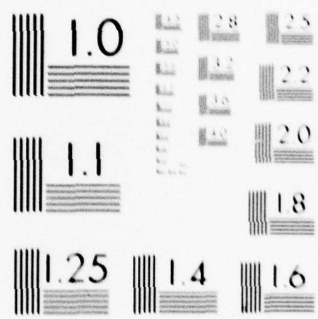
AD-A070 079 OHIO STATE UNIV RESEARCH FOUNDATION COLUMBUS F/G 12/1
IDENTIFICATION OF FINITE STATE MODELS OF HUMAN OPERATORS.(U)
MAR 79 R A MILLER AFOSR-77-3152
UNCLASSIFIED OSURF-760518/784556 AFOSR-TR-79-0703 NL

1 OF 3
AD A070079



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

IDENTIFICATION OF FINITE STATE MODELS OF HUMAN OPERATORS

Final Report
AFOSR Grant #77-3152
OSU Research Project #784556

by

Richard A. Miller
Principal Investigator

SYSTEMS RESEARCH GROUP
Department of Industrial and Systems Engineering
The Ohio State University
Columbus, Ohio 43210

March 1, 1979

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: AFOSR-TR-79-0703
2. GOVT ACCESSION NO.:
3. RECIPIENT'S CATALOG NUMBER:
4. TITLE (and Subtitle): IDENTIFICATION OF FINITE STATE MODELS OF HUMAN OPERATORS.
5. TYPE OF REPORT & PERIOD COVERED: Final 10/1/76-12/31/78
6. PERFORMING ORG. REPORT NUMBER: OSURF-760518/784556 - Final
7. AUTHOR(s): Richard A. Miller
8. GRANT NUMBER(s): AFOSR-77-3152
9. PERFORMING ORGANIZATION NAME AND ADDRESS: The Ohio State University Research Foundation, 1314 Kinnear Road, Columbus, Ohio 43212
10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS: 61102F 2312/A6
11. CONTROLLING OFFICE NAME AND ADDRESS: U.S. Air Force, Office of Scientific Research (NL), Building 410, Bolling Air Force Base, D.C. 20332
12. REPORT DATE: March 1, 1979
13. NUMBER OF PAGES: 243 pp.
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office): Final rept. 1 Oct 76-31 Dec 78
15. SECURITY CLASS. (of this report): Unclassified

15a. DECLASSIFICATION/DOWNGRADING SCHEDULE:

16. DISTRIBUTION STATEMENT (of this Report): "Approved for public release; distribution unlimited." 1 Mar 79 245p.

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

18. SUPPLEMENTARY NOTES:

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

supervisory control, discrete control, decision making, teams of operators, knowledge representation, network models, event based systems

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

The problems of modelling a class of manned systems in which the operator or operators have available only a finite number of decision alternatives which they can use to control the system configuration and mode of operation over time are the focus of this report. An abstract system theoretic structure suitable for representing such discrete control systems is developed and the structure is used to organize the analysis of data obtained from a man-in-the-loop simulation of an AAA system. The simulation was performed by the Aerospace Medical Research Laboratory, Wright-Patterson Air Force Base. The

DD FORM 1473, 1 JAN 73 — EDITION OF 1 NOV 65 IS OBSOLETE — over

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

267 360

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE(When Data Entered)

Block #20 - Abstract (Continued)

analysis was performed at The Ohio State University with funding provided by this grant.

The structure used to define discrete control is a hierarchical/heterarchical network of finite state systems. The nodes in this network represent system components, task and activities. Several levels of abstraction are used which means that both macroscopic and microscopic descriptions are possible. The structure captures the certain aspects of coordination and the flow of decision making activity through the system.

The structure is used to guide the analysis of data obtained from a simulation involving teams of three operators controlling a fairly complex system. The resulting model clearly describes the decision making strategies employed by the various subject teams. Situations of high confusion, rare event type errors, and misunderstandings of the system are all readily detectable from the model.

The feasibility of discrete control modelling is demonstrated. The structural aspects, particularly knowledge representation and the identification of key decision points, seem quite powerful. The statistical and data analysis procedures work successfully but need further refinement if model construction is to be maximally efficient.

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE(When Data Entered)

TABLE OF CONTENTS

CHAPTER		Page
1	INTRODUCTION	1
2	DISCRETE CONTROL MODELLING	4
	A. An Abstract Behavioral Description	5
	B. State Representations	12
	C. Networks of Finite State Systems	22
	D. Some Comments about Modelling Strategy	29
3	THE DISCRETE CONTROL II EXPERIMENT	34
	A. Description of the System	34
	B. Description of the Experiment	41
	C. The Data Base	44
4	A DISCRETE CONTROL MODEL OF AAA TEAM PERFORMANCE	50
	A. Overview of the Modelling Approach	50
	B. System Decomposition	52
	B.1 Decomposition by Variable Type	53
	B.2 Decomposition by Level of Analysis and Description	54
	B.3 Decomposition by System Type	65
	C. Some Block Diagram Representations of the System	68
	D. The Control/Coordination Network	74
	E. The Decision Conditioning Network	85
5	ANALYSIS OF THE DISCRETE CONTROL II DATA	89
	A. Summary of the Discrete Control II Data Base	91
	B. Analysis of Key Decision Controlled Systems	134
	C. Some Additional Analysis	159
6	SUMMARY AND CONCLUSIONS	170
REFERENCES		178

APPENDIX

1 TARGET STATE ASSIGNMENTS 179

2. COMPONENT SYSTEM STATE ASSIGNMENTS 181

3. MISSION STATUS INFORMATION SYSTEM 187

4 COORDINATION/CONTROL NETWORK STATE TRANSITION GRAPHS 189

5 ANALYSIS ROUTINES 210

ABSTRACT

The problems of modelling a class of manned systems in which the operator or operators have available only a finite number of decision alternatives which they can use to control the system configuration and mode of operation over time are the focus of this report. An abstract system theoretic structure suitable for representing such discrete control systems is developed and the structure is used to organize the analysis of data obtained from a man-in-the-loop simulation of an AAA system. The simulation was performed by the Aerospace Medical Research Laboratory, Wright-Patterson Air Force Base. The analysis was performed at The Ohio State University with funding provided by this grant.

The structure used to define discrete control is a hierarchical/heterarchical network of finite state systems. The nodes in this network represent system components, task and activities. Several levels of abstraction are used which means that both macroscopic and microscopic descriptions are possible. The structure captures the certain aspects of coordination and the flow of decision making activity through the system.

The structure is used to guide the analysis of data obtained from a simulation involving teams of three operators controlling a fairly complex system. The resulting model clearly describes the decision making strategies

employed by the various subject teams. Situations of high confusion, rare event type errors, and misunderstandings of the system are all readily detectable from the model.

The feasibility of discrete control modelling is demonstrated. The structural aspects, particularly knowledge representation and the identification of key decision points, seem quite powerful. The statistical and data analysis procedures work successfully but need further refinement if model construction is to be maximally efficient.

CHAPTER 1

INTRODUCTION

This report is the final report on a two year research effort which has focused on the problems of mathematically modelling the discrete control activities in, and performance of, a class of manned systems. The general objectives were to define and develop the structure of discrete control models, to construct methods of representing specific systems within this abstract structure, and to develop the algorithms necessary to analyze and interpret data within this general structure.

This report focuses mainly on the second year of effort. The first year was spent primarily in developing the abstract structure and studying the results of a pilot study in order to gain insight into the problems of discrete control modelling of human operators. Certain basic statistical and methodological issues were addressed at that time as well. The model structure has been substantially refined during the second phase of the effort and the procedures used for data analysis have been similarly refined and upgraded. These improvements are discussed in this report.

During the second phase of the research the effort has focused on theoretical questions of modelling and system representation, and the problems of using a discrete control representation to analyze and interpret the data produced by a rather complex discrete control experiment. The objectives of each major activity are described below.

The modelling questions focus on the problems of capturing in some mathematical representation the way in which team members might decompose a complex problem into simpler parts and how they then manage to coordinate their individual activities and configure the system so that acceptable overall system performance is achieved. The basic questions therefore are questions of knowledge representation, information flow and communication in a complex system. A hierarchical/heterarchical structure which allows for structural coordination of subsystems by upper level components and which utilizes a heterarchical control structure to shift the focus of control to the proper subsystem at the proper time has been developed. This general structure can be used to explain how a system is configured at any given point in time and how the activities of the operators are coordinated over time.

The general structure was used to guide the analysis of a simulated antiaircraft artillery system. A specific realization of the structure was constructed and the data base obtained from experiments using the above mentioned simulator were interpreted using this structure and specially designed analysis routines. The results of this analysis and modelling form a major part of this report.

Detailed discussions of the research are given in the following sections of the report. In Chapter 2 the theoretical questions of modelling and discrete control representation of coordination are discussed. An overview of the discrete control II experiment which was conducted by the Aerospace Medical Research

Laboratory, Wright Patterson Air Force Base and which formed the above mentioned data base is discussed Chapter 3. Chapter 4 contains a detailed discussion of the hierarchical/heterarchical structure which was used to model the discrete control aspects of the antiaircraft artillery system. Results and observations obtained from analysis of the data base are given in Chapter 5.

CHAPTER 2

DISCRETE CONTROL MODELLING

The general class of systems of interest is those in which the operator (operators) of the system has (have) available only a finite number of control or decision alternatives and these are used to directly or indirectly control the behavior of the system over time. An operator might also have other tasks including continuous control tasks to perform, but the issue here is the set of discrete tasks by which the system configuration and mode of operation is established, and the procedures by which the team members' activities are coordinated. In this chapter a general structure suitable for representing discrete control performance is presented in three steps. The first two develop a set theoretic description of the abstract structure which serves to guide the design of data analysis procedures. The third step is the description of a network structure which is used to explore problems of system coordination. The development here is quite abstract and general, but it serves as the foundation for more specific model construction exercises. It will be used in Chapter 4 where a discrete control model of the antiaircraft artillery simulation is constructed. The reader primarily interested in applications and interpretations might want to skip the next section and go directly to section C of this chapter or to the start of Chapter 3.

A. An Abstract Behavioral Description

The only property established by the definition of discrete control cited above is that a finite number of control alternatives are available for use by the operators. The operators presumably change the alternative (control) selected from time to time in response to changing requirements or a changing environment. Basically, the purpose behind constructing a discrete control model is to explain how specific selections are reached from information about the system, environment and the context of the situation. More precisely, the objective is to provide some plausible and empirically testable explanation since many different explanations are possible. In discrete control then the model must attempt to explain some set of output event sequences of the type shown in Figure 2.1. Specific alternatives are selected and used for a period of time and then a switch is then made. The length of time that a given alternative is used need not be, and generally will not be, fixed.

It is assumed throughout this development that the information about the controlled system and the environment which is displayed to the operator, or otherwise provided him, is also discrete. The assumption is that such data are naturally discrete or are used by him in discrete form. A status display showing on/off information is naturally discrete, but a gauge with a needle showing speed is not. In the latter case it is assumed that such continuous information can be categorized in some way, e.g., slow, medium, fast. This

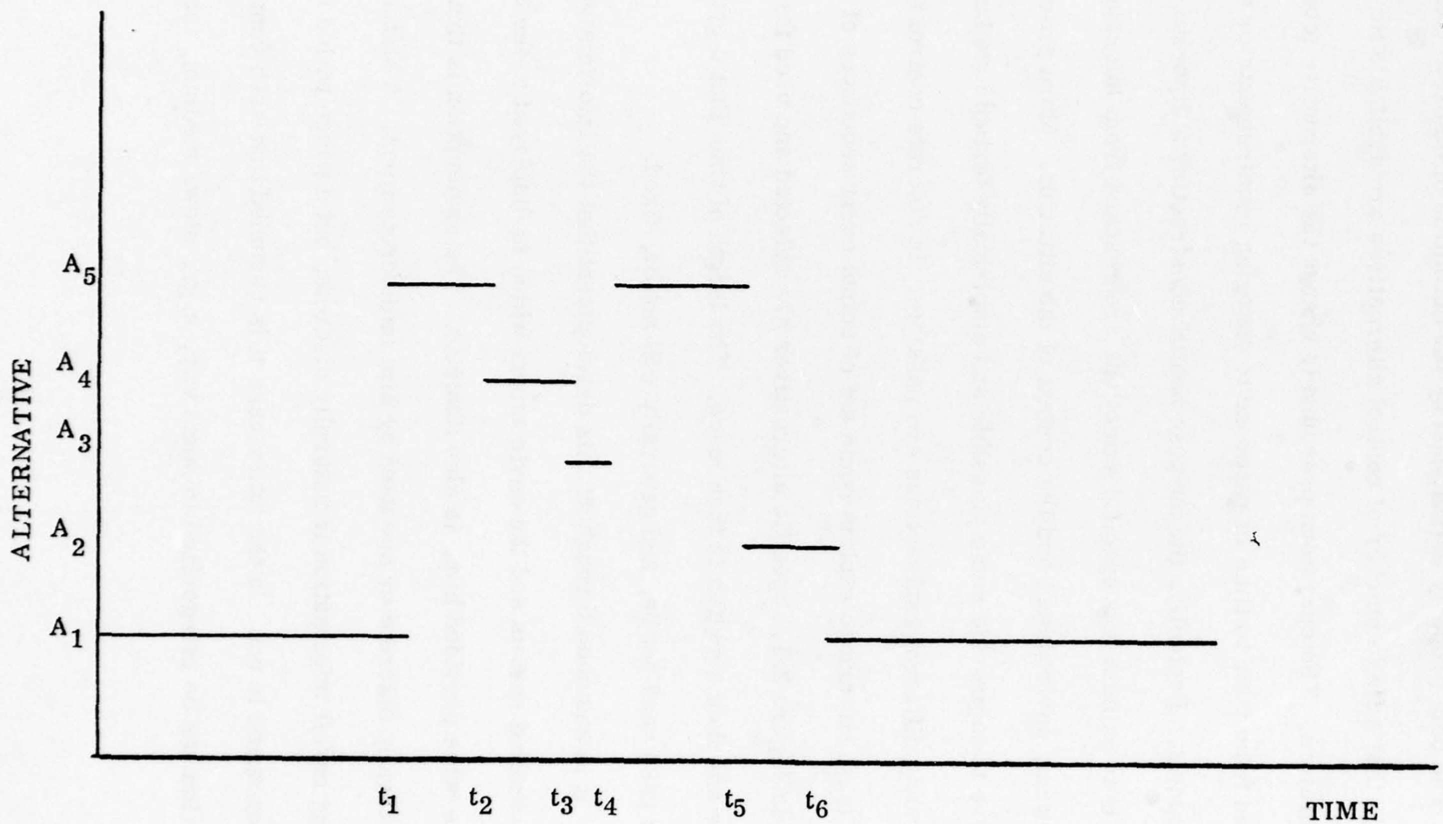


FIGURE 2.1--Typical Output Event Sequence

at first seems like a rather restricting assumption, but many studies have shown that operators tend to abstract higher level information even when the display is of the continuous type (Rasmussen, 1976). The process of representing continuous information in a discrete qualitative format in some sense corresponds to a feature extraction or abstraction process performed by an operator when encoding and internalizing the information provided him. Such processes seem quite reasonable in discrete control situations.

Abstractly then the discrete control context consists of an operator or team of operators receiving information in the form of event sequences and producing some sequence of control selections in response. To formalize the concept a time set, input alphabet and output alphabet are needed. A time set is a some set T together with a binary relation which linearly orders the set. Specifically, let T denote the set and let $<$,

$$< \subset T \times T,$$

denote the relation. The relation $<$ must be defined so that the following are true:

i. $\forall t_1, t_2 \in T, t_1 \neq t_2$ either $t_1 < t_2$ or $t_2 < t_1$

ii. $t_1 < t_2 \Rightarrow t_2 \not< t_1$

iii. given $t_1, t_2, t_3, \in T$ and $t_1 < t_2$ and $t_2 < t_3$, then $t_1 < t_3$.

The set T can be either discrete or continuous.

The input alphabet is simply any set A, here assumed to be a finite set. The elements of A correspond to the discrete pieces of information which are provided the operator or which are used by him. More correctly, in any modelling application the elements of A are the names of equivalence classes of information provided the operator. Each class generally contains many distinct information patterns, which are all deemed equivalent for purposes of modelling.

The output alphabet is some finite set B. Again, the elements of B are in some sense the names of the distinct decision alternatives which are available for use by the operator.

Input and output sequences can now be defined. Let

$$A^T \triangleq \{x \mid x: T \to A\} \tag{1a}$$

$$B^T \triangleq \{y \mid y: T \to A\} \tag{1b}$$

A^T then is the set of all functions with domain T, codomain A. Any input sequence defined on the time set T and taking values in the input alphabet A must be an element of A^T. Similarly, any output sequence must be an element of B^T. In any discrete control experiment an operator is presented with some input sequence, say x_1, $x_1 \in A^T$, and he or she in the process of generating a response constructs an output sequence corresponding to some specific element of B^T, say y_1. The pair $(x_1, y_1) \in A^T \times B^T$ is the input-output data corresponding to the specific trial in question. It should be noted that the input function (i.e., x_1) need not be totally prespecified, but rather it can be generated as a function of the events which take place during the course of the trial. It might, for example, be a function of tracking error.

A behavioral representation of a discrete controller (operator or operators) is then a system S,

$$S \subseteq A^T \times B^T \qquad (2)$$

That is, it is a relation consisting of the pairs of input-output behaviors which are possible. This relation can be thought of as the sample space from which all data items in a discrete control experiment are selected.

Before addressing the modelling issues in detail, a few points about the input and output alphabets must be made. The only constraints defined so far are that A and B must both be finite sets. This does not mean however that these sets are first order. They can for example be relations of the form

$$A \subseteq A_1 \times A_2 \times A_3 \times \ldots \times A_n$$

where each set A_i is a finite set. In this case each set A_i might be the alphabet of a specific display or information channel. The same observation applies to the output alphabet B.

The relation S defined in abstract terms in equation (2) serves as the starting point for any discrete control modelling problem. It is the foundation of any data base constructed by experiment and any model is constructed to explain in more detail the specific elements contained in it. In any stochastic representation S serves as the sample space in the strictest technical sense. Since stochastic representations are required for any realistic discrete control modelling a few of the requirements are briefly reviewed.

A probability space is the basic structure used to represent any stochastic system. In essence, a probability space consists of a sample space, a sigma algebra, and a probability measure. Let S denote the sample space, Ω the sigma algebra and P the probability measure. Technically, Ω is a set of subsets of S which is closed under intersection and union. For our purposes there is little harm done if Ω is assumed to be the power set (the set of all subsets of S). The probability measure P is a function,

$$P: \Omega \rightarrow [0, 1]$$

where

$$[0, 1] \triangleq \{ r \mid r \text{ a real number}, 0 \leq r \leq 1 \}$$

The conditions imposed on P are widely known and available in any good probability or stochastic processes text (see, for example, Cinlar, 1975). Essentially, if $E \in \Omega$, (then $E \subseteq S$), P(E) is the probability associated with the set E (usually called the "event" E).

The probability space (S, Ω, P) is the conceptual structure which any model of such a stochastic system is attempting to describe or explain. This is important here only because it clearly defines where the sources of any probabilistic information must be assumed to reside. The elements of Ω are sets of input function, output function pairs and the basic probabilistic information is really joint information, information about the occurrence of input-output pairs. In modelling the interest is usually conditional probabilities, i.e., the probability that output function y is observed given input function x, or the probability that the output function y is an element in some specified set given the input function x. These conditional probabilities must however follow from the probability space as defined above.

The next task, addressed in the next section, is concerned with the problem of defining sufficient structure so that more than a pure behavioral description of discrete control performance can be provided. The specific task is to introduce the idea of state decomposition as it is used to provide local information about output response given past and present input data.

B. State Representations:

The relation S, equation (2), is a behaviorial representation. It does not provide any explanation or description of how or why a given output function y results given that the input function was x. A state representation is the usual means of providing such information, at least in a technical way. Some general properties of state representations which are needed to fully understand the discrete control structure developed later in the report are presented in this section.

To simplify notation, let

$$X = A^T \tag{3a}$$

$$Y = B^T \tag{3b}$$

and

$$S \subseteq X \times Y \tag{3c}$$

Another set, the state space, must now be introduced. Let C denote the state space and let

$$Z = C^T = \{ z \mid z: T \to C \} \tag{4}$$

The set Z is the set of state trajectories which can be defined on the time set T with state space C. If C is in fact a state space for the system S, it is possible to decompose S using two (2) relations:

$$S_1 \subseteq X \times Z \tag{5a}$$

$$S_2 \subseteq Z \times Y \tag{5b}$$

such that

$$S = S_2 \circ S_1 \qquad (6)$$

where o denotes set composition. It has been shown (Windeknecht, 1971) that, formally at least, such decompositions are always possible. (They are in fact trivially possible). Furthermore S_2 can be constrained to be a function.

$$S_2\colon\ Z \rightarrow Y \qquad (7)$$

This is important for certain theoretical purposes which are discussed shortly.

Given a decomposition of the form shown in (5a) and (7), and given a probability space

$$(S,\ \Omega\ P), \qquad (8)$$

a second probability space

$$(S_1, \Omega_1,\ P_1) \qquad (9)$$

can be derived. This space defines input-state probabilities given the above mentioned decomposition.

To construct (9), first consider the relation

$$f \subseteq S \times S_2 \qquad (10)$$

defined such that

$$((x, y), (x, z)) \in f \Leftrightarrow (z, y) \in S_2$$

Clearly, f need not be a function but f^{-1} always will be given that S_2 is a function. Now, given an event $E \subseteq S$, an event $E_1 | E \subseteq S_1$ can be constructed using f.

Specifically, define

$$f\big|E \stackrel{\Delta}{=} \{((x, y), (x, z)) \big| (x, y) \in E\} \subseteq f.$$

Now, $E_1 \big| E$ is the second projection of $f\big|E$, i.e.

$$E_1\big|E = \{(x, z) \big| \exists (x, y) \ni ((x, y), (x, z)) \in f\big|E\}$$

The probability measure used in the second measure, (9), must then be defined so that

$$P_1 (E_1\big| E) = P(E) \tag{11}$$

The above sketch is by no means a formal proof, but the important point is illustrated. The probability space which must be associated with the input-state component of any given state decomposition follows deductively from the basic probability space defined on the system. The main part of any modelling activity then must be focused on finding an adequate state space C. A major portion of this task is to provide a state space which is sensible and which provides useful substantive information.

Up to this point no assumptions have been made about the structure of the state space C. Abstractly, a state should in some sense summarize the past history of the system or process in sufficient detail that the desired information about future behaviors can be obtained from knowledge of the state and future inputs. These observations provide some insight into the essential structure which must be exhibited by the state space of a discrete control system.

Since the input alphabet and output alphabet are both presumed finite sets, the discrete control system can best be thought of in event terms. That is, both inputs and outputs remain constant for periods of time changing only at isolated points. This suggests that state trajectories might also exhibit the same behavior, although this need not necessarily be the case.

To start the development assume that an operation +,

$$+: T \times T \to T$$

is defined on the time set and that the set T together with the operation + form an abelian monoid (Bobrow and Aribib, 1974). Furthermore, it is assumed that for any two elements $t_1, t_2 \in T$, $t_1 < t_2$, there exists an element $s \in T$ such that

$$t_2 = t_1 + s.$$

This clearly is true for most sets used as time sets. This structure enables discussion of time intervals.

Now, let t_1, $s \in T$ and let

$$Tt_{1s} \triangleq \{t \mid t \in T,\ t_1 < t,\ \ t < t_1 + s\} \quad \cup \quad \{t_1 + s\} \tag{12}$$

Tt_{1s} is that subset of T containing all points greater than t_1 but not greater than $t_1 + s$. The task now is to describe the general state transition properties of event based systems.

Assume that the set C is a finite set. Any state trajectory z,

$$z: T \to C$$

then stands in one-one correspondence with a sequence

$$(t_0, c_0),\ \ (t_1, c_1),\ \ (t_2, c_2)\ldots$$

where

$$z(t) = c_0, \quad t \in Tt_{0\,s0}$$

$$z(t) = c_1, \quad t \in Tt_{1\,s1}, \; t_1 = t_0 + s_0$$

$$z(t) = c_2, \quad t \in Tt_{2s2}, \; t_2 = t_1 + s_1$$

In otherwords, the trajectory can be constructed by concatenating together restricted functions which are constant over the interval of their definition. Similar statements apply to input trajectories x given that the input alphabet A is finite. An input-state pair (x, z) therefore is representable in event form and this form can be used to express the properties required of a state transition function.

If at some point in time one examines a state trajectory graph, three pieces of information are available:

1. the current state occupied,
2. the time at which the occupancy started, and
3. the time since that event.

These three items in some sense form the abstract state of an event based system. The system is said to be time homogeneous if the time of the last state change event, item number two above, need not be included for any probabilistic analysis. The time homogeneous case will be assumed here.

Consider the hypothetical input and state trajectories shown in Figure 2.2. Assuming time homogeneity, the information contained on these graphs is summarized by the two sequences:

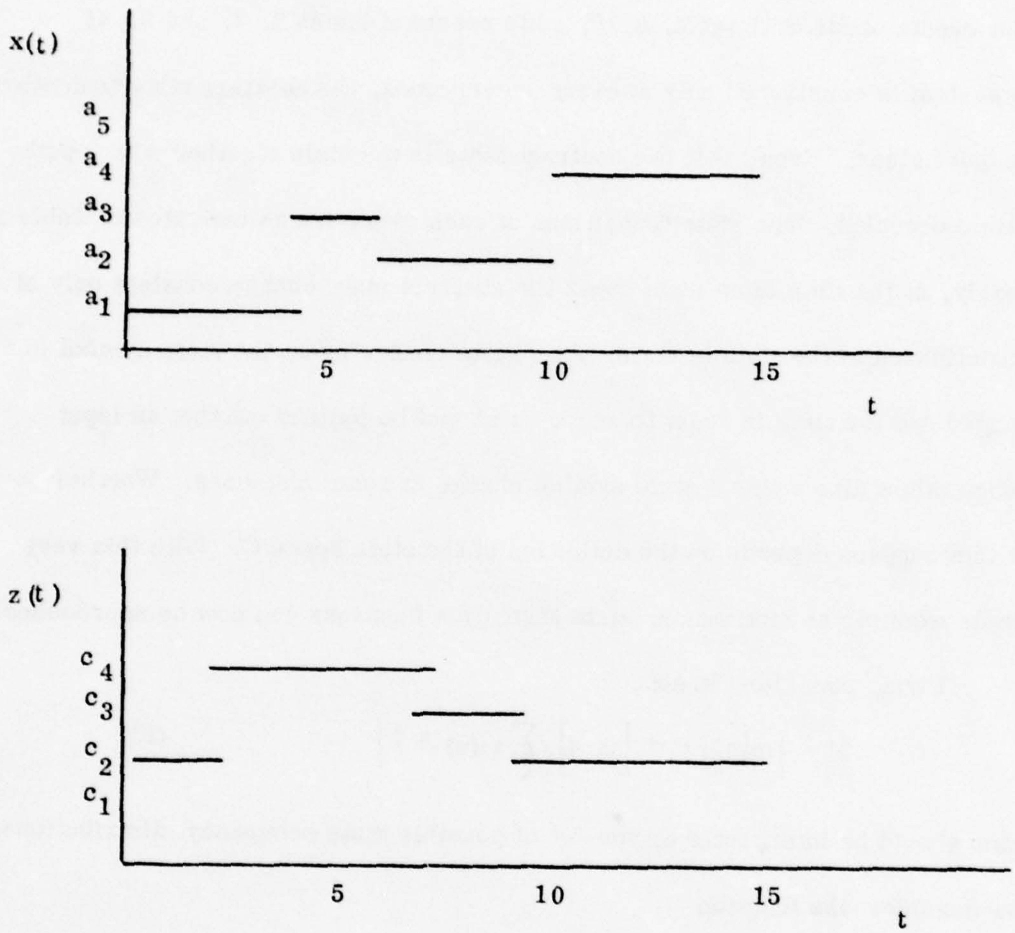


Figure 2. 2. --Hypothetical Input and State Trajectories

$(a_1, 0), (a_3, 4), (a_2, 6), (a_4, 10)$

$(c_2, 0), (c_4, 2), (c_3, 7), (c_2, 9)$

Input events occur at times 4, 6, 10; state events at times 2, 7, and 9. If the system is considered only at event occurrences, the abstract state transitions are quite clear. Recall that the abstract state is th state together with length of time occupied. The state transitions at each event are as indicated in Table 2.1. Clearly, at the time of an input event the abstract state change consists only of a redefinition of the time in state. At a state change event the state symbol is changed and the time is reset to zero. It should be pointed out that an input change might also cause a state symbol change in some instances. Whether or not this happens depends on the definition of the state space C. With this very simple example as motivation, state transition functions can now be approached.

First, consider the set

$$M = \left\{ m \middle| m : C \rightarrow \left[0, 1\right], \sum_{C} m(c) = 1 \right\} \tag{13}$$

which should be interpreted as the set of possible state occupancy distributions. Now consider the function

$$\Phi: \; C \times T \times A \times T \rightarrow M \tag{14}$$

This function is a transition function of a very special kind. Specifically,

$$\Phi\,(c, t, a, s) \in M$$

defines a conditional state occupancy distribution with the conditioning provided by the following information:

Table 2.1

State Transitions From Figure 2.2

Time of Event	State at Last Event	State At This Event	State After This Event
2	$(c_2, 0)$	$(c_2, 2)$	$(c_4, 0)$
4	$(c_4, 0)$	$(c_4, 2)$	$(c_4, 2)$
6	$(c_4, 2)$	$(c_4, 4)$	$(c_4, 4)$
7	$(c_4, 4)$	$(c_4, 5)$	$(c_3, 0)$
9	$(c_3, 0)$	$(c_3, 2)$	$(c_2, 0)$
10	$(c_2, 0)$	$(c_2, 1)$	$(c_2, 1)$

1. the abstract state at the last event, (c, t),

2. the time of the next state event is within s units of the last event*, and

3. the input over the interval from the last event to the next state event is fixed at a.

The function Φ then establishes the probability that a state will be occupied within a specified number of units given the indicated information.

An input event transition function must also be defined,

$$\Psi : C \times T \times A \times T \rightarrow C \times T \tag{15}$$

this function must establish the state upon the occurrence of an input event.

$$(c_2, t_2) = \Psi(c_1, t_1, a, s)$$

is the abstract state after the event given that the abstract state at the previous event was (c_1, t_1), the time since that event was s and the new input is a. In cases where the input does not directly change the state, $c_2 = c_1$, $t_2 = t_1 + s$. In cases where c_2 is not the same as c_1, $t_2 = 0$ with 0 denoting the identity of the monoid.

The last function needed is an output assignment function

$$\lambda : C \rightarrow B \tag{16}$$

which simply assigns the appropriate output symbol given the state symbol. In general, the cardinality of C exceeds that of B which implies that output events in general do not occur at each state event.

---

*M(c) where c was the last state occupied, is the probability of no event, or a transition to the same state.

Expression (14), the transition function, is the key structure in analyzing discrete control and some important observations can be based upon it. First, consider the case in which the state occupancy distribution depends only on the state and not the abstract state. That is,

$$\Phi: C \times A \times T \rightarrow M. \qquad (17)$$

In this case, state occupancy probabilities depend on the time since the last event, not the last state event. This structure does however still have some time dependency inherent in it. A further simplification results if the next state occupancy distribution is totally independent of time, i.e.,

$$\Phi: C \times A \rightarrow M \qquad (18)$$

This is the stochastic automata case (Paz, 1971) and it also is the case in which the time between state change events is exponentially distributed. Which of the various forms that applies in a given problem is in part an empirical question and such issues are addressed in Chapter 5.

In summary, the problem of establishing a state decomposition is one of finding a state space C with the correct properties. A state space is acceptable if a transition function (14), input transition function (15) and an output transition function (16) all exist.

In the next section some network methods are introduced. These methods are particularly suitable for constructing state representations of discrete control.

C. Networks of Finite State Systems

The concept of state which was discussed in the previous section was a highly abstract one and for any complex system requires some additional refinement. In this section, some of the properties of a system constructed from a network of such finite systems is discussed.

Consider the simple diagram shown in Figure 2.3. Each system in the network is a finite state system represented by the usual objects: an input alphabet, output alphabet, state space, transition function and output assignment function. In other words each system in the network inherits all of the properties discussed in the two previous sections. A few special conditions which are discussed below must however be imposed.

At one level of abstraction the network is simply a directed graph with nodes consisting of the systems and arcs consisting of communication links. Formally, let

$$N = \{ i \mid i \text{ a system name} \}$$

and let

$$G \subseteq N \times N \ni (i, j) \in G$$

if and only if system j is connected to system i via a link from i to j. In the example of Figure 2.3,

$$G = \{(1, 2), (1, 3), (2, 3), (2, 4), (3, 5), (3, 6)\}$$

The graph G captures the essence of the communication paths, but there is still some room for ambiguity. For example, system 1 in Figure 2.3

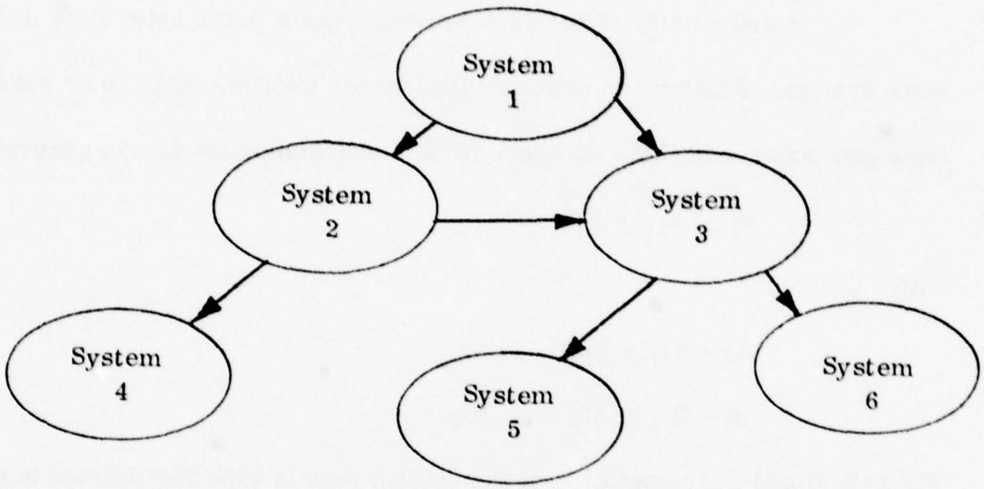


Figure 2.3--Network of Systems

communicates with both system 2 and system 3. Clearly, different information could be sent to each of these. The ambiguity is best resolved by providing a slightly less abstract description of each system.

As stated earlier in the section each system in the network is a finite state system. Further, in order to account for multiple paths in or out the input and output alphabets will have to be multi-dimensionial. In general then,

$$S_i \subseteq A_i^T \times B_i^T$$

with

$$Ai = Ai_1 \times Ai_2 \times \ldots Ai_k$$

$$Bi = Bi_1 \times Bi_2 \times \ldots Bi_e$$

The behavioral representation of the system then is explicity defined in terms of a finite number of input and output channels.

Now, let I denote the set of positive integers and let

$$L: G \rightarrow I \times I$$

be defined so that

$$((i, j), (k, l)) \in L \Leftrightarrow$$

output k of system i is connected to input l of system j. The labelling function L then defines the explicit interconnections which are made and it provides another structural piece of information as well. Specifically, the output alphabets and input alphabets of the interconnected systems must match. That is, if

$$(k, l) = L(i, j)$$

then

$$Bi_k = Aj_l$$

Or in other words, only systems which can communicate are interconnected.

There are a few more conditions dealing with systems that communicate in only one direction which must be addressed. First some more notation. Let

$$G^1 = \left\{ i \,\middle|\, \exists j \ni (i, j) \in G \right\}$$

$$G^2 = \left\{ j \,\middle|\, \exists i \ni (i, j) \in G \right\}$$

These are the source and receiving nodes, respectively. Now, let

$$N_I = \left\{ i \,\middle|\, i \in N,\ i \notin G^2 \right\}$$

$$N_0 = \left\{ i \,\middle|\, i \in N,\ i \notin G^1 \right\}$$

N_I consists of those nodes which do not receive information from any system and N_0 is those which do not send to any system. All systems in N_I are constrained to have no input alphabet, i.e.,

$$j \in N_I \Rightarrow S_j \subseteq B_j^{\ T}.$$

The systems in N_I then in some sense represent the basic information input to the system. The outputs of systems in N_0 on the other hand define the overall system outputs.

Suppose that

$$N_I = \{i_1, i_2, \ldots I_k\}$$

and

$$N_0 = \{ j_1, j_2, \ldots j_e \}$$

Then, from the argument above

$$S \subseteq (Bi_1 \times Bi_2 \times \ldots Bi_k)^T \times (Bj_1 \times Bj_2 \times \ldots Bj_e)^T \qquad (19)$$

forms the behavioral representation of the overall system.

The above define the basic technical constraints which must be met if a network of finite state systems is to be logically consistent. The main reason for using a network of simple systems to construct a more complex system is the simplicity of the state representation which results. In abstract terms the relation (19) is the input-output (behavioral) representation of the system which was the focus of discussion in the previous section. That is, if

$$A = Bi_1 \times Bi_2 \times \ldots \times Bi_k$$

$$B = Bj_1 \times Bj_2 \times \ldots \times Bj_e,$$

the relation

$$S \subseteq A^T \times B^T$$

is the input-output relation and hence the sample space required for a stochestic representation of the system. But, given the network, the state space of S is quite clear. Let

$$Ci, \quad i \in N$$

denote the state space of system i. The state space of S then is the cartesian product of the state spaces of these component system

$$C = \underset{i \in N}{X} \; Ci$$

It should be noted that the above state space includes the state space of input systems as well as internal systems. In some cases it may be desirable to exclude such systems (i. e., those in N_I) when constructing C.

In addition to the state space C, the state transition function and output assignment function required to represent S also follows from the functions associated with the individual elements in the network. Each system in the network is governed by a transition function and any change in state is communicated to the appropriate systems via the communication links defined by the graph G. The system transition function then is really a fairly complex function which is in some sense the product of the appropriate individual functions. But, the important point is that each individual element is quite simple and the rules for constructing the overall state transition function are straight forward. In other words by knowing the properties of the component systems and the road map which defines their interconnection, the system properties immediately follow.

Another advantage, closely related to the above, is the complexity reduction afforded by the network. For example, suppose the system shown in Figure 2.3 has state spaces with the following cardinalities:

System 1	3
System 2	2
System 3	3
System 4	4
System 5	2
System 6	2

The complete system state space thus has $3 \cdot 2 \cdot 3 \cdot 4 \cdot 2 \cdot 2 = 288$ states. Now, if the probabilistic automaton construction is used there are $(288)^2 = 82,944$ transition probabilities required to specify the system. Assuming that the state is the output for each system, the network construction of Figure 2.3 requires only

$$(3 \cdot 3) + 3 \cdot (2 \cdot 2) + 3 \cdot 2 \cdot (3 \cdot 3) + 2 \cdot (4 \cdot 4) + 3 \cdot (2 \cdot 2)$$
$$+ 3 \cdot (2 \cdot 2) = 131$$

transition probabilities which obviously is a very substantial saving. Note that this total is obtained by summing the number of transition probabilities required to describe each system (i.e., the number of states squared) multiplied by the number of distinct inputs which the system can receive. For example, system three has three states and receives inputs from system one which has three states and from system two which has two. The number of inputs is therefore $3 \cdot 2 = 6$ and the number of state transitions which is possible for any given input is $3 \cdot 3 = 9$ which together give 54 transition probabilities.

In addition to the reduction of combinational complexity in state transition probability estimation, the network representation has a distinct substantive advantage. The primary reason for constructing the model in the first place is to help explain how operators perform discrete control tasks. The network is in essence a representation of the intelligence which might be brought to bear on the problem. It is one way of representing a complex problem in a manageable form. Each system in the network represents some important activity or subsystem which the operator must control. By identifying what these component systems are and how they interrelate, the discrete control model is constructed, and network representation follows as a natural by product.

Several simplifications were used in the analysis of the antiaircraft artillery system which is discussed in the remaining chapters of the report. First, all output assignment functions were assumed to be of the form

$$\lambda_i : C_i \to C_i^k$$

where C_i^k is the kth order cartesian product of C_i. The function λ_i simply produced multiple copies of the state, one for each output channel. It is also clear from the structure of λ_i that output alphabets and state spaces were assumed to be identical. In summary then, component system states were communicated to the required systems and no simplification was provided through use of an output assignment function. This simplifies programming for data analysis, but it does create some data analysis problems which will be discussed later.

D. Some Comments About Modelling Strategy

The main effort in discrete control modelling is spent in constructing the network. Once the network is obtained data analysis and similar problems can proceed in a fairly mechanical way, but the analysis must start with the network and the success or failure of the modelling effort depends to some degree on the care which goes into the specification of the structure. A few comments about the overall process of discrete control modelling are provided below.

The first step is to determine all of the discrete outputs which the system is required to specify. These normally are the specific decision alternatives which the operators can select from and typically include items like switch settings and other discrete status indicators. Such items generally can be obtained from a detailed analysis of the system which the operators control. In some cases it may not be necessary or desirable to work at the level of individual switches in which case the analyst must define the proper level and specify in unambiguous terms exactly what the output primitives are to be. The individual items identified in this phase of the analysis determine the system output alphabet.

The second step is to identify the exogenous input variables which in some sense drive the system. These might include things like target trajectories or command information from other systems. Some of this information will probably be in the form of continuous variables in which case rules for interpreting such data in events format must be defined. This step corresponds to some type of feature extraction through which the essential information classes are extracted from the data. For example, targets might be classed as maneuvering or nonmaneuvering as a function of their time behavior. In essence the task is to abstract out a small number of information classes which can then be used for discrete control analysis.

At the same time other nonexogenous continuous information such as tracking errors must be represented in events format. There are no preset procedures for accomplishing this but rules specifically designed to match the problem context must be defined. This in general isn't too difficult to accomplish.

The next step requires that the elements to be used in the network be defined. It is important to note that, for purposes of data analysis, the state of any system defined must be computable from available information. That is, data analysis can not proceed if the state of one or more systems in the network cannot be uniquely specified. With this constraint in mind, the process of defining the required or desired systems proceeds in several states which are often patterned after a level of abstraction hierarchy. First level systems (components) are one level of abstraction away from the primitive data items and they consist of fairly independent subsystems. These can be established on functional grounds or for purposes of forming aggregate information about the primitives. Second level systems are formed in a similar manner from the primitives and the first level systems. These can be formed to provide coordination of the lower level activities and functions, or they can again simply be an aggregation. This procedure of subsystem definition continues until no further systems are needed. The key point is that the states of all systems at each level must be determined from simple logical operations on systems previously defined. Once the component systems have been defined the graph G described in section C must be defined. This is probably best accomplished in two steps because two kinds of information generally flow through the system network. In some cases the state of a given system is said to directly constrain the states which another system can occupy whereas in other situations the information that flows to a system only influences the decisions which specify the state. The two networks should be considered separately. The graph of the first type, the constraint network, should be constructed and the constraints themselves identified. After this has been accomplished the various systems which are decision loci are clearly identifiable and the decision influence network can be established.

It should be noted that in general the decision influence or conditioning network is quite speculative. One can conjecture as to what information might be needed to make a specific decision, but the final network usually must wait for experimentation with the available data.

When the above steps have been completed the model structure is defined and data analysis can proceed. Analysis also requires two major stages. Preprocessing transforms the available data into an events data base suitable for use in the analysis programs. It may be desirable to use two phases of preprocessing. The first puts the primitive data into events form and the second computes the component system states (i. e., systems used in the network mentioned previously) from the primitive events data. The component systems state information is also represented in events format. If the network is fixed in terms of the component systems and their state definition rules, then the two phases mentioned above can be combined. Clearly, preprocessing must be problem specific and the programs necessary for its accomplishment must be written for each problem.

The analysis phase forms estimates of the state transition functions of the component systems. The form assumed is that of equation (17), although (18) can also be used. As mentioned above, all output alphabets are assumed to be the same as the corresponding state space. This is a simplification which can be very easily modified. The analysis program produces the following as output: transition probability estimates, time in state information including, as an option, several time in state histograms. The analysis program details are summarized in Appendix 5.

Understanding discrete control modelling probably requires an example. The remainder of this report consists of such an example in the form of the discrete control II experiment. A description of the experiment and the data base is provided in the next chapter. Modelling and results then follow.

CHAPTER 3

THE DISCRETE CONTROL II EXPERIMENT

Discrete control II refers to an experiment conducted by personnel at the Aerospace Medical Research Laboratory, Wright Patterson Air Force Base to support in part the research reported here. AMRL had primary responsibility for the design and execution of the experiment, but the basic data requirements were defined by the needs of this research.

The design of the experiment was based in part upon knowledge and insights gained from the pilot study performed during 1976-1977 and reported in the annual report of work funded through Grant No. AFOSR-77-3152, (Miller, 1977). Discrete control II was a much more extensive and ambitious experimental exercise and has generally provided a much more comprehensive and usable data base.

A brief review of the experiment and a detailed discussion of the modelling and analysis are presented in this section. More detailed descriptions of the hardware and the experiment can be obtained from AMRL.

A. Description of the System

The system which served as the focus of the study was a man-in-the-loop simulation of an anti-aircraft artillery (AAA) installation. This system consisted of a mock-up of the operators' consoles, including the major controls, switches, and displays; plus the computing equipment required to drive the displays, record data and generally simulate the AAA system and its environment. This particular simulator required a three person team consisting of a range tracker, an angle tracker and a commander. A block diagram illustrating the relationships between major system components is given in figure 3.1. Basically the system consists of two optical sighting systems (left optics, right optics), a radar system with separate displays for the angle operator and range operator, a gun servo system which positions the guns as a function of tracking commands, and a lead angle computer. There are also a variety of switches and controls devices used to control and coordinate the activities of the system. A list of controls and dispiays is given in Table 3.1 through 3.3.

No one would claim that the simulator was a realistic representation of an AAA installation, but it did have a number of unique and interesting features. First, targets could be found and tracked using either optical information (presented via a television system) or a simulated radar return. Second, the tracking and fire control systems could be operated in a variety modes selectable by the team of operators. These will be discussed more fully later in the report. Third, simulated tracer feedback information was available through the optical display system.

Some of the major limitations (in terms of discrete control modelling at least) were the following. First, only one target could be displayed at any time so that the subjects could not be required to pick the most important target from several available. Second, the simulator was fixed based so that certain operating modes which would not generally be used in a real system

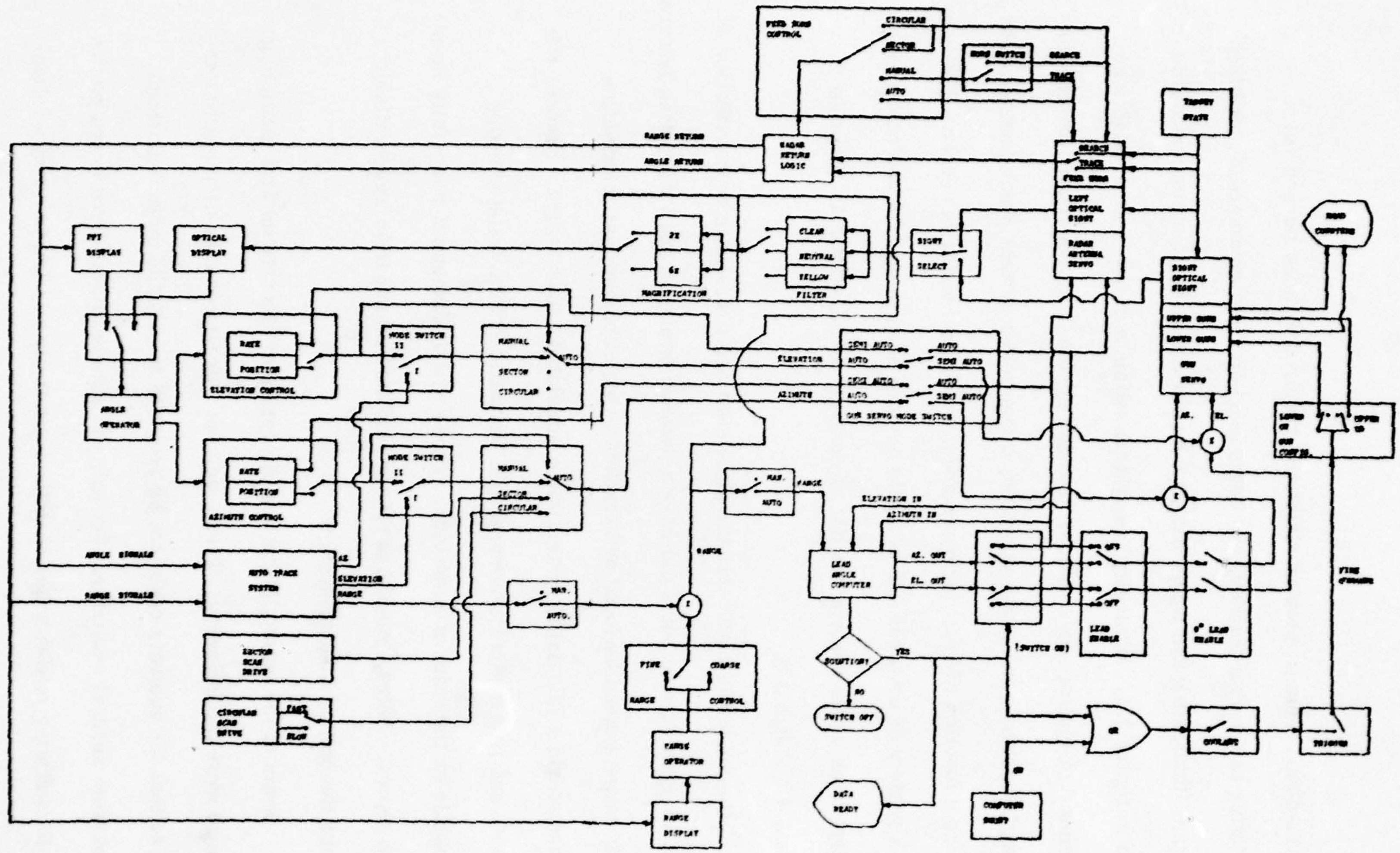


Figure 3.1.-- System Diagram

Table 3.1

Commander's Controls and Displays

Item	Function
Ammunition Counters	Read out of Ammunition Remaining
Firing Lamp	Indicates Trigger Depressed
Data Ready Lamp	Indicates Fire Control Computer has Reached a Solution
Upper Gun Switch	Enables Upper Guns To Fire
Lower Gun Switch	Enables Lower Guns To Fire
Zero Degree Lead Enable Switch	Enables Tracking Without Fire Control Solution
Lead Enable Switch	Enables Fire Control Computer To Control Lead Angle
Computer Shunt	Allows Guns To Be Fired Without Fire Control Computer Solution

Table 3.2

Angle Operator's Controls and Displays

Item	Function
Magnification	Selects 2x or 6x magnification for optical sighting system
Filters	Selects one of three filters for optical sighting system
Optics Left/Right	Used to select left or right sight
Trigger	Used to fire guns
Data Lamp	Indicates fire control computer solution
Elevation Indicator	Indicates antenna elevation
PPI Scope	Displays radar azimuth and elevation return
15/20 Switch	Selects 0-15 or 5-20 slant range
Circular Sweep Switch	Selects Circular Sweep Search Mode
Sector Switch	Selects Sector Search Mode
Fast/Slow Switch	Selects Fast or Slow Circular Sweep

Table 3.2 (Continued)

Item	Function
Sector Width	Control Sector Width During Sector Search
Azimuth Rate/Position	Determines Azimuth Tracking Mode
Elevation Rate/Position	Determines Elevation Tracking Mode
Coolant	Activates Coolant Circuit
Auto Track Switch	Engages Auto Track Mode
Manual Track	Reverts To Full Manual Track
Mode I/II	Selects Range Only or Full Auto Track
Track/Search	Selects Radar Beam Width
Semi-Auto/Auto	Auto: Mode I or II Semi Auto: Manual Control of Gun Motion
Track Control Wheel/Yoke	Used to Track in Azimuth and Elevation

Table 3.3

Range Operator's Controls and Displays

Item	Function
A Scope	Basic Range Display
Range Gate Crank	Used to Track Target Range
Coarse/Fine Adjustment	Changes Crank Sensitivity

because of motion problems could be used in the simulator with no effect or penalty. The team's decision problems were therefore reduced in complexity in the sense that certain switches could be set and forgotten with no performance penalty. Third, there are physical constraints present in AAA systems which were not present in the simulator. For example, in reality gun barrels must be allowed to cool between bursts and a fairly rigorous firing doctrine must be followed. Such was not the case in the simulator since the gun barrels were really artifacts of a computer program.

Even though assumptions and simplifications made during the design of the simulator removed or constrained some of the decision problems faced by AAA system operators, this simulation still serves as a good data source for discrete control modelling. The main objective of the modelling effort is to understand some of the processes and procedures used by operators to accomplish their discrete control functions; i.e., to use the total range of controls available to achieve the proper system mode of operation at any point in time. The AAA simulation provides ample data to explore such issues.

B. Description of the Experiment

The experiment was a highly stylized simulation of the AAA task. Subject teams were required to search for, acquire, and track simulated targets and to try to maximize the hit score attained on each target. The teams were free to select any system mode of operation at any point in time. As mentioned previously, each team consisted of three team members: an angle operator, range operator and commander. Team members specialized in the sense that they played the same role throughout the experiment.

Teams were asked to perform simulated "missions" which consisted of a sequence of 23 targets which they were required to acquire and track. Targets were presented one at a time and the target trajectory ran its full course before a new target was introduced. There was also a period of time between targets during which no target was present. The subject teams however remained actively involved in the search for the next target during this interval. Subject teams were given the approximate coordinates of the next target to maximize the probability of detection. This information simulated the role of an early warning system.

Two of the three subject teams completed a total of 23 missions and the third team completed 22. Therefore, two teams tracked (or attempted to track) 529 targets each and the third team attempted 506.

Four distinct trajectories were used. These were:

1. 2 X 2 Fly by
2. 0 X 1.6 Fly by
3. Fair Pass
4. S Pass

Sketches of these trajectories are given in Figure 3.2. The first was considered a simple "non-threat" type of trajectory, the other three were considered "threat" trajectories by the experimenters.

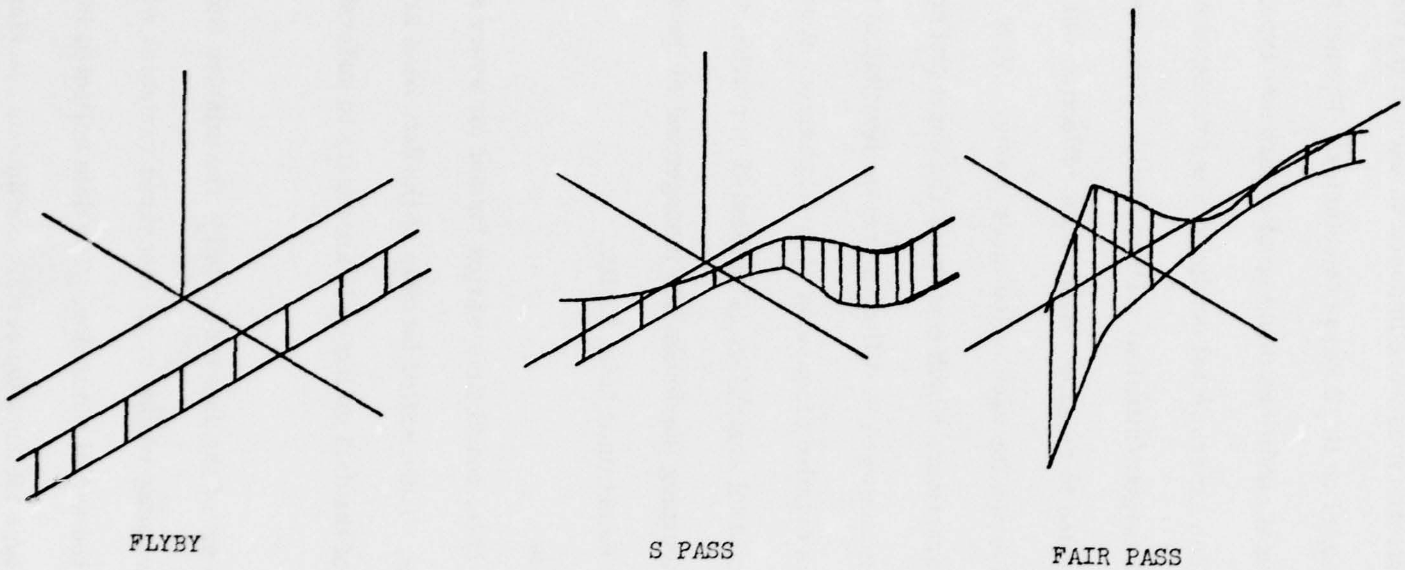


Figure 3.2.--Trajectory Sketches

Six disturbance pattern types were imposed on each of the first three trajectories and five were imposed on the S Pass trajectories. The net result was 23 distinct experimental conditions made up of disturbance type, trajectory combinations. Each experimental condition was presented once during each mission (hence 23 targets/mission). Presentation of the conditions was randomized from session to session. Also, data collection was grouped into blocks of four missions for each of the three teams. After each block was completed, the parameters which controlled the onset and duration of the disturbance conditions were modified to prevent learning of the disturbance patterns. A more detailed discussion of the experimental conditions, as they impact discrete control modelling, is presented in section 4. B. 1 of this report. Other details concerning the design and management of the discrete control II experiment must be obtained from AMRL.

C. The Data Base

Data was collected in time-series format for every simulated mission in the experiment. The detailed formats of the data base are not important here, but an overview of the data collected is necessary to understand the modelling which follows.

The time set of the data collection is the mission time set. This means that a complete running record of all measured variables was collected from the beginning to the end of a mission. The data collected included discrete status indicator type information,certain continuous tracking information, and header information to indicate teams and trajectories. All data was collected at a 30 HZ sampling rate.

The discrete data collected consisted of all switch settings, and status display states, plus certain variables intended to provide information about the activities and performance of the team. The primitive discrete data items are defined in Table 3.4.

It should be noted that items 1 through 11 and 13 through 20 are all physical switch settings or display states. Item 12, display in use, was obtained from a limit switch which was set whenever the angle operator was using the optical sighting system. Items 22 through 30 establish the type of disturbance applied, if any, and a rough indication of tracking performance and fire control performance.

In addition to the above mentioned discrete variables, the continuous variables listed in Table 3.5 were measured. The trajectory data was not recorded on each and every mission data tape. Rather, one trajectory tape was made and the header record of each engagement within a mission contained a flag which defined the trajectory which was used. This saved a substantial amount of storage space with no information loss.

The data base produced by this experiment is very large. It consists of approximately 6,913,500 samples of each of the 35 variables mentioned above, plus trajectory data. This is about 64 hours of tracking and discrete control performance data or about 21 hours of data from each team. The modelling

Table 3.4

List of Primitive Data Items

Component Name	States	Value Assigned
1. Antenna Horn	Search	1
	Track	0
2. Automatic Circular Scan	Fast	1
	Slow	0
3. Radar Mode	Circular Scan	11
	Sector Scan	10
	Manual	01
	Automatic	00
4. Gun Servo Mode	Semiautomatic	1
	Automatic	0
5. 0^{o} Lead Enable	On	1
	Off	0
6. Lead Enable	On	1
	Off	0
7. Mode Switch	II	1
	I	0
8. Computer Shunt	On	1
	Off	0
9. Data Ready Indicator	On	1
	Off	0
10. Coolant	On	1
	Off	0
11. Trigger	On	1
	Off	0
12. Display in Use	Optics	1
	Not Optics	0
13. Azimuth Tracking Control	Rate	1
	Position	0
14. Elevation Tracking Control	Rate	1
	Position	0

Table (Continued)

Component Name	States	Value Assigned
15. Range Tracking Control	Fine	1
	Coarse	0
16. Sight Selector	Left	1
	Right	0
17. Sight Magnification	6X	1
	2X	0
18. Sight Filter	Clear	00
	Neutral	01
	Yellow	10
19. Lower Barrel Enable	On	1
	Off	0
20. Upper Barrel Enable	On	1
	Off	0
21. Target Introduced	Yes	1
	No	0
22. Target on Display	Yes	1
	No	0
23. Azimuth Tracking Error	Locked	1
	Not Locked	0
24. Elevation Tracking Error	Locked	1
	Not Locked	0
25. Range Error	Locked	1
	Not Locked	0
26. Optics Disturbance	Blanked	1
	Not Blanked	0
27. PPI Disturbance	On	1
	Off	0
28. Range Disturbance	On	1
	Off	0
29. Range & PPI Blanking	On	1
	Off	0
30. Tracer Evaluation	In Window	1
	Not in Window	0

Table 3.5
Continuous Variables Measured

Variable Name	Unit of Measurement
1. Azimuth Tracking Error	Milliradians
2. Elevation Tracking Error	Milliradians
3. Range Tracking Error	Meters
4. Upper Counter Reading	# of Rounds
5. Lower Counter Reading	# of Rounds
6. Target Trajectory	
a. Azimuth	Degrees X 10^{-1}
b. Elevation	Degrees X 10^{-1}
c. Range	Meters
d. Roll	Degrees X 10^{-1}
e. Pitch	Degrees X 10^{-1}
f. Yaw	Degrees X 10^{-1}
g. X Co-ordinate	Meters
h. Y Co-ordinate	Meters
i. Z Co-ordinate	Meters

task is to provide an organized, systematic way of exploring this data and to provide some explanation of how the discrete control task was performed by the subject teams.

CHAPTER 4

A DISCRETE CONTROL MODEL OF AAA TEAM PERFORMANCE

In this chapter the model which was constructed to analyze the discrete control II data is defined and discussed. This material provides an example of the style of modelling used for discrete control and it illustrates some of the fundamental problems and advantages of the approach. Before addressing the specific modelling issues, an overview of the general approach is presented.

A. Overview of The Modelling Approach

The operators of the AAA system together must make decisions which determine which activities are to be engaged in and which mode of operation is to be used at each point in time. Their individual activities and decisions must be coordinated if the system performance level is to be maximized. Such coordination is achieved only if certain information flows through the system and each operator performs his tasks accurately and in a timely manner. The model of the AAA system used for analysis of the discrete control II data was carefully structured to capture the coordination and communication requirements as well as quantify individual task performance.

The model was constructed utilizing the general principles of discrete control modelling which were discussed in sections I and II of this report.

Inputs required and utilized for discrete control purposes are assumed to be event oriented. Coordination is achieved by passing information about the occurrence of events between system components. The complexity of the total system makes it necessary to decompose the system into a number of smaller, less complex systems which are responsible for certain specific tasks or serve as the information transfer points necessary for coordination. These smaller systems are placed in a network in which the systems themselves are the nodes and the arcs the communication links. The network serves as the model of the team of operators working together on the discrete control problems imposed by the AAA system.

A few words about the model are necessary. The network is probably best viewed as a related set of internal models. Certain parts of the network are best thought of as the internal models used by specific team members. But, other parts, particularly those which provide common communication points, are best thought of as models shared by two or more of the members. Through training and experience the team members learn the overall mission objectives and they learn what information must be shared, hence developing at some level a common representation of the problem and the system. Decisions made by the operators are presumed to be based on the state of these internal models. Furthermore, the decisions represent the desire to change the state of some component system thus enabling or disabling the occurrence of other events and decisions.

The specific systems used and some rationale for their construction are discussed in the next section.

B. System Decomposition

Several levels of analysis and types of decomposition are needed to construct a representation of a system as complex as the AAA system studied in the discrete control II experiments. The required decomposition takes place along three, not necessarily independent, dimensions:

1. Variable Type

   a. exogenous
   b. endogenous

2. Level of Analysis or Description

   a. primitive component
   b. major components
   c. functional systems
   d. coordination/communication systems
   e. management/command systems

3. System Type

   a. interface or information feedback
   b. decision controlled
   c. event controlled

Each type of decomposition will be defined and explained in some detail. It should be noted that the system structure which results was obtained through a careful analysis of the AAA system, its structure, its functions, and the degrees of freedom possible in its operation. This analysis was performed prior to the collection of any data and it represents what was thought to be a structure adequate to capture and explain the possible behaviors which could be exhibited by the system. As will be seen when the data analysis results are presented, the full range of possible behaviors were not exhibited by the subject teams in the discrete control II experiments and some of the systems defined play little or no role.

B.1 Decomposition by Variable Type

The exogenous variables of the system as defined for this simulation are those which are target related. Specifically, these include the trajectory number, the target position in space, and the disturbance state. Recall that the disturbance state refers to the simulated countermeasures which were used in the experiment.

The position of the target in space is actually a vector of continuous functions. For purposes of the discrete control analysis a simplified finite state representation was used. Target position was defined in terms of seven states:

1. no target;
2. target out of range;
3. target in range, high altitude;
4. target medium range, low altitude;
5. target close range, low altitude;
6. target medium range, medium altitude; and
7. target close range, medium altitude.

The rules used to determine the proper classification are explained in Appendix I. This seven state representation of the target is convenient for an event oriented description of system performance and provides sufficient detail for explaining discrete control behavior.

The disturbances used to simulate countermeasures were the following:

1. blanking of all optical displays;
2. blanking of all radar displays;
3. noise superimposed on the PPI display used by angle operator; and
4. noise superimposed on the A scope display used by the range operator.

A five state representation of the countermeasure condition was constructed. The disturbance system states and the rules of state assignment are also given Appendix 1.

The trajectory number, the target position system, and the disturbance system are the only exogenous variables. These systems are used to provide any target specific information used by other systems in the model and to explain any trajectory specific behavior. All other variables in the system are endogenous and characterize behaviors and decisions produced by the team members in response to the presented targets.

B.2 Decomposition by Level of Analysis and Description

The output of the discrete control model at any point in time is the state of each of the decision controlled discrete data items listed in Table 3.4, Section 3C. These switch and control settings are the primitives of the discrete control problem, the lowest level data available for use in analysis, and the lowest level which must be explained by the model. The explanation of such data however must incorporate knowledge of the AAA system including mission objectives, major tasks, major subsystems and major components. With such knowledge the various primitive data items can be grouped in a logical fashion and a structure which organizes the flow of information can be designed.

In this section a hierarchical system involving five levels is constructed. This hierarchy is based on the above mentioned knowledge and it establishes the elements of the discrete control model of the AAA system. The levels are shown in Figure 4.1. As one moves from the bottom to the top of this hierarchy the view of the AAA system becomes more global and systemic. At the bottom the perspective is that of individual controls and switches, at the top the perspective is that of overall mission objectives. Upper levels define abstract, less detailed views of the system; lower levels fill in the details. By moving from top to bottom any question about system performance and operation can be answered. Each level of the hierarchy is now described starting with the primitive component level.

Level 1, the primitive component level consists of twenty simple systems which correspond to the basic switches and controls used during the discrete control II experiments. The primitive components are listed in Table 4.1. These items are the decision or event controlled elements in the primitive data lists shown in Table 3.4. The state of each component listed in Table 4.1 must be established by some system higher in the system hierarchy.

The system component level contains two types of systems: distinct system components such as the lead angle computer, and pseudo components

Figure 4.1--System Hierarchy

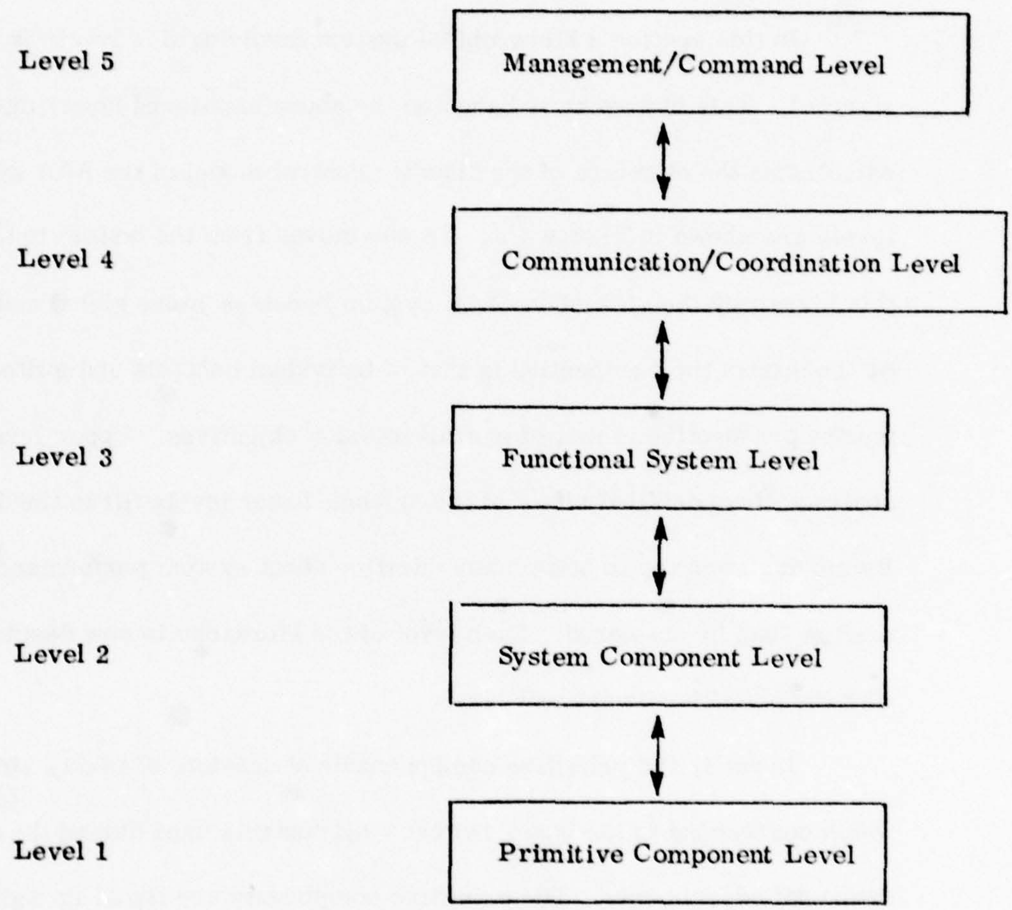


Table 4.1

Primitive Components

Component Name	States
Antenna Horn	Search, Track
Automatic Circular Scan	Fast, Slow
Radar Mode	Circular Scan, Sector Scan, Manual, Automatic
Gun Servo Mode	Semi-Auto, Automatic
0^{o} Lead Enable	On, Off
Lead Enable	On, Off
Mode Switch	Mode I, Mode II
Computer Shunt	On, Off
Data Ready Indicator	On, Off
Coolant	On, Off
Trigger	On, Off
Display in Use	Optics, Not Optics
Azimuth Tracking Control	Rate, Position
Elevation Tracking Control	Rate, Position
Range Tracking Control	Fine, Coarse
Sight Selector	Left, Right
Sight Magnification	2x, 6x
Sight Filter	Clear, Neutral, Yellow
Lower Barrels	On, Off
Upper Barrels	On, Off

defined by grouping certain primitive components which are manipulated together. The level two components are listed in Table 4.2 and the state definitions and state assignment rules are given in Appendix 2.

The sight system establishes the physical configuration of the optical sighting mechanism. The sight selector system is considered separately because of its importance in certain modes of operation (to be discussed when level four systems are discussed).

The range control component is obvious. The gun configuration system defines precisely which barrels are enabled at any point in time and hence determines the maximum rate of fire possible. The angle track controls component establishes the exact configuration of azimuth and elevation controls.

The gun servo enabling network is a pseudo component. The state of this component establishes how tracking information flows through the system to the input of the gun drive mechanisms. Such flow is disabled if the gun servo enabling network is in State 1, it is fully enabled if it is in State 4 and the lead angle computer solution, if obtained, drives the guns. If the gun servo enabling network is in State 5, other components determine the flow of information.

The radar antenna drive component defines the physical mode of operation of the mechanism which controls the motion of the antenna and the radar beam characteristics. State 1 is auto track which means that the track or narrow beam is in use and that the range signals are under automatic control. The

Table 4.2

List of System Components

System #	Component Name
1	Sight System
2	Sight Selector
3	Range Control
4	Gun Configuration
5	Angle Track Controls
6	Gun Servo Enabling Network
7	Radar Antenna Drive
8	Computer

angle signals may also be under automatic control depending on the state of upper level systems. State 2 is manual tracking and states 3 through 6 are the various search modes which the team may use.

The computer is a physical component with three states of interest: standby, settling, operating. Settling refers to the period of time after which the computer is put in use but before a solution is reached. Operating refers to the period during which a lead angle solution is available.

The AAA system has other components, for example the A scope display used by the range operator, but none of these additional components have more than one mode of operation which is of operational significance. The components described above are precisely those which can potentially be used in multiple ways and which reflect the decision making activity of the team members.

Level 3 in the hierarchy is used to abstract out five major systems which perform the several functions which are prerequisite if the system is to meet engagement and mission objectives. These five are listed in Table 4.3. The state assignment rules for these systems, with the exception of the range tracking system, are defined in Appendix 2.

The states of the fire control network are defined to be locked, data enabled, fire enabled. When the fire control system is in the locked state the guns cannot be fired. When in the data enabled state, tracking data is available and with appropriate action by the angle operator the guns can be fired. When the fire enabled state is entered, the guns can be fired at anytime. The

Table 4.3

List of Functional Systems

System #	System Name
9	Fire Control Network
10	Firing System
11	Gun Directing System
12a	Angle Track System
12b	Range Track System

firing system states are simply firing and not firing. The firing state is entered whenever the trigger is depressed, but firing actually occurs only if the fire control network is in the fire enabled state at the time. In other words, these two systems must be coordinated for the overall system to function properly.

The gun directing system characterizes the status of the gun drive mechanism. The states are defined to be standby, 0^o lead tracking, and lead tracking. When the gun directing system is in the standby state the guns are not in motion. When in the 0^o lead state the input to the gun drive servomechanisms comes directly from the angle tracking system. In the lead tracking state the guns are driven by the output of the lead angle computer. The proper state for this system at any point in time depends on a number of factors which are established by the state of other higher and lower level systems.

The angle track and range track systems must provide the target state data which directly or indirectly drive the gun directing mechanism.

The angle track system states are defined to be optics auto, optics manual, radar auto, and radar manual. These define whether or not the angle operator is monitoring the PPI radar display or using one of the optical sighting systems and whether the angle track data available at the time is produced by automatic control or by the angle operator himself via manual control. The range tracking system, although logically a distinct system, has no autonomy in terms of discrete control. There is only one display and whether the mode is automatic or manual is completely controlled by other systems. The details of the range tracking system therefore, need not be considered further for discrete control modelling.

Clearly, the five systems which form the functional systems level of the hierarchy partition into three distinct groups: range tracking and angle tracking; gun directing system; and fire control network and firing system. These groups define the three major functions of the AAA system: tracking targets, aiming the guns, and firing at the targets. Obviously, each of these systems must function properly for the mission objectives to be met.

Level four in the hierarchy is defined to be the communication/coordination level. The only system residing at this level is the engagement status system, system 15 in the list presented in Appendix 2. This system is best thought of as a communication center through which information about the current activities of the system are passed. This information is then appropriately distributed and other system activities are enabled or disabled accordingly. The states of the engagement status system are:

1. search,
2. manual track,
3. settling, and
4. valid data.

These states define the various conditions the system can be in from the beginning to the end of any single engagement. This system, together with other systems soon to be discussed, establishes whether or not things are progressing normally or if some change must be made.

The fifth and highest level in the hierarchy is the management/command level. This level contains one system, the tactics system. This system is the locus of information and decision concerning basic modes of operation. The tactics system, system 13 in the list in Appendix 2, has five states:

1. normal Mode 1,
2. normal Mode 2,
3. Mode 4,
4. emergency Mode 1, and
5. emergency Mode 2.

Mode 1 refers to full automatic operation during tracking. That is, azimuth, elevation and range tracking data are all under full automatic control once the auto track mode (settling or valid data states of the engagement status system) is entered. The guns are directed by data from the lead angle computer in this mode. In Mode 2, only range data is placed under automatic control when the auto track mode is entered. Angle data is produced by manual tracking. The guns, however, are directed by the lead angle computer.

The emergency designation refers to fire control rather than tracking. In the emergency modes the computer shunt is turned on so that the guns can be fired whether or not the lead angle computer has reached a solution.

Mode 4 operation is a full manual mode in which the radar system is not used and the gun drive mechanism is slaved to the angle tracking output. This mode is functional only if the angle track controls are in the rate mode (State 4).

Furthermore, the only display which produces meaningful data in this case is the right optical system.

The state of the tactics system determines how the activities of the major functional systems will be carried out once the engagement status reaches the settling and valid data states. Further, it determines whether the guns can be fired prior to the valid data state. Finally, if mode 4 is selected the normal constraints imposed by the engagement states system are overridden but additional constraints must be imposed on component level systems if the system is to function properly.

The systems which compose the five levels in the level of analysis decomposition have now been described. Decomposition by system type is discussed in the next section.

B.3 Decomposition by System Type

The only additional systems which must be defined are the interface or information feedback systems. There are four such systems:

1. tracking performance,
2. system performance,
3. ammunition balance, and
4. mission status.

The tracking performance system provides feedback about the quality of tracking. It is hierarchically defined in the sense that angle tracking errors are deemed more important than range track errors. The states of the system are:

1. No target on any display;
2. Angles locked;
3. Angles OK, range locked; and
4. Track OK.

No target on any display can occur if the tracking error is very large, or if there is no target to track. Angles locked is any case in which azimuth or elevation error is sufficiently large that the automatic tracking system cannot function. Range locked is a similar condition for range tracking error. The state assignment rules are given by variable 14, Appendix 2.

The system performance feedback system attempts to capture some information about overall system performance. The states of this system are:

1. No data,
2. Off target, and
3. On target.

The state assignment rules are given in item 16, Appendix 2. As designed this measure is a very local measure of system performance. It would be desirable to have a more global measure, but implementation problems prevented the use of such variables during this analysis. This system does however provide information about time on target, time off target and similar data. Clearly, if one or more major functional systems is not performing adequately, system performance state 3 will not be occupied.

The ammunition balance system determines the relative number of rounds in the upper and lower magazines. State assignment rules are given in Appendix 3. In the absence of other information this data can be used to manage the use of ammunition resources.

The mission status system is used to assess overall ammunition resources with respect to the requirements of the remaining portion of the mission. The state assignment rules for this system are defined in Appendix 3. The states are:

1. Mission less than 50% complete, ammunition use high.
2. Mission less than 50% complete, ammunition use OK.
3. Mission between 50% and 80% complete, ammunition use high.
4. Mission between 50% and 80% complete, ammunition use OK.
5. Mission more than 80% complete, ammunition use high.
6. Mission more than 80% complete, ammunition use OK.

The state of this system establishes whether or not special ammunition control (i.e., special concern with firing control) is needed if the mission is to be completed without depletion of resources.

These four systems provide the several systems in the discrete control hierarchy with information about local and global performance. This information, particularly any change in state, is used in part to determine if control actions are required.

All of the systems used in the discrete control model have now been identified. Further description in terms of those systems which are decision controlled and those which are exogenous event controlled lies at the very heart of the modelling. This discussion is presented in Section 4D. But, before developing the details of the model more fully, a set of block diagram representations is presented to better show the levels of abstraction used in the previous decomposition and to illustrate the relationships between system components.

C. Some Block Diagram Representations of the System

Several block diagrams which help explain signal flows through the system are presented in this section. These diagrams, although not essential to the understanding of the discrete control modelling, help identify the several levels of abstraction which were used to decompose the system. Generally speaking, these block diagrams help one understand the structure of the physical system but they do not help much with the decision making aspects of the discrete control model.

Figure 3.1 which was used in Chapter 3 to display the multitude of switches and controls available to the team members is the most detailed diagram of interest. This diagram roughly corresponds to the primitive component level and it clearly shows the many interconnection graphs which can be formed using the complete set of components and other system elements.

Relatively speaking, the diagram is not very abstract and it realistically describes possible signal flows. It does not however provide much insight into the decision problems faced by the team members.

The second level of abstraction is the system component level. Here the level of information is roughly the same as the primitive component level, but many of the details of specific signal paths have been suppressed. Instead of specific switches, system modules are noted. Figure 4.2 is such a diagram. Clearly, a simpler, less cluttered view of the system structure is provided without much loss of information. The main orientation remains however at the level of hardware and physical signal flow.

The third level, the functional systems level suppresses almost all specific detail. The required diagram is shown in Figure 4.3. This diagram is obtained from 4.2 by grouping elements by function. The result is a very simple representation of interdependency among systems. No longer is it true that the diagram displays physical system flows. This is a more abstract representatian from which one can infer some informatin about performance dependences. In terms of the discrete control model this level of abstraction is at the boundary which separates detailed physical descriptions, including switch settings and specific display utilizations, from the more macroscopic decision problems faced by the team members. In essence, Figure 4.3 shows what major systems the team must keep operating in some coordinated fashion.

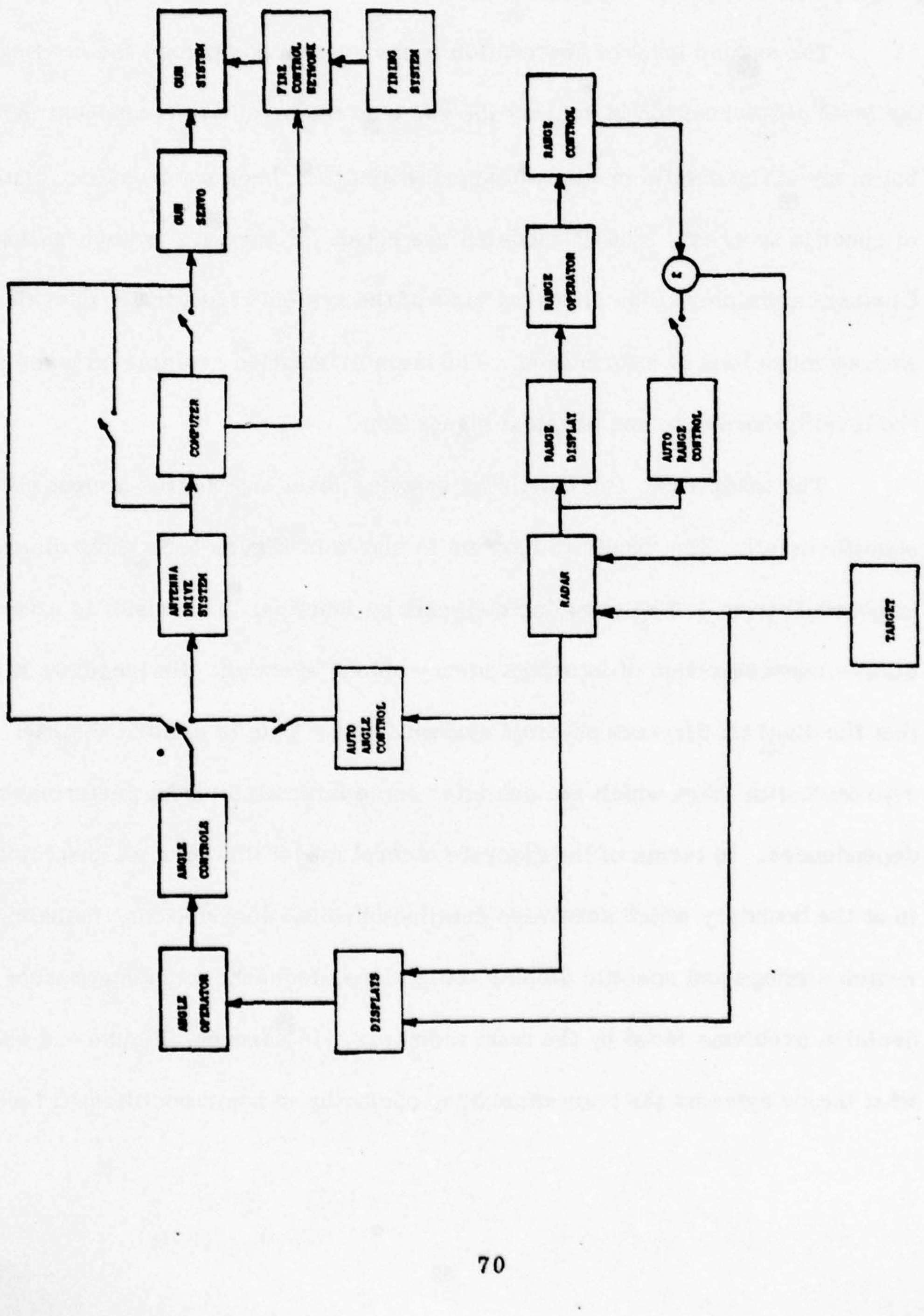


Figure 4.2.--Components Level Description

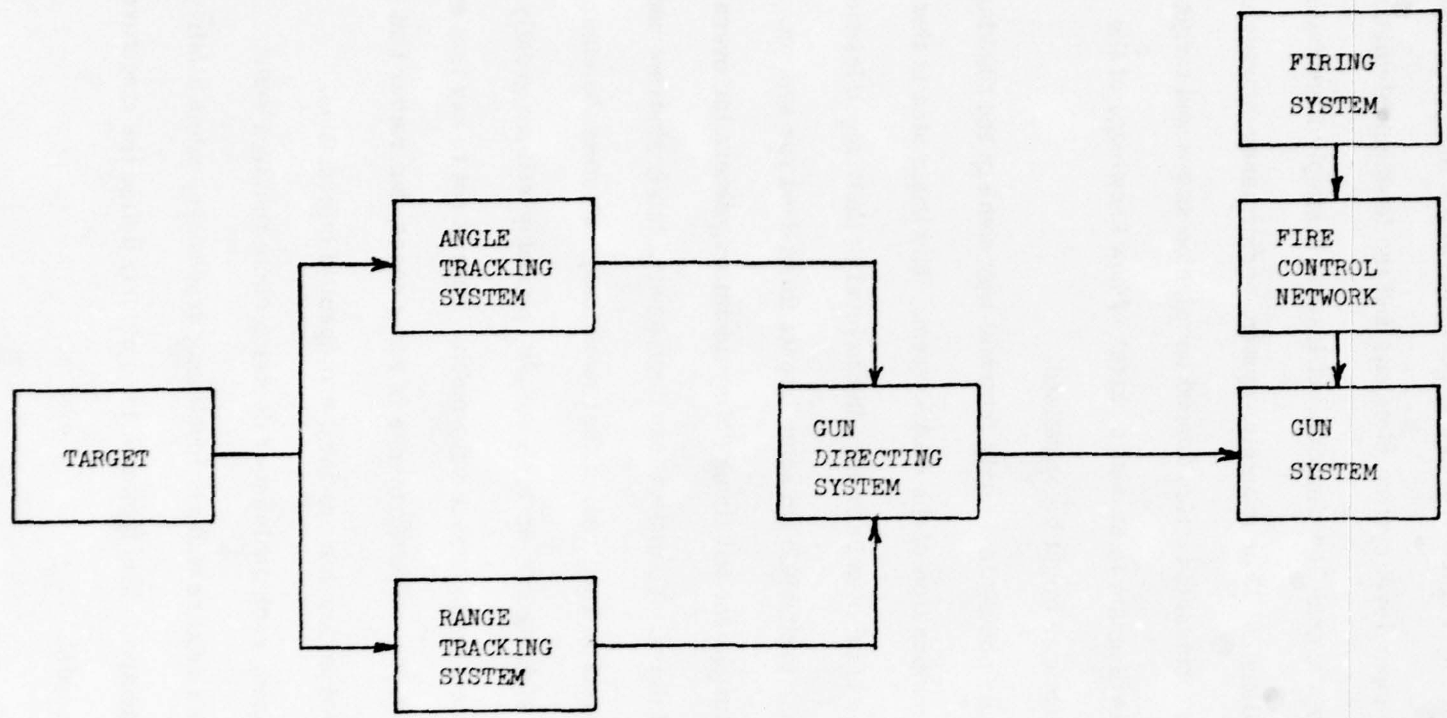


Figure 4.3.--Functional System Level Description

The final diagram, Figure 4.4, is a description of the interrelationships between some of the upper level system elements and the feedback elements. The diagram is not really a signal flow diagram but instead it simply illustrates some possible dependencies. For example, system performance depends on functional system performance; tactics depend on mission status and target state and so on. This diagram is at best a sketch of how knowledge of the system and its performance might be organized.

These diagrams convey in a fairly informal way some of the thinking that went into the decomposition of the AAA system. The basic idea is that decisions made by the team members vary in the level of their significance. Some, such as tactics, probably have some impact throughout the system. Some, such as resource management and firing policy decisions, determine overall mission success or failure. To make these decisions, a fairly abstract and broad view of the system is required. Other decisions, selecting the clear filter for the optical sighting system for example, probably will not greatly influence overall system performance very much. This is not to say that such a decision could not impact on performance in some cases, but rather that such a decision is made based on much more local and specific information.

In summary then, each decision or decision class requires some system context. Some require a fairly localized perspective, some a fairly abstract global perspective. The task now is to clearly define the structure of the discrete control model.

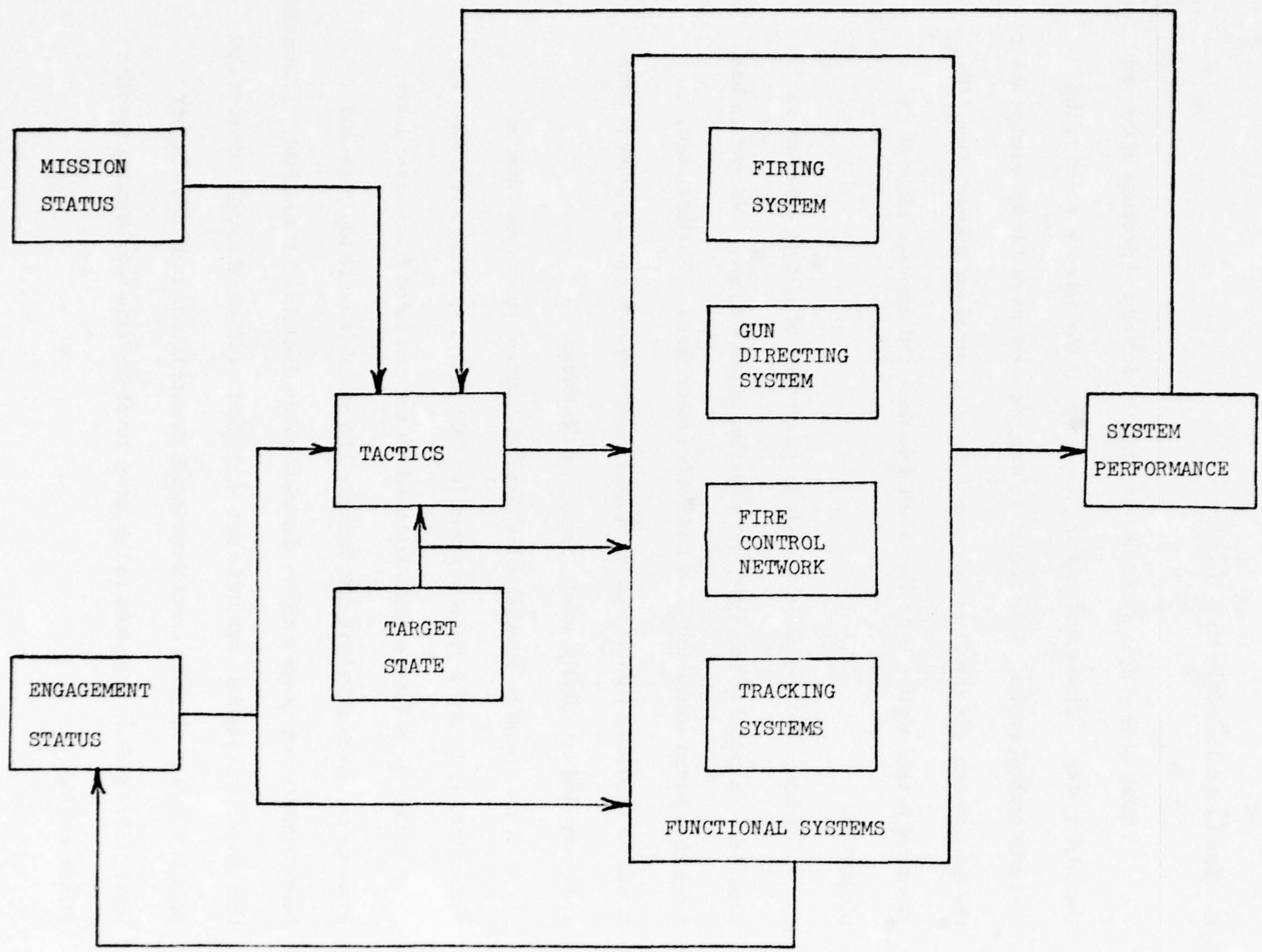


Figure 4.4.--Feedback Dependencies

D. The Control/Coordination Network

Thirty-nine simple systems were defined during the process of decomposing the AAA system. These systems are the nodes in the network which is the discrete control model. This structure organizes the available knowledge about the AAA system and the discrete control tasks required for its operation. The objective in this section is to display this network and examine some of its properties.

Although thirty-nine is a fairly large number of system elements to consider in a model of this type, each system is quite simple. No system has more than seven states and most have only two or three. Furthermore, it will be argued that the states occupied by these systems at any point in time are controlled by a fairly small number of decisions.

It also must be noted that the state assignment rules described in Appendices 1, 2, and 3 are those used for data analysis. They are bottom up rules in that upper level system state values are computed from lower level systems and from primitive data items. The model is more top down and heterarchical in concept with key decision points distributed throughout. This means that eventhough the data available are at the bottom level, the explanation of that data in terms of discrete control decisions proceeds generally from higher level, more global, decisions to the lower level systems representing specific switch settings.

Two types of information must flow through the network. First, constraint information or direct control information and second, conditioning or influence information. The first type is said to actually cause specific state transitions to occur in other systems. Such transitions can be deterministic in that a specific state is occupied after the transition, or they can be nondeterministic in which case the new state is required only to be a member of a specific set. Conditioning information on the other hand does not directly constrain behaviors. Rather, it provides information to a given system about the state of other systems and this information may influence state transitions in the system receiving the information. Any state transitions which take place in this case are the result of a discrete control decision and this decision is based in part on the conditioning information in force at the time.

Several systems are controlled both by external systems and by internal decisions, depending on the situation. In specific situations this type of system's actions may be constrained or controlled by some other system in which case it is directly controlled. But, in other cases such constraints are relaxed and the behaviors of the system under question are decision controlled. This is one of the mechanisms by which overall coordination of the system is achieved and it is also a reason for structuring the system in a hierarchical fashion.

Figure 4.5 is a network diagram which shows the information sources and the receiving systems in the control/coordination network. The arcs (links) in this network should be thought of as communication channels through which the state

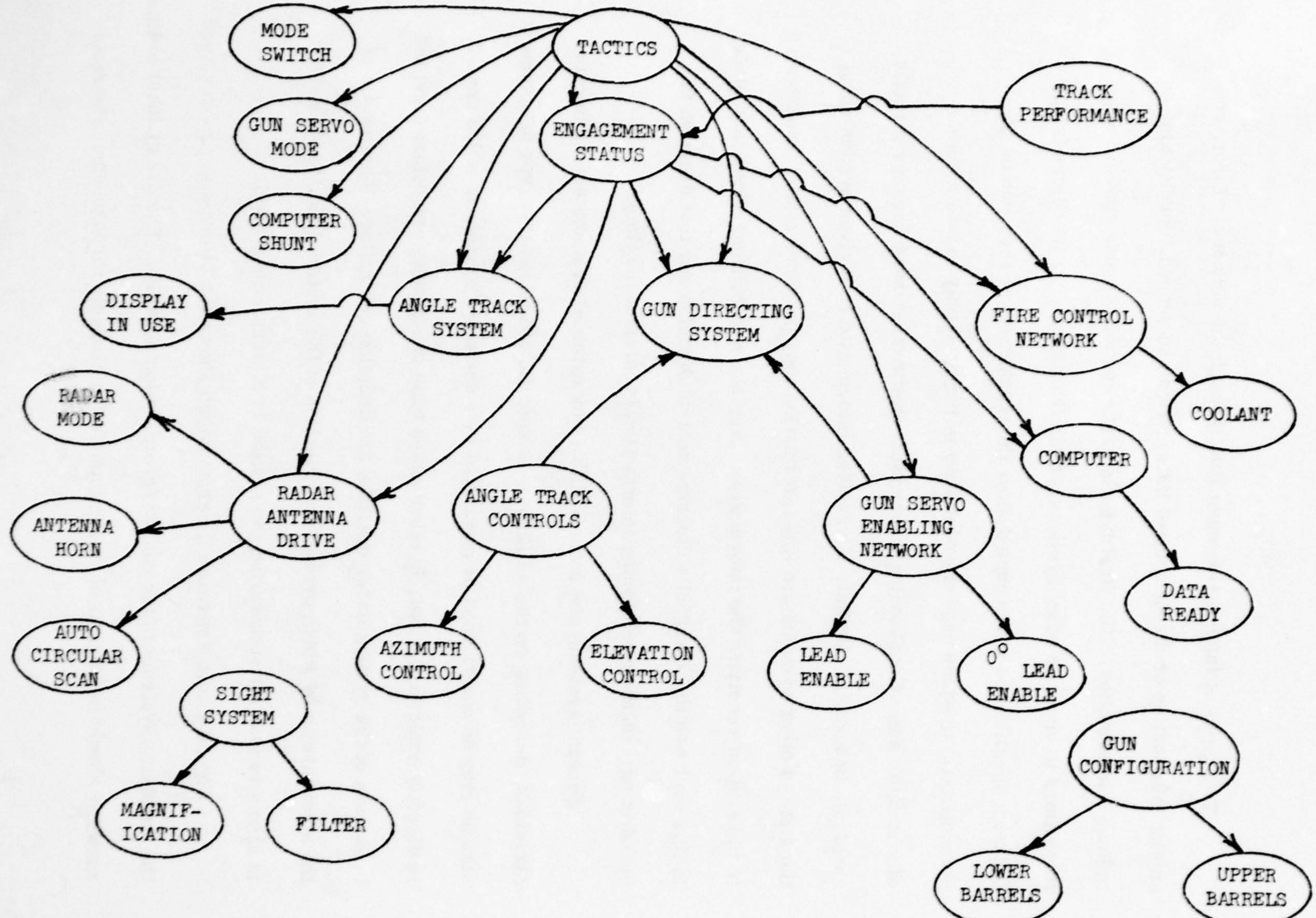


Figure 4.5.--Coordination/Control Network

of the originating node is made known to the receiving node. The state transitions which occur, or which are enabled to occur, in the receiving system are functions of the state of the originating system.

Appendix 4 contains state transition graphs for each system in the network. These graphs clearly show the constraints imposed as a function of the state of the input information.

Several other important properties of the discrete control model can be inferred from the coordination/control network and the corresponding transition graphs. First of all, any system for which all maximum resolution nodes contain single states is completely controlled by external sources. These systems for the most part are the lower level primitive components. The second class of systems is that for which one or more maximum resolution nodes is a set of states. Such systems are, at least under some circumstances, partly decision controlled. The third major class of systems, which with one exception do not appear on Figure 4.4, is event controlled systems. These are the information and feedback systems which interface the finite state systems with the various sources of continuous data. Lists of all three system classes are given in Table 4.4.

Of the systems controlled by decisions, several are effectively controlled by external decisions in the sense that only in specific cases, usually dependent on the tactics state, are they decision controlled. These systems, seven in number, are noted by asterisks in Table 4.4. Five of these; mode switch, automatic circular scan, antenna horn switch, 0^{o} lead enable, and lead enable,

Table 4.4
Breakdown of Systems By Type of Control

Controlled by Internal Decisions	Controlled by External Decisions	Controlled by Exogenous Events
Tactics	Gun Directing System	Tracking Performance
Engagement Status	Computer	System Performance
Angle Track System	Gun Servo Mode	Ammunition Balance
Fire Control Network	Data Ready Indicator	Mission Status
Gun Servo Enabling Net	Display in Use	Target Position
Radar Antenna Drive	Radar Mode	Disturbance
Mode Switch*	Azimuth Control	
Computer Shunt*	Elevation Control	
Coolant*	Sight Filter	
Automatic Circular Scan*	Magnification	
Antenna Horn Switch*	Lower Barrel	
0^{o} Lead Enable*	Upper Barrel	

Table 4.4 (cont.)

Controlled by Internal Decisions	Controlled by External Decisions	Controlled by Exogenous Events
Lead Enable*		
Sight System		
Gun Configuration		
Sight Selected		
Range Control		
Firing System		
Angle Track Controls		

* Denotes systems which are effectively external decision controlled. See discussion in text.

come under decision control only in cases in which their state is of no consequence. Generally, no state change would be made and these systems would remain in the state occupied prior to the occurence of the event which placed them under decision control. The computer shunt is under decision control only when the tactics system is in state number three. In this situation the firing system cannot be operated unless the operator places the shunt in the on state (state 1). Hence, the computer shunt in actuality is constrained to be in state one if the system is to operate in these circumstances. The seventh system listed, the coolant system, is under decision control only when the fire control network is in state one, the locked state. The coolant state will not influence system performance until the fire control network is unlocked, and in that case coolant is controlled from the fire control network.

If the above systems are removed from the list, eleven systems remain in the decision controlled column. These can easily be partitioned in terms of importance.

The tactics system obviously a key element. It interacts with ten other systems and it is the key element in establishing the system configuration. Engagement status is also a key element and it provides information to five systems. The angle track system, fire control network, gun servo enabling network and firing system follow in terms of impact on system configuration and overall performance. The remaining systems, although important, provide alternative means of accomplishing the same tasks and they probably have a lesser impact on total system performance. The decision flexibility in the radar antenna drive, for example, is in establishing the specific mode of automatic search. The major activity would be defined at a higher level.

Some general observations about system coordination can be made at this point. The coordination problem faced by the AAA team members might be defined roughly as follows: to direct each major subsystem into the proper state for each phase of an engagement. What is deemed the proper state will depend on the mission status, resources available and the characteristics of the target.

The network described above clearly illustrates a number of coordination activities. Specifically, the selection of the tactics state defines some major parameters which determine the configuration of the system and also the way in which the tracking phase of an engagement is to be carried out. For example, if the tactics system is placed in state three, the system is greatly simplified and the angle operator is responsible for manually finding and tracking any targets. The radar system, range tracking system, computer and most of the displays are of no interest. The communication of tactics information to the appropriate system elements then defines the set of states which those elements can use and thereby constrains behaviors to be consistent with the objective as defined by the tactics system. This enabling/disabling is clearly apparent in the diagrams given in Appendix 4.

The second point which should be made about coordination concerns the engagement status system. Whereas tactics determines the basic system structure and establishes what activities take place, engagement status provides the vehicle for coordinating the time phasing of these activities. In rough terms, the engagement state defines what each system should be engaged in at a given time. Engagement status is the system through which the focus of control changes as the engagement evolves. During the search phase the focus is in the angle track system and associated subsystems. The status of all other systems is of very little concern during this time. During manual track the focus includes the angle and range tracking systems. During the settling phase, the focus of control is switching from the tracking systems to the gun directing systems and fire control networks. Once the valid data state is reached the focus is in the fire control network and firing systems. During this phase the other system components are involved primarily in monitoring activities, trying to determine whether or not performance is satisfactory.

The heterarchical nature of the system is quite clear given the above perspective. Tactics sets some major constraints and unless a change is needed control flows to engagement status which in some sense directs control at the appropriate time to the tracking systems, gun directing, and fire control systems. A given system retains control until its task is complete or a lower or high level system intervenes and takes control for some reason. When a given system is the focus of control, the various subsystems which define it are active. The states of these subsystems are manipulated to accomplish the task. When a system is not the focus of control, its subsystems are much less active and generally exhibit no state change behaviors.

Errors and mistakes can also be described in terms of this network. The above discussion is based on the assumption that the operator or operators responsible for a specific activity were in fact prepared to carry it out. If control is given to a particular system and the operator whom this system represents in the specific situation fails to perform, he in essence has failed to accept control. This presumably would be detected and corrected at some point, but it certainly represents a deviation from the design condition and from standard procedure.

A second possible source of error exists in the class of systems which were called effectively decision controlled (those marked with an asterisk in Table 4.4). Most of these systems have a nominal or preferred state and if for any reason the system state is changed during a period in which it is inactive, this might not be immediately detected when the system next becomes active. The operators would have to detect a problem and diagnose the source before making corrections and if the system causing the problem happens to be one whose state is seldom changed, this could take some time.

In summary then, the coordination/control network shown in Figure 4.4, together with the state transition diagrams in Appendix 4, define the architecture or organization of the discrete control system. They define what information flows through the system, how activities and behaviors are enabled and disabled and they show how the focus of control is passed from one major system element to another. Furthermore, possible sources of error can be identified. These include the failure of an operator to accept control when it is passed to him and failure to detect an improper system state.

The discussion so far has referred only to control/coordination information flow through the network. In the next section conditioning information flow is examined and a network structure synthesized. Before turning to that development though it should be emphasized that the above discussion of the discrete control network is all based on a specific representation, a specific model. This model defines one organizational scheme which in some sense explains the information flow in the AAA system at a level of abstraction useful for discrete control analysis and explanation. It must be remembered however that it is only a model and as with any model statements should not be taken too literally.

E. The Decision Conditioning Network

The network described in the previous section was constructed from an analysis of the system which identified the major system components, decision types and the relationships between the two. That development did not attempt to explain how the decision controlled systems operate. In this section the task is to construct an information network which to some degree explains the specific behaviors of these decision controlled systems.

The discussion which follows is analytic in the sense that it is based on a fairly detailed analysis of the structure and function of the AAA system. The network which results is therefore a statement about what information an operator might be expected to use in making discrete control decisions. As will be seen in the results section, the subject teams did not in fact exercise all of the flexibility possible and in some sense reduced the complexity of their task.

Rather than present a diagramatic representation of the network which, because of the number of arcs, is quite complex and difficult to read, the decision conditioning network is defined in Table 4.5. This table lists the source systems which provide inputs to the decision controlled systems. Each table entry corresponds to one arc in the network.

The overall discrete control network is a conjunction of the control/ coordination network and the decision conditioning network. The discrete

Table 4.5

Decision Conditioning Network

Receiving System	Source Systems
Tactics	Mission Status Target Position Disturbance System System Performance Gun Directing System
Engagement Status	Tactics Target Position Disturbance System System Performance Track Performance
Angle Track System	Target Position Disturbance System Engagement Status Tracking Performance
Fire Control Network	Engagement Status
Gun Servo Enabling Net	Engagement Status Gun Directing System Target Position Disturbance System
Radar Antenna Drive	Engagement Status Target Position Disturbance System
Sight System	Engagement Status
Gun Configuration	Mission Status Target Position Disturbance System Ammunition Balance

Table 4.5 (cont.)

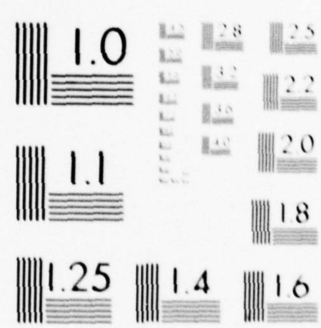
Receiving System	Source Systems
Sight Selected	Gun Directing System Engagement Status Target Position Disturbance System
Range Control	Target Position Disturbance System Track Performance Engagement Status
Firing System	Mission Status Target Position Disturbance System
Angle Track Controls	Engagement Status Track Performance

control model then consists of the nodes which are finite state systems, and the arcs of the network which define what information is input to each finite state system. The systems are non-deterministic automata and those which are decision controlled are represented as stochastic automata.

The control/coordination network is essentially fixed by the way in which the various systems which form its nodes were defined. The decision conditioning network on the other hand is not fixed and must be developed from analysis of the man-in-the loop simulation data. In the next chapter, the results of the data analysis are presented and several conclusions about the model and the simulation are drawn.

AD-A070 079 OHIO STATE UNIV RESEARCH FOUNDATION COLUMBUS F/G 12/1
IDENTIFICATION OF FINITE STATE MODELS OF HUMAN OPERATORS.(U)
MAR 79 R A MILLER AFOSR-77-3152
UNCLASSIFIED OSURF-760518/784556 AFOSR-TR-79-0703 NL

2 OF 3
AD
A070079



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

CHAPTER 5

ANALYSIS OF THE DISCRETE CONTROL II DATA

The discrete control II data were analyzed according to the general rules outlined in Chapter 2. The raw data tapes obtained from AMRL were preprocessed and converted to an events format data base. The specific items included in this data base are listed in Table 5.1. The rules for assigning values to each variable are, with the exception of variables 21, 22, 23, defined in appendices 1, 2, and 3. Before discussing these last three variables, a few general observations concerning the data base are needed.

Time was assumed to run from the start to the end of a mission (23 targets) and data were recorded throughout including the periods between trajectories. An event was said to occur any time the value of one or more of the first 21 variables listed in Table 5.1 changed value. The time into the mission at which any such event occurred was recorded and stored as the value of variable 23. The values of all 23 variables were written upon the occurrence of each event.

Variable 22 was assigned a value equal to the trajectory number of the target whenever that target was available for detection and tracking, and was assigned the value seven between engagements. Variable 21 was a binary variable with values threat, non-threat, which were computed from the trajectory data. Because of noise and processing problems this variable always took on the threat value whenever a target was available for tracking. Because of

Table 5.1

Items Included In The Events Format Data Base

Variable Number	Variable Name
1	Sight State
2	Sight Selected
3	Range Control
4	Gun Configuration
5	Angle Track Controls
6	Gun Servo Enabling Network
7	Radar Antenna Drive
8	Computer
9	Fire Control Network
10	Firing Doctrine
11	Gun Directing System
12	Angle Track System
13	Tactics
14	Tracking Performance
15	Ammo Balance
16	Engagement Status
17	System Performance
18	Mission Status
19	Target Position
20	Disturbance
21	Threat
22	Trajectory
23	Time of Last Event

this problem it was not used in the analysis and the trajectory number was used in its place.

The preprocessing step was very expensive to carry out and it was accomplished with a computer program written for this specific data base. Changes in variable definition or changes in the number of variables were therefore very expensive to make and were kept to a minimum. This meant that the variable definitions which were made during the planning stages of the experiment were used essentially without modification throughout the analysis. The problems which this caused are discussed more fully in the next chapter.

A summary of the data base is provided in the next section and state transition and time in state results for the important variables are presented in section B.

A. Summary of The Discrete Control II Data Base

A number of insights and some perspective can be obtained from a simple summary of the data base and a Markovian analysis of each system. The objective in this section is to point out these results and to provide the foundation needed for further analysis.

Several of the major parameters of the data base are shown in Table 5.2. These data, particularly the average engagement length, are required to properly interpret the following summaries.

Table 5. 2

Data Base Summary

	Team 1	Team 2	Team 3
Missions/Team	22	23	23
Targets/Mission	23	23	23
# of Targets Presented	506	529	529
Total Time Frames* of Data (Seconds)	2, 233, 902 (74, 463)	2, 344, 386 (78, 146)	2, 335, 208 (77, 840)
Average Mission Length (Frames)	101, 541	101, 929	101, 530
Average Engagement Length (Frames)	4, 414	4, 431	4, 414

* 30 Time Frames = 1 second.

Tables 5.3 through 5.20 contain summaries and state transition matrices for each of the important variables. In each case the A subtable lists by team the total number of occupancies of each state, the average length of time that the state was occupied, and the standard deviation of the occupancy time. The B subtable contains the state transition matrix for the system again listed by team. Also included are the number of occupancies of each state and the number of transitions from each state.

For purposes of this summary, an occupancy is counted at the start of each mission (each experimental session) and every time a state change occurs (i.e., the value of the variable changes). For any variable the total number of occupancies for all values must exceed the total number of transitions (changes in values) by 22 for team one and 23 for team two and three. This difference follows because the system must occupy some state at the end of each mission. By comparing the number of occupancies and the number of transitions given in the B subtables, one can determine the states which were occupied at the end of the missions.

The transition matrices are next event matrices. They show the probability the state occupied upon occurrence of the next event will be the j^{th} state given the current state is the i^{th}. Current states are indexed by the row of the matrix, next state by the column. Therefore, p_{ij}, the entry in the i^{th} row, j^{th}, column of a matrix, is the above mentioned probability.

From Table 5.3A it is very clear that the preferred sight configuration for all teams was 2x magnification with the clear filter. Any changes were

Table 5. 3A

Sight System Summary

State #	State Description		Team 1	Team 2	Team 3
1	Clear, 2x	#Of Occupancies	42	29	87
		average time*	49668	80765	26666
		std. dev. of time	40679	36464	29717
2	Clear, 6x		20	7	52
			765	311	187
			969	392	289
3	Neutral, 2x		1	0	7
			21	--	12
			--	--	4
4	Neutral, 6x		0	0	0
			--	--	--
			--	--	--
5	Yellow, 2x		3	1	8
			43675	1	673
			15653	--	1592
6	Yellow, 6x		1	0	3
			1492	--	20
			0	--	1

* Times are recorded in frames, 30 Frames/Second

Table 5.3B

Sight System Transition Matrices

	Matrix					Number of Transitions	Number of Occupancies
Team 1							
	0	.91	.045	.045	0	22	42
	1	0	0	0	0	20	20
	1	0	0	0	0	1	1
	0	0	0	0	1	1	3
	0	0	0	1	0	1	1
Team 2							
	0	1	0	0	0	6	29
	1	0	0	0	0	7	7
	0	0	0	0	0	0	0
	1	0	0	0	0	1	1
	0	0	0	0	0	0	0
Team 3							
	0	.797	.109	.094	0	64	87
	.981	0	0	0	.019	52	52
	1	0	0	0	0	7	7
	.750	0	0	0	.250	8	8
	0	.333	0	.667	0	3	3

generally to the 6x magnification but 6x magnification was used only for short periods of time. The preference for two power optics was somewhat surprising since previous tracking studies showed that operators used the high power sight. Apparently, the additional activities involved in the discrete control exercise call for a qualitatively different style of tracking.

It is clear from Table 5.3B that team three was somewhat more active in changing sight configuration than the other teams but the occupancy times for states other than clear, 2x were very short.

The sight selected summary, Table 5.4A also indicates some unexpected behavior. The right sight was slaved to the guns whereas the left sight was slaved to the radar antenna. Under normal circumstances one would expect that left hand optics would predominate and the right optics would be used only with mode four (tactics in State 3). As will be seen from Table 5.13A tactics State 3 was not often used which means the sight shifting behavior was probably part of the information seeking/control strategy developed by the operators. Note that team one made a switch from left to right about every other trajectory (but with very high variability) and team two and three switched about every run. The left sight was used for significantly longer periods of time and all teams generally ended missions in the left sight configuration.

Range control behavior is also interesting. All three teams showed a very large number of switches, about ten per trajectory. This behavior also

Table 5.4A

Sight Selected Summary

State #	State Description		Team 1	Team 2	Team 3
1	Right	# of Occupancies	224	521	428
		average time	1061	1523	816
		std. dev. of time	830	1217	600
2	Left		241	541	451
			8282	2866	4403
			17130	1339	8103

Table 5.5B

Sight Selected Transition Matrices

	Number of Transitions	Number of Occupancies
Team 1		
$\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$	222 221	224 241
Team 2		
$\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$	518 521	521 541
Team 3		
$\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$	428 428	428 451

Table 5. 5A

Range Control Summary

State #	State Description		Team 1	Team 2	Team 3
1	Coarse	#of Occupancies	4577	5708	5651
		average time	207	185	172
		std. dev. of time	577	483	478
2	Fine		4576.	5707	5655
			281	225	242
			714	569	594

Table 5. 5B

Range Control Transition Matrices

	Number of Transitions	Number of Occupancies
Team 1		
$\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$	4566 4565	4577 4576
Team 2		
$\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$	5699 5693	5708 5707
Team 3		
$\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$	5643 5640	5651 5655

seems to be part of some information seeking strategy. Note that there is not much difference in the length of time that either setting was used.

The gun configuration summary, Table 5.6A, shows that the predominate situation consisted of one barrel on, one off. Team two made substantially fewer switches and team two used both barrels for significantly loner periods of time than did the other two teams. All teams left both barrels off for only one or two seconds. The transition matrices in Table 5.6B show that all teams used basically the same switching pattern.

The angle track controls summary shows expected behavior. Mixed modes (i.e., one rate, one position) tended not to be used. The rate-rate and position-position modes were about evenly used although team one used the rate mode for longer periods of time. The transition matrices again show that all teams used essentially the same switching patterns and the predominance of zeroes in the center core of the matrices indicate that switches from one mixed-mode to the other did not occur. The .001 entry in row two column three of the team three matrix is probably the result of switching errors.

The gun servo enabling network is the system which determines whether or not the guns are in motion and determines in part whether or not the lead angle computer output is used. The predominant mode of operation is full enabled (both lead and 0^o lead enabled). In this mode the guns follow the angle track output until the lead angle computer solution is reached and they then track

Table 5.6A

Gun Configuration Summary

State #	State Description		Team 1	Team 2	Team 3
1	Both off	# of occupancies	139	22	181
		average time	23	31	77
		std. dev. of time	14	30	283
2	Upper on,		177	43	200
	Lower off		6096	29263	6106
			3588	24623	4536
3	Upper Off,		193	51	243
	Lower on		5705	20054	4466
			4433	20170	2884
4	Both on		173	54	141
			293	1158	103
			1076	5967	413

Table 5.6B

Gun Configuration Transition Matrices

					Number of Transitions	Number of Occupancies
Team 1						
	0	.885	.115	0	139	139
	.006	0	.143	.851	175	177
	.726	.117	0	.097	186	193
	0	.125	.875	0	160	173
Team 2						
	0	.789	.211	0	19	22
	.093	0	.070	.837	43	43
	.500	.235	0	.265	34	51
	0	.333	.667	0	51	54
Team 3						
	0	.812	.188	0	181	181
	.092	0	.357	.551	196	200
	.697	.117	0	.126	231	243
	0	.090	.910	0	134	141

Table 5.7A

Angle Track Controls

State #	State Description		Team 1	Team 2	Team 3
1	Az. Position,	# of Occupancies	278	870	1108
	El. Position	avg. time	1683	1100	1085
		std. dev.	843	932	917
2	Az. Position,		206	676	841
	El. Rate		122	86	43
			405	254	178
3	Az. Rate,		196	608	595
	El. Position		42	207	92
			151	470	391
4	Az. Rate,		292	822	926
	El. Rate		5934	1462	1125
			7031	1360	1352

Table 5.7B

Angle Track Controls Transition Matrices

				Number of Transition	Number of Occupancies
Team 1					
0	.513	.131	.356	275	278
.301	0	0	.699	206	206
.821	0	0	.179	195	196
.179	.237	.584	0	274	292
Team 2					
0	.520	.280	.200	862	870
.427	0	0	.573	675	676
.590	0	0	.410	608	608
.265	.281	.454	0	808	822
Team 3					
0	.664	.134	.202	1107	1108
.284	0	.001	.715	841	841
.847	0	0	.153	595	595
.390	.117	.493	0	904	926

that signal. The lead enabled, 0^{O} lead disabled mode (state three) leaves the guns in a standby state until the lead angle computer solution is reached. It is surprising that this mode was used as much as it was (teams two and three used it on an average of once per engagement for a period of about 25 seconds). The transition matrices all show basically the same pattern. There is almost deterministic switching between states three and four. The small probability of switching from state four to state five is simply another indication that the tactics system was not placed in state three very often. The infrequent switches from state three to one and four to two by team one and four to two by team three are probably the result of errors.

The radar antenna drive does not show any real surprises but it does show some interesting differences between teams. Clearly, circular scan modes were not used very much by any of the teams. Sector scan was used about once each engagement by teams two and three and about half that much by team one. Manual search was used heavily by all teams, with team one using it for much longer durations. Autotrack was used by team two on about 50 percent of the trajectories and was used on about 80 percent by the other teams. The transition matrices show generally similar patterns with a few interesting differences. Team one entered autotrack (state one) from manual search whereas team three tended to enter manual track first. Team two tended to go to sector search. This is consistent with the above observation that team two used autotrack less often.

Table 5.8A

Gun Servo Enabling Network

State #	State Description		Team 1	Team 2	Team 3
1	Disabled	# of Occupancies	1	0	0
		avg. time	21	--	--
		std. dev.	--	--	--
2	Lead Disabled,		1	0	10
	0^{o} Lead Enabled		46	--	13
			--	--	5
3	Lead Enabled,		227	516	436
	0^{o} Lead Disabled		994	860	781
			1005	542	558
4	Lead Enabled,		248	548	10
	0^{o} Lead Enabled		8095	3467	389
			17942	1334	829
5	Semi-Auto		1	9	10
			551	58	389
			--	85	829

Table 5.8B

Gun Servo Enabling Network

					Number of Transitions	Number of Occupancies
Team 1						
0	0	1	0	0	1	1
0	0	0	1	0	1	1
.004	0	0	.996	0	226	227
0	.004	.991	0	.004	227	248
0	0	0	1	0	1	1
Team 2						
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	1	0	516	516
0	0	.983	0	.017	525	548
0	0	0	1	0	9	9
Team 3						
0	0	0	0	0	0	0
0	0	0	1	0	10	10
0	0	0	1	0	436	436
0	.022	.956	0	.022	456	479
0	0	0	1	0	10	10

Table 5.9A

Radar Antenna Drive

State #	State Description		Team 1	Team 2	Team 3
1	Auto Track	# of Occupancies	417	252	469
		avg. time	423	860	549
		std. dev.	313	373	451
2	Manual Track		229	366	753
			1291	553	752
			1159	600	970
3	Sector Search		289	653	502
			460	697	680
			329	561	767
4	Slow Circular Scan		2	4	0
			225	61	--
			65	42	--
5	Manual Search		786	1042	1366
			2071	1410	856
			2487	1397	1539
6	Fast Circular Scan		2	1	2
			229	28	201
			82	--	1

Table 5.9B

Radar Antenna Drive Transition Matrices

						Number of Transitions	Number of Occupancies
Team 1							
0	.095	0	0	.905	0	410	417
.241	0	.145	0	.614	0	228	229
0	.097	0	0	.903	0	289	289
0	0	0	0	0	1	2	2
.462	.205	.332	.001	0	0	772	786
0	0	0	.500	.500	0	2	2
Team 2							
0	.175	.004	0	.821	0	246	252
.146	0	.201	0	.654	0	364	366
0	.109	0	0	.891	0	653	653
0	0	.500	0	500	0	4	4
.190	.244	.561	.004	0	.001	1027	1042
0	0	1	0	0	0	1	1
Team 3							
0	.151	.009	0	.840	0	456	468
.229	0	.082	0	.688	.001	752	753
0	.110	0	0	.890	0	502	502
0	0	0	0	0	0	0	0
.214	.464	.321	0	0	.001	1356	1366
0	0	.500	0	.500	0	2	2

The computer transition matrices show some interesting features. Team two used autotrack and hence the computer less often than the other teams; but when they did, it ran full course much more often. Only eight percent of the time did team two force the computer to leave the settling state before a fire control solution was obtained. The other teams forced this 26 and 34 percent of the time. Once a fire control solution was obtained (state three), team one never lost the solution as evidenced by deterministic return to the standby state. Team three had some trouble holding a good track as evidenced by a fairly high probability of returning to the settling state from state three and a high probability of returning to standby from settling. These data seem to show both strategic and performance differences across teams.

The fire control network summary provides some very interesting data. First, since the locked state was almost never entered the computer shunt was turned on by all teams and forgotten. This enabled them to fire the guns without regard to the computer status. Also, team two stayed in the data-enabled state for much longer periods than did team one and for about twice as long as team three. They also stayed in the fire-enabled state (coolant on) for longer periods. This probably indicates that teams one and two were actually using the guns to provide some tracking information and to provide information about how the system was working. This seems to be an attempt to overcome the effects of the various disturbances which were applied. The transition matrices show basically deterministic behavior with some evidence of switching error.

Table 5.10A

Computer Summary

State #	State Description		Team 1	Team 2	Team 3
1	Standby	# of Occupancies	427	265	474
		avg. time	4818	8028	4384
		std. dev.	4245	6866	3346
2	Settling		417	256	523
			89	97	85
			22	13	28
3	Operating		308	236	343
			451	813	619
			261	320	341

Table 5.10B

Computer Transition Matrices

			Number of Transitions	Number of Occupancies
Team 1				
0	1	0	412	427
.260	0	.740	416	417
1	0	0	302	308
Team 2				
0	1	0	248	265
.078	0	.922	256	256
.983	.017	0	230	236
Team 3				
0	1	0	463	474
.344	0	.656	523	523
.834	.166	0	331	343

Table 5.11A

Fire Control Network Summary

State #	State Description		Team 1	Team 2	Team 3
1	Locked	# of Occupancies	2	0	2
		average time	16	--	10
		std. dev.	6	--	2
2	Data Enabled		2439	629	1249
			900	3523	1772
			1984	1885	2146
3	Fire Enabled		2426	611	1237
			15	209	98
			21	159	142

Table 5.11B

Fire Control Network Transition Matrices

				Number of Transitions	Number of Occupancies
Team 1					
	0	1	0	2	2
	.001	0	.999	2428	2439
	0	1	0	2415	2426
Team 2					
	0	0	0	0	0
	0	0	1	611	629
	0	1	0	606	611
Team 3					
	0	1	0	2	2
	.002	0	.998	1239	1249
	0	1	0	1224	1237

The firing system summary shows the same basic trend as fire control. Team two spent much more time on the average in the non-firing state than did the other teams. Also, team two fired bursts which lasted about one second whereas the other teams fired much shorter bursts. The teams clearly did not follow the same firing doctrine or procedures.

The gun directing system summary shows the combined effects of the gun servo enabling network and the computer plus a few additional special cases. The standby state corresponds to the lead disabled case of the gun servo enabling net plus cases in which the gun servo enabling net is set at semi-auto but the angle track controls are not in proper configuration. Lead tracking occurs only if lead tracking is enabled and the computer is in the operating state. Team one had four cases in which the gun servo enabling net was improperly set, team two had one case and team three had none. This follows from the lead tracking occupancies of the gun directing system and the computer operating occupancies. The transition matrices show the realtive tendencies to use autotrack and allow the lead angle computer to direct the guns. As pointed out previously, team two used the automatic mode less often than did the other teams.

The angle track system transition matrices are all quite similar, with one exception. Team one almost never switched from state two to state one. That is they never went into an automatic mode while viewing the target through the optical sighting system.

Table 5.12A

Firing System Summary

State #	State Description		Team 1	Team 2	Team 3
1	Not Firing	# of Occupancies	2433	1318	5145
		average time	903	1743	445
		std. dev.	1989	2248	1330
2	Firing		2422	1299	5132
			15	36	8
			21	31	15

Table 5.12B

Firing System Transition Matrices

	Number of Transitions	Number of Occupancies
Team 1		
$\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$	2422 2411	2433 2422
Team 2		
$\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$	1299 1295	1318 1299
Team 3		
$\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$	5132 5122	5145 5132

Table 5.13A

Gun Directing System Summary

State #	State Description		Team 1	Team 2	Team 3
1	Standby	# of Occupancies	227	520	442
		average time	995	855	771
		std. dev.	1060	544	561
2	0^{o} Lead Tracking		545	772	796
			3434	2214	2238
			3697	1518	2507
3	Lead Tracking		304	235	343
			448	812	619
			262	320	341

Table 5.13B

Gun Directing System Transition Matrices

			Number of Transitions	Number of Occupancies
Team 1				
0	1	0	226	227
.426	0	.574	530	545
0	1	0	298	304
Team 2				
0	1	0	520	520
.689	0	.311	755	772
0	1	0	229	235
Team 3				
0	1	0	442	442
.563	0	.437	785	796
0	1	0	331	343

Tactics, Table 5.15, shows that mode one with the shunt on (state four) is the dominant mode. Team one occasionally used mode two for an engagement and team three used it for short periods, but mode one was clearly dominant.

The tracking performance tables show only minor differences across teams. Team one has a somewhat smaller number of no target occupancies but the occupancies were slightly longer in duration than those of the other teams. Team one also seemed to take about two seconds to reduce angle error to acceptable bounds whereas the other teams took about one second. In general, team one seemed slightly slower in achieving good track data and they achieved it a slightly smaller number of times.

From the ammo balance summary, Table 5.17, it is clear that all teams maintained a balance between upper and lower magazine counts. The non-balanced state was occupied only about 10 or 11 seconds. The length of time that the balanced state was occupied reflects in part a firing policy, i.e., team two fired longer burst and expended rounds at a higher rate than the other two teams and hence had shorter stays in the balanced state. But, the fact that team one stayed in the balanced state somewhat longer on the average than did team three seems to indicate more attention to ammunition use on the part of team one.

The engagement status summary is quite interesting. Teams one and two averaged just over one search occupancy per target and they remained in that state about 75 percent of the time. Team three on the other hand seemed to switch

Table 5.14A

Angle Track System Summary

State #	State Description		Team 1	Team 2	Team 3
1	Radar-Auto	# of Occupancies	889	1238	1348
		avg. time	202	388	364
		std. dev.	267	345	518
2	Radar-Manual		3574	1857	3061
			241	469	437
			478	637	700
3	Optics-Auto		814	727	810
			160	264	133
			191	277	160
4	Optics-Manual		3486	1304	2504
			304	615	158
			526	774	229

Table 5.14B

Angle Track System Transition Matrices

				Number of Transitions	Number of Occupancies
Team 1					
0	.506	.493	.001	887	889
.0911	0	.0003	.9086	3568	3574
.691	.009	0	.300	809	814
0	.982	.108	0	4377	3486
Team 2					
0	.585	.414	.002	1235	1238
.371	0	0	.629	1847	1857
.757	.051	0	.192	724	727
0	.833	.167	0	1297	1304
Team 3					
0	.592	.408	0	1341	1348
.228	0	0	.722	3057	3061
.802	.020	0	.178	805	810
.0004	.8947	.1049	0	2497	2504

Table 5.15A

Tactics Summary

State #	State Description		Team 1	Team 2	Team 3
1	Normal Mode 1	# of Occupancies	2	0	2
		avg. time	16	--	10
		std. dev.	6	--	2
2	Normal Mode 2		0	0	0
			--	--	--
			--	--	--
3	Mode 4		1	9	10
			551	58	389
			--	85	829
4	Emer. Mode 1		38	33	59
			56912	71025	39159
			40278	38759	37220
5	Emer. Mode 2		15	1	24
			4711	27	870
			8291	0	2301

Table 5.14B

Tactics Transition Matrices

	Number of Transitions	Number of Occupancies
Team 1		
$$\begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ .118 & .059 & 0 & .824 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$	2 1 17 14	2 1 38 15
Team 2		
$$\begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & .9 & 0 & .1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$	0 9 10 1	0 9 33 1
Team 3		
$$\begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ .06 & .28 & 0 & .67 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$	2 10 36 24	2 10 59 24

Table 5.16A

Tracking Performance Summary

State #	State Description		Team 1	Team 2	Team 3
1	No Target	# of Occupancies	6974	9446	9398
		avg. time	148	117	125
		std. dev.	365	264	310
2	Angles Locked		4144	6612	4805
			58	35	29
			146	105	91
3	Angles OK,		5148	5408	6305
	Range Locked		154	148	127
			353	359	306
4	Track OK		885	1033	971
			182	195	213
			283	303	325

Table 5.16B

Tracking Performance Transition Matrices

					Number of Transitions	Number of Occupancies
Team 1						
	0	.452	.515	.033	6958	6974
	.736	0	.256	.008	4143	4144
	.693	.186	0	.122	5148	5148
	.381	.049	.570	0	880	885
Team 2						
	0	.580	.379	.041	9430	9446
	.797	0	.196	.007	6619	6612
	.684	.205	0	.111	5407	5408
	.444	.030	.526	0	1029	1033
Team 3						
	0	.428	.536	.036	9383	9398
	.822	0	.173	.005	4804	4805
	.794	.108	0	.097	6305	6305
	.434	.106	.460	0	964	971

Table 5.17A

Ammo. Balance Summary

State #	State Description		Team 1	Team 2	Team 3
1	Balanced	# Occupancies	132	215	159
		avg. time	16656	10552	14357
		std. dev.	19549	9579	15638
2	Not Balanced		118	205	148
			299	369	354
			191	200	203

Table 5.17B

Ammo Balance Transition Matrices

	Number of Transitions	Number of Occupancies
Team 1		
$\begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix}$	118 110	132 118
Team 2		
$\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$	205 192	215 205
Team 3		
$\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$	148 136	159 148

back and forth between search and manual track. This is supported by the large number of manual search occupancies shown by team three (see Table 5.9). It is also clear that team one attempted to go directly to the autotrack mode, and hence enter the settling state more often than the other teams. Team three used autotrack and the computer somewhat more than did the other teams and team two used it or attempted to use it substantially less. This data gives hints about differences in the strategies used by the three teams. This also will be discussed in the next section.

The system performance summary shows very little new information. It basically reflects the firing policy of the teams. The average time on target and the variance of time on target however are interesting. Clearly, no team ever stayed on target for two successive time frames. This result however may be an artifact of the data. The on-target state was established from the tracer evaluation bit in the data base and this may not be a good indicator of performance. The transition matrices show the same things. Team two, probably because it fired longer bursts, had a much higher probability of transitioning from the off-target state to the on-target state. No team went from the no-data (not firing) state to on target with high probability.

The mission state summary clearly shows that ammunition management was not a problem. The summary simply shows a very orderly progression through each mission with no high ammo use states ever being entered.

Table 5.18A

Engagement Status Summary

State #	State Description		Team 1	Team 2	Team 3
1	Search	# Occupancies	558	540	992
		avg. time	3156	3564	1524
		std. dev.	3488	4672	2399
2	Manual Track		229	369	756
			1291	548	748
			1158	599	966
3	Settling		417	256	523
			96	102	85
			71	69	28
4	Valid Data		305	241	350
			449	794	607
			261	337	347

Table 5.18B

Engagement Status Transition Matrices

				Number of Transitions	Number of Occupancies
Team 1					
0	.342	.656	.002	544	558
.759	0	.241	0	228	229
.260	.010	0	.731	416	417
.883	.117	0	0	299	305
Team 2					
0	.619	.371	.010	525	540
.853	0	.144	.003	367	369
.055	.027	0	.918	256	256
.830	.153	.017	0	235	241
Team 3					
0	.699	.296	.005	982	992
.769	0	.228	.003	755	756
.294	.050	0	.656	523	523
.707	.130	.163	0	338	350

Table 5.19A

System Performance Summary

State #	State Description		Team 1	Team 2	Team 3
1	No Data	#Occupancies	2426	1317	5143
		avg. time	906	1745	445
		std. dev.	1996	2249	1331
2	Off Target		3589	2750	5901
			9	16	7
			12	20	11
3	On Target		1357	1545	1216
			1	1	1
			0	0	0

Table 5.19B

System Performance Transition Matrices

	Number of Transitions	Number of Occupancies
Team 1		
$\begin{bmatrix} 0 & .965 & .035 \\ .645 & 0 & .355 \\ .066 & .934 & 0 \end{bmatrix}$	2415 3588 1347	2426 3589 1357
Team 2		
$\begin{bmatrix} 0 & .974 & .026 \\ .450 & 0 & .550 \\ .037 & .963 & 0 \end{bmatrix}$	1298 2748 1543	1317 2750 1545
Team 3		
$\begin{bmatrix} 0 & .959 & .041 \\ .830 & 0 & .170 \\ .188 & .812 & 0 \end{bmatrix}$	5130 5897 1210	5143 5901 1216

Several general comments can now be made. First, it is clear that very little high level decision activity was exhibited. Tactics for the most part were routinized as was control of the ammunition balance. The management of ammunition resources was a trivial task since there was always a sufficient amount to easily complete the mission without depleting the supply. The teams apparently did however develop some special strategies for search and track. The teams also were quite similar in the patterns of their activities. The transition matrices were generally quite similar. The biggest differences seem to be in the time between events.

The transition matrices which were presented in this section were all unconditioned. That is, no input information was assumed which means the decision network consisted of isolated, independent systems with no information flow between nodes. In the next section the key decision controlled systems are examined in more detail using conditioning inputs. This analysis will clarify the strategies used by the teams and give a better picture of the decision-making patterns.

Table 5.20A

Mission Status Summary

State #	State Description		Team 1	Team 2	Team 3
1	Less Than 50% of Targets, Ammo Use High	# Occupancies	0	0	0
		avg. time	--	--	--
		std. dev.	--	--	--
2	Less Than 50% of Targets, Ammo Use OK		22	23	23
			38109	38229	38104
			394	437	387
3	50% to 80% of Targets, Ammo Use High		0	0	0
			--	--	--
			--	--	--
4	50% to 80% of Targets, Ammo Use OK		22	23	23
			27803	27937	27805
			129	106	129
5	Over 80% of Targets, Ammo Use High		0	0	0
			--	--	--
			--	--	--
6	Over 80% of Targets, Ammo Use OK		22	23	23
			35628	35763	35621
			456	386	478

Table 5.20B

Mission Status Transition Matrices

	Number of Transitions	Number of Occupancies
Team 1		
$\begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}$	22 22 0	22 22 22
Team 2		
$\begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}$	23 23 0	23 23 23
Team 3		
$\begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}$	23 23 0	23 23 23

B. Analysis of Key Decision Controlled Systems

In Chapter 4 it was determined that the important decision controlled systems are the following: tactics, engagement status, angle track system, fire control network, firing system, gun servo enabling network, sight system, sight selected, range control, angle track controls, and gun configuration. Other systems are partially decision controlled in certain circumstances, but this list contains the important ones.

Of these systems listed above some can be eliminated from consideration for the discrete control II analysis. It was shown in the previous section that the tactics system showed very little activity. There is not sufficient data to further analyze tactics within the current structure. The gun configuration variable is another one which can be eliminated. Table 5.21 contains cross tabulations of the number of gun configuration events as a function of ammo balance and engagement status. Clearly most events occurred when ammo was balanced and during the search phase of the engagement. This, together with the information about the time between events provided in Table 5.6A, seems to imply that the barrel switch was changed between targets nearly every time in the case of teams one and three, and every four or five times in the case of team two. Gun configuration was apparently not a major factor in tactics or resource management. The transition matrices of Table 5.6B are then the best descriptors of this system.

Table 5.21

Gun Configuration Events

Engagement Status

Team 1		1	2	3	4
Ammo Balance	1	374	55	10	55
	2	144	146	0	16

Team 2		1	2	3	4
Ammo Balance	1	88	9	9	28
	2	9	0	0	4

Team 3		1	2	3	4
Ammo Balance	1	268	95	9	167
	2	136	39	8	23

Of the remaining systems, the overall behavior of the teams can be described in terms of four systems: engagement status, angle track system, fire control network and firing system. The remaining systems provide interesting insight into the detailed search and track strategies but they are not essential for a macroscopic understanding of the total system.

Before moving to the discussion of the decision controlled systems, it is useful to look at the gun directing system, an event controlled system, in order to see some very interesting differences between teams and trajectories. Recall that the gun directing system must be in state two (0° lead tracking) before the guns can be sensibly fired, and state three is entered only if the enabling network is properly set and the computer has reached the operating state. State one is the standby state which is entered only if the enabling network is set to disable motion. Now consider Table 5.22. Note that trajectory number seven is not a real trajectory, but denotes the time interval between trajectories. The large 2, 1 entry of each matrix under trajectory seven indicates that the dominant action, if one was taken, was to put the system in the standby state. The 2, 3 entries correspond to those cases in which the fire control computer reached a solution after the target had disappeared from the displays. It is clear that teams one and three are quite similar on all trajectories. Furthermore, trajectories two and four were treated in a similar manner by teams one and three. Team two is somewhat different, particularly on trajectory four. Whereas team one and three entered

Table 5.22

Gun Directing System Transition Matrices

Team	Trajectory 1	Trajectory 2	Trajectory 3	Trajectory 4	Trajectory 7
1	$\begin{bmatrix} 0 & 1 & 0 \\ .407 & 0 & .593 \\ 0 & 1 & 0 \end{bmatrix}$	$\begin{bmatrix} 0 & 1 & 0 \\ .022 & 0 & .978 \\ 0 & 1 & 0 \end{bmatrix}$	$\begin{bmatrix} 0 & 1 & 0 \\ .213 & 0 & .787 \\ 0 & 1 & 0 \end{bmatrix}$	$\begin{bmatrix} 0 & 1 & 0 \\ .083 & 0 & .917 \\ 0 & 1 & 0 \end{bmatrix}$	$\begin{bmatrix} 0 & 1 & 0 \\ .993 & 0 & .077 \\ 0 & 1 & 0 \end{bmatrix}$
2	$\begin{bmatrix} 0 & 1 & 0 \\ .623 & 0 & .377 \\ 0 & 1 & 0 \end{bmatrix}$	$\begin{bmatrix} 0 & 1 & 0 \\ .035 & 0 & .965 \\ 0 & 1 & 0 \end{bmatrix}$	$\begin{bmatrix} 0 & 1 & 0 \\ .154 & 0 & .846 \\ 0 & 1 & 0 \end{bmatrix}$	$\begin{bmatrix} 0 & 1 & 0 \\ .092 & 0 & .08 \\ 0 & 1 & 0 \end{bmatrix}$	$\begin{bmatrix} 0 & 1 & 0 \\ .994 & 0 & .006 \\ 0 & 1 & 0 \end{bmatrix}$
3	$\begin{bmatrix} 0 & 1 & 0 \\ .400 & 0 & .600 \\ 0 & 1 & 0 \end{bmatrix}$	$\begin{bmatrix} 0 & 1 & 0 \\ .090 & 0 & .910 \\ 0 & 1 & 0 \end{bmatrix}$	$\begin{bmatrix} 0 & 1 & 0 \\ .261 & 0 & .739 \\ 0 & 1 & 0 \end{bmatrix}$	$\begin{bmatrix} 0 & 1 & 0 \\ .067 & 0 & .933 \\ 0 & 1 & 0 \end{bmatrix}$	$\begin{bmatrix} 0 & 1 & 0 \\ .884 & 0 & .116 \\ 0 & 1 & 0 \end{bmatrix}$

state three on trajectory four, team two seldom did. This trajectory contained maneuvers which made the lead angle computer less useful than on other trajectories. Team two apparently knew this, took advantage of the knowledge and tracked manually.

The behavior of all teams, and particularly team two, on trajectory one is interesting. There was a fairly high probability of moving from state two back to state one. Since trajectory one was considered the easiest of the group this is difficult to explain. The time data shown in Table 5.23 provide evidence that in some cases the teams tracked trajectory one using manual tracking and after scoring a hit stopped tracking and waited for the next target. Note that the condition times for both team one and two on transitions from two to one were shorter than for transitions from two to three. It may be the case that manual tracking was used only on targets which were acquired late in the run, but there is no reason to believe that this phenomena would occur only on trajectory one and there is no evidence of it on the other trajectories. Note also that team two which also tracked trajectory four manually stayed in state two much longer on trajectory four than on trajectory one.

Table 5.24 lists tracer hits as a function of engagement status and trajectory number. It is very clear that a substantial amount of firing activity took place when the engagement status system was in state one. Engagement status one is the search state but manual search and manual track are distinguished only by the setting of the radar antenna. *If the optical sighting system is in use,there is no* practical difference between engagement status one and engagement status two.

Table 5.23

Selected Transition Time Data, Gun Directing System

Team	Trajectory	Transition	Avg. Time In State	Standard Deviation	Avg. Condition Time*	Standard Deviation	Count
1	1	1-2	23.9	17.9	11.3	16.3	51
1	1	2-1	57.6	75.6	51.4	53.4	57
1	1	2-3	164.5	135.6	94.6	15.2	83
2	1	1-2	31.4	19.8	18.2	20.3	133
2	1	2-1	66.8	55.9	66.4	55.9	114
2	1	2-3	80.0	281.2	77.6	19.7	69
2	4	1-2	30.6	16.9	14.1	16.7	92
2	4	2-1	104.8	36.3	97.4	13.0	46

* Condition time in this case is the time since the introduction of the target or the last state transition, whichever occured last.

Table 5.24

Summary of Tracer Hits*

Team 1	Engagement Status	Trajectory 1	2	3	4
	1	257	263	133	5
	2	8	37	0	4
	3	0	0	0	0
	4	156	231	263	0
Team 2		**1**	**2**	**3**	**4**
	1	220	157	191	2
	2	8	83	3	0
	3	0	0	0	0
	4	205	214	462	0
Team 3		**1**	**2**	**3**	**4**
	1	245	119	156	8
	2	22	38	36	4
	3	0	0	0	0
	4	164	204	320	0

* System performance in state 3 is deemed a hit.

Table 5.25 contains a summary of fire control network switching events as a function of the angle track system and engagement status states. Note that angle track system state four is optics in use, manual tracking. State three is optics, auto. States one and two then are those in which the optics are not in use. It is very clear from this table that virtually all fire control switch activity which took place during engagement status one (i.e., search) occurred with the angle track system in state four (i.e., optics, manual). The firing activity which took place in the "search" mode was actually during a tracking phase and it was obviously effective.

It is also interesting to note from Table 5.24 that the teams were not at all successful with trajectory four. Particularly note that team two was somewhat less effective than the others even though they developed a consistent strategy to use on this trajectory.

From the above analysis one can draw a fairly clear picture of the way in which the various teams performed and used the options available to them. Teams one and three are quite similar in most respects and team two developed some unique procedures. Because of these facts only teams one and two will be considered throughout the remainder of the section.

The four key decision controlled systems will now be addressed. After considerable testing it was determined that the maximum information about engagement status is provided by the track performance variable. Transition matrices and time summaries for teams one and two are given in Tables 5.26 and 5.27.

Table 5.25

Fire Control Network Event Totals

Team	Angle Track System	Engagement Status	Transition	Number of Events
1	1	3	2-3	22
1	1	3	3-2	22
1	1	4	2-3	252
1	1	4	3-2	257
1	2	1	2-3	67
1	2	1	3-2	73
1	2	2	2-3	10
1	2	2	3-2	10
1	3	3	2-3	13
1	3	3	3-2	11
1	3	4	2-3	1202
1	3	4	3-2	1194
1	4	1	2-3	826
1	4	1	3-2	815
1	4	2	2-3	34
1	4	2	3-2	33
2	1	3	2-3	1
2	1	3	3-2	1
2	1	4	2-3	23
2	1	4	3-2	17
2	2	1	3-2	2
2	3	3	2-3	5
2	3	3	3-2	1
2	3	4	2-3	206
2	3	4	3-2	206
2	4	1	2-3	335
2	4	1	3-2	337
2	4	2	2-3	41
2	4	2	3-2	42

Table 5.25 Cont.

Team	Angle Track System	Engagement Status	Transition	Number of Events
3	1	3	2-3	11
3	1	3	3-2	4
3	1	4	2-3	174
3	1	4	3-2	140
3	2	1	2-3	19
3	2	1	3-2	18
3	2	2	2-3	1
3	2	2	3-2	1
3	3	3	2-3	2
3	3	3	3-2	2
3	3	4	2-3	255
3	3	4	3-2	285
3	4	1	2-3	534
3	4	1	3-2	533
3	4	2	2-3	241
3	4	2	3-2	241

Table 5.26A
Team 1
Engagement Status Transition Matrices

Track Performance	Transition Matrix				Transition Count
1 (No Target)	0	.407	.584	.009	113
	.924	0	.076	0	131
	.615	.038	0	.346	78
	.907	.093	0	0	97
2 (Angle Error Locked)	0	.295	.705	0	88
	0	0	0	0	0
	.900	0	0	.100	40
	.800	.700	0	0	5
3 (Angles OK, Range Locked)	0	.530	.470	0	215
	.722	0	.278	0	72
	0	0	0	0	0
	0	0	0	0	0
4 (Track OK)	0	0	1	0	128
	0	0	1	0	25
	.081	.003	0	.916	298
	.873	.127	0	0	197

Table 5.26B
Team 1
Engagement Status Time In State Data

Track Performance	Transition	Avg. Condition Time (Frames)	Standard Dev.	Count
1	1-2	816	1181	46
1	1-3	3090	3091	66
1	2-1	1016	1166	121
1	3-1	103	184	48
1	4-1	535	298	88
2	1-2	2060	2020	26
2	1-3	3712	4491	62
2	3-1	50	19	36
3	1-2	2212	2534	114
3	1-3	4402	3557	101
3	2-1	1088	809	52
3	2-3	1950	940	20
4	1-3	3872	3726	128
4	2-3	2580	904	25
4	3-1	94	133	24
4	3-4	101	0	273
4	4-1	378	222	172
4	4-2	546	209	25

Table 5.27A
Team 2
Engagement Status Transition Matrices

Track Performance	Transition Matrices				Transition Count
1	0	.892	.108	0	93
	.958	0	.042	0	261
	.455	.364	0	.182	11
	.667	.167	.166	0	6
2	0	.955	.027	.018	111
	0	0	0	0	0
	.25	.75	0	0	4
	1	0	0	0	3
3	0	.805	.177	.018	169
	.738	0	.250	.012	84
	0	0	0	0	0
	1	0	0	0	3
4	0	0	1	0	152
	.045	0	.955	0	22
	.033	0	0	.967	241
	.830	.157	.013	0	223

Table 5.27B
Team 2
Engagement Status Time In State Data

Track Performance	Transitions	Avg. Condition Time (Frames)	Standard Dev.	Count
1	1-2	804	1132	83
1	1-3	3978	2151	10
1	2-1	518	625	250
2	1-2	2567	3245	106
3	1-2	2680	3035	136
3	1-3	5076	4854	30
3	2-1	425	373	62
3	2-3	835	457	21
4	1-3	5121	4604	152
4	2-3	779	644	21
4	3-4	101	0	233
4	4-1	809	296	185
4	4-1	807	267	35

There are no particular surprises in this data. As would be expected, transitions during the track OK situation are basically the same for both teams. However, team two generally shows a longer time in state than team one in this situation. They managed to stay in state four for about 800 time frames which is roughly twice as long as team one.

Transition patterns during the no-target phase are also different. Team two preferred to go into manual track from search if a change was made. Team one on the other hand actually started the lead angle computer a fair number of times in the no-target situation. These transitions however took place after about 100 seconds without a target (3000 frames) and therefore they may correspond to cases in which the target was not detected.

The matrices for the angles locked case also show team one's reluctance to use manual track and a preference for transitioning from search directly to the settling state, state three. In other words, they preferred to try the computer even though tracking errors were large. The entries in row three show the same tendencies.

Angle track system activity as a function of tracking performance is given in Tables 5.28 and 5.29. Both teams in the no-target situation show a preference for the radar-auto state, state one. Apparently search was accomplished with the radar system in an automatic mode (sector search). Most transitions from state one were to state two, the radar-manual state. This transition signals the start of manual search. Transitions from state two in the no-target situation were

Table 5.28A
Team 1
Angle Track System Transition Matrices

Tracking Performance	Transition Matrix				Transition Count
1	0	.716	.284	0	455
	.1361	0	.0005	.8634	2035
	.631	0	0	.369	274
	0	.945	.055	0	1574
2	0	.812	.178	0	16
	.021	0	0	.979	373
	.053	0	0	.947	38
	0	.839	.161	0	336
3	0	1	0	0	2
	.016	0	0	.984	1134
	0	0	0	0	0
	0	.927	.073	0	1427
4	0	.261	.737	.002	414
	.846	0	0	.154	26
	.773	.014	0	.213	497
	0	.064	.936	0	140

Table 5.28B
Team 1
Angle Track System Time In State Data

Tracking Performance	Transitions	Avg. Condition Time (Frames)	Standard Dev.	Count
1	1-2	107	192	326
1	1-3	81	165	129
1	2-1	298	217	277
1	2-3	583	-	1
1	2-4	123	186	1757
1	3-1	41	51	173
1	3-4	69	74	101
1	4-2	105	169	1487
1	4-3	80	167	87
2	1-2	17	23	13
2	1-3	10	12	3
2	2-1	44	59	8
2	2-4	90	156	365
2	3-1	17	22	2
2	3-4	49	29	36
2	4-2	58	119	282
2	4-3	106	139	54
3	1-2	2	1	2
3	2-1	623	999	18
3	2-4	97	189	1116
3	4-2	141	223	1323
3	4-3	320	507	104
4	1-2	96	172	108
4	1-3	100	101	305
4	1-4	115	-	1
4	2-1	8	10	22
4	2-4	10	16	4
4	3-1	126	156	384
4	3-2	272	244	7
4	3-4	212	200	106
4	4-2	16	44	9
4	4-3	9	12	131

Table 5.29A
Team 2
Angle Track System Transition Matrices

Tracking Performance	Transition Matrix				Transition Count
1	0	.627	.372	.001	825
	.594	0	0	.406	1093
	.744	0	0	.256	391
	0	.957	.043	0	507
2	0	.912	.088	0	34
	.021	0	0	.979	188
	.333	0	0	.667	9
	0	.973	.027	0	111
3	0	1	0	0	17
	.033	0	0	.967	548
	0	0	0	0	0
	0	.936	.064	0	516
4	0	.437	.560	.003	359
	.883	0	0	.167	18
	.784	.114	0	.102	324
	0	.031	.969	0	163

Table 5.29B
Team 2
Angle Track System Time In State Data

Tracking Performance	Transitions	Avg. Condition Time (Frames)	Standard Dev.	Count
1	1-2	105	183	517
1	1-3	237	292	307
1	1-4	418	-	1
1	2-1	347	184	649
1	2-4	107	117	444
1	3-1	78	82	291
1	3-4	60	88	100
1	4-2	112	133	485
1	4-3	48	83	22
2	1-2	4	11	31
2	1-3	2	1	3
2	2-1	15	11	4
2	2-4	129	234	184
2	3-1	3	2	3
2	3-4	19	27	6
2	4-2	63	110	108
2	4-3	19	14	3
2	1-2	2	1	17
3	2-1	113	224	18
3	2-4	161	252	530
3	4-2	262	269	483
3	4-3	453	901	33
3	1-2	166	92	157
4	1-3	132	110	201
4	1-4	184	-	1
4	2-1	11	14	15
4	2-4	3	4	3
4	3-1	207	233	254
4	3-2	522	155	37
4	3-4	243	174	33
4	4-2	11	13	5
4	4-3	16	13	158

most often to the optics-manual state for team one which means they were almost always making a display change, probably in an attempt to find the target. Team two had a higher probability of returning to the radar-auto state from radar manual than did team one. It is interesting to note that both teams almost always (probability .979) transitioned from state two to state four, optics manual, in both the angles locked and the range locked cases. They both also showed a high probability of exiting the radar-auto state. These imply a very strong preference to complete the target acquisition phase using the optical sighting system. Furthermore, almost all transitions from state four were to state two in the cases where angle error and/or range error were locked. This means that the activity during manual acquisition of the target consisted of display changes. The track OK matrices show the very definite tendencies to get into auto track and use the optical display system.

It was mentioned earlier that a large number of targets were tracked manually. In those cases the range error would almost certainly be large because the range operator would not be required to track (his output was used only by the lead angle computer). The transition matrices for track performance state three then are the ones which most likely describe the angle track system activity during manual tracking of targets.

The fire control network systems almost never occupied state one, the locked state, and hence almost all transitions were between states two and three. Transition matrices in this case provide no information and therefore only time information is presented. Tables 5.30 and 5.31 contain this information. Fire control network state two is the data enabled state, state three is fire enabled. This network must be in state three before the guns can be fired and this state is occupied only if the coolant button is depressed. The conditioning variables for the fire control network are the angle track system and the engagement status. It is clear from the tables that nearly all team two activity took place in engagement status states one or four and angle track system states three or four. That is, team two almost always used the optics when firing. Team one showed more activity and an unexpected amount with the angle track system in state one (radar, auto). This suggests that they may have on occasion depressed the coolant button before switching to the optical sighting system. This probably did not significantly influence performance. The times are as expected: longer, more variable occupancies of state two, short occupancies of state three. These data also show the differences in firing policy. Team two fixed long bursts, team one short ones.

Tables 5.32 and 5.33 are the time data for the two state firing systems. State one is not firing, state two firing. The condition variables are the fire control netword and engagement status. There is little new information in these tables, but one interesting item can be observed. Team one had a tendency when manually tracking (engagement status one) to try to fire the guns (1-2 transition) without first putting the fire control network into state three. It appears that when

Table 5. 30
Team 1
Fire Control Network Time In State Data

Angle Track System	Engagement Status	Transition	Avg. Condition Time	Standard Dev.	Count
1	3	2-3	45	117	22
1	3	3-2	7.7	5.9	22
1	4	2-3	33	103	252
1	4	3-2	6.7	9.8	257
2	1	2-3	225	735	67
2	1	3-2	16	21	73
2	2	2-3	78	180	10
2	2	3-2	7	4.7	10
3	3	2-3	53	60	13
3	3	3-2	14	9.2	11
3	4	2-3	47	99	1202
3	4	3-2	12	15	1194
4	1	2-3	184	497	826
4	1	3-2	21	24	815
4	2	2-3	110	208	34
4	2	3-2	23	24	33

Table 5.31
Team 2
Fire Control Network Time In State Data

Angle Track System	Engagement Status	Transition	Avg. Condition Time	Standard Dev.	Count
1	4	2-3	145	116	23
1	4	3-2	20	22	17
3	4	2-3	202	126	206
3	4	3-2	330	125	206
4	1	2-3	736	857	335
4	1	3-2	129	111	337
4	2	2-3	198	265	41
4	2	3-2	88	91	42

Table 5.32
Team 1
Firing System Time In State Data

Fire Control Network	Engagement Status	Transitions	Avg. Condition Time	Standard Dev.	Count
2	1	1-2	528	1494	583
2	1	2-1	26	26	490
2	2	1-2	387	881	24
2	2	2-1	20	23	12
2	3	1-2	148	246	17
2	3	2-1	11	6	19
3	1	1-2	749	1725	307
3	1	2-1	15	20	395
3	2	1-2	426	961	20
3	2	2-1	17	23	31
3	3	1-2	20	21	18
3	3	2-1	8	9	14
3	4	1-2	85	158	428
3	4	2-1	9	12	861

Table 5.33
Team 2
Firing System Time In State Data

Fire Control Network	Engagement Status	Transition	Avg. Condition Time (Frames)	Standard Dev.	Count
2	1	1-2	510	1291	28
2	1	2-1	53	32	38
3	1	1-2	1630	1924	464
3	1	2-1	49	32	453
3	2	1-2	543	736	51
3	2	2-1	44	35	45
3	4	1-2	150	199	741
3	4	2-1	27	28	741

involved with the tracking activities they sometimes forgot how the system worked and deviated from the standard procedures. Team two obviously did not have this problem.

This concludes the discussion of the four major decision controlled systems. It should be noted that even though the time data was reported in mean and standard deviation summary form, the analysis procedures can be used to provide time histograms for any of the variables reported here.

Before turning to the task of summarizing, a few more observations about some of the minor variables are made in the next section.

C. Some Additional Analysis

There are a few additional questions concerning the sight system, sight selected, range controls and gun servo enabling net systems that need to be resolved. For the most part these relate to certain unexpected behaviors which took place and which to this point have not been fully addressed.

It was observed previously that there were a fairly large number of switches from the left optical system to the right system for no apparent reason. In order to determine when this activity occurred several summaries were examined. Table 5.34 shows fire control network activity as a function of sight selected. Clearly, the vast majority of the fire control activity took place with the left sight, sight number two, in place. The use of the right sight was therefore probably not part of the firing strategy. A check of right sight selection activity

Table 5. 34

Fire Control Network Event Totals

Team	Sight Selected	Transition	Number of Events
1	1	2-3	12
1	1	3-2	11
1	2	2-3	2414
1	2	3-2	2404
2	1	2-3	24
2	1	3-2	19
2	2	2-3	587
2	2	3-2	587
3	1	2-3	4
3	1	3-2	3
3	2	2-3	1233
3	2	3-2	1221

as a function of the disturbance state showed that most activity occurred during periods in which no disturbance was present. These results are shown in Table 5.35. From Table 5.36 which shows sight selection transitions as a function of engagement status, it is obvious that this activity occurs with engagement status at search. Furthermore, since fire control was negligible when the right sight was used it follows that the sight selection activity actually did occur during search and not tracking and that it was part of some strategy to check out or evaluate the operation of the system.

Sight system activity also seemed to be related only to engagement status. Recall that the sight system defined the filter and magnification used. The summary is given in Table 5.37. These events took place primarily during search or manual track. Therefore, they probably occurred when a target was being tracked manually and the subjects had the motivation and time to switch to the best sight configuration for the activity.

Range control switches also took place primarily during search, but further explanation is provided by tracking performance. Consider Table 5.38. Clearly most events occurred when the tracking performance state was either one or three with a smaller number occurring in state two. States one and three are the no-target and range-error locked states and two is angle-error locked. Range control switching activity was therefore essentially restricted to cases in which the range error was large. It was probably part of the range operators search strategy, but it is not clear why such a pattern was advantageous.

Table 5.35

Sight Selection Events by Disturbance

Team	Disturbance	1-2 Transitions	1-2 Transition
1	None	148	69
	One or More	73	153
2	None	337	177
	One or More	184	341
3	None	369	112
	One or More	59	316

Table 5. 36

Sight Selection Events by Engagement Status

Team	Engagement Status	1-2 Transitions	1-2 Transitions
1	1	199	206
1	2	19	15
1	4	3	1
2	1	402	454
2	2	42	36
2	3	2	0
2	4	75	28
3	1	371	368
3	2	57	56
3	3	0	1
3	4	0	3

Table 5.37

Sight System Events by Engagement Status

Team	Engagement Status	1-2 Transitions	2-1 Transitions
1	1	12	3
1	2	8	2
1	4	0	0
2	1	5	1
2	4	1	0
3	1	18	4
3	2	32	28
3	4	1	0

Table 5.38

Range Control Events by Tracking Performance

Team	Tracking Performance	1-2 Transitions	2-1 Transition
1	1	2748	2732
1	2	400	406
1	3	1324	1324
1	4	94	103
2	1	3904	3867
2	2	556	553
2	3	1192	1217
2	4	47	56
3	1	3889	3926
3	2	387	387
3	3	1324	1286
3	4	43	41

The gun servo enabling network is the next system to be considered. It was pointed out earlier that the lead enabled, 0^o lead disabled state, was used a very large number of times. Table 5.39 lists the relevant transitions as a function of tracking performance. Virtually all 4-3 events occurred during periods when no target was present and most 3-4 events did also. It appears that the system was put into state three between engagements and then returned to state four before the gun system would be required again. There is no obvious explanation for this behavior and it might be the result of a misunderstanding of the system.

As might be expected, nearly all angle track control activity takes place during search and manual track. The transition matrices which were presented in Table 5.7B are the best descriptions of the activities which took place during these periods.

In Chapter 4, Section E, it was argued that the decision conditioning network had to be empirically determined. The required network has now been so established and the result is shown in Figure 5.1. This network summarizes the conditioning which was used in the previous sections. It is clearly much simpler than the network hypothesized in Chapter 4 and shown in Table 4.5. Essentially, tracking performance influences decisions made in the angle track system and engagement status. Engagement status then influences the activities of the firing system and the fire control network as well as some of the lesser systems. Those systems shown in Figure 5.1 which are not interconnected with any other system

Table 5.39

Gun Servo Enabling Net Events by Tracking Performance

Team	Tracking Performance	3-4 Transitions	4-3 Transitions
1	1	171	224
1	2	26	1
1	3	28	1
2	1	405	514
2	2	70	1
2	3	40	1
2	4	1	0
3	1	322	429
3	2	60	0
3	3	53	6
3	4	1	1

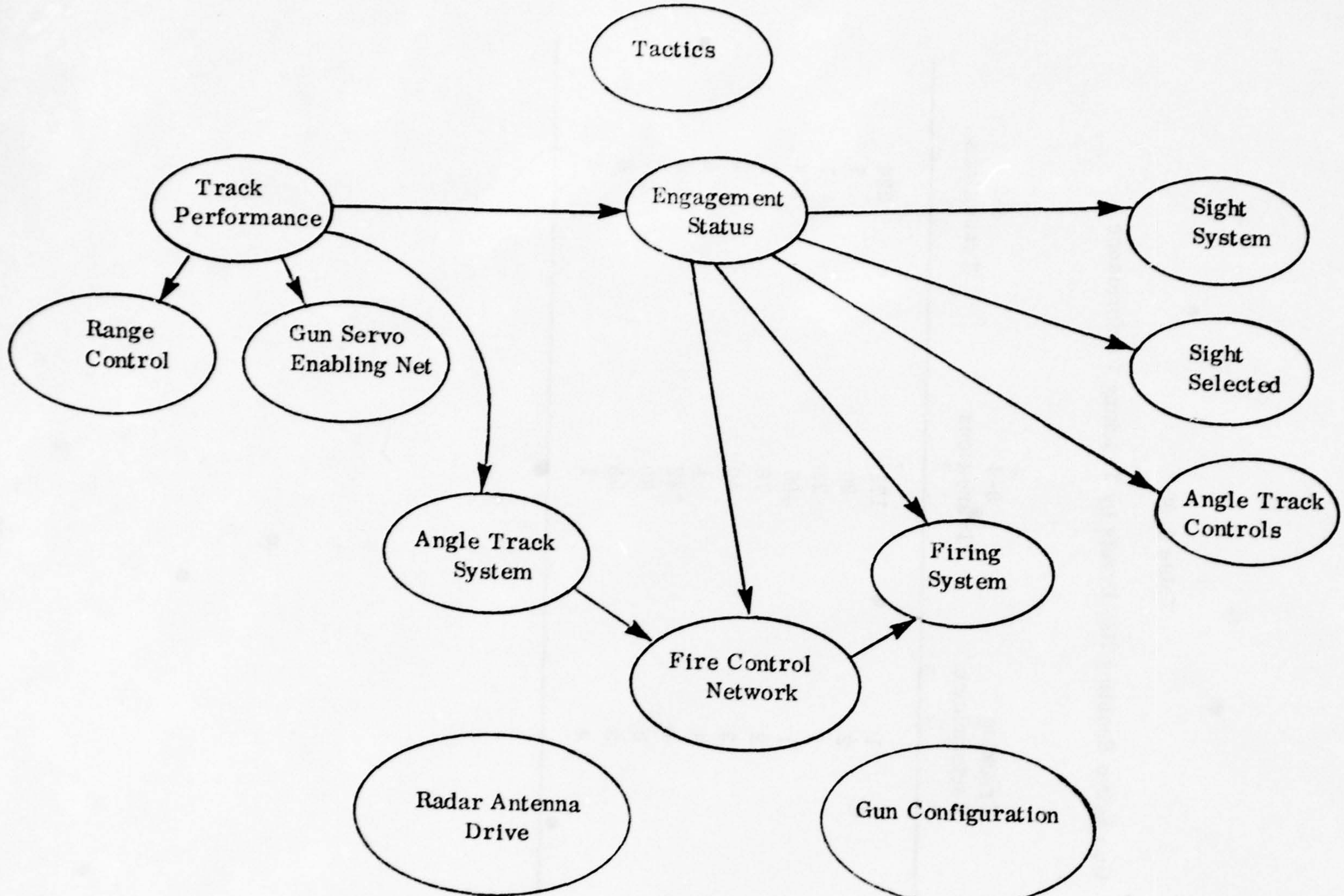


Figure 5.1--Decision Conditioning Network

are those decision controlled systems which were either routined or showed little activity.

Even though one can postulate quite complex decision networks, most of the behaviors shown by the subjects in this experiment can be localized and described by this simple structure. However, there remain a few questions such as the range control switching behavior which it does not answer.

This concludes the presentation of the results of the analysis. In the next and final chapter these results will be briefly summarized and certain technical questions will be raised.

CHAPTER 6

SUMMARY & CONCLUSIONS

The concluding remarks are divided into three classes: comments on the experiment and the results, comments about the simulator, and comments about the discrete control modelling methodology. Each class will be considered in turn.

Based on the previously discussed analysis it is fairly easy to construct a scenario of the way in which the subject teams performed their tasks. First of all, search for targets was generally accomplished using the automatic sector search mode and the primary display was the PPI radar screen. In cases where the target was slow to appear some teams might occasionally switch to the left optics sighting system and less often to the right optical system. As soon as the target appeared the angle operator would switch to the left optics system and switch from the radar driven search mode to a manual mode for acquisition. During this period the range operator was switching his control back and forth between the coarse and fine setting. The commander was resetting the gun servo enabling system to state three and thereby moving the gun directing system from standby to 0^{o} lead tracking. Also during the acquisition phase and the early phases of tracking the angle operator determined which of the four trajectories the current target was following. This determined in part the strategy that he then followed.

Once the target was acquired the system was either put into autotrack or, if the trajectory was particularly easy, tracking often continued in a manual mode. Note, team two also used manual tracking for the most difficult highly maneuvering target as well. If manual tracking was used, firing started within about 20 seconds of acquisition (i.e., when the target came within range). In cases where autotrack was used, firing started shortly after the fire control solution was achieved. Easy targets were disengaged after a few hits. The gun directing system was then generally disabled for a few seconds after the target disappeared from the screen. The angle track system was then reconfigured for search and the cycle started again.

Several interesting observations can be made about the performance of the teams. Team one had some problems with the fire control network interlocks when they used manual tracking. They forgot to enable firing before trying to fire. The other teams had no problem and team one had no problem in the automatic tracking cases. Team one may have become so involved in performing the tracking task that they forgot how the system worked. There were other pieces of evidence which showed that teams one and two infrequently made incorrect switch settings or failed to reconfigure the system quickly enough.

Team two seemed to develop more stylized behavior patterns and in some sense was more predictable. They used some unrealistic practices however (e.g., firing long bursts).

It was a surprise that no team used the six power optical system very much for tracking. Apparently the feedback provided by tracers reduced the need for precise visual information. It definitely seems that a style of tracking was used in this experiment which differed qualitatively from that used in simple tracking studies.

The experiment certainly did not fully exercise all of the potential of the AAA system. Very little high level decision making activity was shown. Only one basic mode of operation was used and the resource management tasks were minimized and routinized. The special tactics improvised by some teams were interesting but they were probably artifacts of the experiment and not representative of behaviors which would be observed in reality.

In general terms the commander's tasks were very trivial. Teams obviously learned the limited number of trajectories which were used and they keyed their actions to the trajectory. The attempt to introduce uncertainty via the disturbances or simulated countermeasures did not seem to have much impact. They may have delayed the start of autotrack, for example, but they did not alter the basic patterns of behavior as represented by the various transition matrices of the discrete control model. The fact that every mission contained exactly the same number of trajectories greatly simplified ammunition management. The subject teams knew that they could and should go after all targets. They did not have to be selective or evaluate the threat potential of any target. There was certainly no risk associated with missing one and there was no significant scoring penalty.

The simplicity of the simulator also helped to trivialize some of the decision tasks. Unrealistic firing policies and continuous use of the computer shunt are two such examples.

The model which seems to best capture the various teams performance is really a set of finite state systems organized into two networks, the coordination/control network (Figure 4.5) and the decision conditioning network (Figure 5.1). The systems included in these networks were established through a detailed analysis of the AAA system, its functions, and the tasks of the operators. By decomposing along several structural dimensions, and particularly by analyzing at several levels of abstraction, an effective and useful representation of the discrete control system was obtained.

In general terms this representation is a model of an organizational structure which the operators might use to reduce the apparent complexity of their task and generally achieve coordinated actions and acceptable performance. It is really just a structured representation of the available knowledge of the system and its functions.

The coordination/control network is basically hierarchical and reflects the constraints on lower level decision making activity imposed by upper level decisions. In the terms of the finite state systems representation, state transitions in lower level systems are disabled, enabled or constrained as a function of the state of upper level systems. The decision conditioning network establishes the information flow patterns which are needed to explain, at least in part, the decisions which are made (i.e., the state transitions which take place). The systems in the decision network are represented by generalized stochastic automata in which state transitions are conditioned by the information flowing into the system from other nodes in the network. The two networks in conjunction form a heterarchical system description in which decision making activity flows from one functional area to another as a function of the established constraints and the environmental situation.

In terms of statistical questions and data analysis procedures there are several points. First of all, state transition matrices, time in state summaries and time in state distributions were all successfully estimated from the data base. The procedures used allow a fairly arbitrary decision network to be specified by the user and they produce as output the above mentioned data. The biggest problem is that it is very easy to produce very much more data than can ever be reasonably absorbed by the analyst. This essentially means that the network must be carefully designed to avoid this data overload problem. Good results were achieved with an iterative network definition process in which only a few key systems were examined at one time.

Further refinements must be made to allow the construction of more sophisticated networks. Specifically, output assignment functions more general than the identity functions used here are required and nested or conditional use of inputs must be incorporated. Both of these improvements will help with the data overload problems and they will enable much more efficient model building.

A portion of the effort during the first year of research funded by this grant was spent in developing procedures to approximate stochastic automata with l^{th} order Markov systems. This work was motivated by the fact that in theory there might be several states of a system which produce the same output and in such cases state transition estimates cannot be obtained. Experience has now shown that intelligently constructed coordination and decision networks constructively specify a state space and a system which is rich enough to explain, at least statistically, the behaviors of interest. Furthermore, these systems are first order. This means that the approximation methods mentioned above have not been required. This is not to say that they will not be needed in the future, but the theoretical and empirical evidence seems to show that the network generated state spaces are adequate.

In general terms then, the discerte control methods so far developed seem to have potential. They can be used to make sense out of complex systems and identify the key decision points. They can discribe quite complex behaviors in terms of a relatively small number of decisions. The structure of the model is quite easy to understand and the individual finite state systems are all simple and intuitive. Grasping the overall view, i. e. all levels simultaneously, is more difficult and amount of statistical information which can be produced is overwhelming. These problems are minimized however if one restricts attention to only the one or two levels which are most important for a given question. The data problems tend to surface only when the level of abstraction is pushed too low and inconsequential system elements are included.

The discrete control structure seems to be a good one to use to define and analyze problems of coordination. It can be used to locate and trace the occurrence of rare events although the causes of such events cannot always be explained. Mistakes or apparent mistakes show up clearly in the transition matrices of the decision controlled system. The network structure can also be used to locate and identify potential problem and error sources. The existing analysis routines allow evaluation of alternate network configurations although the routines need to be made more flexible and interactive if efficiency is to be improved.

In the final analysis the strengths and weaknesses of the ideas and methods reported here have their origin at the same point, i.e., the flexibility and generality of the structure. Discrete control really is not a model but rather is an abstract structure together with a few ideas about modelling. The results in any specific case then will depend on how well the analyst understands the system of interest and how well he or she can abstract out the essentials for inclusion in the model. The discrete control modelling structure enables and facilitates the accomplishment of the task, but it doesn't automatically perform it.

REFERENCES

Bobrow, L. S. and Arbib, M. A. (1974). Discrete Mathematics. Philadelphia: W. B. Saunders.

Cinlar, E. (1975). Introduction to Stochastic Processes. Englewood Cliffs, New Jersey: Prentice Hall.

Miller, R. A. (1977). "Identification of Finite State Models of a Human Operator." Annual report, AFOSR Grant No. -77-3152. Columbus, Ohio: Ohio State University Research Foundation.

Pay, A. (1971). Introduction to Probabilistic Automata. New York, London: Academic Press.

Rasmussen, J. (1976). "Hybrid Model of the Human Operator" in Monitoring Behavior and Supervisory Control; T. B. Sheridan and G. Johannsen, editors. New York, London: Plenum Press.

Windeknecht, T. G. (1971). General Dynamical Processes. New York, London: Academic Press.

APPENDIX 1

TARGET STATE ASSIGNMENTS

This appendix contains the rules used to determine the target state from the trajectory information and from the data items which define the disturbance applied. Several parameters are required in these computations and nominal parameter values are defined.

Data Required:

R = target range

A = target altitude

Parameters:

R_{max} = max range indicator

R_{int} = intermediate range indicator

A_{max} = max altitude indicator

A_{int} = intermediate range indicator

Nominal Parameter Values:

R_{max} = 10,000 meters

R_{int} = 2,000 meters

A_{max} = 1,500 meters

A_{int} = 50 meters

Range:

Condition	Range State
$R > R_{max}$	1
$R_{int} < R \leq R_{max}$	2
$R \leq R_{int}$	3

Altitude:

Condition	Altitude State
$A > A_{max}$	1
$A_{int} < A \leq A_{max}$	2
$A \leq A_{int}$	3

Target Position:

Target Introduced	Range	Altitude	Target State
0	$1 \vee 2 \vee 3$	$1 \vee 2 \vee 3$	1 (No target)
1	1	$1 \vee 2 \vee 3$	2
1	$2 \vee 3$	1	3
1	2	3	4
1	3	3	5
1	2	2	6
1	3	2	7

Disturbance System:

Optics Disturbance	PPI Disturbance	Range Disturbance	Range, PPI Blanked	Disturbance State
0	0	0	0	1
0	1	0	0	4
0	0	1	0	4
0	1	1	0	4
1	$1 \vee 0$	$1 \vee 0$	1	2
1	1	0	0	3
1	0	1	0	3
1	1	1	0	3
1	0	0	0	5

APPENDIX 2

COMPONENT SYSTEM STATE ASSIGNMENTS

The following are tabular displays of the logic functions which assign component system state values for the AAA system model. These may be viewed as truth tables.

The development generally proceeds from lower level system components toward higher levels. The lower level system states are all simple functions of the primitive data items. The higher level states generally are functions of both primitive data items and lower level system states.

1. Sight System

Filter	Magnification	Sight State
00	0	1 (Clear 2x)
00	1	2 (Clear 6x)
01	0	3 (Neutral 2x)
01	1	4 (Neutral 6x)
10	0	5 (Yellow 2x)
10	1	6 (Yellow 6x)

2. Sight Selector

Sight Selector	State
0	1 (right)
1	2 (left)

3. Range Control

Range Control	State
0	1 (coarse)
1	2 (fine)

4. Gun Configuration

Lower Barrel	Upper Barrel	Gun Configuration State
0	0	1 (Both off)
0	1	2
1	0	3
1	1	4 (Both off)

5. Angle Track Controls

Azimuth Tracking Control	Elevation Tracking Control	Angle Track State
0	0	1 (position, position)
0	1	2
1	0	3
1	1	4 (rate, rate)

6. Gun Servo Enabling Network*

Gun Servo Mode	Lead Enable	0° Lead Enable	Gun Servo Enable Network State
0	0	0	1 (disabled)
0	0	1	2
0	1	0	3
0	1	1	4 (full enabled)
1	0 ∨ 1	0 ∨ 1	5 (standby)

7. Radar Antenna Drive

Antenna Horn	Auto Circular Scan	Radar Mode	Radar Antenna Drive State
0 ∨ 1	0 ∨ 1	0	1 (auto track)
0	0 ∨ 1	1	2 (manual track)
0 ∨ 1	0 ∨ 1	2	3 (sector search)
0 ∨ 1	0	3	4 (slow circular scan)
1	0 ∨ 1	1	5 (manual search)
0 ∨ 1	1	3	6 (fast circular scan)

*The symbol ∨ denotes the logical operation "or".

8. Computer*

Gun Servo Mode	Radar Mode	Data Ready Indicator	Computer State
0	¬00	0 ∨ 1	1 (standby)
1	00 ∨ 01 ∨ 10 ∨ 11	0 ∨ 1	1 (standby)
0	00	0	2 (setting)
0	00	1	3 (operating)

9. Fire Control Network

Computer Shunt	Computer State	Coolant	Fire Control Network State
0	¬3	0 ∨ 1	1 (locked)
0	3	0	2 (data enabled)
0	3	1	3 (fire enabled)
1	1 ∨ 2 ∨ 3	0	2
1	1 ∨ 2 ∨ 3	1	3

10. Firing System

Trigger	Firing System State
0	1 (not firing)
1	2 (firing)

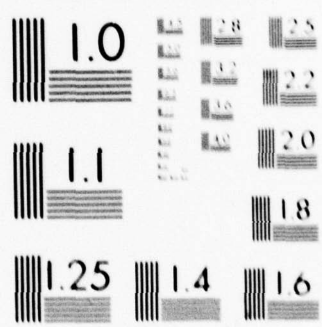
---

*The symbol ¬ used in this table denotes the logical "not".

AD-A070 079 OHIO STATE UNIV RESEARCH FOUNDATION COLUMBUS F/G 12/1
IDENTIFICATION OF FINITE STATE MODELS OF HUMAN OPERATORS.(U)
MAR 79 R A MILLER AFOSR-77-3152
UNCLASSIFIED OSURF-760518/784556 AFOSR-TR-79-0703 NL

3 OF 3
AD A070079

END
DATE FILMED
7-79
DDC



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

11. **Gun Directing System**

Gun Servo Enabling Net State	Angle Track Control State	Computer State	Gun Directing System State
5	¬4	1 ∨ 2 ∨ 3	1 (standby)
5	4	1 ∨ 2 ∨ 3	2 (0^o lead track)
1 ∨ 3	1 ∨ 2 ∨ 3 ∨ 4	1 ∨ 2 ∨ 3	1
2 ∨ 4	1 ∨ 2 ∨ 3 ∨ 4	¬3	2
2	1 ∨ 2 ∨ 3 ∨ 4	3	2
4	1 ∨ 2 ∨ 3 ∨ 4	3	3 (lead track)

12. **Angle Track System**

Gun Servo Mode	Radar Mode	Display In Use Indicator	Angle Track System State
0	¬01	0	1 (radar, auto)
1	00 ∨ 01 ∨ 10 ∨ 11	0	2 (radar, manual)
0	01	0	2
0	¬01	1	3 (optics, auto)
1	00 ∨ 01 ∨ 10 ∨ 11	1	4 (optics, manual)
0	01	1	4

13. **Tactics**

Gun Servo Mode	Mode Switch	Computer Shunt	Tactics State
0	0	0	1 Normal Mode 1
0	1	0	2 Normal Mode 2
1	0 ∨ 1	0 ∨ 1	3 Mode 4
0	0	1	4 Emer. Mode 1
0	1	1	5 Emer. Mode 2

14. Tracking Performance

Target On Display	Azimuth Error	Elevation Error	Range Error	Track Performance State
0	0 ∨ 1	0 ∨ 1	0 ∨ 1	1 (No Target)
1	1	0 ∨ 1	0 ∨ 1	2 (Angle Locked)
1	0	1	0 ∨ 1	2
1	0	0	1	3 (Range Locked)
1	0	0	0	4 (Track OK)

15. Engagement Status

Tactics	Tracking Performance	Radar Antenna Drive System	Gun Directing System	Engagement Status
3	1	1 ∨ 2 ∨ 3 ∨ 4 ∨ 5 ∨ 6	1 ∨ 2 ∨ 3	1 (Search)
3	¬1	1 ∨ 2 ∨ 3 ∨ 4 ∨ 5 ∨ 6	1 ∨ 2 ∨ 3	4
¬3	1 ∨ 2 ∨ 3 ∨ 4	¬(1 ∨ 2)	1 ∨ 2 ∨ 3	1
¬3	1 ∨ 2 ∨ 3 ∨ 4	2	1 ∨ 2 ∨ 3	2 (Manual Track)
¬3	1 ∨ 2 ∨ 3 ∨ 4	1	¬3	3 (Setting)
¬3	1 ∨ 2 ∨ 3 ∨ 4	1	3	4 (Valid Data)

16. System Performance

Fire Control Network	Firing System	Tracer Evaluation	System Performance
1 ∨ 2 ∨ 3	1	0 ∨ 1	1 (No Data)
¬3	2	0 ∨ 1	1
3	2	0	2 (Off Target)
3	2	1	3 (On Target)

APPENDIX 3

MISSION STATUS INFORMATION SYSTEM

This appendix contains a description of the rules by which the mission status states are determined from more detailed information. These computations involve several parameters which must be defined by the analyst. Nominal parameter values are given.

Data Required:

LBC = rounds count in upper magazine

UBC = rounds count in lower magazine

NT = number of targets introduced at a given point into the mission

Parameters Required:

UR = average (or expected) number of rounds expended per target

UT = expected number of targets per mission

MAXR = maximum number of rounds available (# of rounds available at start of mission)

Nominal Parameter Values:

UT = 23
UR = 50
MAXR = 2000

Targets So Far:

Count	State
$NT < .5UT$	0
$.5UT \leq NT < .8UT$	1
$.8UT \leq NT$	2

Ammunition Used:

Count	State
$MAXR - LBC - UBC \geq NT * UR$	0
$MAXR - LBC - UBC < NT * UR$	1

Ammunition Balance:

Count	State
$\lvert UBC - LBC \rvert \leq UR$	0
$\lvert UBC - LBC \rvert > UR$	1

Mission Status System:

Targets So Far	Ammo Used	Mission Status State
0	0	1
0	1	2
1	0	3
1	1	4
2	0	5
2	1	6

APPENDIX 4

COORDINATION/CONTROL NETWORK STATE TRANSITION GRAPHS

This appendix contains graphs which detail the state transitions which occur as a function of information flow in the network show in Figure A. 4.1 (same as Figure 4.5).

The key to reading these diagrams is quite simple, but a few things must be explained. First of all, a given system may be controlled by more than one external system, i.e., it has more than one incoming arrow in Figure A. 4.1. In such cases the inputs are defined as either primary or secondary. The primary input is always listed first in the input list and the primary input on the transition graph is not noted. Secondary inputs are explicitly noted.

The general form of the transition graphs is shown in Figure A.4.2. The elliptical figures represent states or sets of states. The description consists of the state name, where appropriate, plus the numerical value assigned to that state. Sets of states are always identified with a list of specific states enclosed with set brackets.

State transitions are represented by solid lines connecting states and the input conditions which cause the transition to take place are defined by the bracketed symbols displayed on each transition arc (w, x, y, z, u, v, in Figure A. 4.2). These symbols define logical expressions formed from the possible input values and a transition occurs when the appropriate logical expression is "true". To explain this more completely, any time the value of an input variable changes an event is said to occur and this event leads to a state transition in the system receiving the input. The transition which takes place depends on which logical expression is true at the time. Some nodes do not display transitions from themselves to themselves. These correspond to cases in which any input event will produce a transition out of the node in question.

The primary input variable explains transitions on the upper parts of the graph. (w, x, y, z,) then are expressions defined from the primary input values. The effect of secondary and other inputs is shown in those parts of the diagram connected to the primary via dashed lines. Secondary inputs are always used to provide a more detailed explanation of information in the primary diagram. They explain which state or states of the many allowed under the primary condition are actually occupied. Any node connected to a primary node with a dashed line (a secondary conditioning variable) should be viewed as a more detailed representation of the primary node.

The state transition graphs follow.

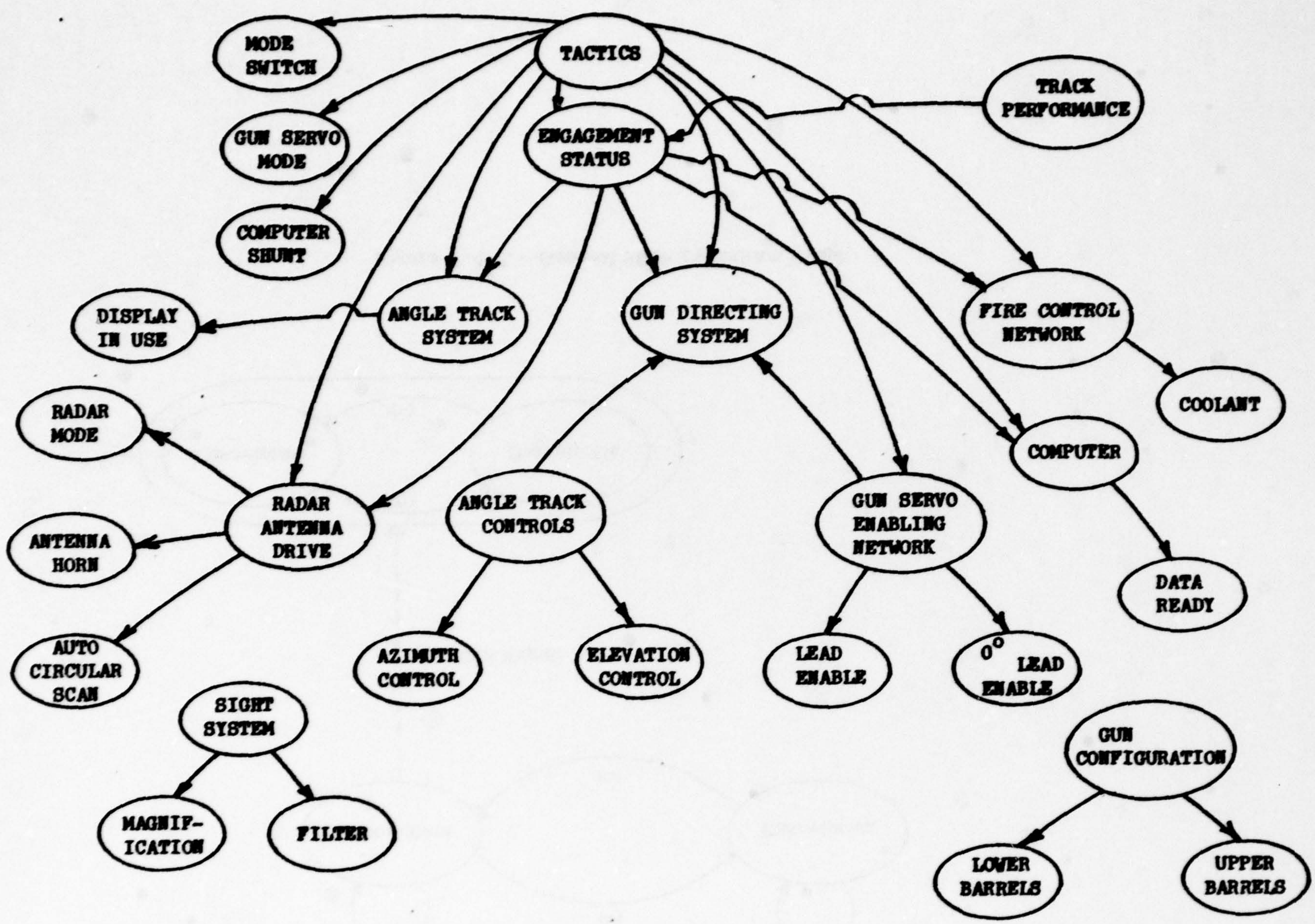


Figure A.4.1.--Coordination/Control Network

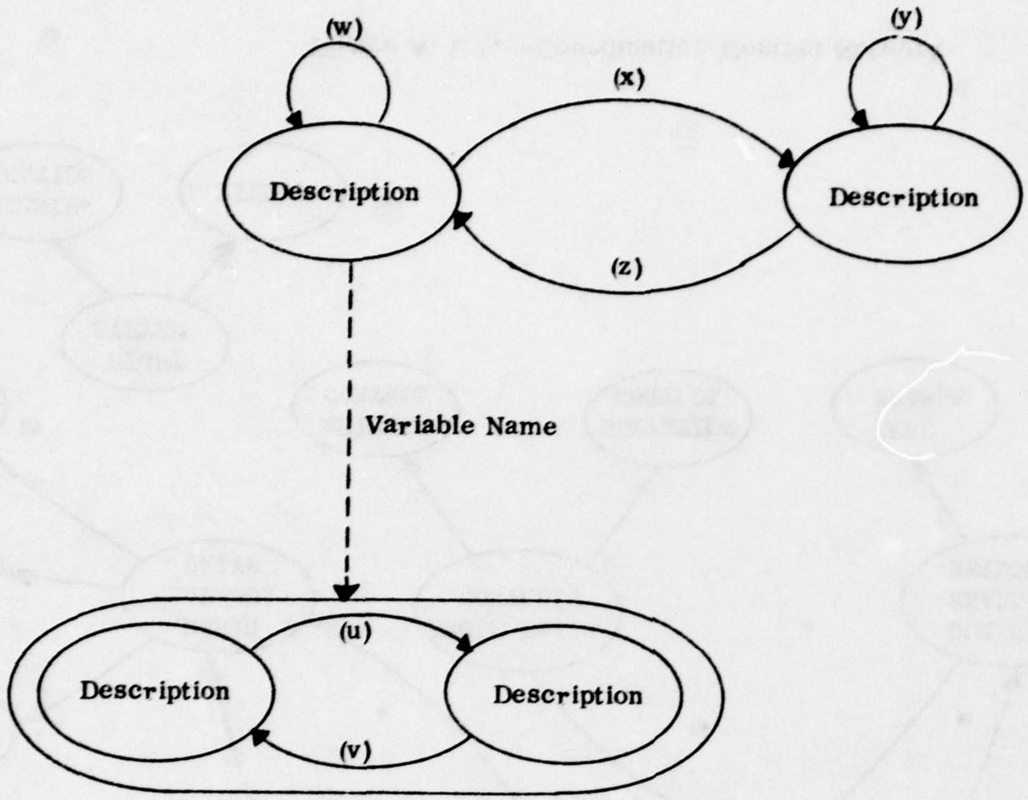
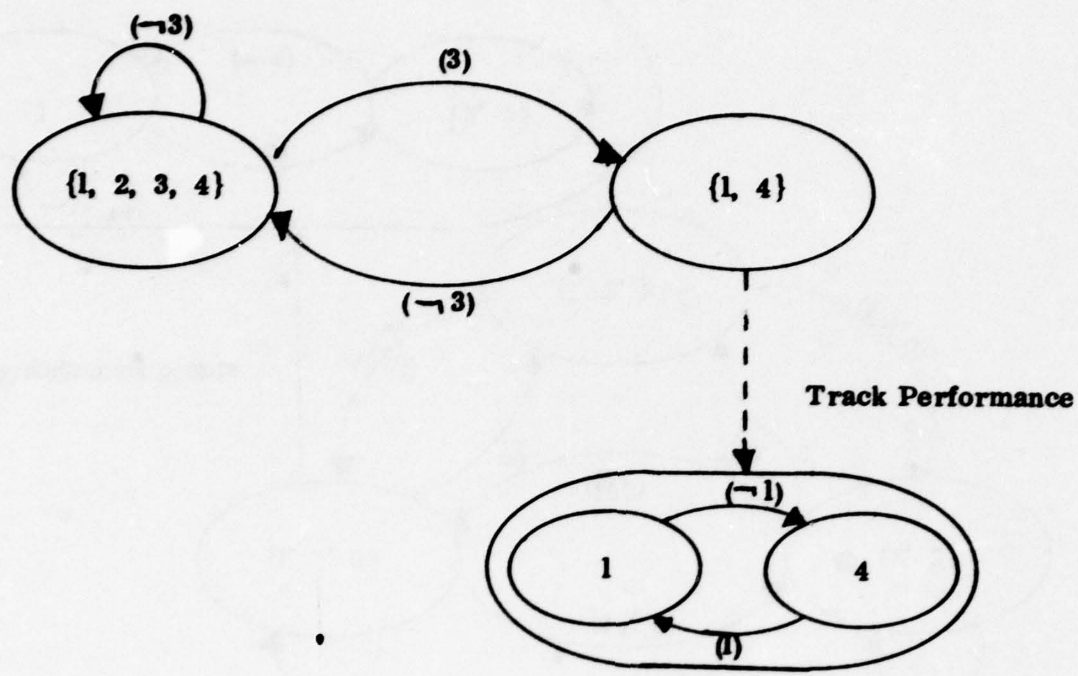


Figure A.4.2.--General State Transition Graph

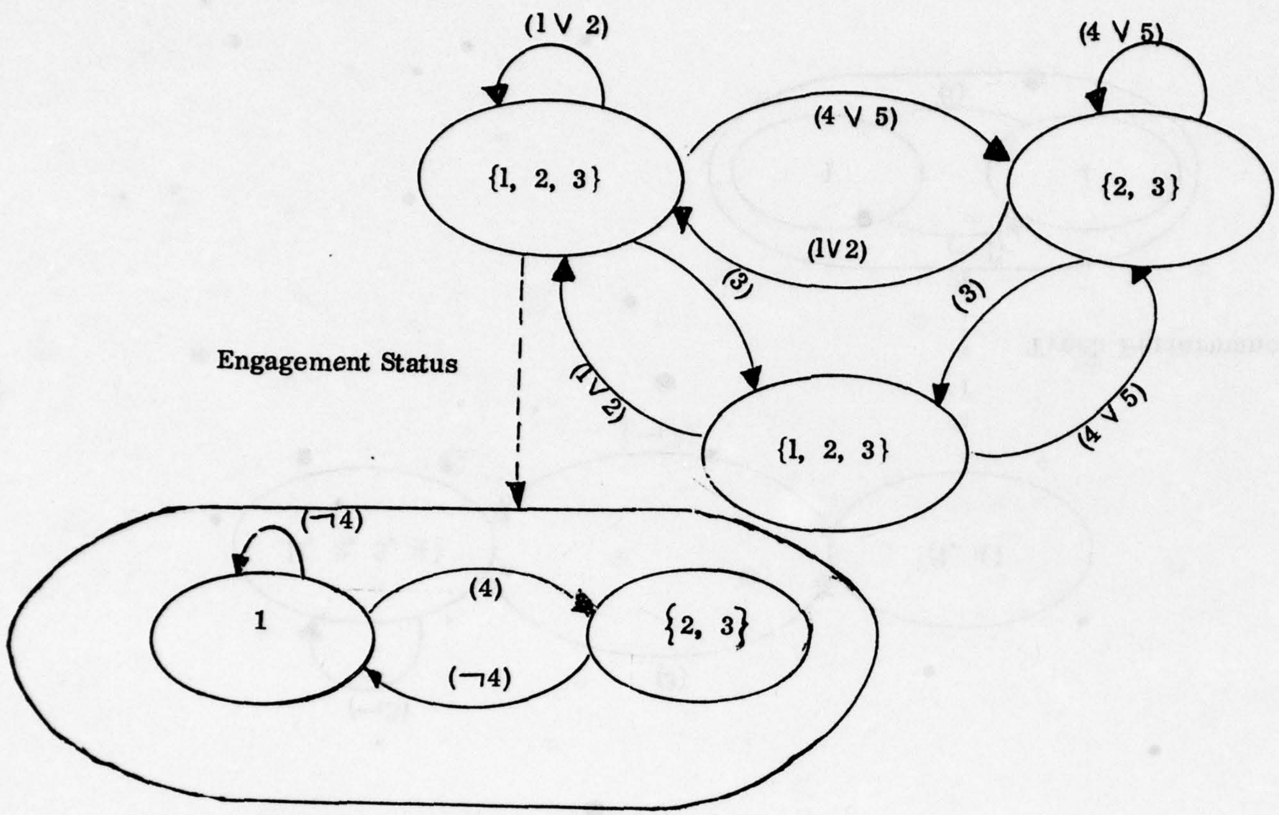
SYSTEM NAME: Engagement Status

INPUT SYSTEMS: Tactics, Track Performance



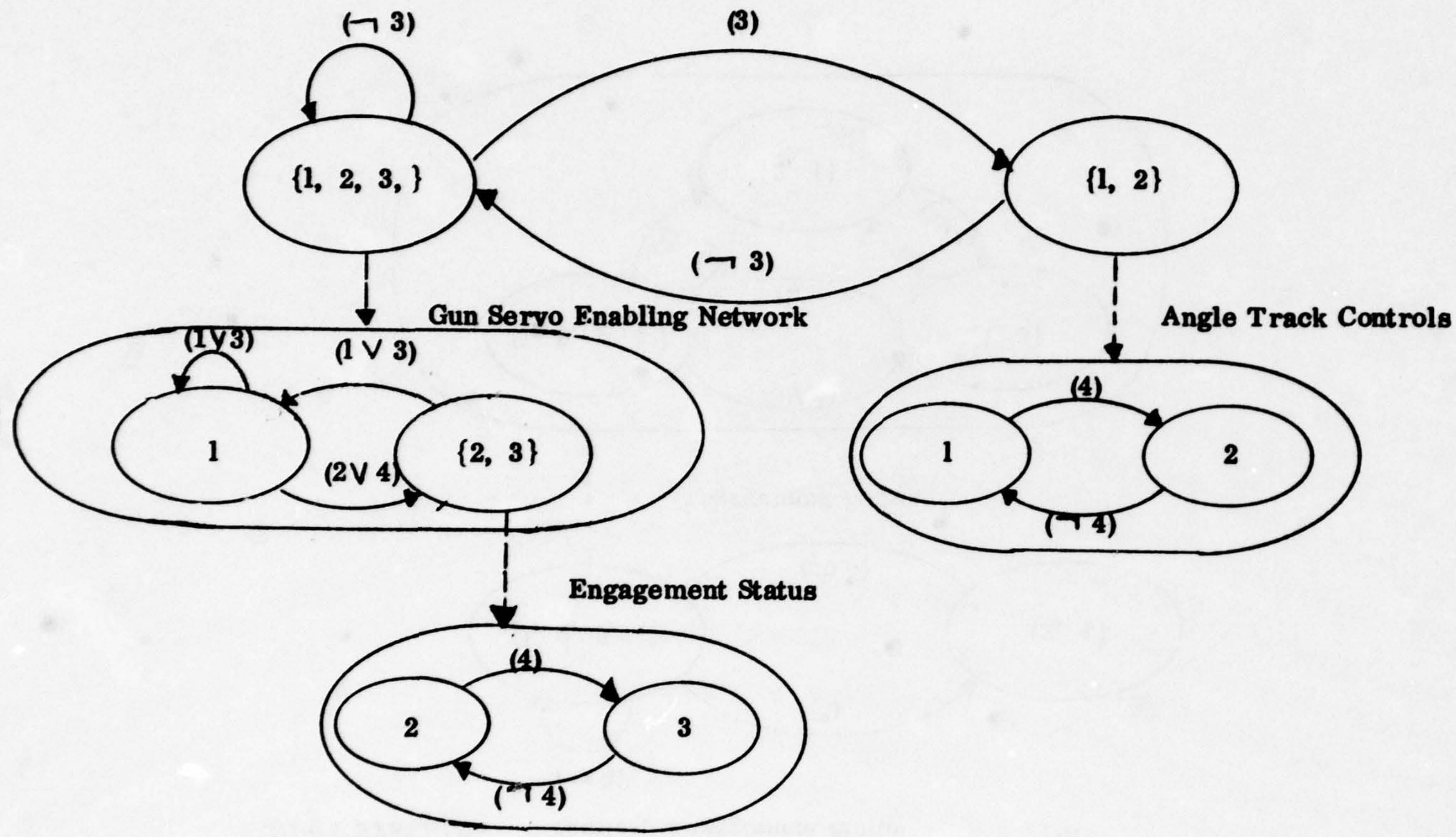
SYSTEM NAME: Fire Control Network

INPUT SYSTEM: Tactics, Engagement Status



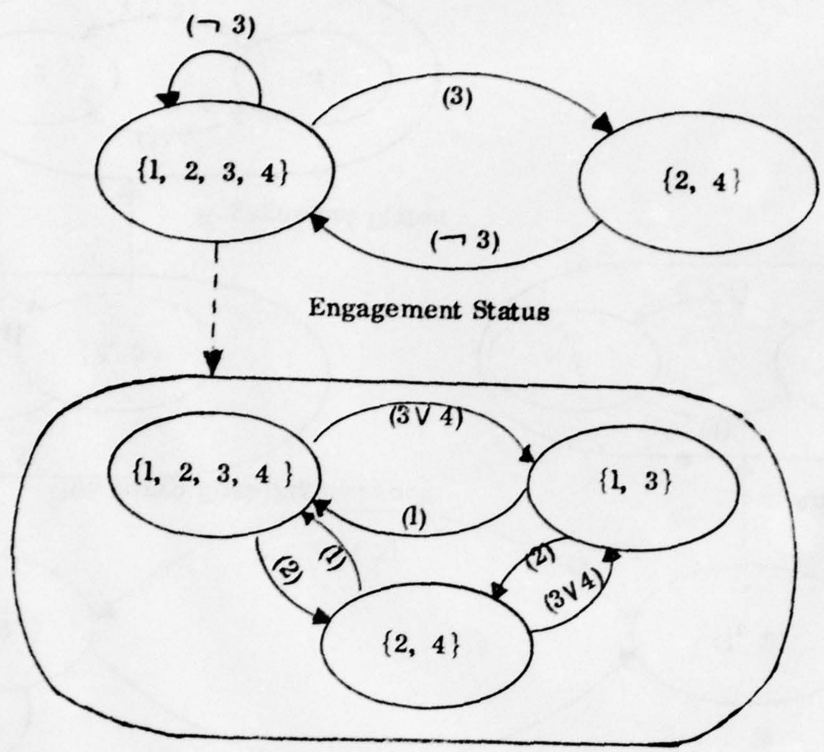
SYSTEM NAME: Gun Directing System

INPUT SYSTEMS: Tactics, Gun Servo Enabling Net, Angle Track Controls, Engagement Status



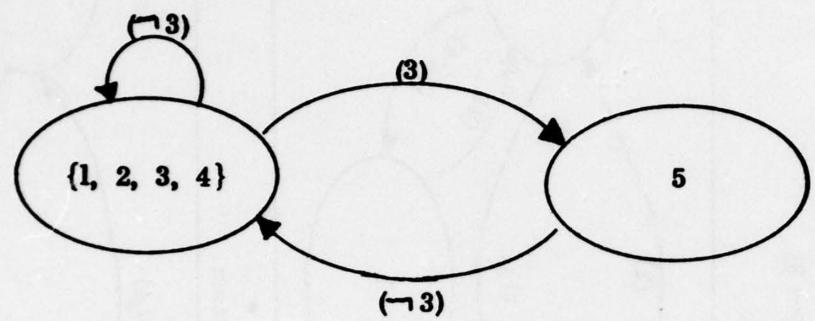
SYSTEM NAME: Angle Track System

INPUT SYSTEMS: Tactics, Engagement Status



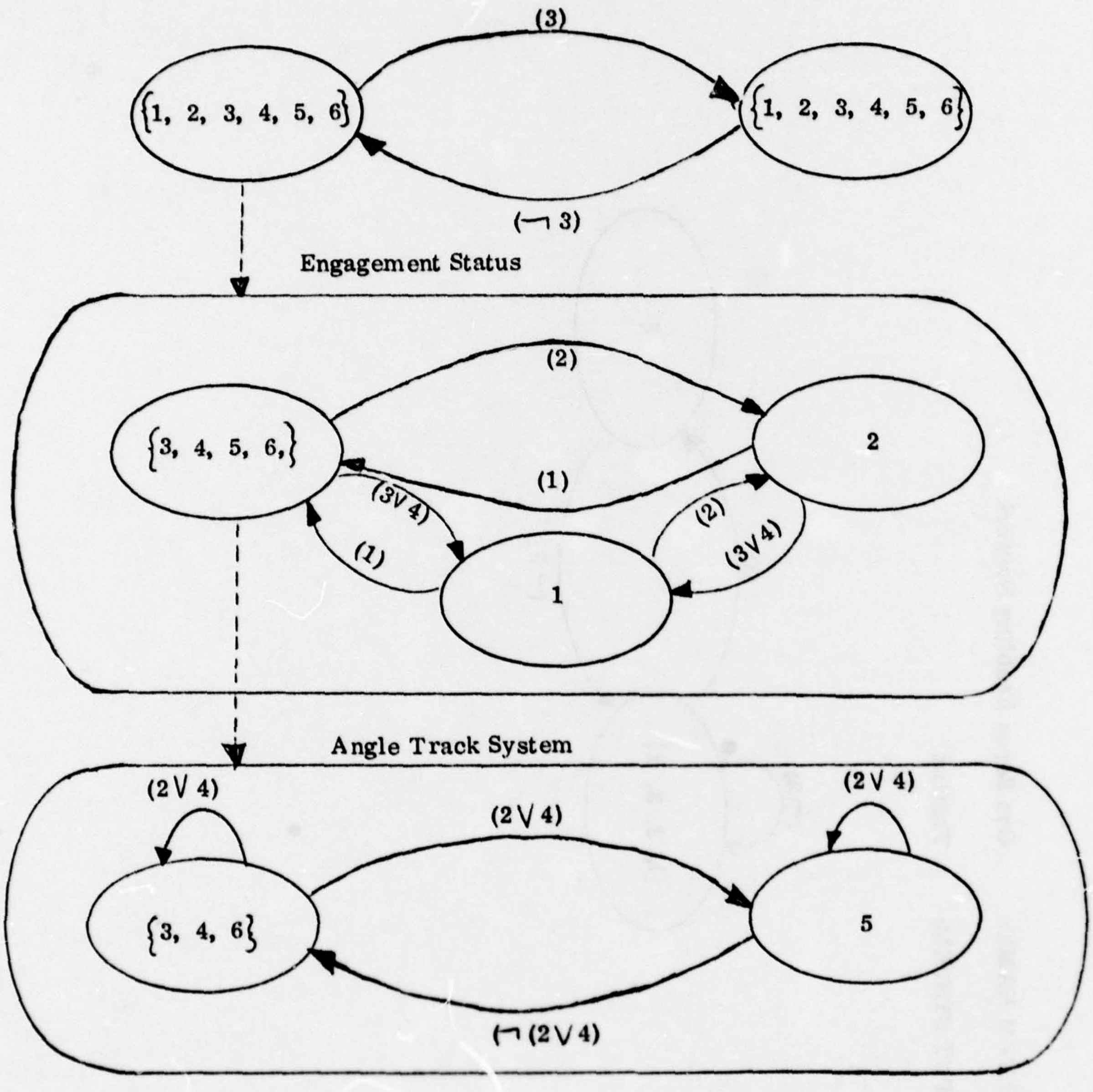
SYSTEM NAME: Gun Servo Enabling Network

INPUT SYSTEM: Tactics



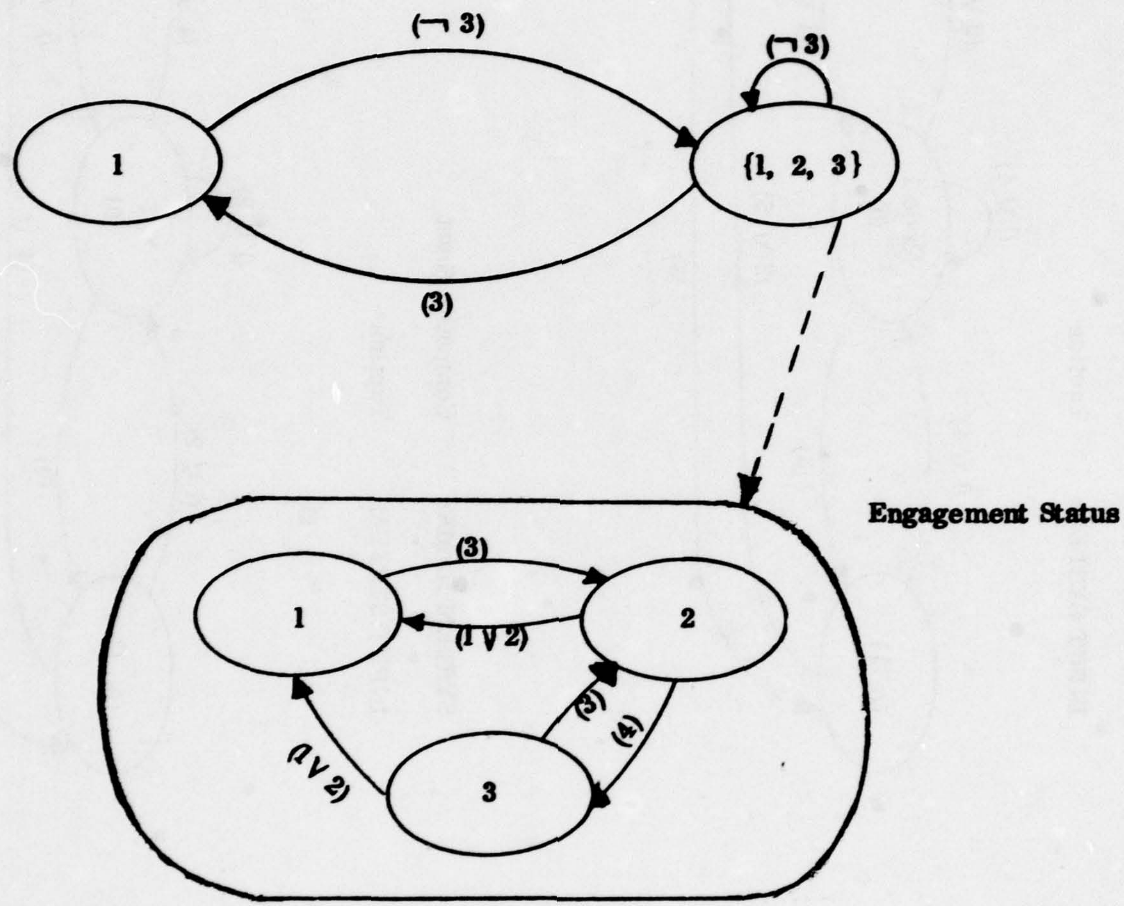
SYSTEM NAME: Radar Antenna Drive

INPUT SYSTEMS: Tactics, Engagement Status, Angle Track System



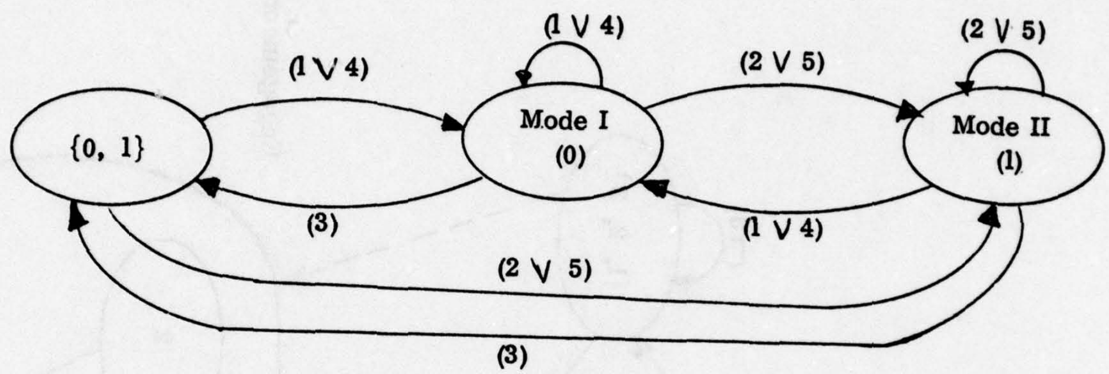
SYSTEM NAME: Computer

INPUT SYSTEMS: Tactics, Engagement Status



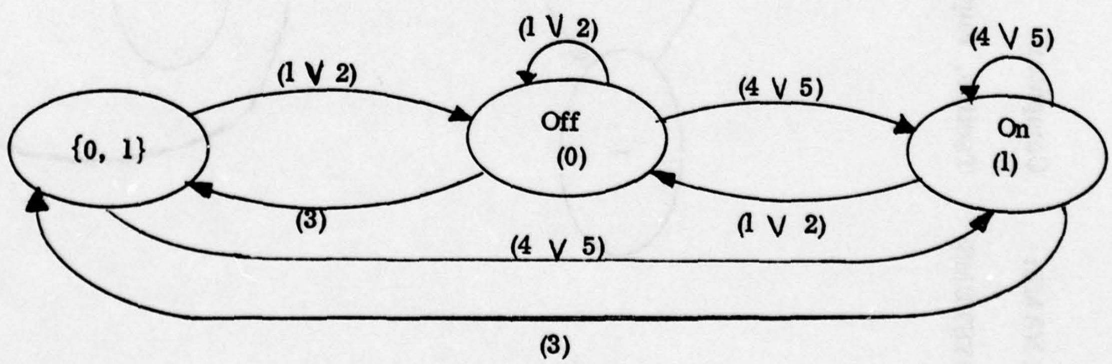
SYSTEM NAME: Mode Switch

INPUT SYSTEM: Tactics



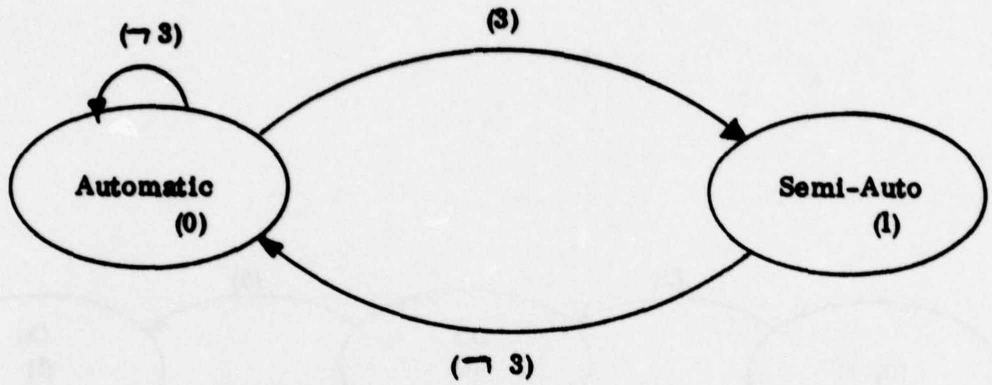
SYSTEM NAME: Computer Shunt

INPUT SYSTEM: Tactics



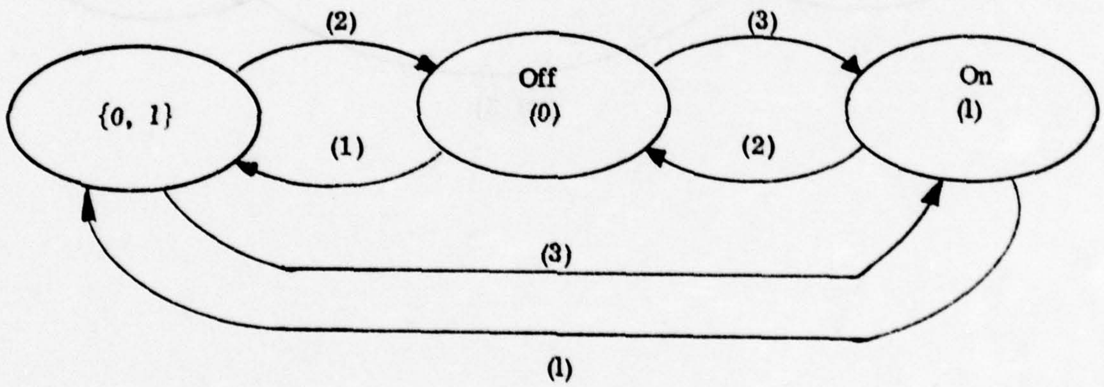
SYSTEM NAME: Gun Servo Mode

INPUT SYSTEM: Tactics



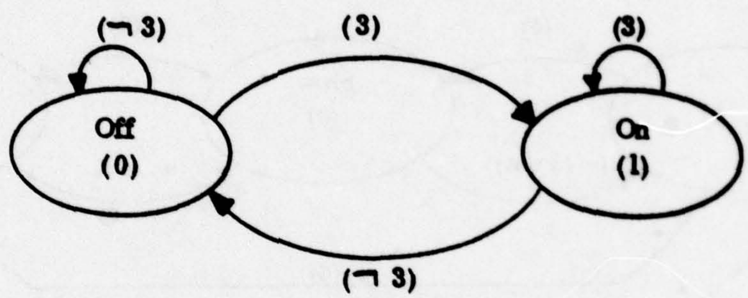
SYSTEM NAME: Coolant

INPUT SYSTEM: Fire Control Network



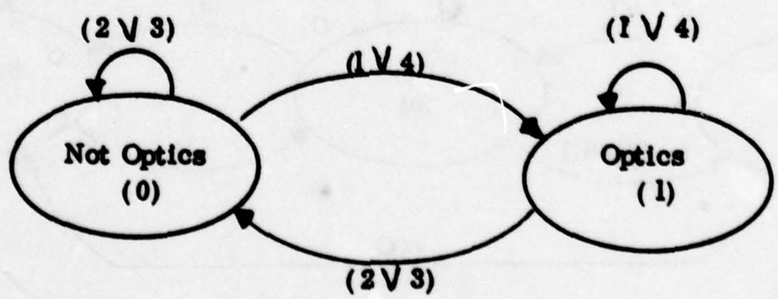
SYSTEM NAME: Data Ready Indicator

INPUT SYSTEM: Computer



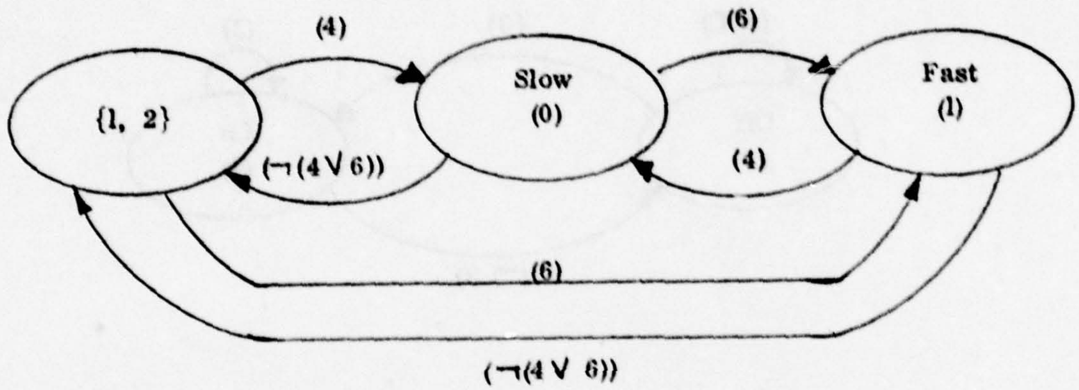
SYSTEM NAME: Display in Use

INPUT SYSTEM: Angle Track System



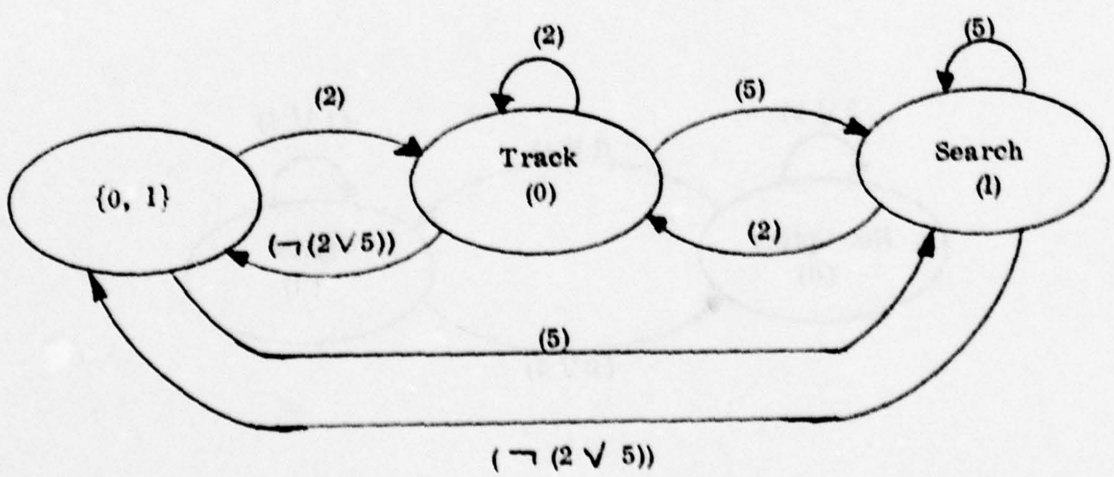
SYSTEM NAME: Automatic Circular Scan

INPUT SYSTEM: Radar Antenna Drive



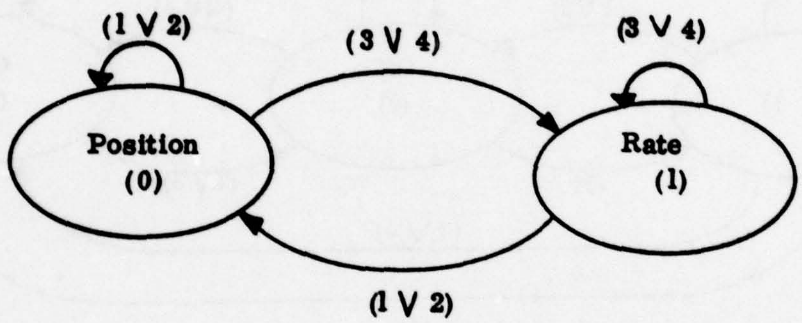
SYSTEM NAME: Antenna Horn Switch

INPUT SYSTEM: Radar Antenna Drive



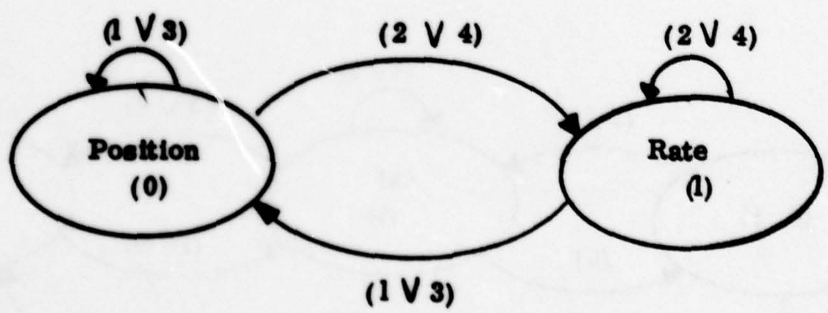
SYSTEM NAME: Azimuth Control

INPUT SYSTEM: Angle Track Controls



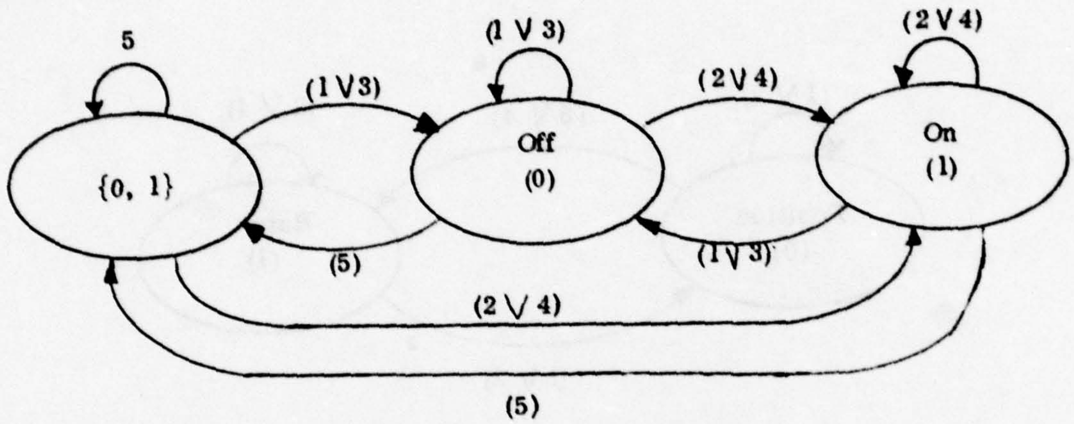
SYSTEM NAME: Elevation Control

INPUT SYSTEM: Angle Track Controls



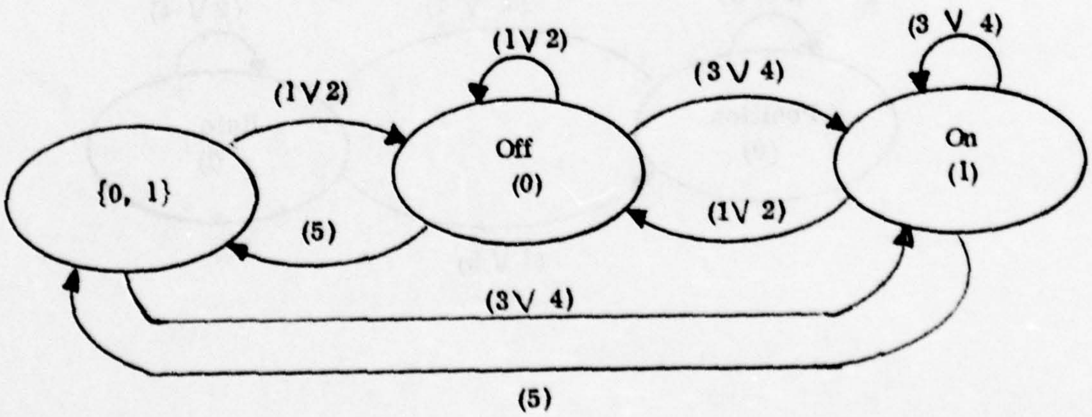
SYSTEM NAME: 0^{o} Lead Enable

INPUT SYSTEM: Gun Servo Enabling Net



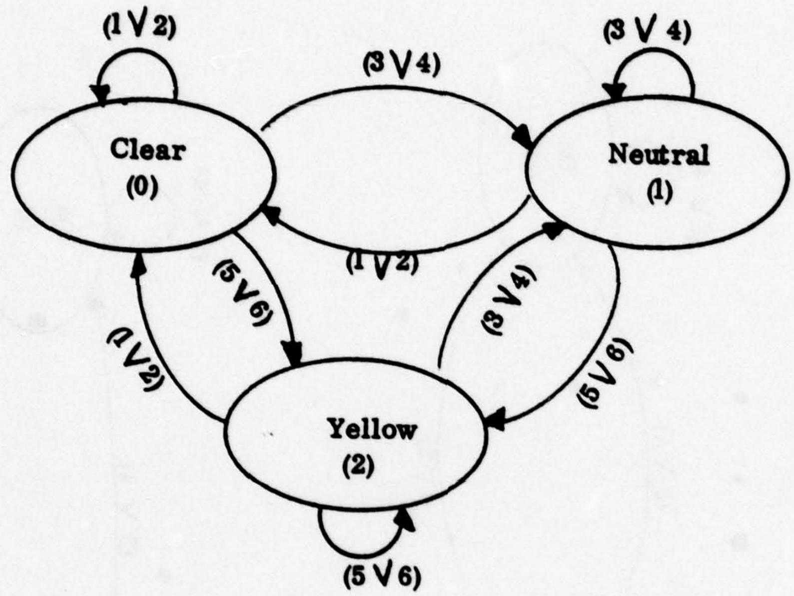
SYSTEM NAME: Lead Enable

INPUT SYSTEM: Gun Servo Enabling Net



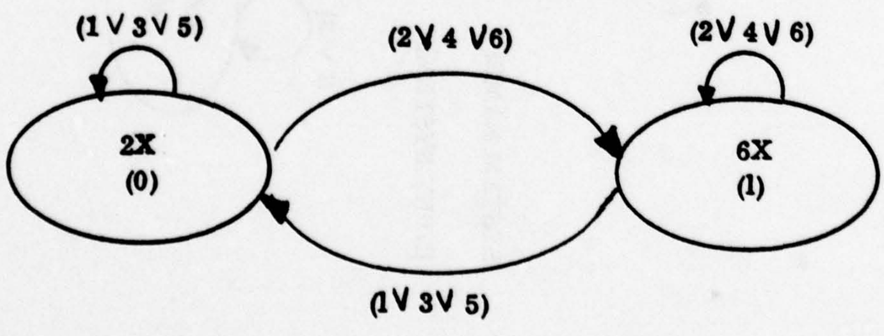
SYSTEM NAME: Sight Filter

INPUT SYSTEM: Sight System



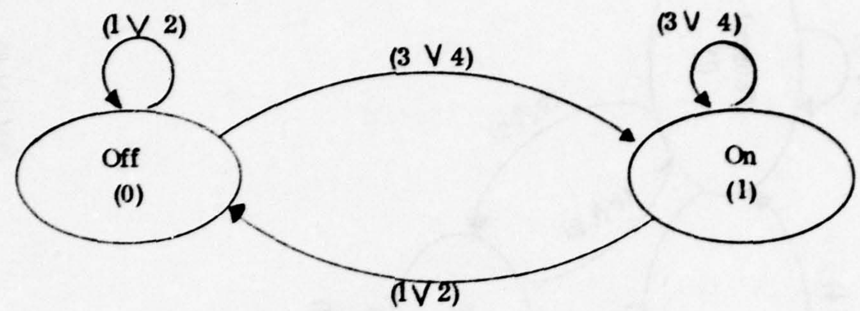
SYSTEM NAME: Magnification

INPUT SYSTEM: Sight System



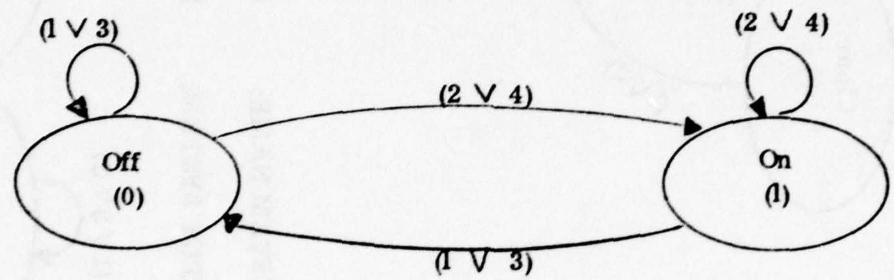
SYSTEM NAME: Lower Barrel

INPUT SYSTEM: Gun Configuration



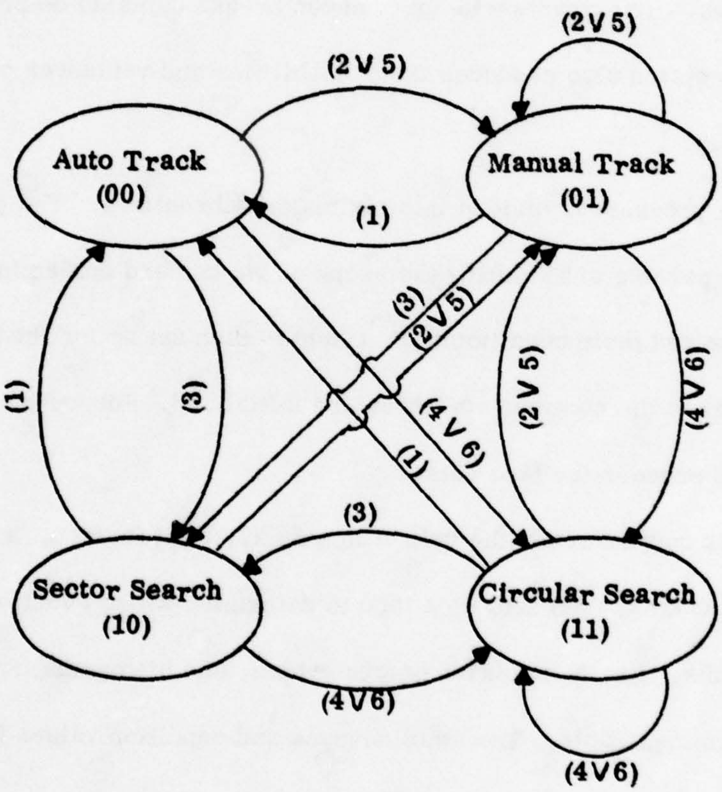
SYSTEM NAME: Upper Barrel

INPUT SYSTEM: Gun Configuration



SYSTEM NAME: Radar Mode

INPUT SYSTEM: Radar Antenna Drive



APPENDIX 5

ANALYSIS ROUTINES

The time average and variances between events for up to 22 variables can be calculated by this program. It also can have each variable conditioned by up to four variables and then calculate time and variance in conditional states between events. Histograms with up to seven breaks can also be printed. The histograms option also produces the probabilities and variances of the occurrence of each event.

The program is divided into six major subroutines. The program first reads user parameters limiting the scope of the run and determining the number of variables and their conditioning. A loop is then set up for the number of teams or runs. Then the common variables are initialized. Subroutine INPUT is called to read and process the tape data.

This routine reads the values in a 50(22I2, I5) format. It checks the range of the data values, then sets up a loop to determine which event occurred. Time before events, time in condition before events, and histogram input values are tallied when applicable. The state changes and condition values for each variable are coded and stored in a table with a pointer to indicate the correct array location. After all events are processed, these tables are sorted into descending order.

Subroutine OUTPUT takes the sorted values and calculates the appropriate averages, variances and counts for each variable at each condition and state. The values are printed with proper headings on the first pass. Correct averages and deviations are stored for histogram processing. The histograms are printed the second pass.

HISTIN is a simple routine that determines the correct information to save for the histograms. HISTOT, called by OUTPUT, prints out the histogram values. A flow chart of each subroutine and a list of common variables follows.

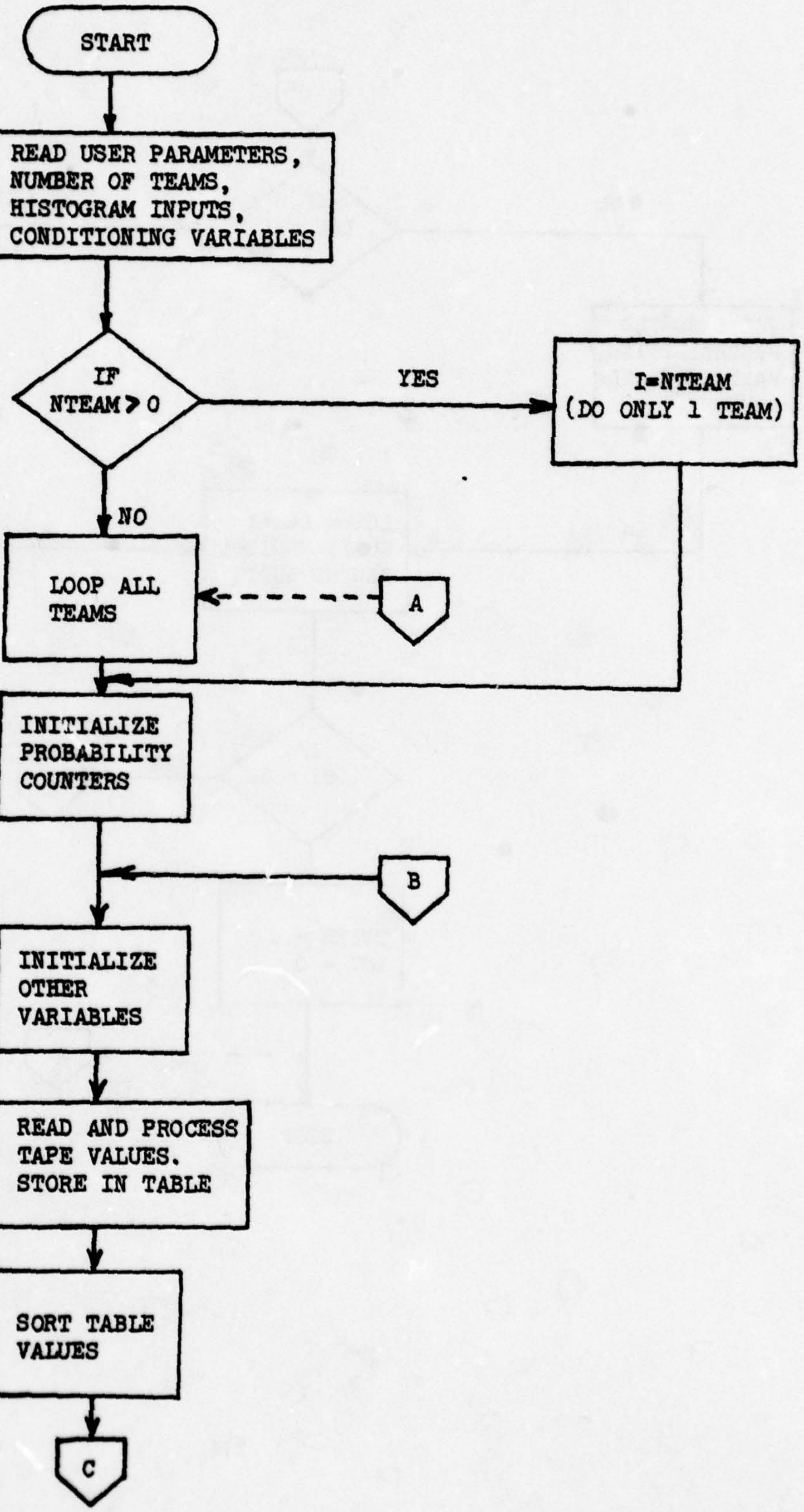
User information inputted in PARAM is important. The first data card (6I5 format) sets the program scope. KPARA is the number of variables to be processed. MARKOV is the order of the Markov chain (assumed one). NTEAM is the number of teams or runs and is used for tape mounts. NBLOCK indicates the number of events per data block. LZ lists the number of histogram breaks, and LX determines which histogram to process ($> 0 \Rightarrow$ Times; $> 0 \Rightarrow$ Times in condition).

If a histogram is to be processed (LX > 0), LZ histogram breaks are read in on the next data card. Up to seven breaks in a 7F5.2 format can be entered. Break values will become the sigma coefficients added to the means and used as data limits.

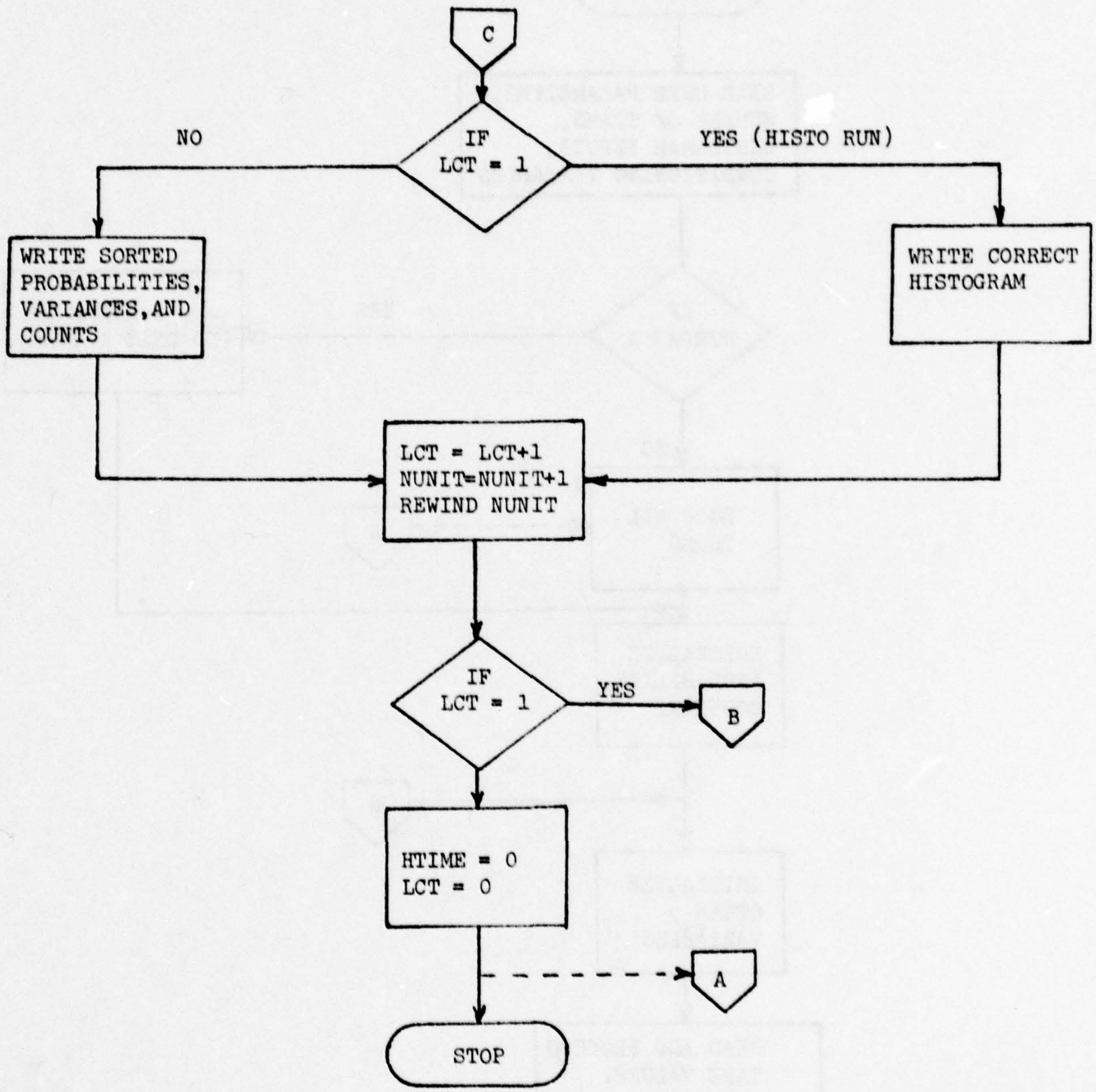
Conditional variables are entered next. These values are in the format of the variable, its maximum state and up to four conditioning variables (6I5 format). Unconditioned variables have maximum state values set to seven. A blank data card delimits.

This program was designed to be as general as possible. Data storage in coded form is by tables by variable for convience. Many checks exist to limit user or data errors. These checks and the common variable's dimensions need to be changed only when the program asks for them.

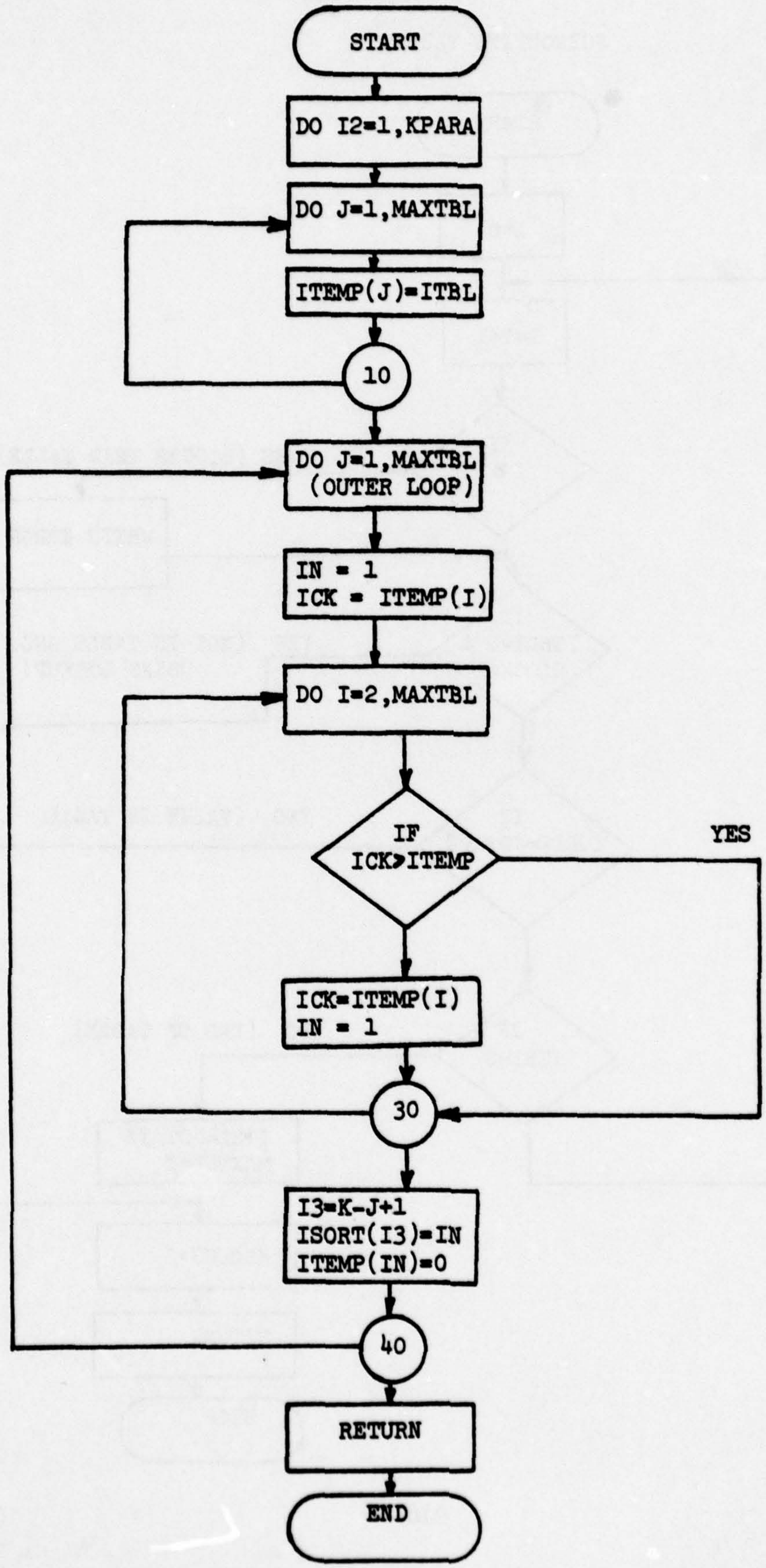
MAIN


START
READ USER PARAMETERS, NUMBER OF TEAMS, HISTOGRAM INPUTS, CONDITIONING VARIABLES
IF NTEAM > 0
YES
I=NTEAM (DO ONLY 1 TEAM)
NO
LOOP ALL TEAMS
A
INITIALIZE PROBABILITY COUNTERS
B
INITIALIZE OTHER VARIABLES
READ AND PROCESS TAPE VALUES. STORE IN TABLE
SORT TABLE VALUES
C

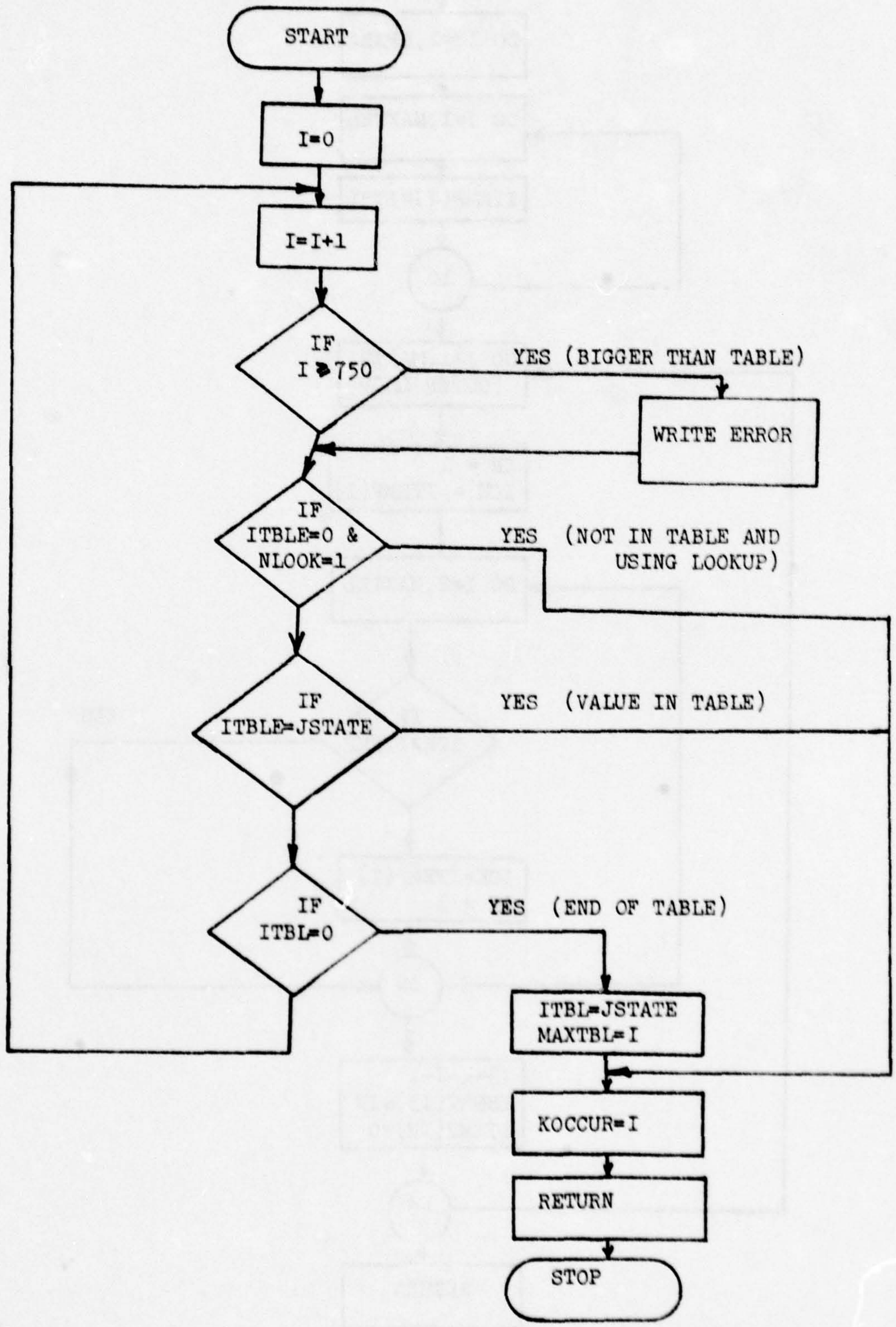
MAIN CONT.


C
IF
LCT = 1
NO
YES (HISTO RUN)
WRITE SORTED PROBABILITIES, VARIANCES, AND COUNTS
WRITE CORRECT HISTOGRAM
LCT = LCT+1
NUNIT=NUNIT+1
REWIND NUNIT
IF
LCT = 1
YES
B
HTIME = 0
LCT = 0
A
STOP

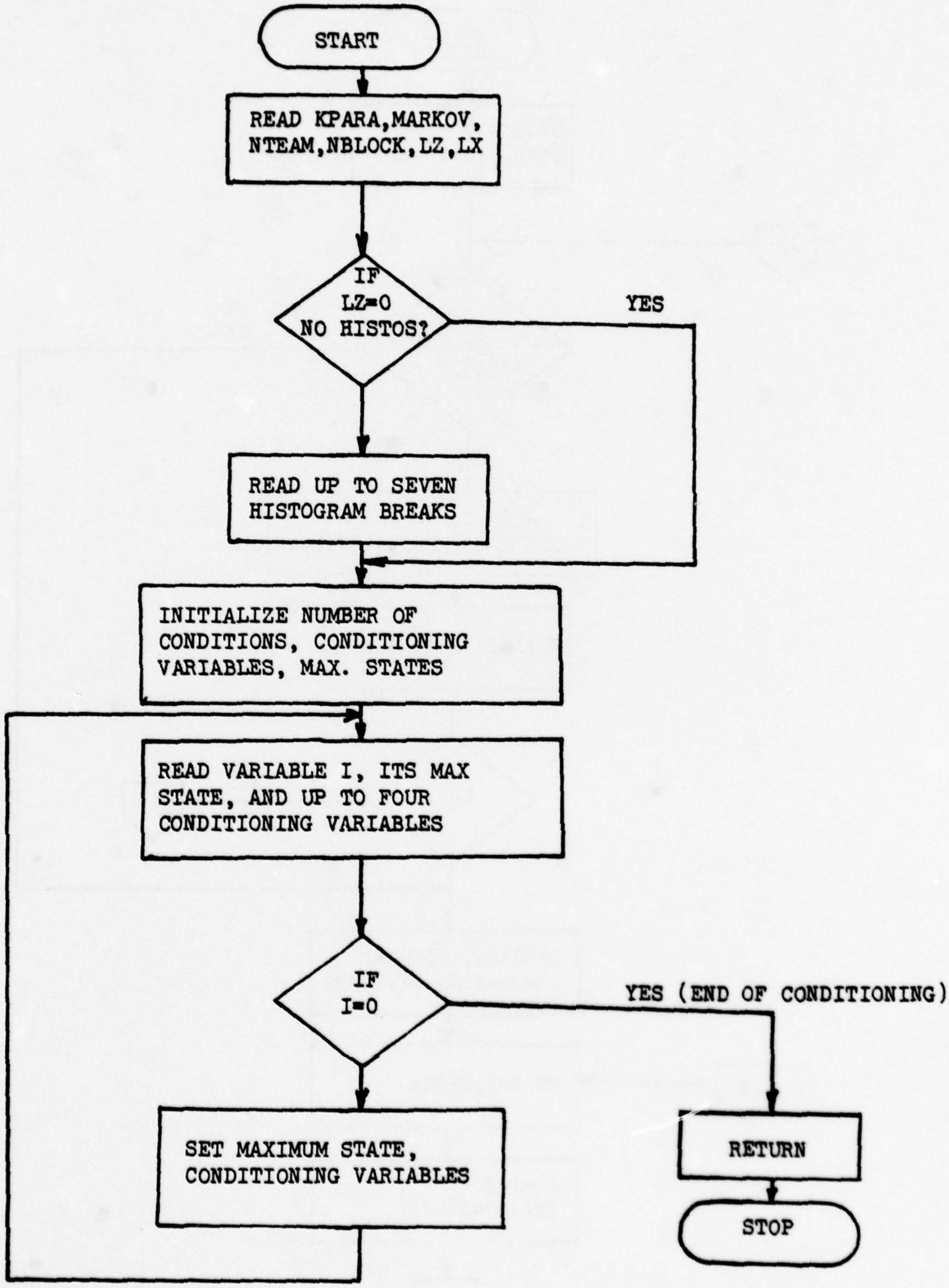
SUBROUTINE SORT



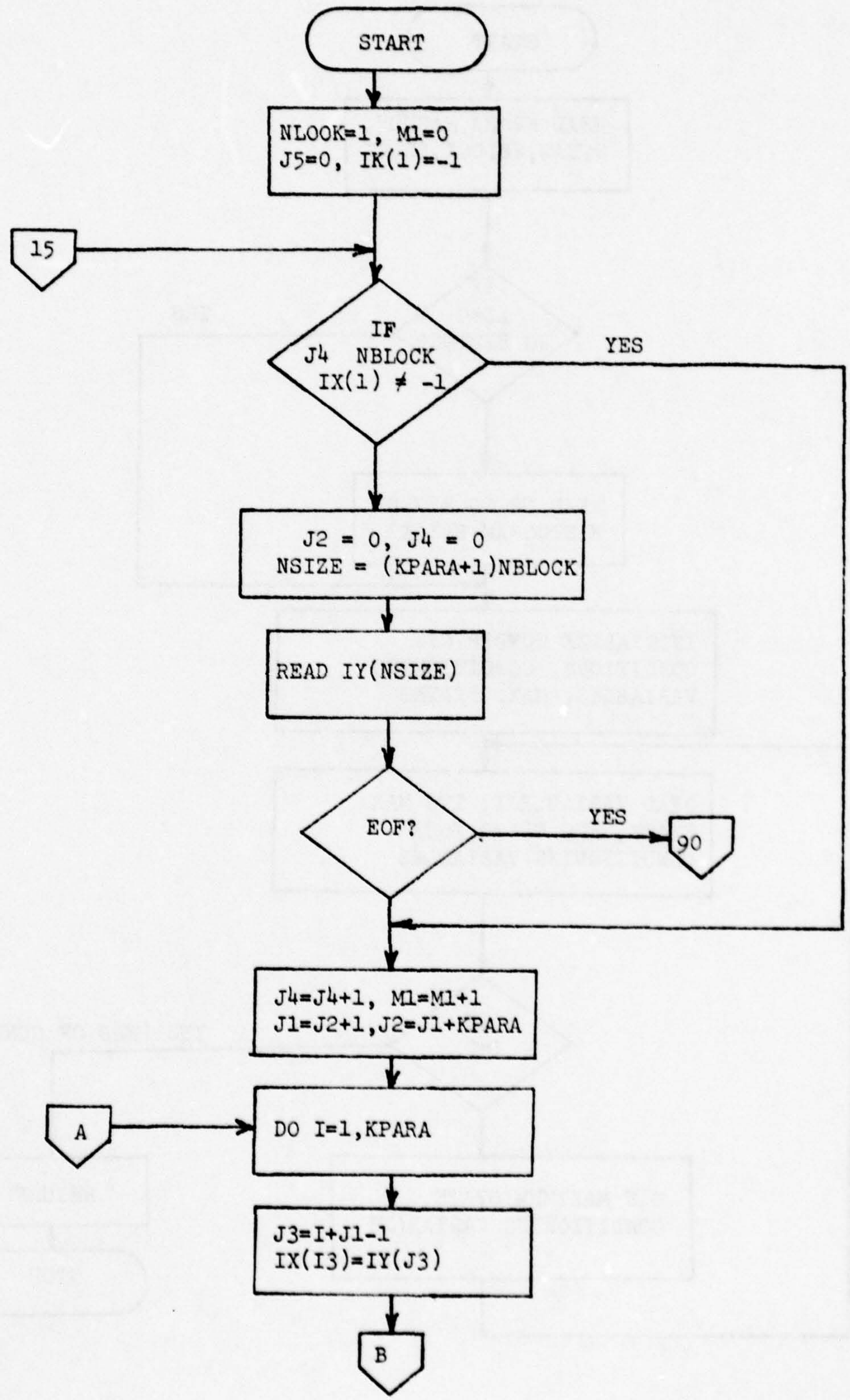
SUBROUTINE TABLE



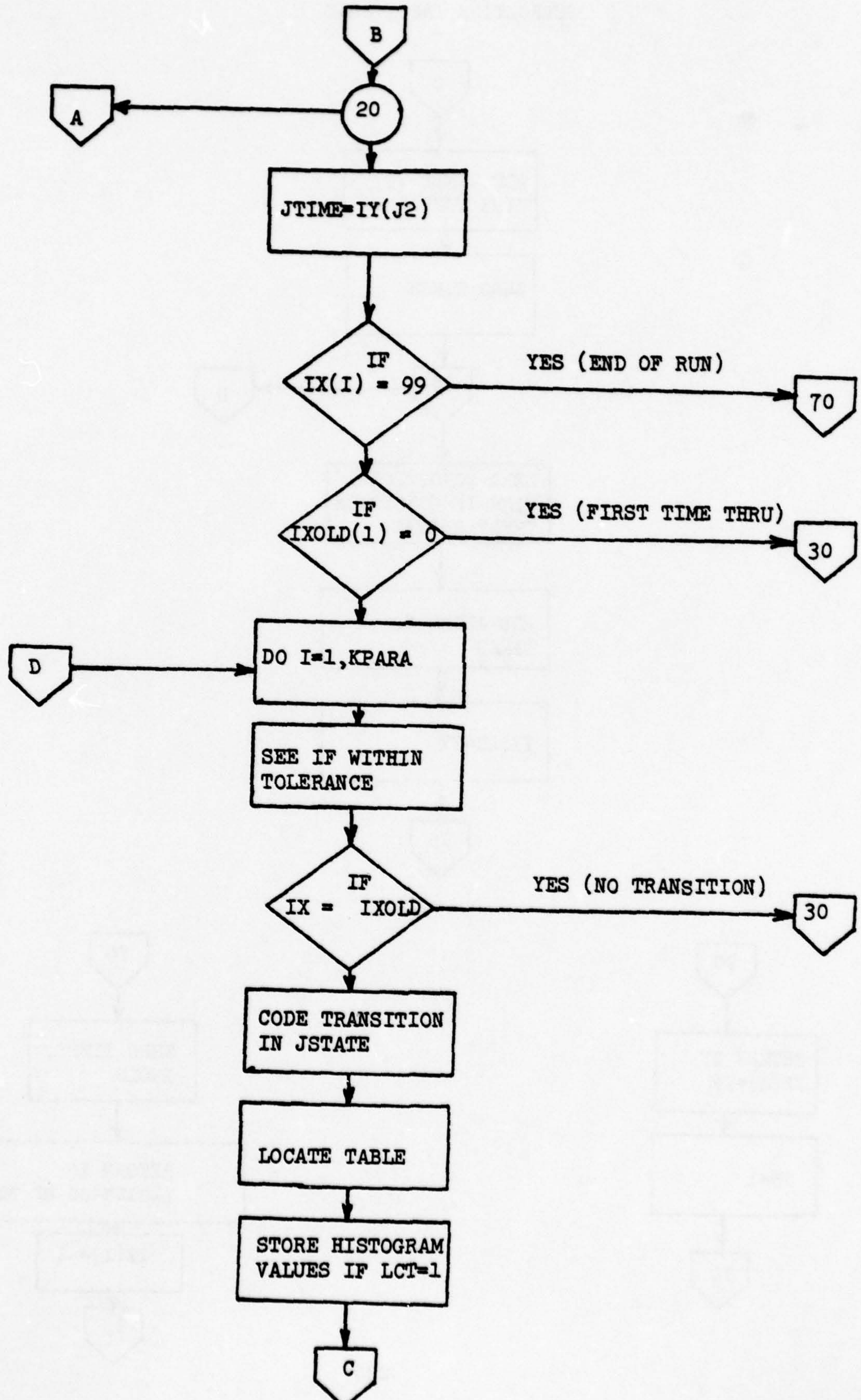
SUBROUTINE PARAM



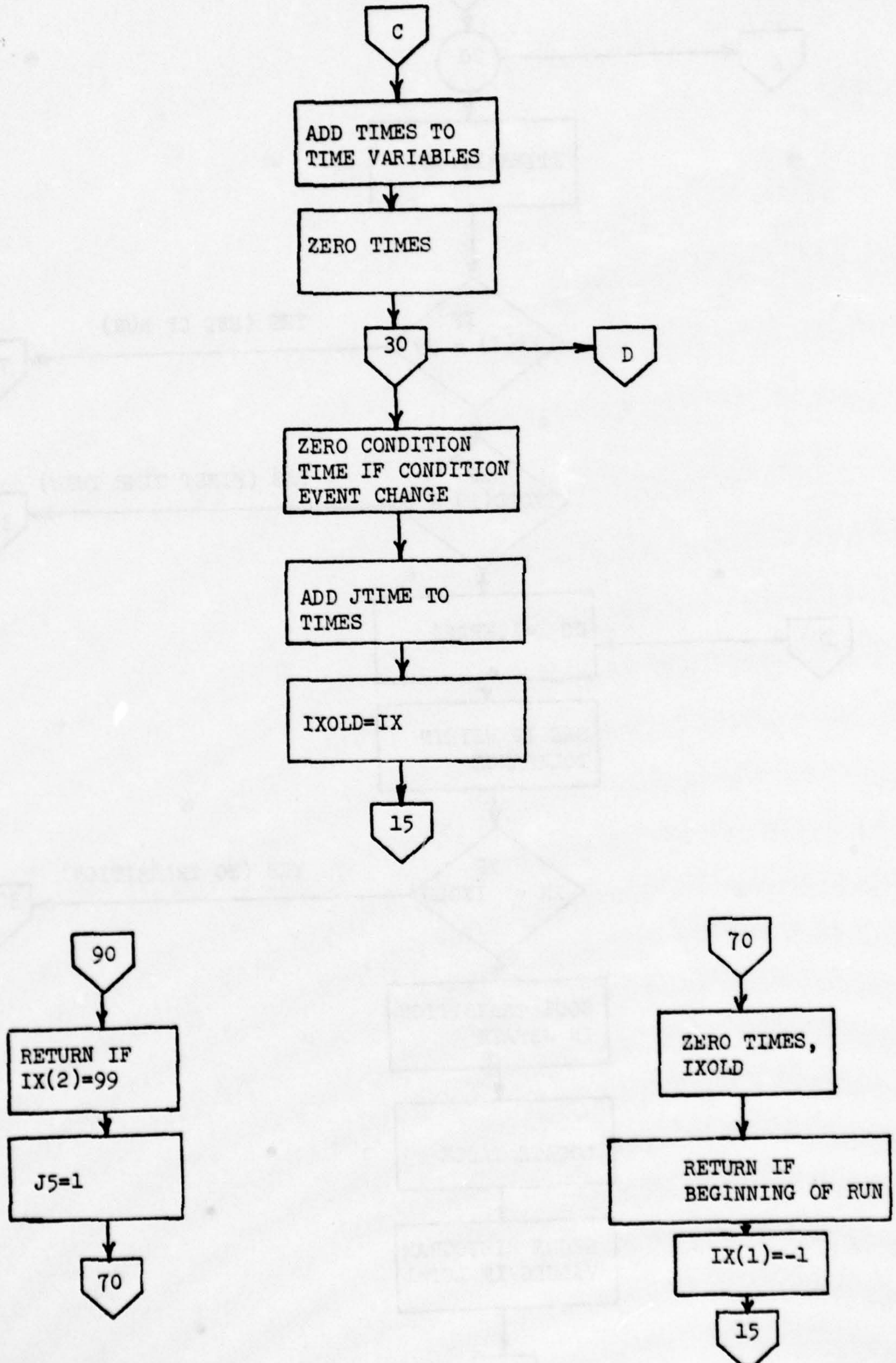
SUBROUTINE INPUT



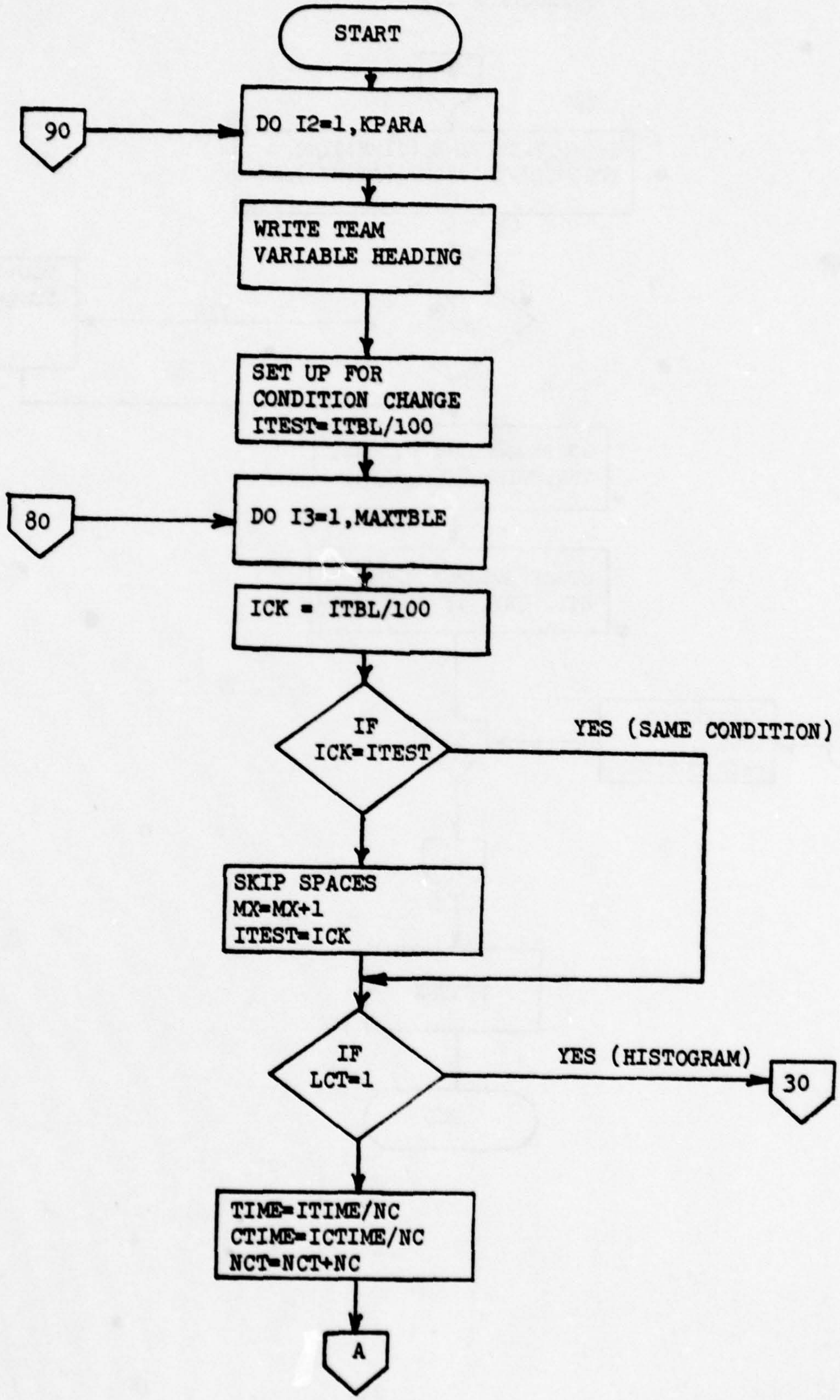
SUBROUTINE INPUT CONT.



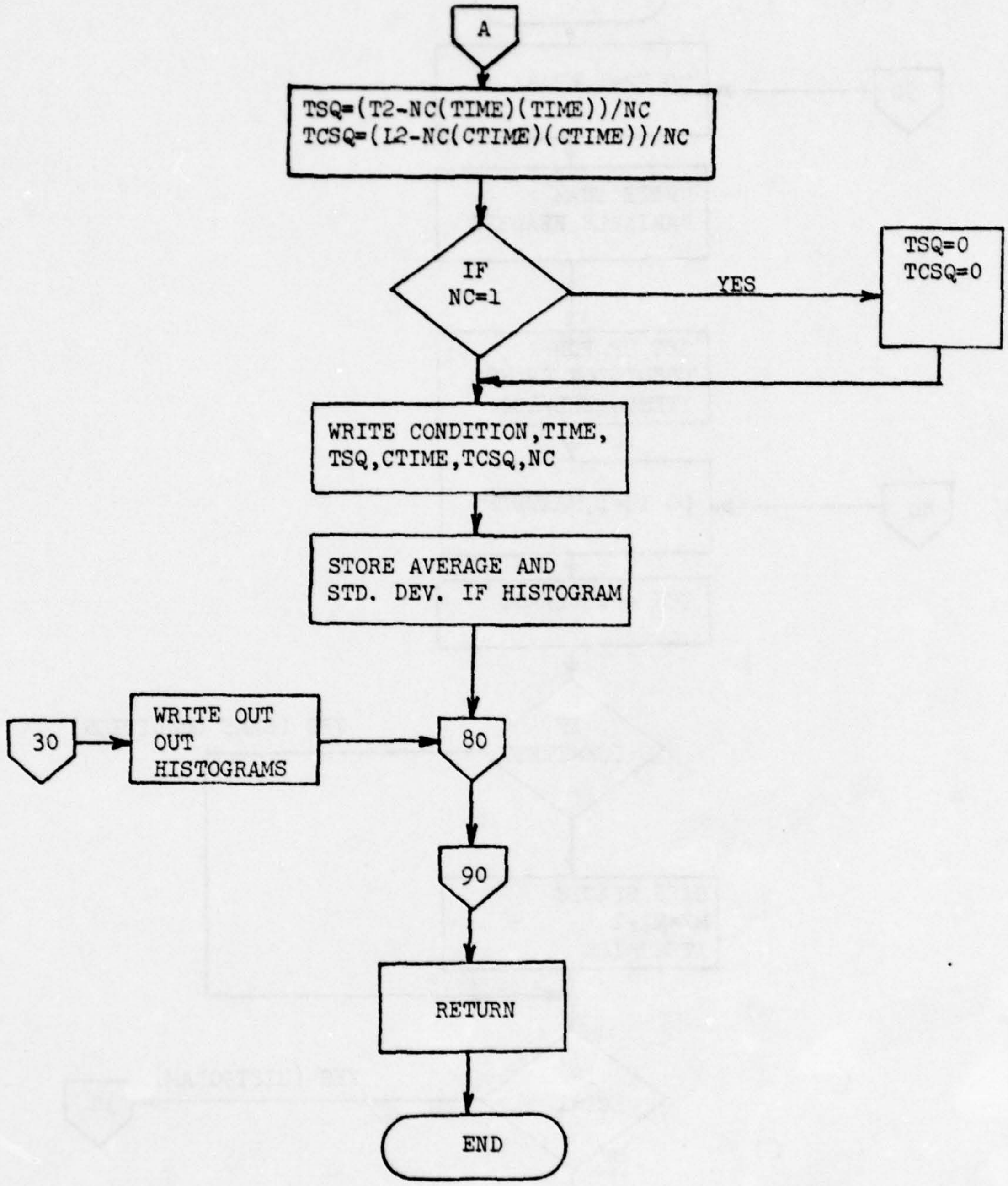
SUBROUTINE INPUT CONT.



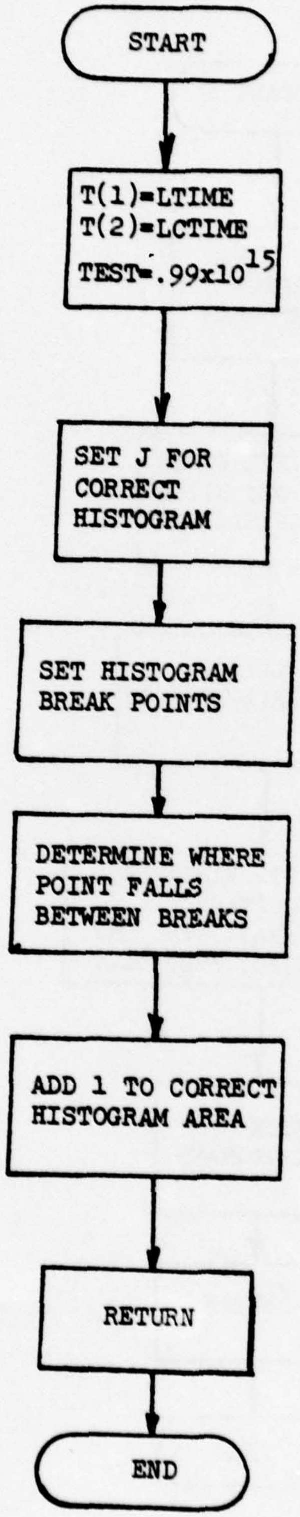
SUBROUTINE OUTPUT



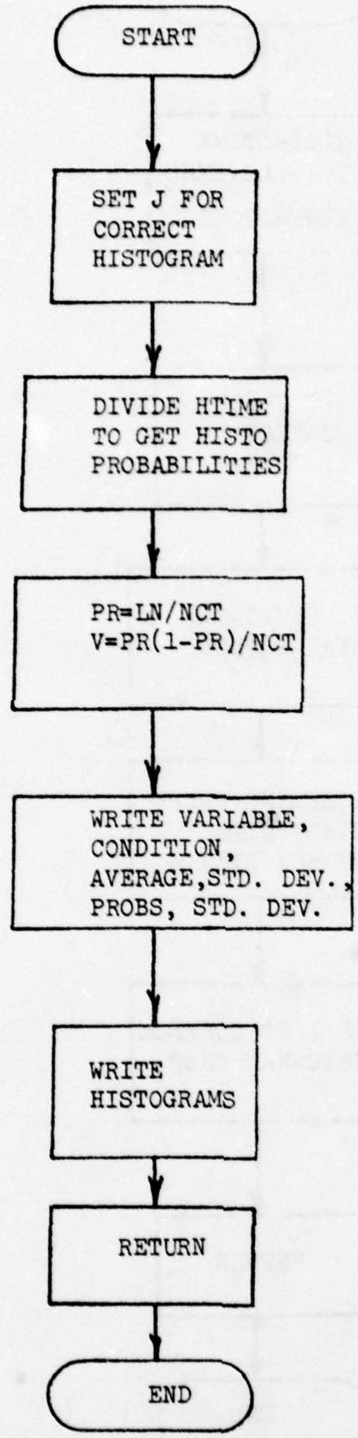
SUBROUTINE OUTPUT CONT.



SUBROUTINE HISTIN



SUBROUTINE HISTOT



```
C      MAIN PROGRAM
C              THE FOLLOWING IS A LIST OF COMMON VARIABLES
C
C            ELEMENTS OF COMMON /A/
C                      (VALUES ENTERED IN PARAM)
C      NBLOCK - # OF ELEMENTS PER TAPE BLOCK (KPARA+1)*NBLOCK=BLKSIZE
C      KPARA - NUMBER OF PARAMETERS
C      KCOND(22) - NUMBER OF CONDITIONING VARIABLES FOR VARIABLE I
C      JST(22,4) - CONDITIONING VARIABLE FOR VARIABLE I
C      IXMAX(22) - MAXIMUM STATE SIZE OF VARIABLE I (ASSUMED=7)
C      MARKOV - NUMBER OF MARKOV CHAIN - NOT USED
C      NTEAM - TEAM OR DATA SET; <O=> ALL THREE
C
C          ELEMENTS OF COMMON /B/
C      ITIME(22,250) - TIME IN STATE
C      T2(22,250) - TIME SQUARED IN STATE
C      NC(22,250) - TIME COUNTS
C      ICOND(22,250) - TIME IN CONDITION STATE
C      IC2(22,250) - TIME SQUARED IN CONDITION
C
C              ELEMENTS IN COMMON /C/
C      JSTATE - CONDITION OF VARIABLE
C      ITBL(3,22,250) - TABLE OF CONDITIONS
C      MAXTBL(3,22) - SIZE OF TABLE I
C      ISORT(3,22,250) - SORTED TABLE VALUES
C      NLOOK - 1=> TABLE LOOK-UP  OTHERWIZE=> TABLE ADD TO
C
C
C            ELEMENTS IN COMMON /D/
C      **    USED IN THE HISTOGRAM ROUTINES   **
C      AVE(22,750) - AVERAGE TIME VALUE
C      DEV(22,750) - STANDARD DEVIATION
C      HTIME(22,750,8) - UP TO 8 BREAKS FOR HISTOGRAM
C      LCT - HITOGRAM INDICATOR
C      H(8) - HISTOGRAM BREAKS (TIMES DEVIATIONS)
C      LZ - NUMBER OF HISTOGRAM BREAKS (<8)
C      LX - <0 => CONDITIONED HISTOS; >0 => NORMAL HISTOS
C      NCT(22,500,7) - PROBABILITY COUNTER
C      MX - MARKER FOR PROBABLITY EVENT
C
       COMMON/A/ NBLOCK,KPARA,KCOND(22),JST(22,4),IXMAX(22),MARKOV,NTEAM
       COMMON/B/ITIME(22,750),T2(22,750),NC(22,750),ICOND(22,750)
       COMMON/C/JSTATE,ITBL(2,22,750),MAXTBL(2,22),ISORT(2,22,750),NLOOK
       COMMON/D/AVE(22,750),DEV(22,750),HTIME(22,750,8),LCT,H(8),LX,LZ
       COMMON C2(22,750),NCT(22,500,7),MX
       REAL ITIME,ICOND
       LCT=99
C  -   -  CALL THE PARAMETER INPUT ROUTINE
       CALL PARAM
C  -   -    LOOP FOR CORRECT TEAM; SET UNIT NUMBER
       I=NTEAM
       IF(NTEAM.GT.0)GO TO 12
       NT=-NTEAM
       DO 10 I=1,NT
   12  CONTINUE
       NTEAM=I
C  -   -    INITIALIZE PROB COUNTER
       DO 15 I2=1,22
       DO 15 I3=1,500
       DO 15 I4=1,7
       NCT(I2,I3,I4)=0
   15  CONTINUE
```

```
      M=0
C - -   CALL INITIALIZATION ROUTINE
      CALL INIT
C - -   CALL TAPE READ ROUTINE
      CALL INPUT(M)
C - -   CALL TABLE SORT ROUTINE
      CALL SORT
C - -   CALL CALCULATION AND OUTPUT ROUTINE
      CALL OUTPUT(I)
C - -   WRITE TEAM PARAMETER SUMMARY
      WRITE(6,100)I,M
  100 FORMAT(/////15X,'PARAMETER SUMMARY FOR TEAM',I3,
     1 ///10X,'THE TOTAL NUMBER OF EVENTS OCCURRING EQUALS',I10)
C - -   HISTOGRAM SETUP; SKIP IF NO HISTOGRAMS
      IF(LZ.EQ.0)GO TO 10
      LCT=LCT+1
      NUNIT=7+I
      REWIND NUNIT
      IF(LCT.EQ.1) GO TO 15
C - -   INIT HISTOGRAM VARIABLES, THEN LOOP BACK
      DO 20 I2=1,8
      DO 20 I3=1,750
      DO 20 I4=1,22
   20 HTIME(I4,I3,I2)=0
      LCT=0
   10 CONTINUE
      STOP
      END
```

```
      SUBROUTINE PARAM
C - -     THIS ROUTINE READS IN THE PARAMETERS FOR THE RUN
      COMMON/A/ NBLOCK,KPARA,KCOND(22),JST(22,4),IXMAX(22),MARKOV,NTEAM
      COMMON/D/AVE(22,750),DEV(22,750),HTIME(22,750,8),LCT,H(8),LX,LZ
      DIMENSION L(4)
C - -    NO MORE THAN 22 VARIABLES WITH UP TO 4 CONDITIONS EACH CAN
C        BE USED WITHOUT CHANGING THE COMMONS
C
C        READ IN:  KPARA - # OF VARIABLES,  MARKOV - MARKOV ORDER
C        NTEAM - # OF TEAMS OR DATA SETS,  NBLOCK - # EVENTS PER TAPE 8
C       **  NTEAM<O=>ALL 3 TEAMS; >O=> ONLY THAT TEAM NUMBER
C         LZ -# OF HISTO BREAKS,    LX - WHICH HISTO
      READ(5,100)KPARA,MARKOV,NTEAM,NBLOCK,LZ,LX
      WRITE(6,120)KPARA,MARKOV,NTEAM,NBLOCK,LZ,LX
  120 FORMAT('1',///10X,'PARAMETER INPUT',///
     1 /5X,'NUMBER OF PARAMETERS =',I5,
     2 /5X,'ORDER OF MARKOV CHAIN=',I5,
     3 /5X,'NUMBER OF TEAMS OR RUNS=',I5,
     4 /5X,'BLOCK SIZE=',I5,
     5/5X,'SIZE OF HISTOGRAM=',I5,
     6  /5X,'WHICH HISTOGRAM=',I5///)
C - -    READ IN HISTOGRAM INFORMATION
      IF(LZ.EQ.0) GO TO 40
      LCT=0
      READ(5,110)(H(I),I=1,LZ )
  110 FORMAT(7F5.2)
      WRITE(6,130)(H(I),I=1,LZ)
  130 FORMAT(//10X,'HISTOGRAM LIMITS',/7F5.2,///10X,'CONDITIONING',///)
   40 CONTINUE
C - -    INITIALIZE
      DO 35 I=1,22
      KCOND(I)=0
      IXMAX(I)=7
      DO 30 J=1,4
   30 JST(I,J)=0
   35 CONTINUE
C - -    READ VARIABLES, MAXIMUM STATE VALUE, AND UP TO 4 CONDITIONING
C  **  VARIABLES NOT READ IN WILL HAVE MAX STATE EQUAL TO 7  **
   10 READ(5,100)I,IXMAX(I),(L(J),J=1,4)
      WRITE(6,100)I,IXMAX(I),(L(J),J=1,4)
  100 FORMAT(6I5)
      IF(I.EQ.0) GO TO 20
C - -    RECORD COND VARS & COUNT # OF COND
      DO 15 J=1,4
      IF(L(J).EQ.0) GO TO 15
      KCOND(I)=KCOND(I)+1
      JST(I,J)=L(J)
   15 CONTINUE
      GO TO 10
   20 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE INIT
C  -  - THIS ROUTINE INITIALIZES THE COMMON VARIABLES TO ZERO
      COMMON/A/ NBLOCK,KPARA,KCOND(22),JST(22,4),IXMAX(22),MARKOV,NTEAM
      COMMON/B/ITIME(22,750),T2(22,750),NC(22,750),ICOND(22,750)
      COMMON/C/JSTATE,ITBL(2,22,750),MAXTBL(2,22),ISORT(2,22,750),NLOOK
      COMMON C2(22,750),NCT(22,500,7),MX
      REAL ITIME,ICOND
      DO 40 I=1,22
      DO 30 J=1,750
      ITIME(I,J)=0.
      T2(I,J)=0.
      NC(I,J)=0
      ICOND(I,J)=0.
      C2(I,J)=0.
      DO 20 K=1,2
      ITBL(K,I,J)=0
      MAXTBL(K,I)=0
      ISORT(K,I,J)=0
   20 CONTINUE
   30 CONTINUE
   40 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE TABLE(KOCCUR,JTAB,IVAR)
C - -    THIS ROUTINE CHECKS THE CODED VALUES AGAINST A TABLE TO DETERM
C        IF IT HAS OCCURRED BEFORE.  KOCCUR - TABLE POSITION
C        JTAB - WHICH TABLE   IVAR - WHICH VARIABLE
      COMMON/C/JSTATE,ITBL(2,22,750),MAXTBL(2,22),ISORT(2,22,750),NLOOK
C - -    SET COUNTER TO ZERO THEN STEP THROUGH TABLE UNTIL CORRECT VALU
C        IS REACHED
      I=0
      IF(JTAB.GT.2.OR.IVAR.GT.22)WRITE(6,110)JTAB,IVAR
   10 I=I+1
C - -    WITHIN TABLE LIMITS?
      IF(I.GE.750)WRITE(6,100)JTAB,IVAR
C - -    NOT IN TABLE AND USING A LOOK-UP? -> RETURN A ZERO
      IF(ITBL(JTAB,IVAR,I).EQ.0.AND.NLOOK.EQ.1) GO TO 20
C - -    VALUE IN TABLE? => RETURN POSITION
      IF(ITBL(JTAB,IVAR,I).EQ.JSTATE) GO TO 20
C - -    VALUE NOT IN TABLE? =>RETURN NEXT OPEN POSITION
      IF(ITBL(JTAB,IVAR,I).EQ.0) GO TO 15
      GO TO 10
C - -    ADD A NEW VALUE TO THE TABLE AND INCREASE TABLE SIZE
   15 ITBL(JTAB,IVAR,I)=JSTATE
      MAXTBL(JTAB,IVAR)=I
C - -    RETURN TABLE POSITION
   20 KOCCUR=I
  100 FORMAT(///' **  WARNING!!  **',//' TABLE',I2,' VARIABLE',I3,
     1   ' OVERFLOW')
  110 FORMAT('1'///' **  FATAL ERROR  **',///' TABLE OR VARIABLE DOES',
     1   ' NOT EXIST:',2I8)
      RETURN
      END
```

```
      SUBROUTINE INPUT(M)
C - - THIS ROUTINE READS THE DATA TAPES, DETERMINES THE TRANSITIONS,
C - - SUMS THE TIMES
      COMMON/A/ NBLOCK,KPARA,KCOND(22),JST(22,4),IXMAX(22),MARKOV,NTEAM
      COMMON/B/ITIME(22,750),T2(22,750),NC(22,750),ICOND(22,750)
      COMMON/C/JSTATE,ITBL(2,22,750),MAXTBL(2,22),ISORT(2,22,750),NLOOK
      COMMON/D/AVE(22,750),DEV(22,750),HTIME(22,750,8),LCT,H(8),LX,LZ
      COMMON C2(22,750),NCT(22,500,7),MX
      REAL ITIME,ICOND
      DIMENSION IX(22),IY(1150),IXOLD(22),IT(22),ICT(22)
      DATA IT,ICT,IXOLD/66*0/
C     IX - VARIABLE VALUES        IXOLD - OLD VARIABLE VALUES
C     IY - TAPE VALUES     IT - TIME IN STATE COUNTER
C     ICT - TIME IN CONDITION COUNTER
C - - INITIALIZE READ COUNTERS, EVENTS, & TABLE
      NLOOK=-1
      M1=0
      J5=0
      IX(1)=-1
   15 CONTINUE
C - - SKIP IF NOT TIME TO READ FROM TAPE
      IF(J4.LT.NBLOCK.AND.IX(1).NE.-1)GO TO 10
C - - INITIALIZE READ COUNTERS, THEN READ TAPE, JUMP IF EOF
      J2=0
      J4=0
      NSIZE=(KPARA+1)*NBLOCK
      N=7+NTEAM
      READ(N,110,END=90)(IY(I),I=1,NSIZE)
C - - FORMAT ASSUMES NBLOCK=50 & KPARA=22
  110 FORMAT(50(22I2,I4))
C - - UPDATE READ AND EVENT COUNTERS
   10 J4=J4+1
      M1=M1+1
      J1=J2+1
      J2=J1+KPARA
C - - TRANSFER CORRECT IY VALUES TO IX
      DO 20 I=1,KPARA
      J3=I+J1-1
   20 IX(I)=IY(J3)
C - - STORE TIME
      JTIME=IY(J2)
C - - SKIP IF END OF RUN
      IF(IX(1).EQ.99) GO TO 70
C - - SKIP IF FIRST TIME THROUGH
      IF(IXOLD(1).EQ.0) GO TO 30
C - - LOOP FOR ALL VARIABLES
      DO 30 I=1,KPARA
C - - SEE IF VALUE WITHIN LIMITS
      IF(IX(I) .GE. 0 .AND. IX(I) .LE. IXMAX(I)) GO TO 17
      WRITE (6,120) I,IX(I),IXMAX(I)
  120 FORMAT(/// 10X,'*** WARNING! ***', ///. 'VARIABLE',
     CI4,' HAS VALUE OF',I4,'VALUE SET TO:',I5 ///)
      IX(I)=IXMAX(I)
   17 CONTINUE
C - - SKIP IF NO TRANSITION
      IF(IX(I).EQ.IXOLD(I)) GO TO 30
C - - CODE VALUES AND CALL TABLE FOR EACH TRANSITION
      JSTATE=IX(I)+IXOLD(I)*10
      IF(KCOND(I).EQ.0) GO TO 26
      K=KCOND(I)
      DO 25 J=1,K
      L=K-J+1
```

```
25     JSTATE=JSTATE+IXOLD(JST(I,L))*10*(10**J)
  26   CONTINUE
       CALL TABLE(KT,1,I)
C  -  -    SUM UP TIME VALUES
       IF(LCT.EQ.1)CALL HISTIN(IT(I),ICT(I),KT,I)
       ITIME(I,KT)=ITIME(I,KT)+IT(I)
       T2(I,KT)=T2(I,KT)+FLOAT(IT(I))*FLOAT(IT(I))
       ICOND(I,KT)=ICOND(I,KT)+ICT(I)
       C2(I,KT)=C2(I,KT)+FLOAT(ICT(I))*FLOAT(ICT(I))
       NC(I,KT)=NC(I,KT)+1
C  -  -    SET TIME POINTERS TO ZERO
       IT(I)=0
       ICT(I)=0
  30   CONTINUE
C  -  -    UPDATE TIME POINTERS
       DO 40 I=1,KPARA
C  -  -    SKIP IF FIRST TIME, ZERO COND POINTER IF CHANGE OCCURRED
       IF(IX(I).EQ.0) GO TO 35
C  -  -    ZERO OUT COND TIME IF TRANSITION
       K=KCOND(I)
       IF(K.EQ.0) GO TO 35
       DO 35 J=1,K
       IF(IX(JST(I,J)).NE.IXOLD(JST(I,J)))ICT(I)=0
  35   CONTINUE
C  -  -    ADD TIMES TO POINTERS
       IT(I)=IT(I)+JTIME
       ICT(I)=ICT(I)+JTIME
       IF(IT(I).GT.1000000) WRITE(6,150)IT(I),I
 150   FORMAT('  TIME OVERFLOW:',I8,' AT VARIABLE',I3)
  40   IXOLD(I)=IX(I)
       GO TO 15
  70   CONTINUE
C  -  -    END OF TRAJECTORY - ZERO OUT TIME POINTERS & IXOLD
       DO 75 I=1,KPARA
       IT(I)=0
       ICT(I)=0
  75   IXOLD(I)=0
C  -  -    COUNT EVENTS
       M=M+M1
       M1=0
C  -  -    RETURN IF EOF
       IF(J5.EQ.1) RETURN
C  -  -    IF NOT, GO BACK AND READ NEXT RUB
       IX(1)=-1
       GO TO 15
  90   CONTINUE
C  -  -    EOF -- HAS EVENT COUNTER BEEN UPDATED?
       IF(IX(2).EQ.99)RETURN
       J5=1
       GO TO 70
       END
```

```
      SUBROUTINE OUTPUT(ITEAM)
C - -   THIS ROUTINE PRINTS OUT THE CALCULATED TIME VALUES
      COMMON/A/ NBLOCK,KPARA,KCOND(22),JST(22,4),IXMAX(22),MARKOV,NTEAM
      COMMON/B/ITIME(22,750),T2(22,750),NC(22,750),ICOND(22,750)
      COMMON/C/JSTATE,ITBL(2,22,750),MAXTBL(2,22),ISORT(2,22,750),NLOOK
      COMMON/D/AVE(22,750),DEV(22,750),HTIME(22,750,8),LCT,H(8),LX,LZ
      COMMON C2(22,750),NCT(22,500,7),MX
      REAL ITIME,ICOND
C - -   LOOP FOR ALL VARIABLES
      DO 90 I2=1,KPARA
C - -   WRITE HEADING FOR EACH VARIABLE
      K=KCOND(I2)
      WRITE(6,100)I2,ITEAM,(JST(I2,J),J=1,K)
  100 FORMAT('1  ***',10X,'VARIABLE',I3,' FOR TEAM',I2,
     1  ' IS CONDITIONED BY:',4I4)
      WRITE(6,105)
  105 FORMAT(/////)
      MX=1
C - -   LOOP FOR TABLE VALUES - ONE
      I1=1
C - -   INITIALIZE TEST FOR CONDITION CHANGE
      J1=ISORT(I1,I2,1)
      ITEST=ITBL(I1,I2,J1)/100
C - -   LOOP FOR ELEMENTS IN TABLE
      K=MAXTBL(I1,I2)
      DO 80 I3=1,K
C - -   GET TABLE ORDER - J1 POSITION MARKER
      J1=ISORT(I1,I2,I3)
C - -   J2 DESCRIPTION OF ELEMENT
      J2=ITBL(I1,I2,J1)
C - -   CHECK T O SEE IF CONDITION CHANGED
      ICK=ITBL(I1,I2,J1)/100
      IF(ICK.EQ.ITEST) GO TO 10
C - -   IF CHANGED, SKIP SPACES ON OUTPUT
      WRITE(6,110)
  110 FORMAT(///)
      MX=MX+1
      ITEST=ICK
C - -   CALCULATE TIME AVERAGES AND TIME VARIANCES
   10 NUM=NC(I2,J1)
C - -   WRITE OUT ONLY HISTO INFO IF SECOND TIME THRU
      IF(LCT.EQ.1)  GO TO 30
      TIME=ITIME(I2,J1)/FLOAT(NUM)
      CTIME=ICOND(I2,J1)/FLOAT(NUM)
C - -   CALCULATE PROB COUNTERS
      NX=ITBL(I1,I2,J1)/10
      NZ=NX/10
      NZ=NZ*10
      NX=NX-NZ
      IF(NX.GE.IXMAX(I2).OR.NX.LE.0)NX=IXMAX(I2)
      NCT(I2,MX,NX)=NCT(I2,MX,NX)+NC(I2,J1)
C - -   CHANGE NUM IF =1 BECAUSE OF ZERO DIVIDE IN VARIANCE
      IF(NUM.EQ.1)NUM=-4
      NUM=NUM-1
      TSQ=(T2(I2,J1)-NC(I2,J1)*TIME*TIME)/FLOAT(NUM)
      TCSQ=(C2(I2,J1)-NC(I2,J1)*CTIME*CTIME)/FLOAT(NUM)
      IF(NUM.NE.-5) GO TO 20
      TSQ=0.
      TCSQ=0.
   20 WRITE(6,130)ITBL(I1,I2,J1),TIME,TSQ,CTIME,TCSQ,NC(I2,J1)
  130 FORMAT(3X,'STATE=',I8,'  TIME AVE=',E10.4,'  TIME VAR=',E10.4,
     1  '  CD TIME=',E10.4,'  CD VAR=',E10.4,'  COUNTS=',I6)
```

```
C  -  -   STORE AVERAGE AND DEVIATION FOR HISTO
          AVE(I2,J1)=TIME
          DEV(I2,J1)=ABS(TSQ)**0.5
          IF(LX.GE.0)GO TO 80
          AVE(I2,J1)=CTIME
          DEV(I2,J1)=ABS(TCSQ)**0.5
          GO TO 80
   30     CONTINUE
C  -  -   WRITE OUT HISTOGRAMS
          CALL HISTOT(I2,J1,NUM,ITBL(I1,I2,J1))
   80     CONTINUE
   90     CONTINUE
          RETURN
          END
```

```
      SUBROUTINE HISTIN(LTIME,LCTIME,LST,LV)
C - -      THIS ROUTINE TAKES THE TIME VALUES AND SEPERATES THEM
C         FOR THE HISTOGRAM
C         LTIME - TIME VALUE      LCTIME - CONDITIONED TIME VALUE
C         LST - CONDITION OR STATE     LV - VARIABLE NUMBER
      COMMON/D/AVE(22,750),DEV(22,750),HTIME(22,750,8),LCT,H(8),LX,LZ
      DIMENSION TEST(8),T(2)
      DATA TEST/8*.999E15/
C - - -   INITIALIZE VARIABLES
      T(1)=LTIME
      T(2)=LCTIME
C - - -    LOOP FOR TIMES & CONDITIONED TIMES
      J=1
      IF(LX.LT.0)J=2
      IF(LZ.GT.8)LZ=8
C -  -   SET UP LIMITS
      DO 10 I=1,LZ
      TEST(I)=AVE(LV,LST)+H(I)*DEV(LV,LST)
   10 CONTINUE
C - - -   CHECK TO SEE WHERE TIME LIMITS FALL
      I=0
   20 I=I+1
      IF(T(J).LE.TEST(I)) GO TO 30
      IF(I.GE.8)GO TO 30
      GO TO 20
   30 CONTINUE
      HTIME(LV,LST,I)=HTIME(LV,LST,I)+1.0
   40 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE HISTOT(LV,LST,LN,LC)
C  -  -   THIS ROUTINE PRINTS OUT THE HISTOGRAM INFORMATION
C        K=# OF HISTOGRAM FRAMES;  HTIME BECOMES THE PROB IN EACH FRAME
      COMMON/D/AVE(22,750),DEV(22,750),HTIME(22,750,8),LCT,H(8),LX,LZ
      COMMON C2(22,750),NCT(22,500,7),MX
      K=LZ+1
      I=1
      IF(LX.LT.0)I=2
      DO 10 J=1,K
      HTIME(LV,LST,J)=HTIME(LV,LST,J)/FLOAT(LN)
   10 CONTINUE
C  -  -     CALCULATE PROBABILITIES
      NX=LC/10
      NZ=NX/10
      NX=NX-NZ*10
      IF(NX.GT.7.OR.NX.LE.0)NX=7
      IF(NCT(LV,MX,NX).EQ.0)NCT(LV,MX,NX)=-1
      PR=FLOAT(LN)/FLOAT(NCT(LV,MX,NX))
C  -  -   CALCULATE VARIANCE
      V=PR*(1.-PR)/NCT(LV,MX,NX)
      WRITE(6,100)LV,LC,AVE(LV,LST),DEV(LV,LST),LN,PR,V
  100 FORMAT(/5X,'VARIABLE',I4,' AT CONDITION',I8,' HAS AVE & DEV=',
     1  2E12.5,8X,'CNT=',I6,'  PR=',F7.5,'  V=',E10.4)
      WRITE(6,110)(HTIME(LV,LST,J),J=1,K)
  110 FORMAT(10X,8F10.5)
      RETURN
      END
/*
```

```
      SUBROUTINE SORT
C  -  -   THIS ROUTINE DETERMINES THE ORDER FOR THE TABLE VALUES
      COMMON/A/ NBLOCK,KPARA,KCOND(22),JST(22,4),IXMAX(22),MARKOV,NTEAM
      COMMON/C/JSTATE,ITBL(2,22,750),MAXTBL(2,22),ISORT(2,22,750),NLOOK
C   **  IF DIMENSIONS CHANGED ABOVE 999, ITEMP ALSO NEEDS UPPED.      **
      DIMENSION ITEMP(999)
      DATA ITEMP/999*0/
C  -  -   LOOP FOR # OF TABLES
      I1=1
C  -  -   LOOP FOR # OF VARIABLES
      DO 90 I2=1,KPARA
C  -  -   PUT VALUES INTO TEMPORARY FILE
      K=MAXTBL(I1,I2)
      DO 10 J=1,K
   10 ITEMP(J)=ITBL(I1,I2,J)
C  -  -   LOOP FOR ALL TABLE ELEMENTS
      DO 40 J=1,K
C  -  -   INITIALIZE TEST VARIABLE AND POSITION HOLDER
      IN=1
      ICK=ITEMP(1)
C  -  -   INNER LOOP TO CHECK SIZE
      DO 30 I=2,K
C  -  -   SKIP IF BIGGER
      IF(ICK.GT.ITEMP(I)) GO TO 30
      ICK=ITEMP(I)
      IN=I
   30 CONTINUE
C  -  -   STORE SMALLEST VALUE AND ZERO OUT
      I3=K-J+1
      ISORT(I1,I2,I3)=IN
      ITEMP(IN)=0
   40 CONTINUE
   90 CONTINUE
      RETURN
      END
```

LIST OF COMMON VARIABLES

COMMON A (most set in SUBROUTINE PARAM)

NBLOCK - Number of events in tape block; (KPARA + 1) NBLOCK = BLKSIZE
KPARA - Number of variables
KCOND(22) - Number of conditioning variables for variable I (≤ 4)
JST(22, 4) - Condition variables of variable I
IXMAX(22) - Maximum state size of variable I (assumed ≤ 7)
MARKOV - Number of Markov chain (assumed = 1)
NTEAM - Team or data set; $< 0 \Rightarrow$ all three (3)

COMMON B (used in SUBROUTINES INPUT and OTPUT)

ITIME(22, 750) - Tally for time in state
T2(22, 750) - Tally for time-squared in state
NC(22, 750) - Time counts
ICOND(22, 750) - Tally for time in conditional state

COMMON C (used in TABLE routines)

JSTATE - Condition of variable
ITBL(2, 22, 750) - Table of conditions
MAXTBL(2, 22) - Size of table I
ISORT(2, 22, 750) - Sorted table values
NLOOK - $1 \Rightarrow$ Table look-up; otherwise $\Rightarrow$ table add-to

COMMON D (used in HISTOGRAM routines)

AVE(22, 750) - Average time values
DEV(22, 750) - Standard deviation time values
HTIME(22, 750, 8) - Stores up to eight (8) breaks for histogram
LCT - INTERNAC histogram indicator
H(8) - Histogram breaks (H(I)* DEV(I, J) + AVE(I, J)
LZ - Number of histogram breaks (< 8)
LX - $< 0 \Rightarrow$ time in conditional histogram $0 \Rightarrow$ normal histogram

COMMON (BLANK)

C2(22, 750) - Tally for time-squared in conditional state
NCT(22, 500, 7) - Probability counter
MX - marker for probability