digits ▲ | three digits ▲ | five digits ▲ |

| ave.morale | presence | tag |
|---|---|---|
| three digits ▲ | two digits ▲ | one |



RATTLERS
Weight Profile for the
Variable "Attitude"
Shaded Area Indicates
Cutoff Level
Relative Weight - Units
120
100
80
60
40
20
0
0
10
20
30
40
Code Digit Number


Figure 15

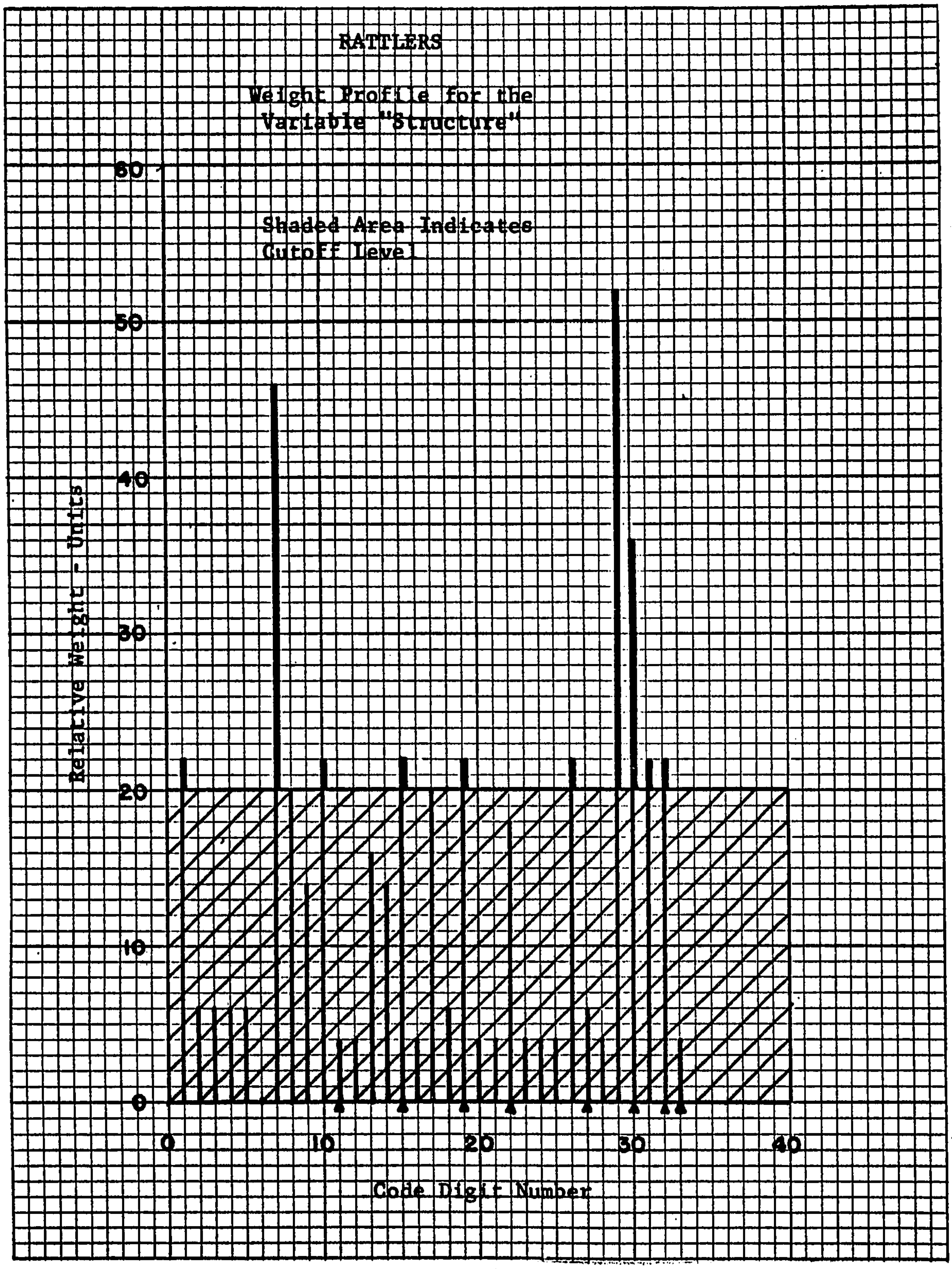

RATTLERS
Weight Profile for the
Variable "Structure"
Shaded Area Indicates
Cutoff Level
60
50
40
30
20
10
0
Relative Weight - Units
0
10
20
30
40
Code Digit Number

Figure 16

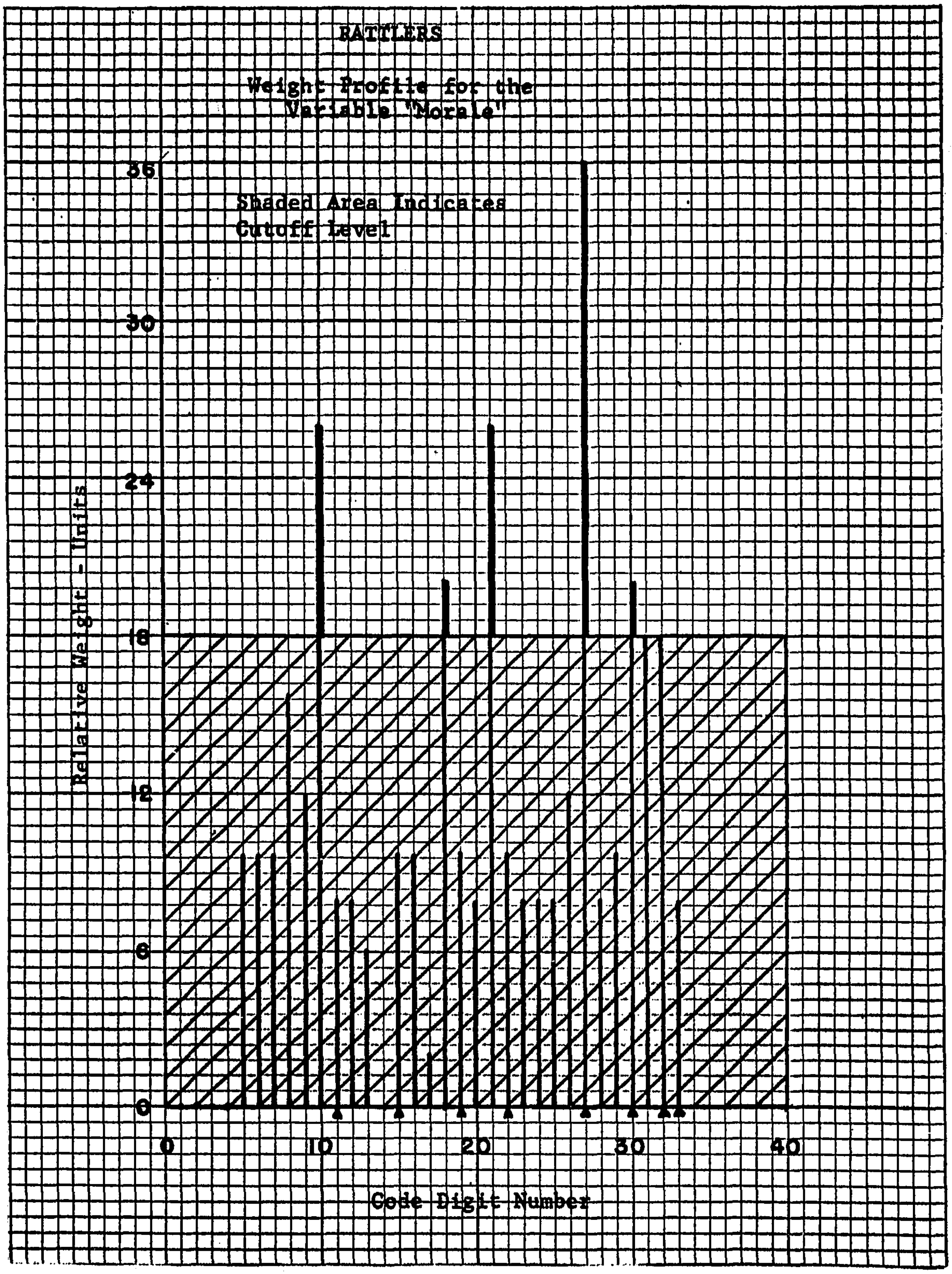

RATTLERS
Weight Profile for the
Variable "Morale"
Shaded Area Indicates
Cutoff Level
Relative Weight - Units
36
30
24
18
12
6
0
0
10
20
30
40
Code Digit Number

Figure 17

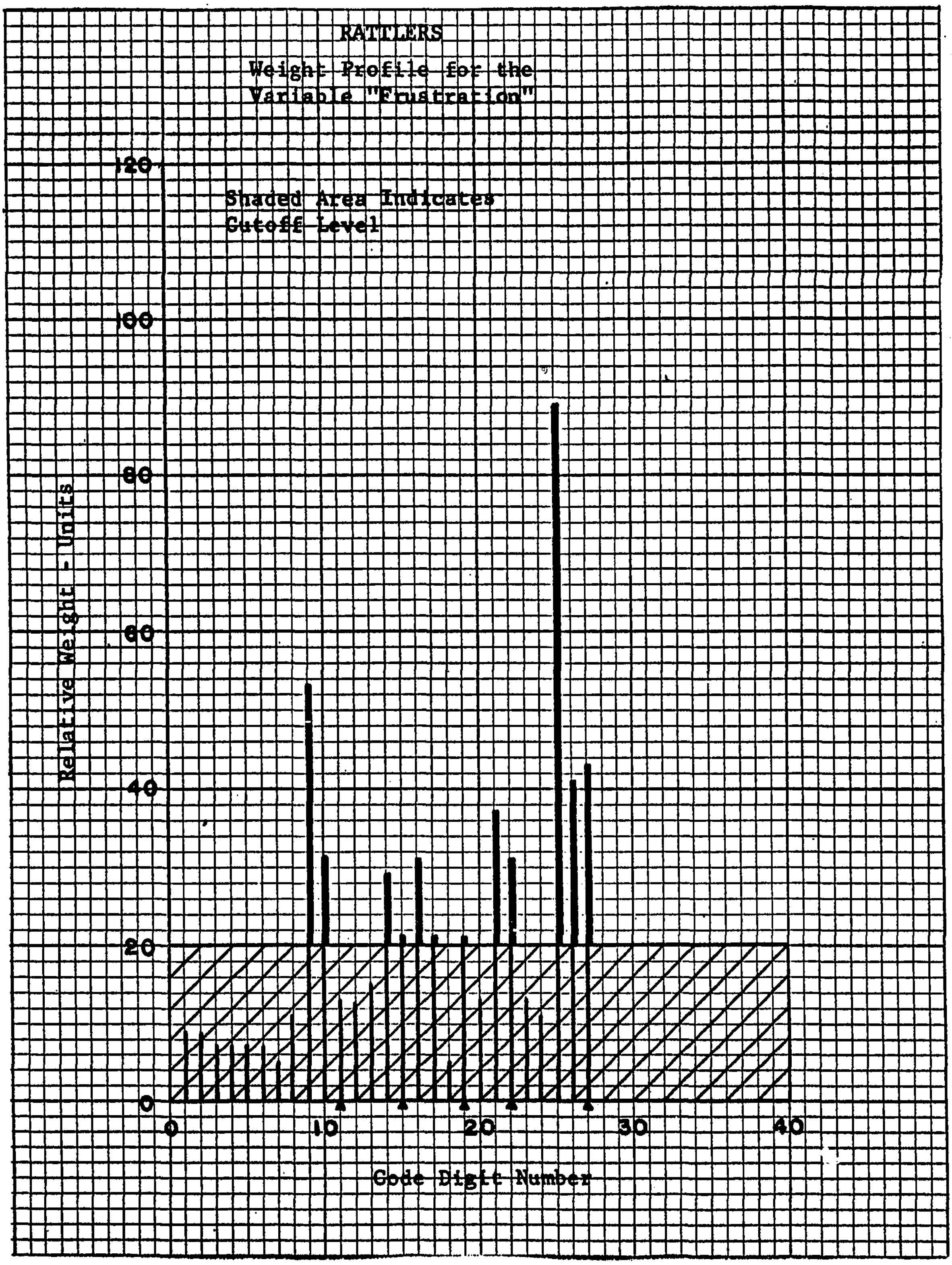

RATTLERS
Weight Profile for the
Variable "Frustration"
Shaded Area Indicates
Cutoff Level
Relative Weight - Units
120
100
80
60
40
20
0
0
10
20
30
40
Code Digit Number

Figure 18

Frustration

| no. times | type | value | attainment | last frust. |
|---|---|---|---|---|
| eleven digits ▲ | four digits ▲ | four digits ▲ | three digits ▲ | five digits ▲ |

The digit which receives a large weight may then be related to the physical situation by reference to the original procedure given in Table 9. One may, for instance, compare Figures 15 and 19 and notice both similarities and differences between the two groups concerning predictors influencing their attitude. Both are influenced by repeated goals of the same type, the Rattlers responding most to the second and third occurrences but the Eagles placing greater emphasis on persistence, i.e., the fifth occurrence is weighted heavily. Both respond to cooperative goals, the Eagles being the more responsive of the two in a relative sense. Both are influenced by low and medium value goals. This influence could either be in a negative (don't care) or positive (care) sense since only influence is indicated by the predictors. Both are influenced to a certain extent by goal attainment, the emphasis being placed more heavily on ties. The attitude of each group toward the other is influenced most strongly by their frustration level, the Rattlers being most influenced by high frustration values while the Eagles are more influenced by average or near to average frustration levels. At this point the groups differ. The Eagles seem to place some emphasis on their past attitude if it was neutral and on the presence of another group, while the Rattlers place no emphasis on these variables.

A set of statements concerning the other three variables could be made similar to the preceeding statements if one were to compare Figures 16 and 20, 17 and 21, and 18 and 22.

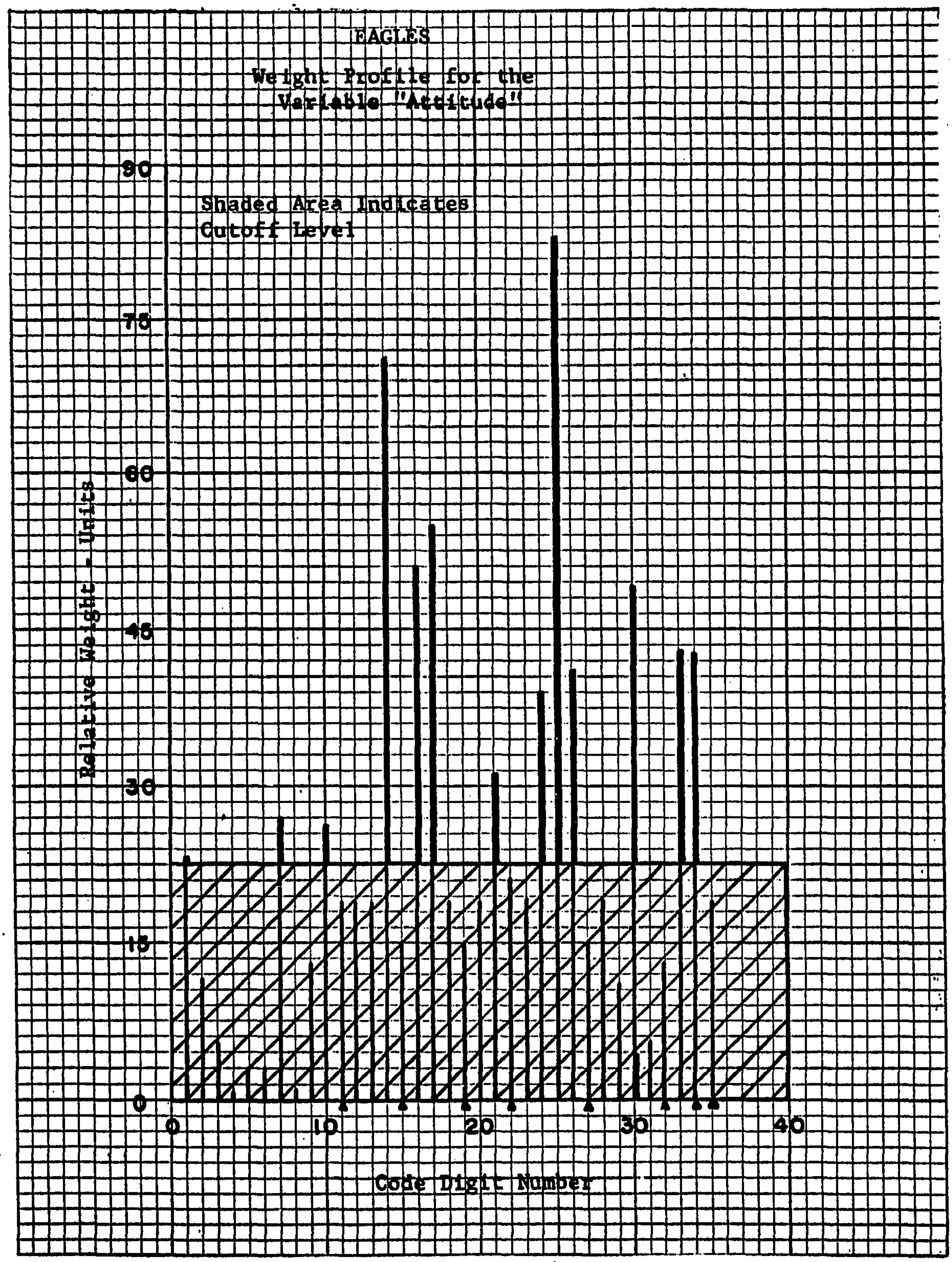

EAGLES
Weight Profile for the
Variable "Attitude"
Shaded Area Indicates
Cutoff Level
Relative Weight - Units
Code Digit Number
0
15
30
45
60
75
90
0
10
20
30
40

Figure 19

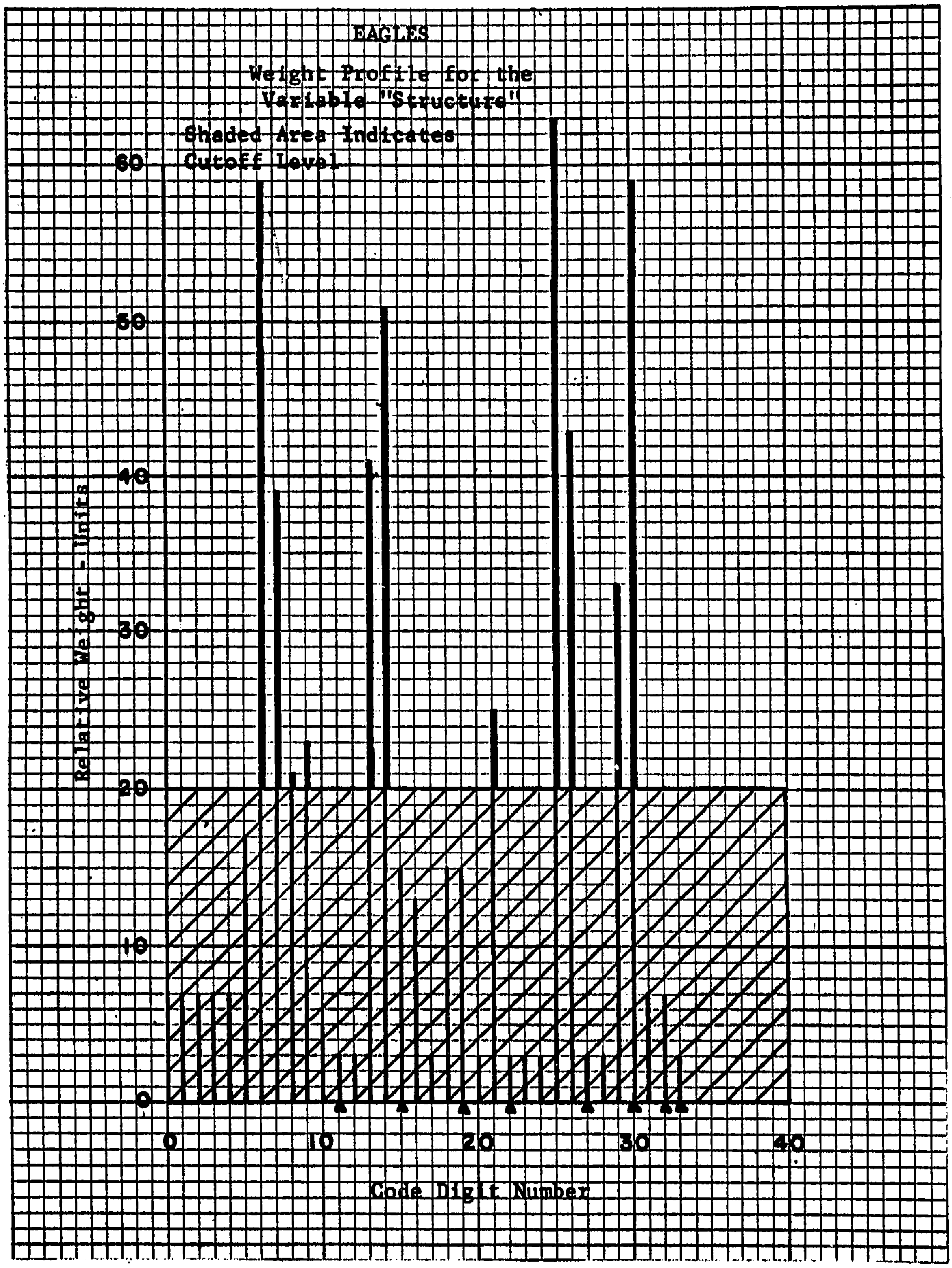
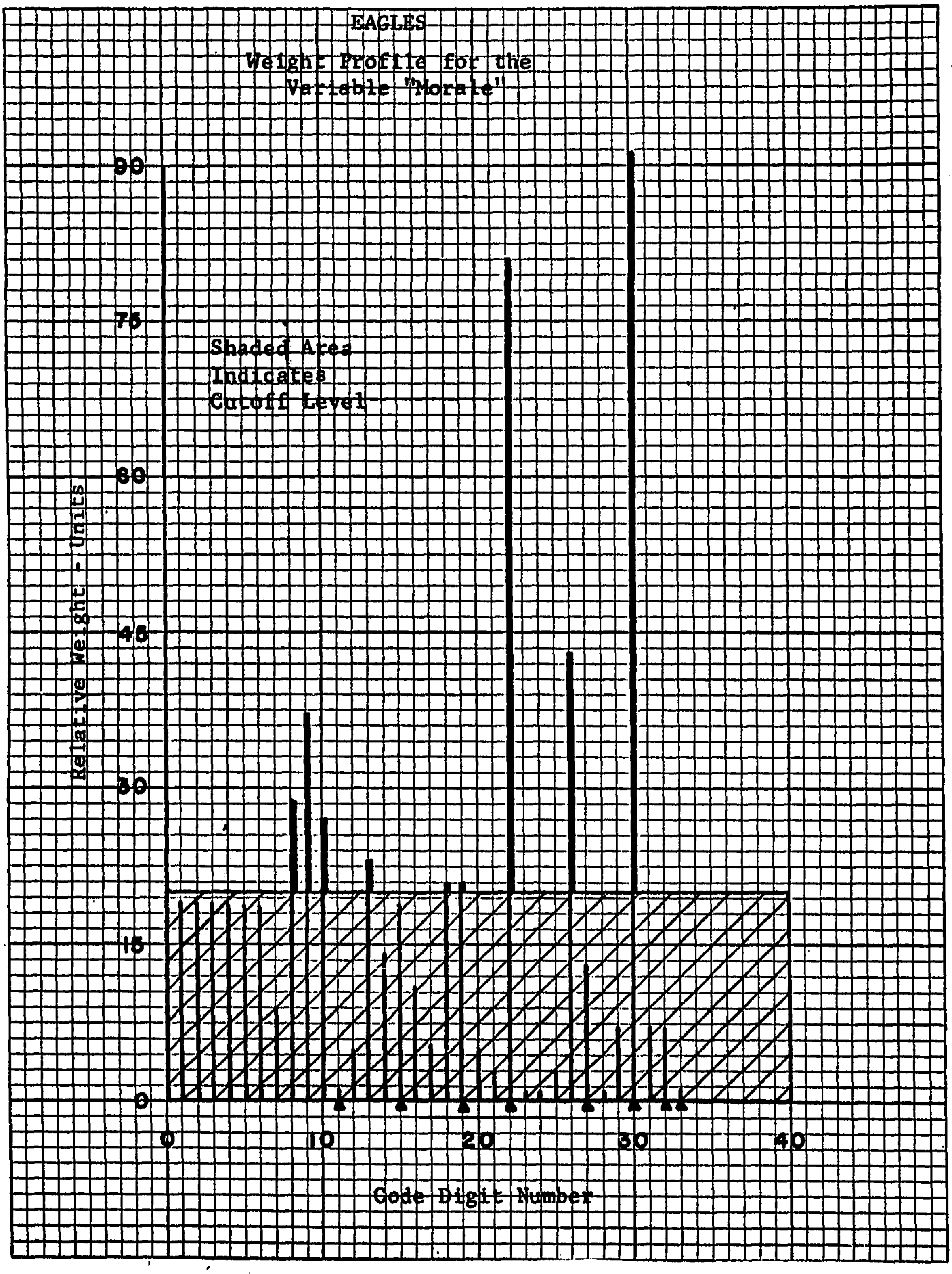

EAGLES
Weight Profile for the
Variable "Structure"
Shaded Area Indicates
Cutoff Level
60
50
40
30
20
10
0
Relative Weight - Units
0
10
20
30
40
Code Digit Number

Figure 20



EAGLES
Weight Profile for the Variable "Morale"
Shaded Area Indicates Cutoff Level
Relative Weight - Units
90
75
60
45
30
15
0
0
10
20
30
40
Code Digit Number


Figure 21

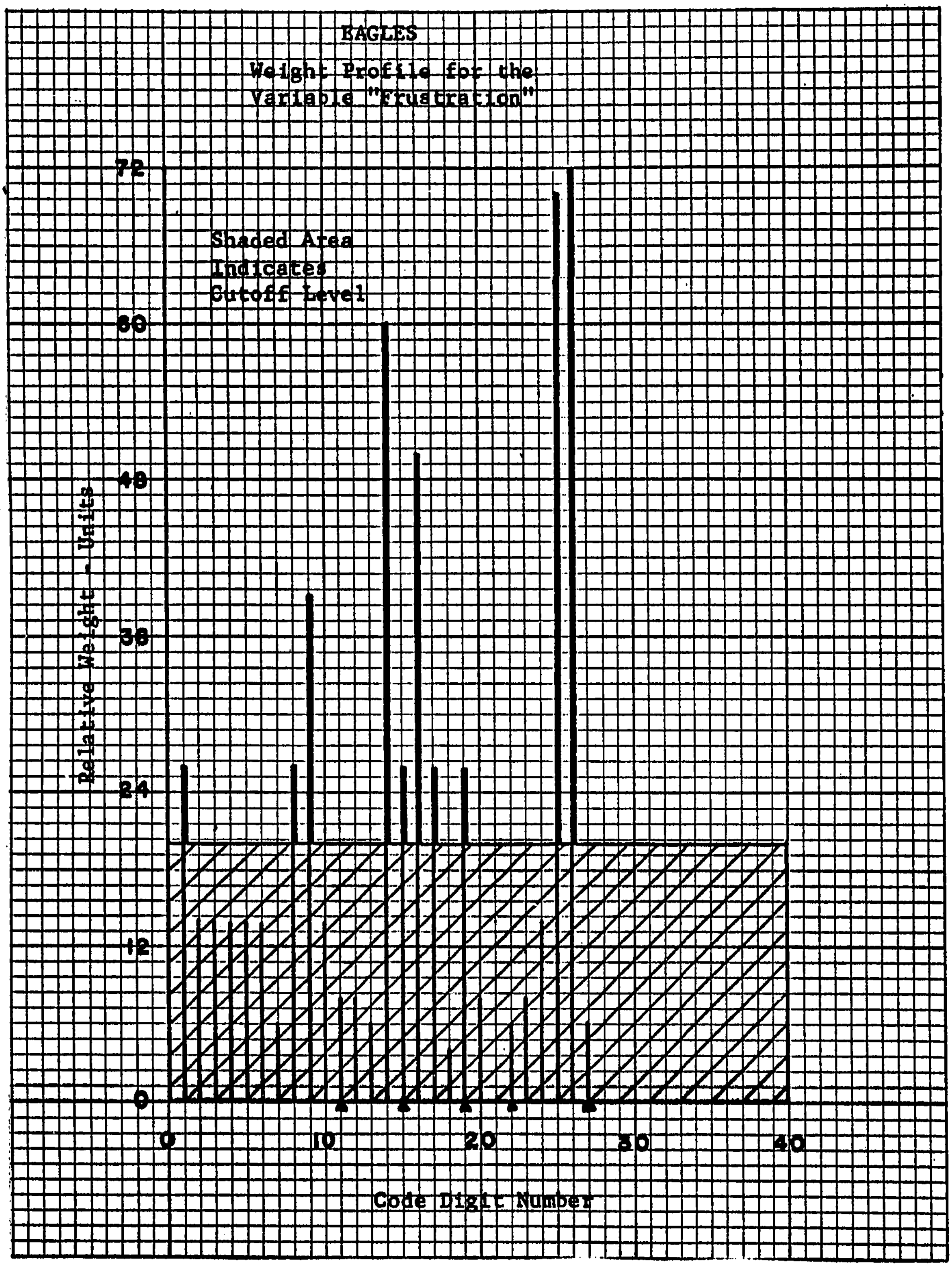

EAGLES
Weight Profile for the
Variable "Frustration"
Shaded Area
Indicates
Cutoff Level
72
60
48
36
24
12
0
Relative Weight - Units
0
10
20
30
40
Code Digit Number

Figure 22

The second phase of this simulation concerns the reaction of the two groups to a set of inputs consisting only of independent variables. In this instance these variables are goal type, goal value, and presence of both groups in the same place. The groups are characterized by the sets of weights resulting from Phase I.

The simulation program, which is contained also in Appendix B, receives a set of independent variable values as a situation input. From this set of input variables the values of all other necessary values are computed, past averages are computed and a set of four situation vectors are classified and a value for the four dependent variables is computed. Once this is done the next situation is presented and the group reaction computed. In this manner one may generate hypotheses concerning the behavior of the group when a certain string of inputs is presented.

In this phase of the simulation the groups were first confronted with the same sequence of inputs as in the original experiment. This was done in order to provide a check on the accuracy of the simulation. The results are shown in Figures 23 and 24. It will be noticed that the simulation fills in the blanks in the original code book and reproduces the original situation with a maximum of approximately 12% error. In Figures 25 and 26 the inputs have been interchanged. That is to say, the Rattlers receive all 41 of the Eagles original inputs in the same sequence and vice versa for the Eagles. The reaction of the groups is decidedly different with the Eagles seeming to react more slowly to changes in their fortunes than the Rattlers did in the same situation.

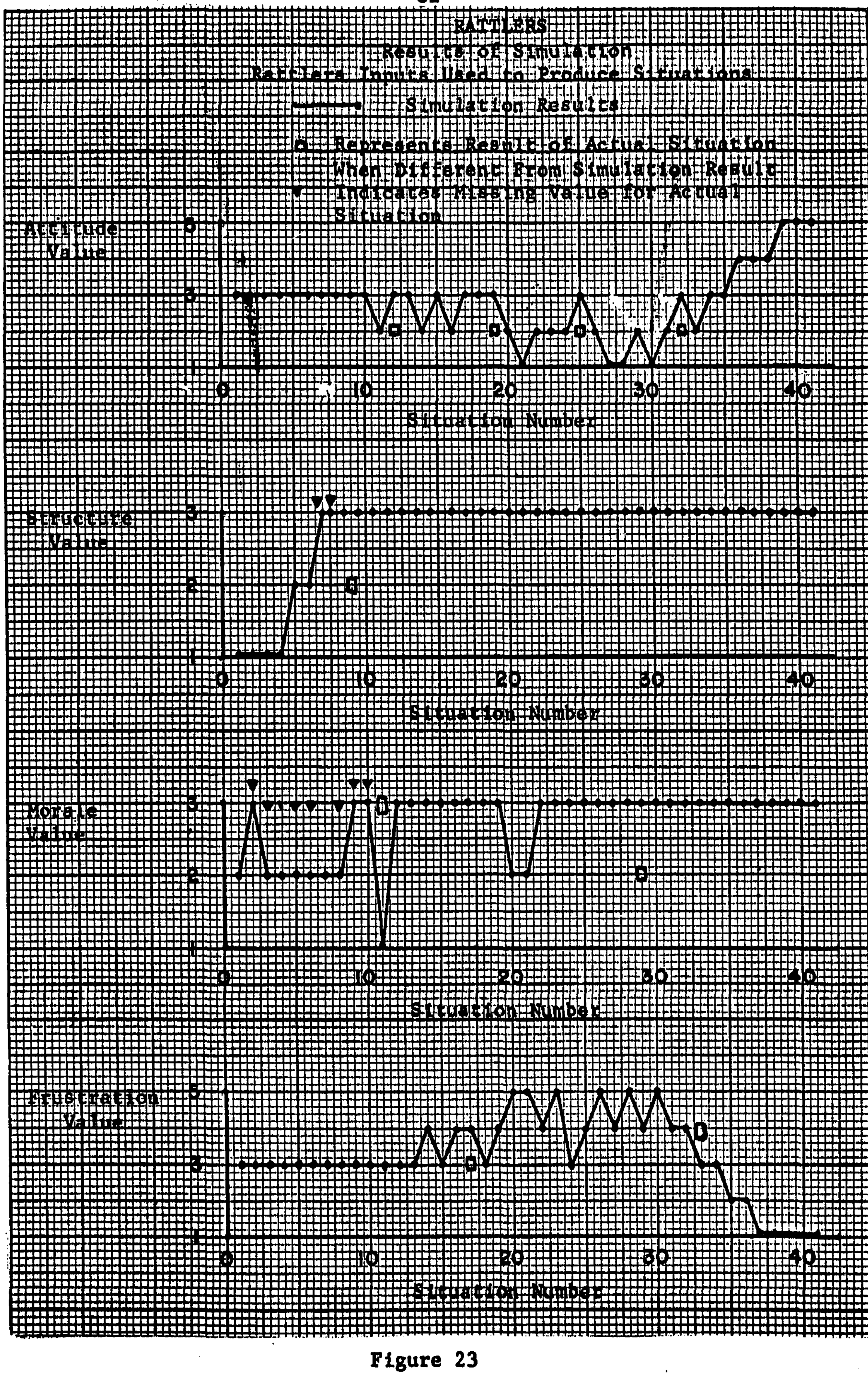

RATTLERS
Results of Simulation
Rattlers Inputs Used to Produce Situations
Simulation Results
Represents Result of Actual Situation When Different From Simulation Result
Indicates Missing Value for Actual Situation
Attitude Value
Situation Number
Structure Value
Situation Number
Morale Value
Situation Number
Frustration Value
Situation Number
0
10
20
30
40

Figure 23

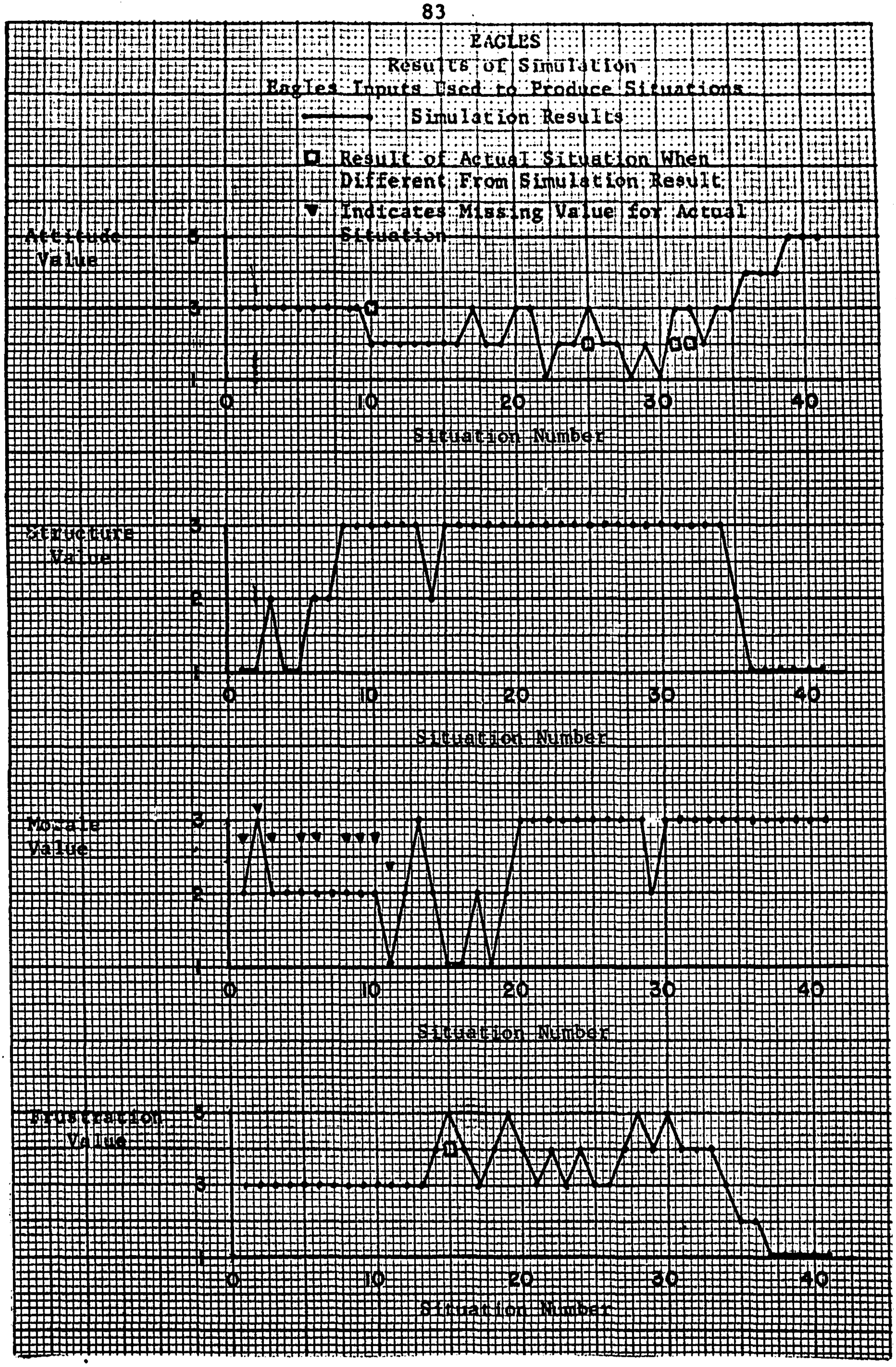

EAGLES
Results of Simulation
Eagles Inputs Used to Produce Situations
Simulation Results
Result of Actual Situation When Different From Simulation Result
Indicates Missing Value for Actual Situation
Attitude Value
Situation Number
Structure Value
Situation Number
Morale Value
Situation Number
Frustration Value
Situation Number

**Figure 24**

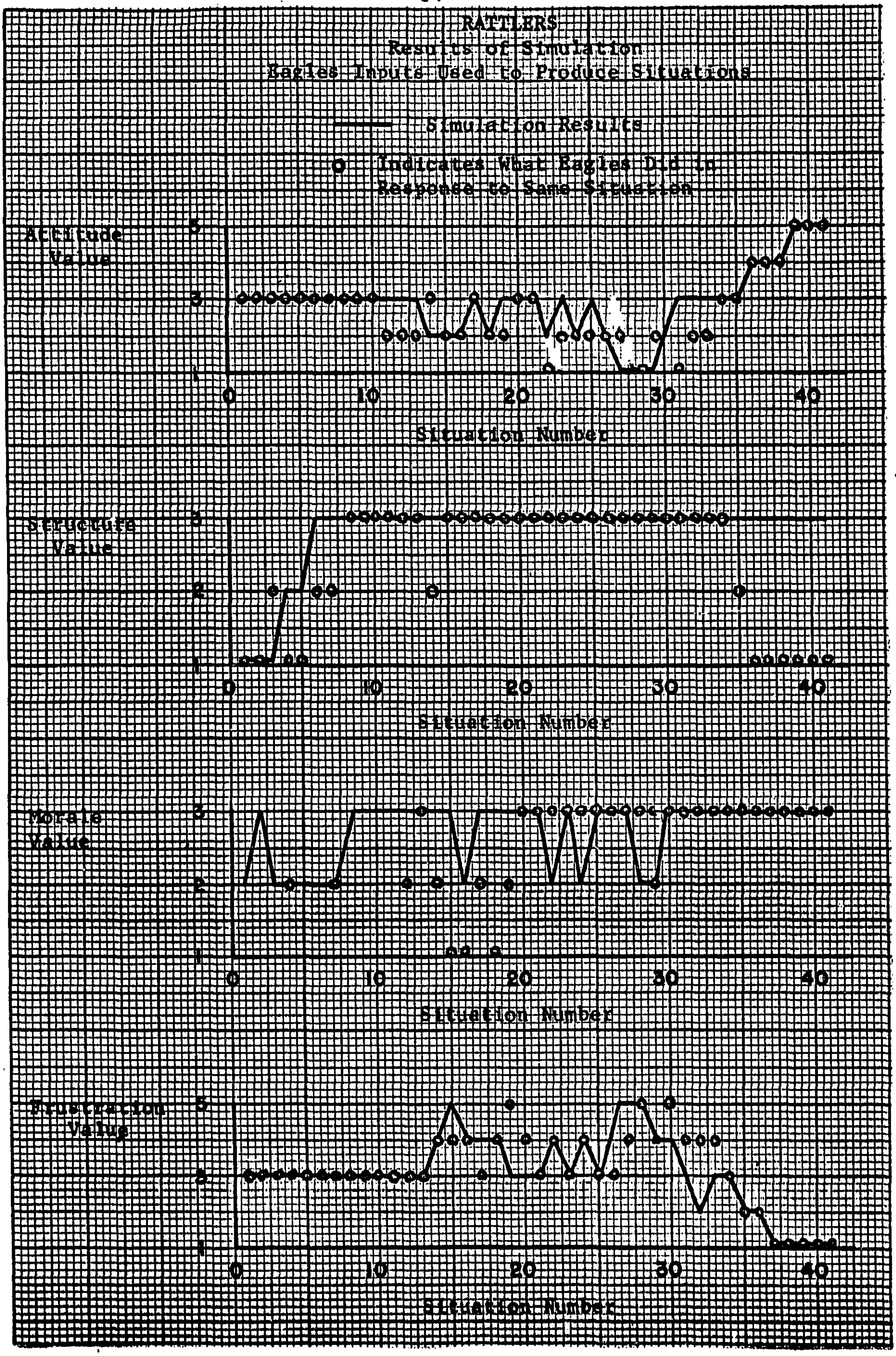

RATTLERS
Results of Simulation
Eagles Inputs Used to Produce Situations
Simulation Results
o Indicates What Eagles Did in Response to Same Situation
Attitude Value
Situation Number
Structure Value
Situation Number
Morale Value
Situation Number
Frustration Value
Situation Number

Figure 25

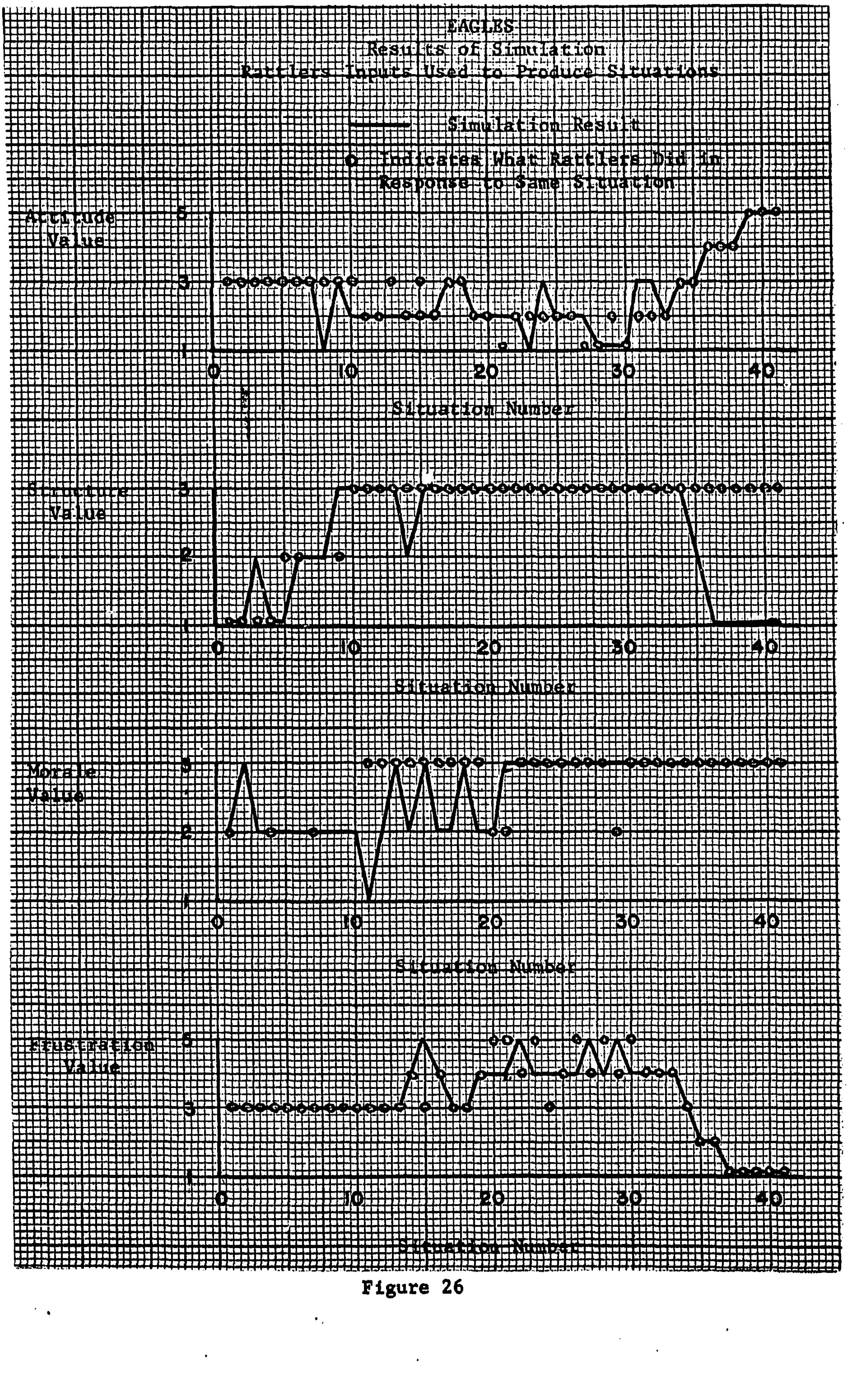

EAGLES
Results of Simulation
Rattlers Inputs Used to Produce Situations
Simulation Result
o Indicates What Rattlers Did in Response to Same Situation
Attitude Value
Situation Number
Structure Value
Situation Number
Morale Value
Situation Number
Frustration Value
Situation Number

Figure 26

## CHAPTER VII

## CONCLUSIONS AND RECOMMENDATIONS

Previous investigators (16, 3, 4, 9) have concluded that adaptive processing provides results which are at least comparable with the results gained by linear regression analysis. They have also found that the running time of the average adaptive processing program during the training phase has been shorter than the comparable linear programming problem. These results were obtained by processing data which yielded to both types of manipulation.

This paper has presented a coding scheme which will enable one to look for patterns or functional dependencies in information which may be characterized as "yes" or "no" statements, objective or subjective gross estimates (high, low, medium, etc.) and standard numeric information. Data concerning a given phenomenon may be mixed in one, two or all of these forms and may still be encoded into a single set of statements concerning that particular phenomenon. No a priori statements about importance of any variables need be made since all information concerning the state of the process at one point in time is processed in one fell swoop and the result of the processing is a statement about the relative importance of the various variables. In the examples presented it was shown that the predictors resulting from this calculation are easily recognizable and may be found from the weights each TLU in the MFL machine assigns to each digit in the input code by simply algebraically adding the weights corresponding to each digit.

These calculations may be made using a relatively small computer if it is fast and can effectively transfer information from bulk storage. The initial training runs and all of the investigation concerning output code sensitivity were made using a Burroughs 205 with all arithmetic being done in fixed point. All of the codes were stored in memory using a two digit word length but there was no loss of accuracy since all of the numbers concerning each TLU are either binary numbers or small integers less than say, 1000. The remaining training runs and the simulation runs were made using an IBM 1410 with a fixed and floating point word length of 4. This did not result in any loss of accuracy in the final result and speeded up the running time by about a factor of 3. It is easily seen, then that a short word length computer could be used to process code words of any length provided sufficient bulk storage was available.

It is not the author's intention to imply that this technique of information processing should replace all others. Rather, since pattern recognizing learning machines exist they should be employed to recognize something besides hand written characters. Some thought should be given to the recognition of regularities in other kinds of information and information about social systems is of a type in which patterns are often sought and less frequently found.

McWhinney (11) has mentioned that data concerning communication net experiments is only superficially processed since it is not amenable to processing by standard means. An attempt should be made to obtain and process information of this type using a learning machine. Also, an "on line" experiment conducted concurrently with an experiment such as the Robbers Cave experiment would be very helpful in providing a clearer understanding of the relationship between the weights in the learning machine and the actual process.

In conclusion let it be said that engineering technology has made available in the computer a highly flexible computational tool. Effort should now be expended to integrate this machine into the scientific experimental process as a decision making tool as well.

## REFERENCES


1. Chow, C. K., "A Recognition Method Using Neighbor Dependence," IRE Transactions on Electronic Computers, Vol. EC-11, No. 5, October 1962.

2. Coe, R. M., "Conflict, Interference and Aggression: Computer Simulation of a Social Process," Behavioral Science, Vol. 9, No. 2, April 1964.

3. Duda, R. O., J. W. Machanik, R. C. Singleton, "Function Modeling Experiment," SRI Project 3605 Technical Report, Stanford Research Institute, Menlo Park, California, December 1963, AD 427 108.

4. Farley, B. G., and W. A. Clark, "Generalization of Pattern Recognition in a Self-Organizing System," Proceedings of the Western Joint Computer Conference, Los Angeles, 1955

5. Farley, B. G., L. S. Frishkopf, W. A. Clark, J. T. Gilmore, "Computer Techniques for the Study of Patterns in the Electroencephologram," Lincoln Laboratory Technical Report No. 165, Massachusetts Institute of Technology.

6. Guetzkow, H., and W. R. Dill, "Factors in the Organizational Development of Task Oriented Groups," Sociometry, Vol. 20, 1957, pp. 175-204.

7. Gullahorn, J. T., and J. E. Gullahorn, "A Computer Model of Elementary Social Behavior," Behavioral Science, Vol. 8, No. 4, October 1963.

8. Homans, G. C., Social Behavior: Its Elementary Forms, Harcourt, Brace, 1961.

9. Lorenz, E. N., "Nonlinear versus Linear Objective Weather Prediction," IRE International Convention Record, Part 2, 1960, pp. 78-87.

10. McCullock, W. S., and W. Pitts, "A Logical Calculus of the Ideas Imminent in Nervous Activity," Bulletin of Mathematical Biophysics, Vol. 5, 1943, pp. 115-133.

11. McWhinney, W. H., "Simulating the Communication Network Experiments," Behavioral Science, Vol. 9, No. 1, January 1964.

12. Peterson, W. W., Error Correcting Codes, Academic Press, Inc., 1959.

13. Rosenblatt, F., Principles of Neurodynamics, Spartan Books, Washington, D. C., 1962.

14. Sherif, M., O. J. Harvey, B. J. White, W. R. Hood, and C. W. Sherif, Intergroup Conflict and Cooperation, the Robbers Cave Experiment, Institute of Group Relations, The University of Oklahoma, 1961.

15. Singleton, R. C., "Preliminary Report on Function Approximation with Adaptive Systems," Stanford Research Institute, Menlo Park, California, June 1962.

16. Widrow, B., "Generalization and Information Storage in Networks of Adaline 'Neurons'," Self Organizing Systems 1962, ed: M.C. Yovitts, G. T. Jacoby, and G. O. Goldstein, Spartan Books, Washington, D. C., 1962, pp. 435-461.

## BIBLIOGRAPHY

Anonymous, "New Perceptron Shown", Electronic Design News, Vol. 5, July 1960.

Braverman, D. J. "Machine Learning and Automatic Pattern Recognition", Stanford Electronics Labs, Stanford University, Stanford, California, Technical Report No. 2003-1, February 1961.

Feldman, J., and E. A. Feigenbaum, et al., Computers and Thought, ed: Feigenbaum, E. A. and J. Feldman, McGraw Hill, Inc., New York, N.Y., 1963.

Fukunaga, K., and T. Ito, "Learning Effect in Self Organizing Systems", Mitsubishi Denki Laboratory Reports, Amagasaki, Japan, Vol. 6, No. 2, April 1965.

Goode, H. H., and R. E. Machal, System Engineering, McGraw Hill, Inc., New York, N.Y., 1957.

Green, B. F., Jr., "Computer Languages for Symbol Manipulation", IRE Transactions on Human Factors in Electronics, Vol. HFE-2, March 1961.

Hartmanis, J., and C. J. Munsey, "A Magnetic Integrator for the Preception Program", Aeronutronic, Annual Summary Report, April 1, 1959 to May 31, 1960, July, 1960.

Highleyman, W. H., "Linear Decision Functions With Application to Pattern Recognition", Proceedings of the IRE, Vol. 50, pp. 1501-1514, June 1962.

Hu, M. K., E. Kletsky, S. Golman, E. Stabler, and M. Krieger, Theory of Adaptive Mechanisms", Rome Air Development Center Technical Report RADC-TDR-63-334, December 1963 AD 429 935.

Kemeny, J. G., and J. L. Snell, Mathematical Models in the Social Sciences, Ginn and Company, 1962.

Kimura, T., and T. Kuroda, "An Experiment on Compounding Recognition Systems", Mitsubishi Denki Laboratory Reports, Amagasaki, Japan, Vol. 5, No. 3, July 1964.

Lee, R. J., "Theory of Probability State Variable Systems", Adaptronics Inc., Alexandria, Virginia, Technical Report ASD-TDR-63-664, Vol. 2, December 1963. AD 428 099.

Margolis, M., "On the Theory of Process Control Systems, the Learning Model Approach", University of California at Los Angeles, Report No. 60-32, May 1960.

Mattson, R. L., O. Firschein, and M. Fischler, "An Experimental Investigation of a Class of Pattern Recognition Synthesis Algorithms", IRE Transactions on Electronic Computers, Vol. EC-13, pp. 586-593, October 1964.

Mays, C. H., "Adaptive Threshold Logic", Technical Report No. 1557-1, Stanford Electronics Labs, Stanford University, Stanford, California, April 1963.

Mosteller, F., R. R. Bush, "Stochastic Models for Learning", John Wiley and Sons, New York, N.Y., 1955.

Mosteller, H. W., "Learning Control Systems", General Electric Electronics Laboratory Report, Syracuse, N.Y., November 1963.

Murray, A. E., "Perceptron Applicability to Photointerpretation (Project PICS)" Cornell Aeronautical Lab., Buffalo, N.Y., Report No. VE-1346-G-1, November 1960.

Rosenblatt, F., "On the Convergence of Reinforcement Procedures in Simple Perceptrons", Cornell Aeronautical Lab., Buffalo, N.Y., Report No. VG-1196-G-4, February 1960.

Rosenblatt, F., "Perceptron Simulation Experiments", Proceedings of the IRE, Vol. 48, pp. 301-309, March 1960.

Sebestyn, G. S., Decision Making Processes in Pattern Recognition, The MacMillan Co., New York, N.Y., 1962.

Stearns, S. D. "A Method for the Design of Pattern Recognition Logic", IRE Transactions on Electronic Computers", Vol. EC-9, pp. 48-53, March 1960.

Tou, J. T., P. D. Joseph, and J. B. Lewis, "A Study of Digital Adaptive Control Systems, School of Electrical Engineering, Purdue University, Lafayette, Indiana, Technical Report No. 1, September 1960.

Uhr, T., and C. Vossler, "Suggestions for Self Adapting Computer Models of Brain Functions", Behavioral Science, Vol. 6, pp. 91-97, January 1961.

Von Neumann, J., et al., Automata Studies, ed. C. E. Shannon and J. McCarthy, Princeton University Press, Princeton, New Jersey, 1956.

Widrow, B. and M. E. Hoff, "Adaptive Switching Circuits", IRE WESCON Convention Record, Part 4, 1960.

# APPENDIX A

This Appendix contains a duplicate of the original content analysis which was used in forming the code words processed in Example 2. The number in the left hand margin (when there is one) indicates the situation number assigned to that statement or group of statements. These situation numbers correspond to the ones in Tables 7 and 8.

# ANALYSIS OF ROBBERS' CAVE EXPERIMENT

## RATTLERS

### Stage 1:

A. System objects

1. Goal-oriented interaction → hierarchical structure
2. Structure → In-group identity
3. In-group identity → norms
4. Norms → sanctions

| B. System inputs | | C. System outputs |
|---|---|---|
| 1. Discovery of swimming hole | | 1. a) Group selects largest boy (Brown) to make out swim buddy list<br>b) Brown and Mills lead in improving swimming hole |
| 2. Meal must be prepared | | 2. Simpson leads meal preparation |
| 3. Canoe must be carried | | 3. Brown and Simpson lead canoe carrying operation |
| 4. Latrine must be dug | "Toughness" norm | 4. Brown hands shovel to Simpson; all cooperate in digging latrine |
| 5. Discovery of the dam | | 5. Mills organizes dam climbing game |
| 6. Canteen list required | | 6. Mills writes up canteen list |
| 7. Discovery of paper cups | | 7. Resentment that "outsiders" had been there |
| 8. Tent pitching practice | | 8. |

Analysis of Robbers' Cave Experiment (continued)

<u>Rattlers</u>

| | |
|---|---|
| 9. Group treasure hunt | 9. <u>Mills</u> proposes hardball equipment |
| 10. Baseball practice | 10. <u>Mills</u> nominates <u>Simpson</u> for Captain (elected) and chooses own position; <u>Mills</u> the recognized leader |
| 11. "Discovery" of the Eagles | 11. a) <u>Mills</u> chooses "Rattlers" name for group<br>b) Challenge of the Eagles to play ball game<br>c) Enthusiasm in possible competitive activities<br>d) Non-swimmers helped to swim - increase in solidarity and build-up of morale |

# ANALYSIS OF ROBBERS' CAVE EXPERIMENT

## EAGLES

### Stage 1:

A. System objects:

1. Goal-oriented interaction → hierarchical structure
2. Structure → In-group identity
3. In-group identity → norms
4. Norms → sanctions

B. System inputs

1. Campfire
2. Sign left by earlier campers
3. Canoe carrying
4. Bridge building
5. Screen requirement
6. Softball workout

C. System outputs

1. a) Myers leads in building fire
   b) Craig stops Myers from "bossing"
2. Myers decides name of camp
3. Craig leads in canoe carrying
4. a) Mason leads bridge building
   b) Cutler walks bridge, to the surprise of all - (emerging status relationships)
5. a) Craig says screens unnecessary; group disagrees - votes for them
   b) Craig leads in putting up screens
6. a) Mason the best ballplayer
   b) Craig and Davis scold Myers for clowning

Analysis of Robbers' Cave Experiment (continued)

<u>Eagles</u>

| | |
|---|---|
| 7. Campout requested by boys (reservoir trip) | 7. a) <u>Myers</u> astounds all by carrying 3 tents<br>b) <u>Mason</u> says reservoir not good for swimming; <u>Davis</u> supports <u>Mason's</u> opinion<br>c) <u>Craig</u> directs transportation of supplies back home<br>d) <u>Myers</u> pelted with stones for leaving swimsuit and holding up gang<br>e) <u>Craig's</u> song called "our song". |
| 8. Canteen list | 8. <u>Craig</u> and <u>Davis</u> take lead in drawing up canteen list |
| 9. Treasure hunt | 9. <u>Craig</u> given notes to read; <u>Craig</u> backs <u>Davis'</u> proposal for hardball equipment and instructs boys to sign petition for it. <u>Craig</u> lectures on how to play baseball |
| 10. Baseball practice | 10. <u>Craig</u> assigns each boy a number; <u>Myers</u> ignored when he objects to a decision; <u>Craig</u> not ignored when he objects to a decision, and gives in |
| 11. Discovery of the Rattlers | 11. <u>Craig</u> instructs staff to challenge Rattlers to a game |
| Homesickness of Davis and Boyd | <u>Davis</u> drops in status; <u>Davis</u> and <u>Boyd</u> leave for home |
| Practice at wrestling, tumbling, tent pitching, baseball | Enthusiasm in practice<br><u>Craig's</u> suggestion of name "Eagles" supported<br><u>Craig</u> requires <u>Mason</u> to accept stencil on shirt<br><u>Craig</u> tells staff the group has decided to sleep out in a tent |

## ANALYSIS OF ROBBERS' CAVE EXPERIMENT

(Continued)

**Stage 2:**

A. System objects:

1 - 4 as in Stage 1.

5. Territorial dispute → inter-group hostility

6. Intergroup hostility → challenge to competition

7. Challenge to competition → intergroup interaction

8. Intergroup interaction → frustration

9. Frustration → unfavorable stereotype (of out-group)

10. Frustration/success → in-group solidarity

11. Frustration → change in group structure (+ norms, sanctions)

12. Frustration → reprisals

B. System inputs

12. Mutual challenge to play baseball

13. Day 1: Tournament announced

C. System outputs

12. Enthusiasm in both groups

13. a) Rattler response: Full confidence in victory; spend time in improving "their own" ball field; put Rattler flag on backstop; practice for events

    b) Eagle response: Cautious enthusiasm; mild expression of confidence; attach Eagle flag to a pole; practice for events

<u>Stage 2</u> (Continued)

| | |
|---|---|
| 14. Day 2: First baseball game | 14. a) Rattlers show <u>possession</u> of ballfield - they were first to arrive (i.e., as "home team") <br> b) Eagles arrive with flag and menacing song <br> c) Mutual derogatory name calling, increasing as the game proceeds |
| 15. | 15. d) Craig (E) tries to control Myers (E) with words about sportsmanship <br> e) Mason (E) begins to displace Craig (E); chooses own pitching replacement; Craig accepts <br> f) Rattlers win; and display "good sportsmanship toward Eagles; Mason threatens his team if they do not try harder; Craig (E) throws Rattler glove into the water |
| 16. Day 2: Lunch together | 16. a) Mutual name calling and razing <br> b) Meyers (E), in saying grace, suggests Eagles' fear that others (like Davis and Boyd) might get homesick and go home |
| 17. Day 2: Practice for other events of tournament | 17. a) Rattlers: show stronger in-group feeling (e.g.,deciding to wear their stenciled shirts at every game) <br> b) Eagles: Mason takes over leadership of group (lectures on how to win; directs practice for various events); controls group by threatening to go home |

Stage 2 (Continued)

| | |
|---|---|
| 18. Day 2: First tug-of-war | 18. a) Mason (E) assumes job of captain for tug-of-war; Simpson (R) assumes same job for Rattlers<br>b) Craig (E) withdraws when he perceives Eagles are losing<br>c) Rattlers win; show sportsmanship (3 cheers for the Eagles)<br>d) Eagles' morale drops (Mason cries, says Rattlers are at least 8th graders, that he is going home, that he will <u>fight</u>)<br>e) Eagles burn the Rattlers' flag |
| 19. Day 3: Rattlers' discovery of their burnt flag | 19. a) Rattlers' response: noise and resentment; plan to fight the Eagles<br>b) Rattlers and Eagles skirmish; Craig (E) admits that <u>all</u> the Eagles burned the flag |
| 20. Day 3: Second ball game | 20. a) Eagles win the game, with cheers for the losers; ascribe their victory to prayers; ascribe Rattlers' loss to their "bad cussing"; decide to avoid cussing, and <u>not even to talk</u> to Rattlers who are poor sports and "bad cussers".<br>b) Rattlers response to loss of game: much internal friction, mutual recrimination; Brown (R) and Allen (R) write letters to ask to go home; |

Stage 2 (Continued)

| | |
|---|---|
| | Mills, the Rattler leader patches things up in group |
| 21. Day 3: Second tug-of-war | 21. a) Eagles employ strategy which brings contest to a tie; Rattlers are fatigues more than Eagles (who sat down and dug-in)<br>b) Eagles considered the contest short, the Rattlers considered it long (p. 115) |
| 22. Day 3: Rattlers raid Eagles' cabin | 22. a) Eagles response: First astonishment, then anger; Craig pretends to sleep through raid<br>b) Rattlers response: heroic and jubilant self-expression |
| 23. Day 4: Eagles raid Rattlers' cabin | 23. a) Rattlers furious at mess in their cabin<br>b) Eagles prepare to meet retaliation |
| 24. Day 4: Touch football game | 24. a) Rattlers win game by narrow margin, and win tent-pitching; morale goes up<br>b) Eagles do not feel too bad at losing by narrow margin; Craig walks away from contest |
| 25. Day 4: Third Baseball game | 25. a) Eagles win; morale goes high; refrain from bragging in presence of Rattlers<br>b) Rattlers explain loss because of the heaviness of the bats they used; show increase in |

<u>Stage 2</u> (Continued)

| | |
|---|---|
| | in-group solidarity at breakfast next morning (plan to put flags on their property in the camp) |
| 26. Day 5: Remaining contests | 26. a) Rattlers win third tug-of-war<br>b) Eagles win second tent-pitching<br>c) Eagles win third tent-pitching |
| 27. | 27. d) Rattlers decide to raid Eagles cabin if they (Rattlers) <u>win</u>, but not if they lose<br>e) Eagles win the treasure hunt |
| 28. | 28. f) Rattlers raid Eagles' cabin and take away prizes won by Eagles<br>g) Mason (E) wants to fight; Craig (E), Bryan (E) and McGraw (E) return to cabin<br>h) Mason (E) calls Craig and Bryan "yellow"<br>i) Fight between groups breaks out and is stopped by counselors<br>j) Neither group wants further association with the other; negative attitudes and social distance <u>standardized</u> in each group in relation to the other |
| 29. Day 6: "Test situation" (p.113); Rattlers explore Eagle territory | 29. Display of persisting negative attitudes by both groups |

Stage 2 (Continued)

| | | |
|---|---|---|
| Day 7: | Stereotype ratings (pp. 137-138) | Tendency (pp. 137-138) to rate in-group favorably and out-group unfavorably; the first tendency more pronounced than the second |
| Day 7: | Performance estimates (bean toss) | Performance of in-group judged significantly higher than that of out-group<br><br>Rattlers (the losers of the tournament) overestimated their own performance less than the Eagles overestimated theirs<br><br>Rattlers underestimated Eagles' performance, while Eagles overestimated Rattlers' performance (p. 146)<br><br>For both groups, performance of in-group members judged significantly higher than that of out-group members<br><br>Results reflect in-group solidarity and negative attitudes toward out-group |

## ANALYSIS OF ROBBERS' CAVE EXPERIMENT
(Continued)

Stage 3:

A. System objects:

1 - 4 as in Stage 1

5. Frustration → intergroup friction

6. Intergroup friction → resistance to cooperation

7. Superordinate goals → cooperation

8. Cooperation → decrease of friction

B. System inputs

30. Seven contact situations

31. Superordinate Goal (SG) 1: The drinking water problem

C. System outputs

30. Intergroup friction remains: Name-calling, accusations, insults, food-throwing fight

31. a) Boys choose to search in small segregated teams - Eagles going with Eagle staff members, Rattlers going with Rattler staff members

    b) Cooperation and communication between groups while trying to extract sack from faucet

    1) Several members of each group take turns

    2) Craig (E) gives advice to both E's and R's

Stage 3 (Continued)

| | |
|---|---|
| | 3) Everett (R) asks Eagle participant observer to try it; staff completes the job |
| | c) Common rejoicing; Rattlers (who had canteens) let Eagles (who did not have canteens) drink first, with no protest or name calling; some intermingling afterwards |
| | d) Negative attitudes persist after this SG: insults and food throwing at supper; lack of enthusiasm for idea of a trip together to Cedar Lake |
| 32. SG 2: The problem of securing a movie | 32. a) Groups decide without argument on choice of movie |
| | b) Mills (R leader) and Myers (E) lead in reaching a compromise solution for allocating costs between the groups |
| | c) Mason (E leader) does not participate in solution |
| | d) McGraw (E) and Martin (R) rend list of contributors |
| | e) Negative attitudes weaken: some scuffling among a few; one near-fight between Simpson (R) and Mason (E); seating at movie along group lines, _with some exceptions_; groups reach agreement on taking turns at entering dining room first for meals |

Stage 3 (Continued)

| | |
|---|---|
| 33. SG 3: Camp-out at Cedar Lake | 33. a) Mason (E leader) does not wish to go to Lake with Rattlers; the Rattlers had raised some objections previous day |
| | b) The two groups swim at same spot, but interact very little |
| 34. SG 4: The truck stalls | 34. a) Mills (R leader) suggests a "tug-of-war" against the stalled truck |
| | b) Clarke (E) and Mills (R) attach rope to bumper |
| | c) Harrison (R) suggests each group pull on a separate piece of rope |
| | d) Barton (R) says it doesn't make any difference who pulls where |
| | e) Groups pull on separate pieces, except that Swift (big R) joins Eagles as anchor man, and Craig (E) is next to Brown (R anchor man) |
| 35. | 35. f) Following successful starting of truck, there is much intermingling and friendly talk between groups; Mills (R), Hill (R), Craig (E) and Bryan (E) pump water for each other |
| 36. SG 5: Meal Preparation | 36. a) Rattlers prefer to alternate meal preparation with Eagles - at Mills (R) suggestion |

<u>Stage 3</u> (Continued)

b) Eagles are split on issue:

   1) Low status members (Clarke, Cuttler, Lane) favor alternating with Rattlers

   2) High status members support Mason's (E leader) opposition to alternating

   3) In actual fact, while the two groups discussed their preferences about alternating, <u>food preparation together</u> began. (This was a result, in part, of having bulk food which had to be divided)

c) Negative attitudes decrease:

   1) No one scolded Myers (E) for salting the Kool Aid by mistake; Harrison (R) even exonerated him

   2) Low status members on each side particularly active in preparing and distributing food

   3) Eating takes place in partly integrated fashion

   4) At swim following this (p.173), the groups are brought closer together by seeing a water moccasin; groups mix together in water

Stage 3 (Continued)

37. SG 6: Tent pitching

37. a) Cooperation in sorting out tent parts

b) Lack of competition - Rattlers lacked a mallet and Eagles had uneven site

c) No negative attitudes in evidence

38. SG 7: The truck stalls again

38. a) Rattlers initiate pushing truck, which gets rolled into a tree

b) Mills (R leader) gets tug-of-war rope; groups mix in pullin on same lines

c) Cooperative pattern established

1) in tug-of-war against the truck - groups mix

2) in meal preparation - Rattlers and Eagles work side by side

3) All eat together, with no group lines evident

4) Engage in friendly water fight, but not along group lines (p. 175)

39. SG 8: The trip to the border

39. a) McGraw (E) suggests that the Rattlers go to Arkansas first in their truck, and that Eagles go when they have returned

b) Mason (E leader) wants to return instead to Robbers Cave; Craig (E) supports him

<u>Stage 3</u> (Continued)

c) Mills (R) proposes they all go together in Rattler truck; Allen (low status R) and several Eagles support him; Simpson (R) says "let's don't".

d) Mills (R) and Clarke (E) settle the matter by leading a rush to get into the Rattler truck

e) Negative attitudes give way to cooperation, group singing; both groups instruct Myers (E) and Martin (R), who volunteered to draw up ice-cream lists, to <u>do it together</u>; they generally approve plan to ride bus together back to Oklahoma City (with Harrison (R) dissenting; and Mills (R) supporting it after he saw the rest wanted it)

| | |
|---|---|
| 40. SG 9: Last evening in camp | 40. a) Groups enter mess hall separately (after discussion, the Eagles went first by agreement), and get food separately<br>b) Table arrangement made it difficult to sit by groups - they mixed with remarking on it<br>c) They agree (under Mill's (R) influence) to joint campfire at Stone Corral (part of R's territory) |

Stage 3 (Continued)

| | |
|---|---|
| | d) They take turns at entertaining one another all evening |
| | e) No evidence of negative attitudes remains |
| 41. SG 10: The trip home | 41. a) They decide to ride bus together |
| | b) Seating arrangement does not follow group lines, as they go together on the bus |
| | c) Mills (R) suggest R's \$5.00 prize be spent to buy malts for all boys, both R's and E's. Rattlers approve |
| | d) Negative attitudes gone! (Note: on pp. 185-196 are results of tests showing attitude changes under conditions of Stage 3). |

# APPENDIX B

This Appendix contains the FORTRAN listings of the Adaptive Decoder used in Phase I and the simulation program used in Phase II of the Robbers' Cave example.

```
C       RATTLERS AND EAGLES; SIMULATED CONFLICT SITUATION
C       #1   E - EAGLES (REG)    R - RATTLER (REG)
C       F. J. KERN

        DIMENSION   W1(35,17),W2(33,9),W3(33,9)
                    W4(27,17),KMO(4),KSTR(4),KATT(4)
                    KFRU(2),X(35),Y(17),Z(17),
                    VAL(5),V1(17,5),V2(9,3),V3(9,3),V4(17,5)

        KCLA=0
        DO1L =1,17
1       READ2,(W1(I,L),I=1,35)
2       FORMAT(9F7.1)
        READ1001,((V1(L,M),L=1,17), M=1,5)
1001    FORMAT(15F4.1)
        DO3L-1,9
3       READ2,(W2(I,L),I=1,33)
        READ2001,((V2(L,M),L-1,9),M-1,3)
2001    FORMAT(15F4.1)
        DO4L=1,9
4       READ2,(W3(I,L),I-1,33)
        READ3001,((V3(L,M),L=1,9),M=1,3)
3001    FORMAT(15F4.1)
        DO5L=1,17
5       READ2,(W4(I,L),I=1,27)
        READ4001,((V4(L,M),L=1,17),M-1,5)
4001    FORMAT(15F4.1)
        READ6,(KMO(J1),J1-1,3),(KSTR(L1),L1=1,3)
               (KATT(M1),M1=1,3),KFRU(1)
6       FORMAT(13I2)
        K=1
8       READ650,I1,I2,I3,I4
650     FORMAT(4I2)
        IF(I1-KCLA)10,9,10
9       KCLKO=KCLKO+1
        GO TO 11
10      KCLKO=1
11      KCLA=I1
        DO12I=1,11
        IF(11-I-KCLKO)14,14,13
13      X(I)=-1.0
        GO TO 12
```

```
14        X(I)=+1.0
12        CONTINUE
          DO15I=12,15
          IF(12-I+I1)16,16,17
16        X(I)=-1.0
          GO TO 15
17        X(I)=+1.0
15        CONTINUE
          DO18I=16,19
          IF(16-I+I2)19,19,20
19        X(I)=-1.0
          GO TO 18
20        X(I)=+1.0
18        CONTINUE
          DO21I=20,22
          IF(20-I+I3)22,22,23
22        X(I)=-1.0
          GO TO 21
23        X(I)=+1.0
21        CONTINUE
          DO24I=23,27
          IF(23-I+KFRU(1))25,25,26
25        X(I)=-1.0
          GO TO 24
26        X(I)=+1.0
24        CONTINUE
          KTOT=KATT(1)+KATT(2)+KATT(3)
          TOT=FLOATF(KTOT)
          IF(TOT/3.0-1.5)27,27,28
27        XTOT=+1.0
          GO TO 35
28        IF(TOT/3.0-2.5)29,29,30
29        XTOT=+2.0
          GO TO 35
30        IF(TOT/3.0-3.5)31,31,32
31        XTOT=+3.0
          GO TO 35
32        IF(TOT/3.0-4.5)33,33,34
33        XTOT=+4.0
          GO TO 35
34        XTOT=+5.0
35        ITOT=XFIXF(XTOT)
          DO36I=28,32
          IF(28-I+ITOT)37,37,38
37        X(I)=-1.0
          GO TO 36
38        X(I)=+1.0
36        CONTINUE
          IF(I4-1)39,40,40
39        X(33)=-1.0
          X(34)=-1.0
          GO TO 41
```

```
40      X(33)=+1.0
        X(34)=+1.0
41      X(35)=+1.0
        PRINT42,K
42      FORMAT(41X,19H ATTITUDE SITUATION,1X,I2)
        PRINT43,(X(I),I=1,35)
43      FORMAT(1X,35F 3.0)
        DO501L=1,17
        Y(L)=0.0
        DO501I=1,35                                  KATT(4)
        SUM=X(I)*W1(I,L)
501     Y(L)=Y(L)+SUM
        DO502L=1,17
        IF(Y(L))504,503,503
503     Z(L)=+1.0
        GO TO 502
504     Z(L)=-1.0
502     CONTINUE
        DO505M=1,5
        VAL(M)=0.0
        DO505L=1,17
        G=Z(L)*V1(L,M)
505     VAL(M)=VAL(M)+G
        DO506M=1,5
        IF(M-1)509,509,508
508     IF(VAL(M)-AA)506,506,509
509     AA=VAL(M)
        KATT(4)=M
506     CONTINUE
        PRINT44,KATT(4),ITOT
44      FORMAT(5X,20H CLASSIFIED AS LEVEL,1X,I2,10X,I2)
        KTOT=KSTR(1)+KSTR(2)+KSTR(3)
        TOT=FLOATF(KTOT)
        IF(TOT/3.0-1.5)45,45,46
45      XTOT=+1.0
        GO TO 49
46      IF(TOT/3.0-2.5)47,47,48
47      XTOT=+2.0
        GO TO 49
48      XTOT=3.0
49      ITOT=XF1XF(XTOT)
        DO50I=28,30
        IF(28-I+ITOT)51,51,52
51      X(I)=-1.0
        GO TO 50
52      X(1)=+1.0
50      CONTINUE
        IF(I4-1)53,54,54
53      X(31)=-1.0
        X(32)=-1.0
        GO TO 55
```

```
54      X(31)=+1.0
        X(32)=+1.0
55      X(33)=+1.0
        PRINT56,K
56      FORMAT(39X,21H STRUCTURE SITUATION,1X,I2)
        DO601L=1,9
        Y(L)=0.0
        DO601I=1,33
        SUM=X(I)*W2(I,L)
601     Y(L)=Y(L)+SUM
        DO602L=1,9
        IF(Y(L))604,603,603
603     Z(L)=+1.0
        GO TO 602
604     Z(L)=-1.0
602     CONTINUE
        DO605M=1,3
        VAL(M)=0.0
        DO605L=1,9
        G=Z(L)*V2(L,M)
605     VAL(M)=VAL(M)+G
        DO606M=1,3
        IF(M-1)609,609,608
608     IF(VAL(M)-AA)606,606,609
609     AA=VAL(M)
        KSTR(4)=M
606     CONTINUE
        PRINT57,(X(I),I=1,33)
57      FORMAT(LX,33F3.0)
        PRINT58,KSTR(4),ITOT
58      FORMAT(5X,20H CLASSIFIED AS LEVEL, LX,I2,10X,I21)
        KTOT=KMO(1)+KMO(2)+KMO(3)
        TOT=FLOATF(KTOT)
        IF(TOT/3.0-1.5)59,59,60
59      XTOT=+1.0
        GO TO 63
60      IF(TOT/3.0-2.5)61,61,62
61      XTOT=+2.0
        GO TO 63
62      XTOT=3.0
63      ITOT=XF1XF(XTOT)
        DO64I=28,30
        IF(28-I+ITOT)65,65,66
65      X(I)=-1.0
        GO TO 64
66      X(I)=+1.0
64      CONTINUE
        PRINT67,K
67      FORMAT(40X,17H MORALE SITUATION,1X,I2)
        PRINT68,(X(I),I=1,33)
```

```
68      FORMAT(1X,33F3.0)
        DO701L=1,9
        Y(L)=0.0
        DO701I=1,33
        SUM=X(I)*W3(I,L)
701     Y(L)=Y(L)+SUM
        DO702L=1,9
        IF(Y(L)704,703,703
703     Z(L)=+1.0
        GO TO 702
704     Z(L)=-1.0
702     CONTINUE
        DO705M=1,3
        VAL(M)=0.0
        DO705L=1,9
        G=Z(L)*V3(L,M)
705     VAL(M)=VAL(M)+G
        DO706M=1,3
        IF(M-1)709,709,708
708     IF(VAL(M)-AA)706,706,709
709     AA=VAL(M)
        KMO(4)=M
706     CONTINUE
        PRINT69,KMO(4),ITOT
69      FORMAT(5X,20H CLASSIFIED AS LEVEL, 1X,I2,10X,I21)
        PRINT70,K
70      FORMAT(28X,22H FRUSTRATION SITUATION,1X,I2)
        PRINT71,(X(I),I=1,27)
71      FORMAT(1X,27F3.0)
        DO801L=1,17
        Y(L)=0.0
        DO801I=1,27
        SUM=X(I)*W4(I,L)
801     Y(L)=Y(L)+SUM
        DO802L=1,17
        IF(Y(L))804,803,803
803     Z(L)=+1.0
        GO TO 802
804     Z(L)=-1.0
802     CONTINUE
        DO805M=1,5
        VAL(M)=0.0
        DO805L=1,17
        G=Z(L)*V4(L,M)
805     VAL(M)=VAL(M)+G
        DO806M=1,5
        IF(M-1)809,809,808
808     IF(VAL(M)-AA)806,806,809
809     AA=VAL(M)
        KFRU(2)=M
806     CONTINUE
        PRINT72,KFRU(2),KFRU(1)
```

```
72    FORMAT(5X,20H CLASSIFIED AS LEVEL,1X,I2,10X,I2)
      IF(K-41)73,73,74
73    STOP
74    K=K+1
      KFRU(1)=KFRU(2)
      DO75I=1,3
      KMO(I)=KMO(I+1)
      KATT(I)=KATT(I+1)
75    KSTR(I)=KSTR(I+1)
      GO TO 8
      END
```

```
C       GENERALIZED ADAPTIVE DECODER

C       F. J. KERN

        DIMENSIONX(35,41),W(35,17),Y(17),NUT(41),Z(17),
        V(17,5),VAL(5),NDD(17),SUM(17), MARK(17),NRD(17),B(17),DUM(10)

        READ1,N1,N2,N3,N4,DELTA,D2,ALPHA
1       FORMAT(4I3,3F5.2)
        READ2,((X(I,J),I=1,N1),J-1,N2)
        READ2,((V(L,M),L=1,N3),M-1,N4)
2       FORMAT(15F4.1)
        READ100,(NUT(J),J=1,N2)
100     FORMAT(22I2)
        INDEX=1
        NUM=1
        MISS=0
        MSE=0
        DO3L=1,N3
        DO3I=1,N1
3       W(I,L)=1.0
4       PRINT51
51      FORMAT(48X,10H CODE WORD, 48X,6H CLASS,2X,5H REAL)
        DO16J=1,N2
        DO5L=1,N3
        Y(L)=0.0
        DO5I=1,N1
        F=X(I,J)*W(I,L)
5       Y(L)=Y(L)+F
        DO7L-1,N3
        IF(Y(L))9,8,8
8       Z(L)=1.0
        GO TO 7
9       Z(L)=-1.0
7       CONTINUE
        DO10M=1,N4
        VAL(M)=0.0
        DO10L=1,N3
        G=Z(L)*V(L,M)
10      VAL(M)=VAL(M)+G
        DO12M=1,N4
        IF(M-1)14,14,13
13      IF(VAL(M)-AA)12,12,14
14      AA=VAL(M)
        KLASS=M
```

```
12      CONTINUE
        KMAX=(35-N1)
        IF(KMAX)200,200,201
201     DO202K=1,KMAX
202     DUM(K)=0.0
        PRINT15,(X(I,J),I=1,N1),(DUM(K),K=1,KMAX),KLASS,NUT(J)
        GOTO16
200     PRINT15,(X(I,J),I=1,N1),KLASS,NUT(J)
15      FORMAT(1X,35F3.0,3X,I2,5X,I2)
16      CONTINUE
        IF(INDEX-500)53,52,52
52      DELTA=D2
53      IF(INDEX-1000)55,54,54
54      GOTO44
55      J=NUM
17      DO18L=1,N3
        Y(L)=0.0
        DO18I=1,N1
        F=X(I,J)*W(I,L)
18      Y(L)=Y(L)+F
        DO20L=1,N3
        IF(Y(L))22,21,21
21      Z(L)=1.0
        GO TO 20
22      Z(L)=-1.0
20      CONTINUE
        DO24M=1,N4
        VAL(M)=0.0
        DO24L=1,N3
        G=Z(L)*V(L,M)
24      VAL(M)=VAL(M)+G
        DO25M=1,N4
        IF(M-1)27,27,26
26      IF(VAL(M)-AA)25,25,27
27      AA=VAL(M)
        KLASS=M
25      CONTINUE
        IF(KLASS-NUT(J))56,40,56
56      MISS=MISS+XABSF(KLASS-NUT(J))
        MSE=MSE+(KLASS-NUT(J))*(KLASS-NUT(J))
        N=1
        M=NUT(J)
        DO28L=1,N3
        IF(V(L,M)+Z(L))28,29,28
29      NDD(N)=L
        SUM(N)-ABSF(Y(L))
        N=N+1
28      CONTINUE
        LIST=N-1
        FILE=FLOATF(N-1)
        M=1
        IF(LIST-1)35,35,34
```

```
34      B(M)=SUM(1)
        MARK(M)=1
        NRD(M)=NDD(1)
        DO30N=2,LIST
        IF(B(M)-SUM(N))30,30,31
31      B(M)=SUM(N)
        MARK(M)=N
        NRD(M)=NDD(N)
30      CONTINUE
        DO32N=1,LIST
        IF(N-MARK(M))32,32,33
33      SUM(N-1)=SUM(N)
        NDD(N-1)=NDD(N)
32      CONTINUE
        M=M+1
        LIST=LIST-1
        IF(LIST-1)35,35,34
35      MARK(M)=1
        B(M)=SUM(1)
        NRD(M)=NDD(1)
        M=1
36      L=NRD(M)
        TOT=FLOATF(N1)
        FUN=(B(M)+ALPHA)/TOT
        ENJK=1.0
37      IF(ENJK-FUN)59,38,38
59      ENJK=ENJK+1.0
        GO TO 37
38      DO39I=1,N1
39      W(I,L)=W(I,L)-ENJK*Z(L)*X(I,J)
        RANGE=FLOATF(M)
        IF(RANGE-DELTA*FILE)57,40,40
57      M=M+1
        GO TO 36
40      J=XMOD(J,N2)
        J=J+1
        IF(XMOD(INDEX,10))43,41,43
43      INDEX=INDEX+1
        GO TO 17
41      PRINT42,INDEX,MISS,MSE
42      FORMAT(2X,9HWORD NO.,I4,5X,24H ABS ERROR LAST 10 WORDS,I4,23H
       SQ ERROR LAST 10 WORDS,I4)
        MISS=0
        MSE=0
        IF(XMOD(INDEX,500))43,50,43
50      NUM=J
        INDEX=INDEX+1
        GO TO 4
44      Do45L=1,N3
        PRINT46,L
```

```
46      FORMAT(10X,12H TLU. NUMBER,I4)
        PRINT47,(W(I,L),I=1,N1)
47      FORMAT(5F7.1)
        PUNCH48,(W(I,L),I=1,N1)
48      FORMAT(9F7.1)
45      CONTINUE
        STOP
        END
```

## APPENDIX C

This section contains the final weights of all Threshold Logic units used in the Robbers Cave Simulation.

## WEIGHTS FOR RATTLERS ATTITUDE

### ERRORS

| Word No. | Classed As | Should Be |
|---|---|---|
| 12 | 3 | 2 |
| 19 | 3 | 2 |
| 25 | 3 | 2 |
| 32 | 3 | 2 |

TLU. NUMBER 1

| | | | | |
|---|---|---|---|---|
| 1.0 | 1.0 | 1.0 | 1.0 | 1.0 |
| 1.0 | 1.0 | 7.0 | 3.0 | 5.0 |
| -1.0 | -1.0 | -5.0 | -1.0 | 3.0 |
| -5.0 | -5.0 | 3.0 | 3.0 | -1.0 |
| -5.0 | -1.0 | -1.0 | -1.0 | -1.0 |
| 7.0 | 13.0 | -1.0 | -5.0 | -1.0 |
| 3.0 | 3.0 | 1.0 | 1.0 | -1.0 |

TLU. NUMBER 2

| | | | | |
|---|---|---|---|---|
| .0 | .0 | .0 | .0 | .0 |
| .0 | .0 | 2.0 | 2.0 | 6.0 |
| .0 | .0 | -4.0 | -2.0 | 2.0 |
| -6.0 | -6.0 | 2.0 | 2.0 | .0 |
| -10.0 | -6.0 | .0 | .0 | .0 |
| 10.0 | 10.0 | .0 | -4.0 | -2.0 |
| 2.0 | 2.0 | 2.0 | 2.0 | .0 |

TLU. NUMBER 3

| | | | | |
|---|---|---|---|---|
| .0 | .0 | .0 | .0 | .0 |
| .0 | 2.0 | 8.0 | 4.0 | 8.0 |
| .0 | .0 | -6.0 | -2.0 | 2.0 |
| -6.0 | -6.0 | .0 | 2.0 | .0 |
| -6.0 | -2.0 | .0 | .0 | .0 |
| 8.0 | 16.0 | .0 | -8.0 | -4.0 |
| 2.0 | 2.0 | 2.0 | 2.0 | .0 |

TLU. NUMBER 4

| | | | | |
|---|---|---|---|---|
| .0 | .0 | .0 | .0 | .0 |
| .0 | .0 | 4.0 | 2.0 | 6.0 |
| .0 | .0 | -6.0 | -2.0 | 2.0 |
| -4.0 | -4.0 | 4.0 | 2.0 | .0 |
| -4.0 | -4.0 | .0 | .0 | .0 |
| 6.0 | 8.0 | .0 | -4.0 | -4.0 |
| 2.0 | 2.0 | 2.0 | 2.0 | .0 |

TLU. NUMBER 5

| | | | | |
|---|---|---|---|---|
| -2.0 | -2.0 | -2.0 | -2.0 | -2.0 |
| -2.0 | .0 | 10.0 | -10.0 | 4.0 |
| .0 | .0 | -4.0 | 8.0 | 2.0 |
| -2.0 | -4.0 | 6.0 | 2.0 | .0 |
| -2.0 | 18.0 | .0 | .0 | -4.0 |
| 6.0 | .0 | .0 | .0 | -4.0 |
| 2.0 | 2.0 | -8.0 | -8.0 | .0 |

TLU. NUMBER 6

| | | | | |
|---|---|---|---|---|
| 4.0 | 4.0 | 4.0 | 4.0 | 4.0 |
| 4.0 | 2.0 | -6.0 | 6.0 | 2.0 |
| .0 | .0 | -2.0 | -4.0 | 2.0 |
| -4.0 | -2.0 | -6.0 | 2.0 | .0 |
| -4.0 | -10.0 | .0 | .0 | 12.0 |
| .0 | 18.0 | .0 | -4.0 | 4.0 |
| 2.0 | 2.0 | 6.0 | 6.0 | .0 |

TLU. NUMBER 7

| | | | | |
|---|---|---|---|---|
| 4.0 | 2.0 | 2.0 | .0 | .0 |
| .0 | -2.0 | -4.0 | 4.0 | 2.0 |
| .0 | .0 | 6.0 | .0 | 2.0 |
| -6.0 | -4.0 | -6.0 | 2.0 | .0 |
| -4.0 | -6.0 | .0 | 2.0 | 10.0 |
| 12.0 | 12.0 | .0 | .0 | .0 |
| .0 | 2.0 | 6.0 | 6.0 | .0 |

TLU. NUMBER 8

| | | | | |
|---|---|---|---|---|
| 1.0 | 1.0 | 1.0 | -1.0 | -1.0 |
| -1.0 | -1.0 | -9.0 | 17.0 | 3.0 |
| 1.0 | 1.0 | 11.0 | -5.0 | 1.0 |
| -3.0 | -3.0 | -9.0 | 1.0 | 1.0 |
| -9.0 | -11.0 | 1.0 | 1.0 | 9.0 |
| 7.0 | 13.0 | 1.0 | -1.0 | 5.0 |
| 1.0 | 1.0 | 5.0 | 5.0 | 1.0 |

Weights for Rattlers Attitude (Continued)

TLU. NUMBER 9

| | | | | |
|---|---|---|---|---|
| .0 | .0 | .0 | -2.0 | -2.0 |
| -2.0 | -2.0 | -6.0 | 6.0 | 4.0 |
| 2.0 | 2.0 | .0 | -4.0 | .0 |
| .0 | 4.0 | -6.0 | .0 | 2.0 |
| -2.0 | -12.0 | 2.0 | 2.0 | .0 |
| 2.0 | 10.0 | 2.0 | -4.0 | 2.0 |
| .0 | .0 | 10.0 | 10.0 | 2.0 |

TLU. NUMBER 10

| | | | | |
|---|---|---|---|---|
| .0 | .0 | .0 | .0 | .0 |
| .0 | .0 | .0 | -4.0 | .0 |
| 2.0 | 2.0 | .0 | -6.0 | .0 |
| .0 | -2.0 | 2.0 | .0 | 2.0 |
| .0 | .0 | 2.0 | 2.0 | 10.0 |
| 2.0 | .0 | 2.0 | .0 | -2.0 |
| .0 | .0 | -4.0 | -4.0 | 2.0 |

TLU. NUMBER 11

| | | | | |
|---|---|---|---|---|
| .0 | .0 | .0 | .0 | .0 |
| .0 | .0 | .0 | -4.0 | .0 |
| 2.0 | 2.0 | .0 | -6.0 | .0 |
| .0 | -2.0 | 2.0 | .0 | 2.0 |
| .0 | .0 | 2.0 | 2.0 | 10.0 |
| 2.0 | .0 | 2.0 | .0 | -2.0 |
| .0 | .0 | -4.0 | -4.0 | 2.0 |

TLU. NUMBER 12

| | | | | |
|---|---|---|---|---|
| .0 | .0 | .0 | .0 | .0 |
| .0 | .0 | .0 | -4.0 | .0 |
| 2.0 | 2.0 | .0 | -6.0 | .0 |
| .0 | -2.0 | 2.0 | .0 | 2.0 |
| .0 | .0 | 2.0 | 2.0 | 10.0 |
| 2.0 | .0 | 2.0 | .0 | -2.0 |
| .0 | .0 | -4.0 | -4.0 | 2.0 |

TLU. NUMBER 13

| | | | | |
|---|---|---|---|---|
| .0 | .0 | .0 | .0 | .0 |
| .0 | .0 | .0 | .0 | 4.0 |
| 2.0 | 2.0 | .0 | 2.0 | .0 |
| 2.0 | 2.0 | 2.0 | .0 | 2.0 |
| 2.0 | 4.0 | 2.0 | .0 | 6.0 |
| .0 | .0 | 2.0 | 2.0 | .0 |
| 2.0 | .0 | -2.0 | -2.0 | 2.0 |

TLU. NUMBER 14

| | | | | |
|---|---|---|---|---|
| .0 | .0 | .0 | .0 | .0 |
| .0 | .0 | .0 | 4.0 | .0 |
| 2.0 | 2.0 | .0 | -2.0 | .0 |
| 2.0 | 2.0 | 2.0 | .0 | 2.0 |
| 2.0 | 4.0 | 2.0 | 6.0 | 4.0 |
| .0 | .0 | 2.0 | 2.0 | .0 |
| -4.0 | .0 | .0 | .0 | 2.0 |

TLU. NUMBER 15

| | | | | |
|---|---|---|---|---|
| .0 | .0 | .0 | .0 | .0 |
| .0 | .0 | .0 | 4.0 | .0 |
| 2.0 | 2.0 | .0 | -2.0 | .0 |
| 2.0 | 2.0 | 2.0 | .0 | 2.0 |
| 2.0 | 4.0 | 2.0 | 6.0 | .0 |
| .0 | .0 | 2.0 | 2.0 | .0 |
| -4.0 | .0 | .0 | .0 | 2.0 |

TLU. NUMBER 16

| | | | | |
|---|---|---|---|---|
| .0 | .0 | .0 | .0 | .0 |
| .0 | .0 | .0 | 2.0 | .0 |
| 2.0 | 2.0 | .0 | -2.0 | .0 |
| .0 | .0 | 2.0 | .0 | 2.0 |
| .0 | 2.0 | 2.0 | 6.0 | 4.0 |
| 2.0 | .0 | 2.0 | 2.0 | .0 |
| -4.0 | .0 | 2.0 | 2.0 | 2.0 |

TLU. NUMBER 17

| | | | | |
|---|---|---|---|---|
| .0 | .0 | .0 | .0 | .0 |
| .0 | .0 | .0 | .0 | .0 |
| 2.0 | 2.0 | .0 | .0 | .0 |
| 2.0 | 2.0 | 2.0 | .0 | 2.0 |
| 2.0 | .0 | 2.0 | 2.0 | 2.0 |
| .0 | .0 | 2.0 | 2.0 | 2.0 |
| .0 | .0 | 2.0 | 2.0 | 2.0 |

## WEIGHTS FOR RATTLERS STRUCTURE

### NO ERRORS

#### TLU. NUMBER 1

| | | | | |
|---|---|---|---|---|
| 3.0 | 3.0 | 3.0 | 3.0 | 3.0 |
| 1.0 | -13.0 | -5.0 | -3.0 | -5.0 |
| -1.0 | -1.0 | 1.0 | 3.0 | 3.0 |
| 1.0 | 3.0 | -1.0 | -1.0 | -1.0 |
| 1.0 | 3.0 | -1.0 | -1.0 | -1.0 |
| -5.0 | 1.0 | -1.0 | -7.0 | -7.0 |
| -3.0 | -3.0 | -1.0 | | |

#### TLU. NUMBER 2

| | | | | |
|---|---|---|---|---|
| 3.0 | 3.0 | 3.0 | 3.0 | 3.0 |
| 1.0 | -13.0 | -5.0 | -3.0 | -5.0 |
| -1.0 | -1.0 | 1.0 | 3.0 | 3.0 |
| 1.0 | -5.0 | -1.0 | 3.0 | -1.0 |
| 1.0 | 3.0 | -1.0 | -1.0 | -1.0 |
| -5.0 | 1.0 | -1.0 | -7.0 | -7.0 |
| -3.0 | -3.0 | -1.0 | | |

#### TLU. NUMBER 3

| | | | | |
|---|---|---|---|---|
| 2.0 | 2.0 | 2.0 | 2.0 | 2.0 |
| 2.0 | -8.0 | -2.0 | -2.0 | -4.0 |
| .0 | .0 | 2.0 | 2.0 | 2.0 |
| .0 | -4.0 | -2.0 | 2.0 | .0 |
| 2.0 | 2.0 | .0 | .0 | .0 |
| -4.0 | .0 | .0 | -4.0 | -4.0 |
| -2.0 | -2.0 | .0 | | |

#### TLU. NUMBER 4

| | | | | |
|---|---|---|---|---|
| 1.0 | 1.0 | 1.0 | 1.0 | 1.0 |
| -1.0 | -5.0 | -5.0 | -3.0 | -3.0 |
| 1.0 | 1.0 | 3.0 | 1.0 | 1.0 |
| 1.0 | -1.0 | -1.0 | 1.0 | 1.0 |
| 3.0 | 3.0 | 1.0 | 1.0 | 1.0 |
| -1.0 | 1.0 | 1.0 | -1.0 | -1.0 |
| -1.0 | -1.0 | 1.0 | | |

#### TLU. NUMBER 5

| | | | | |
|---|---|---|---|---|
| 1.0 | 1.0 | 1.0 | 1.0 | 1.0 |
| 1.0 | -5.0 | -1.0 | -1.0 | -1.0 |
| 1.0 | 1.0 | 1.0 | 1.0 | 1.0 |
| 1.0 | -3.0 | 1.0 | 1.0 | 1.0 |
| 1.0 | 1.0 | 1.0 | 1.0 | 1.0 |
| 1.0 | 1.0 | 1.0 | 1.0 | 1.0 |
| 1.0 | 1.0 | 1.0 | | |

#### TLU. NUMBER 6

| | | | | |
|---|---|---|---|---|
| 3.0 | -1.0 | -1.0 | -1.0 | -1.0 |
| -1.0 | 1.0 | -1.0 | -1.0 | -1.0 |
| -1.0 | -1.0 | 1.0 | -1.0 | 3.0 |
| -1.0 | -1.0 | 1.0 | 3.0 | -1.0 |
| -1.0 | -1.0 | -1.0 | -1.0 | -1.0 |
| -3.0 | -1.0 | -1.0 | -7.0 | -3.0 |
| -3.0 | -3.0 | -1.0 | | |

#### TLU. NUMBER 7

| | | | | |
|---|---|---|---|---|
| 3.0 | -1.0 | -1.0 | -1.0 | -1.0 |
| -1.0 | -1.0 | 1.0 | 1.0 | -1.0 |
| -1.0 | -1.0 | 1.0 | 3.0 | 3.0 |
| -1.0 | 1.0 | -1.0 | 3.0 | -1.0 |
| -1.0 | 1.0 | -1.0 | -1.0 | -1.0 |
| -3.0 | 1.0 | -1.0 | -9.0 | -5.0 |
| -1.0 | -1.0 | -1.0 | | |

#### TLU. NUMBER 8

| | | | | |
|---|---|---|---|---|
| 3.0 | -1.0 | -1.0 | -1.0 | -1.0 |
| -1.0 | -1.0 | -1.0 | -1.0 | -1.0 |
| -1.0 | -1.0 | 3.0 | 1.0 | 3.0 |
| 1.0 | -1.0 | -1.0 | 3.0 | -1.0 |
| -1.0 | 3.0 | -1.0 | -1.0 | -1.0 |
| -1.0 | 1.0 | -1.0 | -9.0 | -5.0 |
| -5.0 | -5.0 | -1.0 | | |

Weights for Rattlers Structure (Continued)

TLU. NUMBER 9

| | | | | |
|---|---|---|---|---|
| 3.0 | -1.0 | -1.0 | -1.0 | -1.0 |
| -1.0 | -1.0 | -1.0 | -1.0 | -1.0 |
| -1.0 | -1.0 | 3.0 | 1.0 | 3.0 |
| 1.0 | -1.0 | -1.0 | 3.0 | -1.0 |
| -1.0 | 3.0 | -1.0 | -1.0 | -1.0 |
| -1.0 | 1.0 | -1.0 | -9.0 | -5.0 |
| -5.0 | -5.0 | -1.0 | | |

WEIGHTS FOR RATTLERS MORALE

NO ERRORS

TLU. NUMBER 1

| | | | | |
|---|---|---|---|---|
| 2.0 | 2.0 | 2.0 | 2.0 | .0 |
| .0 | .0 | .0 | .0 | .0 |
| .0 | .0 | .0 | 2.0 | 2.0 |
| .0 | .0 | 2.0 | 2.0 | .0 |
| .0 | .0 | .0 | .0 | .0 |
| 2.0 | 2.0 | .0 | .0 | 2.0 |
| 2.0 | 2.0 | .0 | | |

TLU. NUMBER 2

| | | | | |
|---|---|---|---|---|
| 2.0 | 2.0 | 2.0 | 2.0 | .0 |
| .0 | .0 | .0 | .0 | .0 |
| .0 | .0 | .0 | 2.0 | 2.0 |
| .0 | .0 | 2.0 | 2.0 | .0 |
| .0 | .0 | .0 | .0 | .0 |
| 2.0 | 2.0 | .0 | .0 | 2.0 |
| 2.0 | 2.0 | .0 | | |

TLU. NUMBER 3

| | | | | |
|---|---|---|---|---|
| 2.0 | 2.0 | 2.0 | 2.0 | .0 |
| .0 | .0 | .0 | .0 | .0 |
| .0 | .0 | .0 | 2.0 | 2.0 |
| .0 | .0 | 2.0 | 2.0 | .0 |
| .0 | .0 | .0 | .0 | .0 |
| 2.0 | 2.0 | .0 | .0 | 2.0 |
| 2.0 | 2.0 | .0 | | |

TLU. NUMBER 4

| | | | | |
|---|---|---|---|---|
| 2.0 | 2.0 | 2.0 | 2.0 | .0 |
| .0 | .0 | .0 | .0 | .0 |
| .0 | .0 | .0 | 2.0 | 2.0 |
| .0 | .0 | 2.0 | 2.0 | .0 |
| .0 | .0 | .0 | .0 | .0 |
| 2.0 | 2.0 | .0 | .0 | 2.0 |
| 2.0 | 2.0 | .0 | | |

TLU. NUMBER 5

| | | | | |
|---|---|---|---|---|
| 1.0 | 1.0 | 1.0 | 1.0 | -1.0 |
| -1.0 | 7.0 | -5.0 | -1.0 | 17.0 |
| 1.0 | 1.0 | 5.0 | 5.0 | 1.0 |
| 3.0 | 5.0 | -3.0 | 1.0 | 1.0 |
| 15.0 | -1.0 | 1.0 | 1.0 | 1.0 |
| 3.0 | -11.0 | 1.0 | 5.0 | -5.0 |
| 9.0 | 9.0 | 1.0 | | |

TLU. NUMBER 6

| | | | | |
|---|---|---|---|---|
| -4.0 | -4.0 | -4.0 | -4.0 | .0 |
| .0 | -6.0 | 6.0 | 2.0 | -10.0 |
| 2.0 | 2.0 | .0 | -4.0 | .0 |
| -4.0 | -6.0 | 2.0 | .0 | 2.0 |
| -16.0 | .0 | 2.0 | 2.0 | 2.0 |
| 2.0 | 6.0 | 2.0 | -6.0 | 4.0 |
| -12.0 | -12.0 | 2.0 | | |

Weights for Rattlers Morale (Continued)

TLU. NUMBER 7

| | | | | |
|---|---|---|---|---|
| -2.0 | -2.0 | -2.0 | -2.0 | 2.0 |
| 2.0 | -4.0 | 2.0 | .0 | -8.0 |
| .0 | .0 | 4.0 | -2.0 | 2.0 |
| -8.0 | -2.0 | 8.0 | 2.0 | .0 |
| -12.0 | 4.0 | .0 | .0 | .0 |
| 4.0 | 16.0 | .0 | -6.0 | 2.0 |
| -6.0 | -6.0 | .0 | | |

TLU. NUMBER 8

| | | | | |
|---|---|---|---|---|
| .0 | .0 | .0 | .0 | 8.0 |
| 8.0 | -2.0 | 8.0 | 8.0 | -12.0 |
| 2.0 | 2.0 | 2.0 | -2.0 | .0 |
| 2.0 | 4.0 | 4.0 | .0 | 2.0 |
| -8.0 | 4.0 | 2.0 | 2.0 | 2.0 |
| -6.0 | 4.0 | 2.0 | -2.0 | 4.0 |
| -10.0 | -10.0 | 2.0 | | |

TLU. NUMBER 9

| | | | | |
|---|---|---|---|---|
| -3.0 | -3.0 | -3.0 | -3.0 | 1.0 |
| 1.0 | -5.0 | 5.0 | 3.0 | -13.0 |
| 3.0 | 3.0 | -5.0 | -5.0 | -1.0 |
| -3.0 | -3.0 | 1.0 | -1.0 | 3.0 |
| -5.0 | 3.0 | 3.0 | 3.0 | 3.0 |
| 1.0 | 13.0 | 3.0 | -1.0 | 7.0 |
| -7.0 | -7.0 | 3.0 | | |

## WEIGHTS FOR RATTLERS FRUSTRATION

ERRORS

| Word No. | Classed As | Should Be |
|---|---|---|
| 17 | 4 | 3 |
| 33 | 3 | 4 |

TLU. NUMBER 1

| | | | | |
|---|---|---|---|---|
| 1.0 | 1.0 | 1.0 | 1.0 | 1.0 |
| 1.0 | -1.0 | -1.0 | -5.0 | -5.0 |
| -1.0 | -1.0 | 3.0 | 1.0 | 3.0 |
| -1.0 | -1.0 | -3.0 | 3.0 | -1.0 |
| -1.0 | -5.0 | -1.0 | -5.0 | -11.0 |
| 1.0 | 1.0 | | | |

TLU. NUMBER 2

| | | | | |
|---|---|---|---|---|
| 1.0 | 1.0 | 1.0 | 1.0 | 1.0 |
| 1.0 | -1.0 | -1.0 | -5.0 | -5.0 |
| -1.0 | -1.0 | 3.0 | 1.0 | 3.0 |
| -1.0 | -1.0 | -3.0 | 3.0 | -1.0 |
| -1.0 | -5.0 | -1.0 | -5.0 | -11.0 |
| 1.0 | 1.0 | | | |

TLU. NUMBER 3

| | | | | |
|---|---|---|---|---|
| 1.0 | 1.0 | 1.0 | 1.0 | 1.0 |
| 1.0 | -1.0 | -1.0 | -5.0 | -5.0 |
| -1.0 | -1.0 | 3.0 | 1.0 | 3.0 |
| -1.0 | -1.0 | -3.0 | 3.0 | -1.0 |
| -1.0 | -5.0 | -1.0 | -5.0 | -11.0 |
| 1.0 | 1.0 | | | |

TLU. NUMBER 4

| | | | | |
|---|---|---|---|---|
| .0 | .0 | .0 | .0 | .0 |
| .0 | -2.0 | -2.0 | -2.0 | -2.0 |
| .0 | .0 | 2.0 | 2.0 | 2.0 |
| .0 | .0 | -2.0 | 2.0 | .0 |
| 2.0 | -4.0 | .0 | -4.0 | -10.0 |
| -2.0 | -2.0 | | | |

Weights for Rattlers Frustration (Continued)

TLU. NUMBER 5

| | | | | |
|---|---|---|---|---|
| -1.0 | -1.0 | 1.0 | 1.0 | 1.0 |
| 1.0 | 1.0 | 1.0 | 1.0 | -5.0 |
| 3.0 | 3.0 | 1.0 | -7.0 | -1.0 |
| 3.0 | 3.0 | -5.0 | -1.0 | 3.0 |
| 3.0 | -3.0 | 3.0 | 3.0 | -1.0 |
| 1.0 | 1.0 | | | |

TLU. NUMBER 6

| | | | | |
|---|---|---|---|---|
| 1.0 | 1.0 | 1.0 | 1.0 | 1.0 |
| 1.0 | -1.0 | -1.0 | 1.0 | 7.0 |
| -1.0 | -1.0 | 3.0 | 7.0 | 3.0 |
| -1.0 | -1.0 | 5.0 | 3.0 | -1.0 |
| 1.0 | -3.0 | -1.0 | -3.0 | -9.0 |
| -5.0 | 1.0 | | | |

TLU. NUMBER 7

| | | | | |
|---|---|---|---|---|
| 1.0 | 1.0 | -1.0 | -1.0 | -1.0 |
| -1.0 | -3.0 | -3.0 | -3.0 | 3.0 |
| -1.0 | -1.0 | 5.0 | 9.0 | 3.0 |
| -1.0 | -1.0 | 3.0 | 3.0 | -1.0 |
| 1.0 | -3.0 | -1.0 | -1.0 | -9.0 |
| -3.0 | 1.0 | | | |

TLU. NUMBER 8

| | | | | |
|---|---|---|---|---|
| 2.0 | 2.0 | .0 | .0 | .0 |
| .0 | .0 | -2.0 | -2.0 | 6.0 |
| .0 | .0 | 4.0 | 8.0 | 2.0 |
| .0 | 2.0 | 2.0 | 2.0 | .0 |
| .0 | -4.0 | .0 | .0 | -8.0 |
| -4.0 | .0 | | | |

TLU. NUMBER 9

| | | | | |
|---|---|---|---|---|
| -2.0 | -2.0 | -2.0 | -2.0 | -2.0 |
| -2.0 | -2.0 | -2.0 | -6.0 | 2.0 |
| 4.0 | 4.0 | 10.0 | 6.0 | -2.0 |
| -4.0 | 4.0 | 8.0 | -2.0 | 4.0 |
| -4.0 | -14.0 | 4.0 | 4.0 | .0 |
| 6.0 | 6.0 | | | |

TLU. NUMBER 10

| | | | | |
|---|---|---|---|---|
| 2.0 | 2.0 | 2.0 | 2.0 | 2.0 |
| 2.0 | 2.0 | .0 | -6.0 | -8.0 |
| .0 | .0 | -16.0 | .0 | 2.0 |
| 8.0 | .0 | 6.0 | 2.0 | .0 |
| 4.0 | 22.0 | .0 | -2.0 | -6.0 |
| -2.0 | -16.0 | | | |

TLU. NUMBER 11

| | | | | |
|---|---|---|---|---|
| 3.0 | 3.0 | 3.0 | 3.0 | 3.0 |
| 3.0 | 3.0 | 3.0 | 3.0 | -5.0 |
| -1.0 | -1.0 | -17.0 | -1.0 | 3.0 |
| 5.0 | 1.0 | 3.0 | 3.0 | -1.0 |
| -1.0 | 17.0 | -1.0 | -1.0 | -3.0 |
| -5.0 | -13.0 | | | |

TLU. NUMBER 12

| | | | | |
|---|---|---|---|---|
| .0 | .0 | .0 | .0 | .0 |
| .0 | .0 | 2.0 | -4.0 | -6.0 |
| 2.0 | 2.0 | 4.0 | 2.0 | .0 |
| 8.0 | 2.0 | 2.0 | .0 | 2.0 |
| 2.0 | 12.0 | 2.0 | .0 | -8.0 |
| -4.0 | -14.0 | | | |

TLU. NUMBER 13

| | | | | |
|---|---|---|---|---|
| 1.0 | 1.0 | 1.0 | 1.0 | 1.0 |
| 1.0 | 1.0 | -3.0 | -5.0 | -5.0 |
| 1.0 | 1.0 | -1.0 | 1.0 | 1.0 |
| 7.0 | 5.0 | -1.0 | 1.0 | 1.0 |
| 5.0 | 11.0 | 1.0 | -1.0 | -7.0 |
| 1.0 | -13.0 | | | |

TLU. NUMBER 14

| | | | | |
|---|---|---|---|---|
| .0 | .0 | .0 | .0 | .0 |
| .0 | .0 | .0 | -4.0 | .0 |
| 2.0 | 2.0 | 2.0 | .0 | .0 |
| 2.0 | 2.0 | -2.0 | .0 | 2.0 |
| 6.0 | 4.0 | 2.0 | 2.0 | 2.0 |
| -8.0 | .0 | | | |

Weights for Rattlers Frustration (Continued)

TLU. NUMBER 15

| | | | | |
|---|---|---|---|---|
| .0 | .0 | .0 | .0 | .0 |
| .0 | .0 | .0 | -4.0 | .0 |
| 2.0 | 2.0 | 2.0 | .0 | .0 |
| 2.0 | 2.0 | -2.0 | .0 | 2.0 |
| 6.0 | 4.0 | 2.0 | 2.0 | 2.0 |
| -8.0 | .0 | | | |

TLU. NUMBER 16

| | | | | |
|---|---|---|---|---|
| .0 | .0 | .0 | .0 | .0 |
| .0 | .0 | .0 | -4.0 | -2.0 |
| 2.0 | 2.0 | 4.0 | .0 | .0 |
| 2.0 | 2.0 | -2.0 | .0 | 2.0 |
| 8.0 | 4.0 | 2.0 | 2.0 | .0 |
| -6.0 | 2.0 | | | |

TLU. NUMBER 17

| | | | | |
|---|---|---|---|---|
| -1.0 | -1.0 | -1.0 | -1.0 | -1.0 |
| -1.0 | -1.0 | -1.0 | -3.0 | -1.0 |
| 3.0 | 3.0 | 3.0 | -1.0 | -1.0 |
| 3.0 | 3.0 | -1.0 | -1.0 | 3.0 |
| 7.0 | 3.0 | 3.0 | 3.0 | 1.0 |
| -5.0 | 1.0 | | | |

# WEIGHTS FOR EAGLES ATTITUDE

## ERRORS

| Word No. | Classed As | Should Be |
|---|---|---|
| 10 | 2 | 3 |
| 13 | 2 | 3 |
| 25 | 3 | 2 |
| 31 | 3 | 2 |
| 32 | 3 | 2 |

### TLU. NUMBER 1

| | | | | |
|---|---|---|---|---|
| .0 | .0 | .0 | .0 | .0 |
| .0 | 8.0 | -2.0 | -2.0 | 4.0 |
| .0 | .0 | 2.0 | .0 | 2.0 |
| -6.0 | -6.0 | .0 | 2.0 | .0 |
| -4.0 | 4.0 | .0 | .0 | .0 |
| 2.0 | -4.0 | .0 | .0 | 10.0 |
| 2.0 | 2.0 | 4.0 | 4.0 | .0 |

### TLU. NUMBER 2

| | | | | |
|---|---|---|---|---|
| .0 | .0 | .0 | .0 | .0 |
| .0 | 8.0 | -2.0 | -2.0 | 4.0 |
| .0 | .0 | 2.0 | .0 | 2.0 |
| -6.0 | -6.0 | .0 | 2.0 | .0 |
| -4.0 | 4.0 | .0 | .0 | .0 |
| 2.0 | -4.0 | .0 | .0 | 10.0 |
| 2.0 | 2.0 | 4.0 | 4.0 | .0 |

### TLU. NUMBER 3

| | | | | |
|---|---|---|---|---|
| 1.0 | 1.0 | 1.0 | 1.0 | 1.0 |
| 1.0 | 7.0 | -1.0 | -1.0 | 5.0 |
| -1.0 | -1.0 | 1.0 | 1.0 | 3.0 |
| -7.0 | -7.0 | 1.0 | 3.0 | -1.0 |
| -11.0 | 5.0 | -1.0 | -1.0 | 1.0 |
| 1.0 | 3.0 | -1.0 | -1.0 | 9.0 |
| 3.0 | 3.0 | 3.0 | 3.0 | -1.0 |

### TLU. NUMBER 4

| | | | | |
|---|---|---|---|---|
| .0 | .0 | .0 | .0 | .0 |
| .0 | 4.0 | 2.0 | 2.0 | 8.0 |
| .0 | .0 | -2.0 | -2.0 | 2.0 |
| -8.0 | -8.0 | -4.0 | 2.0 | .0 |
| .0 | 2.0 | .0 | .0 | .0 |
| 4.0 | -4.0 | .0 | 2.0 | -2.0 |
| 2.0 | 2.0 | 4.0 | 4.0 | .0 |

### TLU. NUMBER 5

| | | | | |
|---|---|---|---|---|
| -12.0 | -8.0 | -4.0 | -4.0 | -2.0 |
| -2.0 | 6.0 | 6.0 | -10.0 | .0 |
| .0 | .0 | -8.0 | 8.0 | 2.0 |
| -6.0 | -10.0 | -4.0 | 2.0 | .0 |
| .0 | 10.0 | .0 | .0 | -4.0 |
| -6.0 | 4.0 | .0 | 6.0 | 10.0 |
| 2.0 | 2.0 | -6.0 | -6.0 | .0 |

### TLU. NUMBER 6

| | | | | |
|---|---|---|---|---|
| 8.0 | 4.0 | 2.0 | .0 | .0 |
| .0 | -2.0 | -2.0 | 8.0 | 6.0 |
| .0 | .0 | 10.0 | -16.0 | 2.0 |
| -10.0 | -6.0 | .0 | 2.0 | .0 |
| 10.0 | -2.0 | .0 | .0 | 8.0 |
| 14.0 | -4.0 | .0 | -12.0 | .0 |
| 2.0 | 2.0 | 10.0 | 10.0 | .0 |

### TLU. NUMBER 7

| | | | | |
|---|---|---|---|---|
| 12.0 | 6.0 | 2.0 | 2.0 | .0 |
| .0 | .0 | .0 | 8.0 | 2.0 |
| .0 | .0 | 4.0 | -12.0 | 2.0 |
| -12.0 | -6.0 | .0 | 2.0 | .0 |
| -4.0 | -6.0 | .0 | .0 | 10.0 |
| 6.0 | .0 | .0 | -6.0 | 2.0 |
| 2.0 | 2.0 | 10.0 | 10.0 | .0 |

### TLU. NUMBER 8

| | | | | |
|---|---|---|---|---|
| 4.0 | 2.0 | .0 | -2.0 | -2.0 |
| -2.0 | -2.0 | .0 | 6.0 | 6.0 |
| 2.0 | 2.0 | 12.0 | -12.0 | .0 |
| -2.0 | .0 | 4.0 | .0 | 2.0 |
| -4.0 | -2.0 | 2.0 | 4.0 | 10.0 |
| 6.0 | -2.0 | 2.0 | -10.0 | .0 |
| -2.0 | .0 | 6.0 | 6.0 | 2.0 |

Weights for Eagles Attitude (Continued)

TLU. NUMBER 9

| | | | | |
|---|---|---|---|---|
| 10.0 | 6.0 | 4.0 | 2.0 | .0 |
| .0 | -2.0 | .0 | 10.0 | -2.0 |
| 2.0 | 2.0 | 6.0 | -18.0 | .0 |
| .0 | 6.0 | -2.0 | .0 | 2.0 |
| -6.0 | -10.0 | 2.0 | 2.0 | -2.0 |
| -2.0 | -4.0 | 2.0 | -6.0 | 10.0 |
| .0 | .0 | 10.0 | 10.0 | 2.0 |

TLU. NUMBER 10

| | | | | |
|---|---|---|---|---|
| .0 | .0 | .0 | .0 | .0 |
| .0 | .0 | .0 | -2.0 | .0 |
| 2.0 | 2.0 | -2.0 | -4.0 | .0 |
| .0 | -4.0 | 4.0 | .0 | 2.0 |
| 2.0 | .0 | 2.0 | 2.0 | 10.0 |
| 2.0 | .0 | 2.0 | 2.0 | .0 |
| .0 | .0 | -2.0 | -2.0 | 2.0 |

TLU. NUMBER 11

| | | | | |
|---|---|---|---|---|
| .0 | .0 | .0 | .0 | .0 |
| .0 | .0 | .0 | -2.0 | .0 |
| 2.0 | 2.0 | -2.0 | -4.0 | .0 |
| .0 | -4.0 | 4.0 | .0 | 2.0 |
| 2.0 | .0 | 2.0 | 2.0 | 10.0 |
| 2.0 | .0 | 2.0 | 2.0 | .0 |
| .0 | .0 | -2.0 | -2.0 | 2.0 |

TLU. NUMBER 12

| | | | | |
|---|---|---|---|---|
| .0 | .0 | .0 | .0 | .0 |
| .0 | .0 | .0 | -2.0 | .0 |
| 2.0 | 2.0 | -2.0 | -4.0 | .0 |
| .0 | -4.0 | 4.0 | .0 | 2.0 |
| 2.0 | .0 | 2.0 | 2.0 | 10.0 |
| 2.0 | .0 | 2.0 | 2.0 | .0 |
| .0 | .0 | -2.0 | -2.0 | 2.0 |

TLU. NUMBER 13

| | | | | |
|---|---|---|---|---|
| .0 | .0 | .0 | .0 | .0 |
| .0 | .0 | .0 | -4.0 | .0 |
| 2.0 | 2.0 | -2.0 | -4.0 | .0 |
| 2.0 | .0 | 4.0 | .0 | 2.0 |
| 2.0 | -4.0 | 2.0 | .0 | 6.0 |
| .0 | .0 | 2.0 | 2.0 | 4.0 |
| 2.0 | 2.0 | .0 | .0 | 2.0 |

TLU. NUMBER 14

| | | | | |
|---|---|---|---|---|
| .0 | .0 | .0 | .0 | .0 |
| .0 | .0 | .0 | .0 | -2.0 |
| 2.0 | 2.0 | .0 | -2.0 | .0 |
| 2.0 | .0 | 2.0 | .0 | 2.0 |
| 2.0 | 4.0 | 2.0 | 6.0 | 6.0 |
| 2.0 | .0 | 2.0 | 2.0 | .0 |
| -4.0 | .0 | .0 | .0 | 2.0 |

TLU. NUMBER 15

| | | | | |
|---|---|---|---|---|
| .0 | .0 | .0 | .0 | .0 |
| .0 | .0 | .0 | .0 | -2.0 |
| 2.0 | 2.0 | .0 | -2.0 | .0 |
| 2.0 | .0 | 2.0 | .0 | 2.0 |
| 2.0 | 4.0 | 2.0 | 6.0 | 6.0 |
| 2.0 | .0 | 2.0 | 2.0 | .0 |
| -4.0 | .0 | .0 | .0 | 2.0 |

TLU. NUMBER 16

| | | | | |
|---|---|---|---|---|
| .0 | .0 | .0 | .0 | .0 |
| .0 | .0 | .0 | 2.0 | -2.0 |
| 2.0 | 2.0 | .0 | .0 | .0 |
| .0 | .0 | 2.0 | .0 | 2.0 |
| .0 | 6.0 | 2.0 | 8.0 | 6.0 |
| 2.0 | .0 | 2.0 | 2.0 | -2.0 |
| -6.0 | -2.0 | 2.0 | 2.0 | 2.0 |

TLU. NUMBER 17

| | | | | |
|---|---|---|---|---|
| .0 | .0 | .0 | .0 | .0 |
| .0 | .0 | .0 | 2.0 | -2.0 |
| 2.0 | 2.0 | .0 | .0 | .0 |
| .0 | .0 | 2.0 | .0 | 2.0 |
| .0 | 6.0 | 2.0 | 8.0 | 6.0 |
| 2.0 | .0 | 2.0 | 2.0 | -2.0 |
| -6.0 | -2.0 | 2.0 | 2.0 | 2.0 |

## WEIGHTS FOR EAGLES STRUCTURE

### NO ERRORS

TLU. NUMBER 1

| | | | | |
|---|---|---|---|---|
| 2.0 | 2.0 | 2.0 | 2.0 | 2.0 |
| -10.0 | -6.0 | 8.0 | .0 | -6.0 |
| .0 | .0 | 8.0 | 4.0 | 2.0 |
| .0 | 4.0 | 2.0 | 2.0 | .0 |
| 2.0 | 2.0 | .0 | .0 | -14.0 |
| -4.0 | 2.0 | .0 | -2.0 | -6.0 |
| -2.0 | -2.0 | .0 | | |

TLU. NUMBER 2

| | | | | |
|---|---|---|---|---|
| 3.0 | 3.0 | 3.0 | 3.0 | 3.0 |
| -7.0 | -3.0 | 9.0 | -1.0 | -7.0 |
| -1.0 | -1.0 | 5.0 | 3.0 | 3.0 |
| 1.0 | 5.0 | -1.0 | 3.0 | -1.0 |
| -1.0 | 3.0 | -1.0 | -1.0 | -13.0 |
| -3.0 | 1.0 | -1.0 | -3.0 | -5.0 |
| -3.0 | -3.0 | -1.0 | | |

TLU. NUMBER 3

| | | | | |
|---|---|---|---|---|
| 2.0 | 2.0 | 2.0 | 2.0 | -2.0 |
| -10.0 | -6.0 | 6.0 | .0 | -6.0 |
| .0 | .0 | 6.0 | 2.0 | 2.0 |
| 2.0 | 4.0 | 2.0 | 2.0 | .0 |
| .0 | 2.0 | .0 | .0 | -12.0 |
| -4.0 | 2.0 | .0 | -6.0 | -6.0 |
| -2.0 | -2.0 | .0 | | |

TLU. NUMBER 4

| | | | | |
|---|---|---|---|---|
| 2.0 | 2.0 | 2.0 | 2.0 | -2.0 |
| -8.0 | -4.0 | 4.0 | -2.0 | .0 |
| .0 | .0 | 4.0 | 2.0 | 2.0 |
| .0 | .0 | 2.0 | 2.0 | .0 |
| 2.0 | 2.0 | .0 | .0 | -4.0 |
| -6.0 | .0 | .0 | -8.0 | -4.0 |
| -4.0 | -4.0 | .0 | | |

TLU. NUMBER 5

| | | | | |
|---|---|---|---|---|
| .0 | .0 | .0 | .0 | 4.0 |
| -6.0 | -2.0 | .0 | -8.0 | -8.0 |
| 2.0 | 2.0 | .0 | .0 | .0 |
| 4.0 | 8.0 | .0 | .0 | 2.0 |
| -4.0 | 4.0 | 2.0 | 2.0 | -2.0 |
| -4.0 | .0 | 2.0 | 6.0 | 4.0 |
| 2.0 | 2.0 | 2.0 | | |

TLU. NUMBER 6

| | | | | |
|---|---|---|---|---|
| .0 | .0 | .0 | .0 | -6.0 |
| -4.0 | -4.0 | .0 | 6.0 | .0 |
| .0 | .0 | 4.0 | 6.0 | 2.0 |
| .0 | -6.0 | 2.0 | 2.0 | .0 |
| 4.0 | -2.0 | .0 | .0 | -4.0 |
| -2.0 | 2.0 | .0 | -6.0 | -6.0 |
| .0 | .0 | .0 | | |

TLU. NUMBER 7

| | | | | |
|---|---|---|---|---|
| -1.0 | -1.0 | -1.0 | -1.0 | -5.0 |
| -5.0 | -5.0 | -5.0 | 7.0 | 7.0 |
| 1.0 | 1.0 | 5.0 | 7.0 | 1.0 |
| 1.0 | -7.0 | 3.0 | 1.0 | 1.0 |
| 7.0 | -5.0 | 1.0 | 1.0 | -3.0 |
| -9.0 | -1.0 | 1.0 | -5.0 | -11.0 |
| -1.0 | -1.0 | 1.0 | | |

TLU. NUMBER 8

| | | | | |
|---|---|---|---|---|
| 1.0 | 1.0 | 1.0 | 1.0 | -5.0 |
| -3.0 | -3.0 | 1.0 | 11.0 | 7.0 |
| 1.0 | 1.0 | 7.0 | 13.0 | 1.0 |
| 3.0 | -3.0 | 1.0 | 1.0 | 1.0 |
| 7.0 | -3.0 | 1.0 | 1.0 | -5.0 |
| -7.0 | -1.0 | 1.0 | -5.0 | -13.0 |
| -1.0 | -1.0 | 1.0 | | |

TLU. NUMBER 9

| | | | | |
|---|---|---|---|---|
| -2.0 | -2.0 | -2.0 | -2.0 | -6.0 |
| -6.0 | -6.0 | -2.0 | 10.0 | 8.0 |
| .0 | .0 | 2.0 | 14.0 | 2.0 |
| 2.0 | -2.0 | 4.0 | 2.0 | .0 |
| 8.0 | -6.0 | .0 | .0 | -6.0 |
| -4.0 | -2.0 | .0 | -4.0 | -12.0 |
| 4.0 | 4.0 | .0 | | |

## WEIGHTS FOR EAGLES MORALE

### NO ERRORS

**TLU. NUMBER 1**

| | | | | |
|---|---|---|---|---|
| 3.0 | 3.0 | 3.0 | 3.0 | 3.0 |
| 3.0 | -3.0 | 1.0 | 1.0 | -1.0 |
| -1.0 | -1.0 | 1.0 | 3.0 | 3.0 |
| 1.0 | 3.0 | -3.0 | 3.0 | -1.0 |
| -5.0 | -15.0 | -1.0 | -1.0 | -1.0 |
| 5.0 | -3.0 | -1.0 | 3.0 | -11.0 |
| 3.0 | 3.0 | -1.0 | | |

**TLU. NUMBER 2**

| | | | | |
|---|---|---|---|---|
| 3.0 | 3.0 | 3.0 | 3.0 | 3.0 |
| 3.0 | 1.0 | 3.0 | 1.0 | -1.0 |
| -1.0 | -1.0 | -.30 | 3.0 | 3.0 |
| 1.0 | 3.0 | -5.0 | 3.0 | -1.0 |
| -3.0 | -13.0 | -1.0 | -1.0 | -1.0 |
| 7.0 | -5.0 | -1.0 | 1.0 | -13.0 |
| 3.0 | 3.0 | -1.0 | | |

**TLU. NUMBER 3**

| | | | | |
|---|---|---|---|---|
| 4.0 | 4.0 | 4.0 | 4.0 | 4.0 |
| 4.0 | 2.0 | 4.0 | 2.0 | .0 |
| -2.0 | -2.0 | .0 | 4.0 | 4.0 |
| -2.0 | .0 | .0 | 4.0 | -2.0 |
| .0 | -4.0 | -2.0 | -2.0 | -2.0 |
| 2.0 | .0 | -2.0 | 2.0 | -2.0 |
| 2.0 | 2.0 | -2.0 | | |

**TLU. NUMBER 4**

| | | | | |
|---|---|---|---|---|
| 3.0 | 3.0 | 3.0 | 3.0 | 3.0 |
| 3.0 | -3.0 | 3.0 | 1.0 | 1.0 |
| -1.0 | -1.0 | 3.0 | 3.0 | 3.0 |
| 1.0 | 1.0 | -7.0 | 3.0 | -1.0 |
| -5.0 | -11.0 | -1.0 | -1.0 | -1.0 |
| 7.0 | -5.0 | -1.0 | -3.0 | -13.0 |
| 1.0 | 1.0 | -1.0 | | |

**TLU. NUMBER 5**

| | | | | |
|---|---|---|---|---|
| 2.0 | 2.0 | 2.0 | 2.0 | 2.0 |
| 2.0 | 2.0 | 6.0 | -10.0 | 8.0 |
| .0 | .0 | -6.0 | 10.0 | 2.0 |
| 2.0 | 4.0 | -8.0 | 2.0 | .0 |
| 2.0 | 6.0 | .0 | .0 | -4.0 |
| 2.0 | -8.0 | .0 | 8.0 | 2.0 |
| 8.0 | 8.0 | .0 | | |

**TLU. NUMBER 6**

| | | | | |
|---|---|---|---|---|
| 1.0 | 1.0 | 1.0 | 1.0 | 1.0 |
| 1.0 | -3.0 | 5.0 | 13.0 | -7.0 |
| 1.0 | 1.0 | 9.0 | -3.0 | 1.0 |
| 5.0 | 1.0 | -1.0 | 1.0 | 1.0 |
| -1.0 | -15.0 | 1.0 | 1.0 | 1.0 |
| 5.0 | 3.0 | 1.0 | -5.0 | -13.0 |
| -9.0 | -9.0 | 1.0 | | |

**TLU. NUMBER 7**

| | | | | |
|---|---|---|---|---|
| .0 | .0 | .0 | .0 | .0 |
| .0 | -2.0 | -2.0 | 10.0 | -8.0 |
| 2.0 | 2.0 | 8.0 | -4.0 | .0 |
| .0 | -4.0 | -4.0 | .0 | 2.0 |
| 2.0 | -14.0 | 2.0 | 2.0 | 4.0 |
| 4.0 | -2.0 | 2.0 | -4.0 | -10.0 |
| -4.0 | -4.0 | 2.0 | | |

**TLU. NUMBER 8**

| | | | | |
|---|---|---|---|---|
| 2.0 | 2.0 | 2.0 | 2.0 | 2.0 |
| 2.0 | -4.0 | .0 | 8.0 | -8.0 |
| .0 | .0 | 6.0 | -4.0 | 2.0 |
| .0 | -2.0 | .0 | 2.0 | .0 |
| .0 | -14.0 | .0 | .0 | 2.0 |
| 8.0 | 6.0 | .0 | -4.0 | -16.0 |
| -6.0 | -6.0 | .0 | | |

**TLU. NUMBER 9**

| | | | | |
|---|---|---|---|---|
| 1.0 | 1.0 | 1.0 | 1.0 | 1.0 |
| 1.0 | 1.0 | -1.0 | 11.0 | -9.0 |
| 1.0 | 1.0 | 5.0 | -3.0 | 1.0 |
| -1.0 | -1.0 | -1.0 | 1.0 | 1.0 |
| 1.0 | -15.0 | 1.0 | 1.0 | 5.0 |
| 3.0 | 1.0 | 1.0 | -5.0 | -15.0 |
| -5.0 | -5.0 | 1.0 | | |

## WEIGHTS FOR EAGLES FRUSTRATION

### NO ERRORS

TLU. NUMBER 1

| | | | | |
|---|---|---|---|---|
| 3.0 | 1.0 | 1.0 | 1.0 | 1.0 |
| 1.0 | 1.0 | -1.0 | -3.0 | -3.0 |
| -1.0 | -1.0 | 3.0 | 3.0 | 3.0 |
| 1.0 | 3.0 | -3.0 | 3.0 | -1.0 |
| 1.0 | -3.0 | -1.0 | -3.0 | 9.0 |
| 1.0 | 1.0 | | | |

TLU. NUMBER 2

| | | | | |
|---|---|---|---|---|
| 3.0 | 1.0 | 1.0 | 1.0 | 1.0 |
| 1.0 | 1.0 | -1.0 | -3.0 | -3.0 |
| -1.0 | -1.0 | 3.0 | 3.0 | 3.0 |
| 1.0 | 3.0 | -3.0 | 3.0 | -1.0 |
| 1.0 | -3.0 | -1.0 | -3.0 | -9.0 |
| 1.0 | 1.0 | | | |

TLU. NUMBER 3

| | | | | |
|---|---|---|---|---|
| 3.0 | 1.0 | 1.0 | 1.0 | 1.0 |
| 1.0 | 1.0 | -1.0 | -3.0 | -3.0 |
| -1.0 | -1.0 | 3.0 | 3.0 | 3.0 |
| 1.0 | 3.0 | -3.0 | 3.0 | -1.0 |
| 1.0 | -3.0 | -1.0 | -3.0 | -9.0 |
| 1.0 | 1.0 | | | |

TLU. NUMBER 4

| | | | | |
|---|---|---|---|---|
| 3.0 | 1.0 | 1.0 | 1.0 | 1.0 |
| 1.0 | 1.0 | -1.0 | -1.0 | -1.0 |
| -1.0 | -1.0 | 1.0 | 3.0 | 3.0 |
| 1.0 | 3.0 | -1.0 | 3.0 | -1.0 |
| 3.0 | -3.0 | -1.0 | -3.0 | -9.0 |
| -1.0 | -1.0 | | | |

TLU. NUMBER 5

| | | | | |
|---|---|---|---|---|
| .0 | .0 | .0 | .0 | .0 |
| .0 | .0 | 2.0 | 2.0 | -4.0 |
| 2.0 | 2.0 | .0 | -8.0 | .0 |
| 2.0 | .0 | -4.0 | .0 | 2.0 |
| 2.0 | -4.0 | 2.0 | 2.0 | .0 |
| 2.0 | .0 | | | |

TLU. NUMBER 6

| | | | | |
|---|---|---|---|---|
| 3.0 | 1.0 | 1.0 | 1.0 | 1.0 |
| 1.0 | 1.0 | -1.0 | -5.0 | 1.0 |
| -1.0 | -1.0 | 5.0 | 9.0 | 3.0 |
| -1.0 | 5.0 | 3.0 | 3.0 | -1.0 |
| 1.0 | -5.0 | -1.0 | -3.0 | -9.0 |
| -9.0 | -1.0 | | | |

TLU. NUMBER 7

| | | | | |
|---|---|---|---|---|
| 3.0 | 1.0 | 1.0 | 1.0 | 1.0 |
| 1.0 | 1.0 | -1.0 | -3.0 | 3.0 |
| -1.0 | -1.0 | 3.0 | 7.0 | 3.0 |
| -1.0 | 1.0 | 1.0 | 3.0 | -1.0 |
| -1.0 | 1.0 | -1.0 | -3.0 | -5.0 |
| -7.0 | -3.0 | | | |

TLU. NUMBER 8

| | | | | |
|---|---|---|---|---|
| 2.0 | 2.0 | 2.0 | 2.0 | 2.0 |
| 2.0 | 2.0 | .0 | .0 | 4.0 |
| .0 | .0 | 4.0 | 6.0 | 2.0 |
| .0 | 4.0 | 2.0 | 2.0 | .0 |
| .0 | .0 | .0 | -2.0 | -6.0 |
| -6.0 | -2.0 | | | |

TLU. NUMBER 9

| | | | | |
|---|---|---|---|---|
| 1.0 | 1.0 | 1.0 | 1.0 | 1.0 |
| 1.0 | 1.0 | 5.0 | 5.0 | -1.0 |
| 1.0 | 1.0 | 7.0 | 11.0 | 1.0 |
| -9.0 | 3.0 | -1.0 | 1.0 | 1.0 |
| 9.0 | -13.0 | 1.0 | -1.0 | -1.0 |
| 1.0 | 7.0 | | | |

TLU. NUMBER 10

| | | | | |
|---|---|---|---|---|
| 3.0 | 3.0 | 3.0 | 3.0 | 3.0 |
| 3.0 | 5.0 | -3.0 | -9.0 | 5.0 |
| -1.0 | -1.0 | -3.0 | -5.0 | 3.0 |
| 7.0 | -3.0 | 9.0 | 3.0 | -1.0 |
| -9.0 | 17.0 | -1.0 | -1.0 | -3.0 |
| -7.0 | -7.0 | | | |

Weights for Eagles Frustration (Continued)

TLU. NUMBER 11

| | | | | |
|---|---|---|---|---|
| 2.0 | 2.0 | 2.0 | 2.0 | 2.0 |
| 2.0 | 4.0 | .0 | -2.0 | 2.0 |
| .0 | .0 | -4.0 | 4.0 | 2.0 |
| 6.0 | -2.0 | 8.0 | 2.0 | .0 |
| 6.0 | 4.0 | .0 | -2.0 | -6.2 |
| -10.0 | -6.0 | | | |

TLU. NUMBER 12

| | | | | |
|---|---|---|---|---|
| 1.0 | 1.0 | 1.0 | 1.0 | 1.0 |
| 1.0 | 3.0 | -5.0 | -3.0 | 9.0 |
| 1.0 | 1.0 | 5.0 | 9.0 | 1.0 |
| 13.0 | 3.0 | 1.0 | 1.0 | 1.0 |
| -3.0 | 11.0 | 1.0 | -1.0 | -7.0 |
| -5.0 | -5.0 | | | |

TLU. NUMBER 13

| | | | | |
|---|---|---|---|---|
| 1.0 | 1.0 | 1.0 | 1.0 | 1.0 |
| 1.0 | 5.0 | -1.0 | -3.0 | -3.0 |
| 1.0 | 1.0 | -1.0 | -9.0 | 1.0 |
| 5.0 | -3.0 | 9.0 | 1.0 | 1.0 |
| -9.0 | 9.0 | 1.0 | -1.0 | -7.0 |
| -1.0 | -17.0 | | | |

TLU. NUMBER 14

| | | | | |
|---|---|---|---|---|
| -1.0 | -1.0 | -1.0 | -1.0 | -1.0 |
| -1.0 | -5.0 | -3.0 | -3.0 | -1.0 |
| 3.0 | 3.0 | -1.0 | 5.0 | -1.0 |
| 5.0 | 3.0 | -1.0 | -1.0 | 3.0 |
| 1.0 | -1.0 | 3.0 | 3.0 | 3.0 |
| -9.0 | 3.0 | | | |

TLU. NUMBER 15

| | | | | |
|---|---|---|---|---|
| .0 | .0 | .0 | .0 | .0 |
| .0 | -4.0 | -4.0 | -2.0 | 2.0 |
| 2.0 | 2.0 | -4.0 | 6.0 | .0 |
| 4.0 | .0 | -4.0 | .0 | 2.0 |
| -2.0 | -4.0 | 2.0 | 2.0 | 2.0 |
| -8.0 | 4.0 | | | |

TLU. NUMBER 16

| | | | | |
|---|---|---|---|---|
| .0 | .0 | .0 | .0 | .0 |
| .0 | -8.0 | -8.0 | -6.0 | 4.0 |
| 2.0 | 2.0 | -10.0 | 6.0 | .0 |
| 6.0 | 2.0 | -4.0 | .0 | 2.0 |
| .0 | -4.0 | 2.0 | 2.0 | 2.0 |
| -6.0 | 10.0 | | | |

TLU. NUMBER 17

| | | | | |
|---|---|---|---|---|
| -1.0 | -1.0 | -1.0 | -1.0 | -1.0 |
| -1.0 | -3.0 | -3.0 | -1.0 | 3.0 |
| 3.0 | 3.0 | -5.0 | 7.0 | -1.0 |
| 9.0 | 1.0 | -5.0 | -1.0 | 3.0 |
| -3.0 | -5.0 | 3.0 | 3.0 | 3.0 |
| -9.0 | 9.0 | | | |