AN ABSTRACT OF THE THESIS OF

CHUNG-DI CHOU for the degree of Master of Science in Industrial Engineering presented on May 5, 1986.

TITLE: THE EFFECT OF UNCERTAINTY REPRESENTATION ON INFORMATION INTEGRATION PERFORMANCE IN A SIMULATED OCEAN SURVEILLANCE TASK

Abstract approved: Redacted for Privacy

Dr. Kenneth H. Funk II

Human operator performance was evaluated using two types of displays in a simulated ocean surveillance task. The two types of displays studied included a display in which position uncertainty information was given (polygon display) and a display in which uncertainty was not explicitly shown (line display). The experimental task was an association task in which subjects associated simulated radar emissions with the platforms (ships) under surveillance.

Thirty-two subjects participated with the study. Each subject performed the task using one of the displays. The results indicated that there was no difference in performance in terms of accuracy or speed for these two displays.

THE EFFECT OF UNCERTAINTY REPRESENTATION
ON INFORMATION INTEGRATION PERFORMANCE
IN A SIMULATED OCEAN SURVEILLANCE TASK

by

Chung-Di Chou

A THESIS

submitted to

Oregon State University

in partial fulfillment of
the requirements for the
degree of

Master of Science

Completed May 6,1986

Commencement June, 1986

APPROVED:

Redacted for Privacy

Assistant Professor of Industrial and General Engineering in Charge of Major

Redacted for Privacy

Head of Department of Industrial and General Engineering

Redacted for Privacy

Dean of Graduate School

Date thesis is presented May 5, 1986

Typed by the author

ACKNOWLEDGEMENTS

My most sincere gratitude is extended to Dr. Kenneth Funk for his guidence and encouragement throughout my study at Oregon State University.

I wish to thank many members of the Department of Industrial Engineering for their experience and expertise, especially Dr. Edward McDowell and Dr. Eugene Fichter.

I am grateful to Dr. James Riggs for his financial support during the period of my study. I also want to thank Dr. Sabah Randhawa for his friendship.

Special thanks are due my wife, Huei-Yong, for her patience and understanding. I also wish to thank my parents for their moral and financial support.

TABLE OF CONTENTS

APPENDICES

LIST OF FIGURES

LIST OF TABLES

THE EFFECT OF UNCERTAINTY REPRESENTATION ON INFORMATION INTEGRATION IN A SIMULATED OCEAN SURVEILLANCE TASK

CHAPTER 1

INTRODUCTION

Human factors is the discipline that tries to optimize the relationship between technology and the human. Wherever we find technology and people interacting, there is a need for human factors [Kantowitz and Sorkin, 1983]. The initial focus of human factors was mainly on designing military equipment for human use. It has been widely applied in many nonmilitary areas in present days, but applications in military systems are still stimulating much research on the subject. Some of the most important research is in the area of command, control, and communications.

"Modern warfare is electronic warfare. The side which controls the electromagenetic spectrum and denies its use to the enemy will be the winner of the next war. A ship unable to use its electronic sensors is helpless, unable to strike at the enemy or repel his blow." [Warship Commander, 1979] As improvements in weapons technology increase the range from which missiles can be fired, it is essential that the effective range of sensor technology increase to compensate. Therefore, sensors must be lumped with command, control, and communication systems under the term "C^3". This term is valuable because it emphasizes the unity of a system of command and control links by sensors [Friedman, 1977].

Command and control is the process whereby a set of military assets is managed in order to achieve an objective; a C^3 system is a collection of men and machines which support this process.

The theory of C^3 began to emerge in the last decade. It has not been possible to carry out a complete set of calculations which show how a C^3 system will, or should behave [Lawson, 1982], but one extremely important aspect about C^3 was introduced by Friedman [1977]: the highest level of naval C^3 is ocean-wide surveillance. The axiom of a fleet commander is to so maneuver his fleet as to destroy his enemy's and thus to gain command of the sea. This statement conceals a major problem: how to know where the enemy fleet is in the first place.

The basic task of tactical ocean surveillance is to maintain a continuous picture of the position, identity, course and speed, and in some cases, the behavior patterns, of all platforms in the vicinity of a ship [Novak and Walker, 1978].

In general, it is difficult to maintain an accurate surveillance picture; only a partially complete, and somewhat delayed picture can be maintained. The task of ocean surveillance is itself very complex, and is characteristically one of maintaining the best possible world picture based on bits and pieces of partial information [Novak and Walker, 1978]. Hence, the task of ocean surveillance can become very difficult to perform, and depends on information from many sources that vary widely in the types of information they provide, and the area covered.

Multisource tactical information integration (MTII) in the context of tactical ocean surveillance is to determine the position, identity, course, and speed of all platforms in the vicinity of a ship [Novak and Walker, 1978]. It is one of the problems involved in the tactical ocean surveillance task being investigated in this thesis. Problems of the MTII type contain a great deal of uncertainty and are hard to resolve. The uncertainty includes many things such as identity of a platform, moving speed of a platform, radar emissions of a platform, and to a certain extent, the course of a platform. Novak and Walker [1978] presented the results of their research on the MTII problem in a tactical ocean surveillance environment. The approach used in their research was artificial intelligence. They tried to find a set of rules which would apply to some of the subproblems in the

MTII problem in an information processing sense. They found out that AI is presently not an appropriate field to be applied in the MTII task. Also, they examined and reported that human performance on the MTII problem was very poor.

Figure 1-1 shows the relationship between C^3, ocean surveillance task, and the MTII problem, in which the C^3 term is a more general term than the others. We can see that "MTII" is a more specific term than "ocean surveillance task".

This thesis is motivated by Novak and Walker's proposal for explicitly showing uncertainty in the display interface between the human operator and the machine. This thesis presents the results of a study of human capabilities in performing the ocean surveillance task of radar emission/platform association. The study was accomplished by comparing human performance under two types of display interfaces. The association tasks were performed while monitoring simulated, digitized, and time-compressed sensor returns displayed on a computer-graphics display.

The contents are outlined as follows. First, Chapter 2 provides descriptions of the background of the MTII problem. Along with this, a simplified model and an

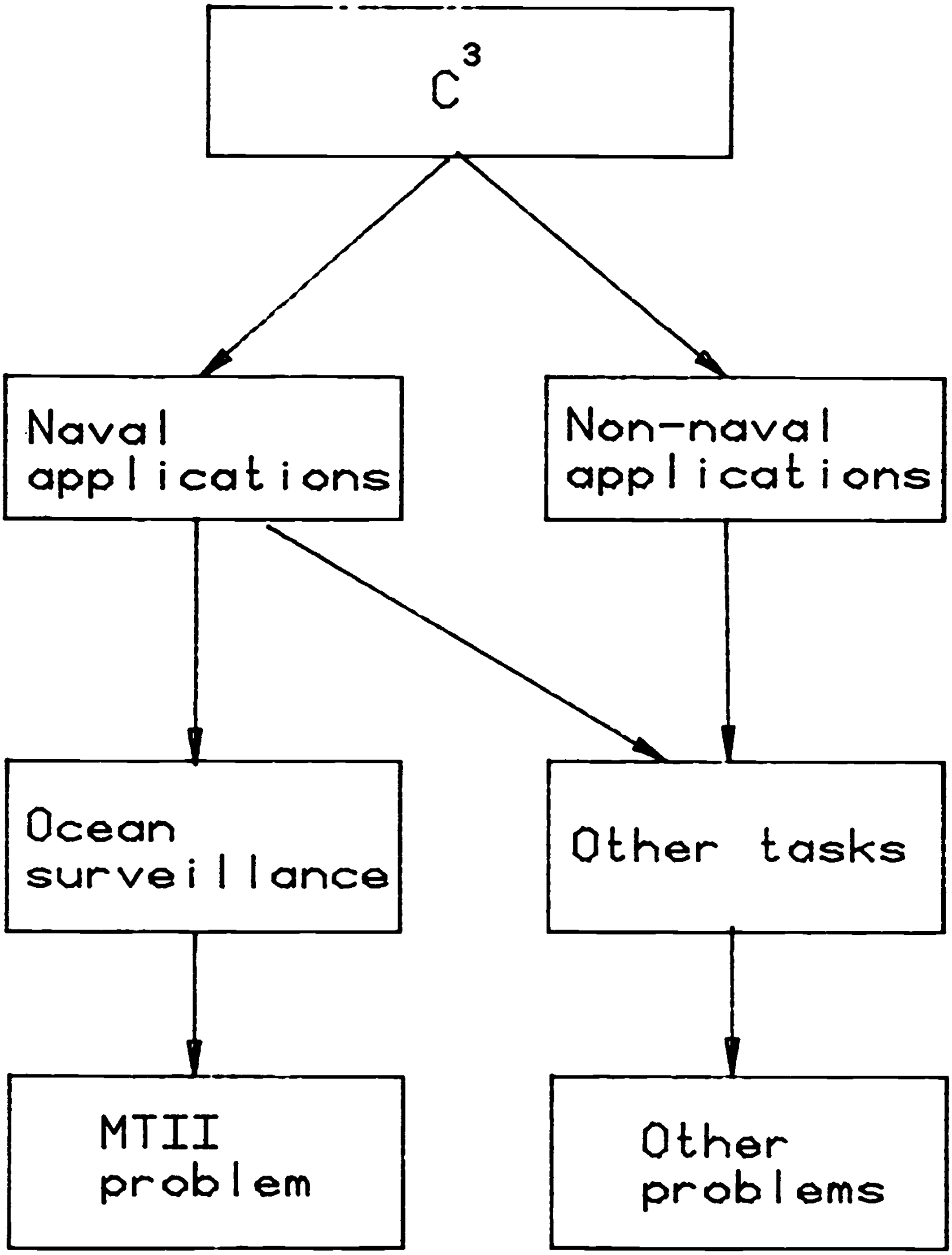


Figure 1-1 Relationship between C^3, ocean surveillance, and MTII.

example are presented to illustrate the central idea. Also, the two displays under comparison and expected results are discussed. Chapter 3 describes the methodology used to achieve the objective of the study. The experiment being employed is described in detail. Next, the results and statistical analysis of the experiment are presented in Chapter 4. There follows a discussion of results in Chapter 5. The conclusions and recommendations for future research are presented in Chapter 6.

CHAPTER 2

BACKGROUND

INTRODUCTION TO MTII

As mentioned before, multisource tactical information integration (MTII) in the context of ocean surveillance is to determine the position, identity, course and speed, of objects in the vicinity of a ship. Let us consider a simplified model of the MTII task. As illustrated in Figure 2-1, the left half of the model represents the state before information is received aboard ship. The right half of the model shows that information from different sources is integrated and displayed to the operator. The general features of this model are described in the following paragraphs.

Pieces of information such as position, identity, radar emissions, and speed of a platform must first be brought together through several communication channels at a central location. This is a serious problem due to the physical separation of the areas where different sensor reports are received aboard ship. In practice, a ship will actively gather additional information if necessary to locate and identify potentially high-interest platforms.

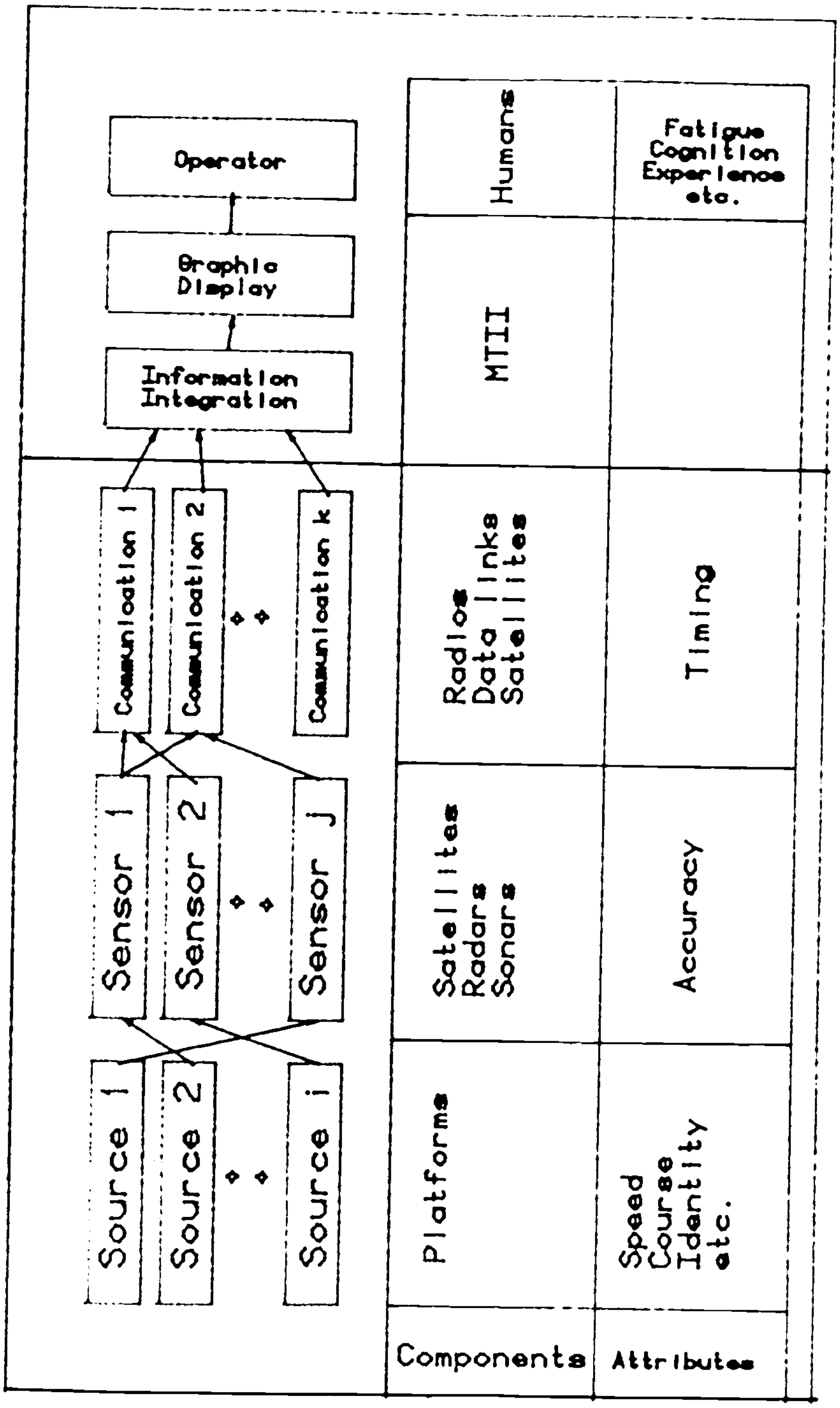


Figure 2-1 A simplified MTII model.

However, even when sensors of the same type are involved it is necessary to translate their reports into a common frame so they can be integrated; therefore, the MTII problem is made more difficult when more information sources are to be integrated. Conversion of information to locally usable form may involve such things as coordinate transformations to convert positions relative to another ship to ownship coordinates.

Existing computer systems using various mathematical techniques have generally been able to deal with only a very few kinds of information, and they are adequate only for a few very narrowly defined parameters.

The association of an input datum with the correct existing object is heavily dependent on the accuracy and timing of the report. If the association and identification information are sufficiently imprecise to allow associations of the new report with more than one existing object, the probability of an erroreous decision is high. [Kullback and Owens,1972]

After the association operation, estimates are made of position, identification, course and speed of objects. Bookkeeping of the previously analyzed data is necessary for later retrieval, for a surveillance task generally extends over a long time period. It is of course necessary to be able to retrieve information and display it for the operator.

It can be easily seen that all the operations in an MTII task are more or less correlated to one another, and that each operation plays an important role. Therefore, the MTII problem contains several subproblems itself. As noted by Novak and Walker, these problems range from relatively easy problems to impossible (unsolvable) problems. Hence, the MTII problem consists of a class of problems. Since there are presently no sensors that can provide position and identification information at the same time, it is worthwhile to investigate the MTII problem.

AI AND MTII

Artificial Intelligence (AI) is a field of inquiry within computer science that attempts to increase the reasoning and perceptual abilities of computer systems.

Novak and Walker were among the first to explore the applicability of AI technology to the MTII problem. In their work, they examined the MTII problem from information processing views and discussed the possible AI technology which might be applied to it. They selected one of the most frequently used knowledge representation and inferences systems of complex AI systems - the rule-based system - to explore the applicability of AI to MTII.

Novak and Walker originally thought they could come up with a knowledge base of rules which could be applied to the MTII task. After thorough research, they concluded that AI is not appropriate for the MTII task at the present time. They did come up with a few rules, but were not able to improve the current system significantly. The main reasons that AI is not applicable to MTII are: (1) The lack of sufficient information for the MTII problem is the main obstacle. (2) MTII problem is ill-defined; there is not such a thing as a single, well-defined "MTII" problem. (3) The most critical subproblems of MTII are not problems of the type with which AI has dealt. Although they found some subproblems which AI can be applied to, the cost was high and payoff was low, hence it made no operational sense to them to use AI in MTII.

According to Novak and Walker's report, people perform the MTII task rather poorly. This introduces an interesting problem: how well does the human perform an MTII task, and what are the human's limitations in an MTII task? It seems to make more sense to think in terms of a system that has a human in the loop rather than a fully automatic system. Since AI is not applicable to MTII at the present time, the human still must play a vital role in such a task.

HUMAN PERFORMANCE IN THE ASSOCIATION TASK

The single most critical subproblem in MTII is the association task [Novak and Walker, 1978]. This is the most critical problem since if reports are incorrectly associated, not only are errors in position and identification likely, but a single platform may be reported as multiple platforms, or multiple platforms may be reported as a single platform.

Smith [1979] presented a computer-aided manual technique for performing tracking and multisensor correlation of ocean surveillance data. In his research, some computer algorithms were used to correlate a large fraction of ship sightings. A human operator was assigned to resolve the ambiguous situations in which the computer could not decide. His results showed that better performance was obtained by having a human operator resolve ambiguous situations.

Novak and Walker pointed out that human operator performance in an MTII task is poor. From the above discussion, it is easy to see that human operator can resolve problems in a specific range of difficulty. Some characteristics of human operator are outlined by Funk

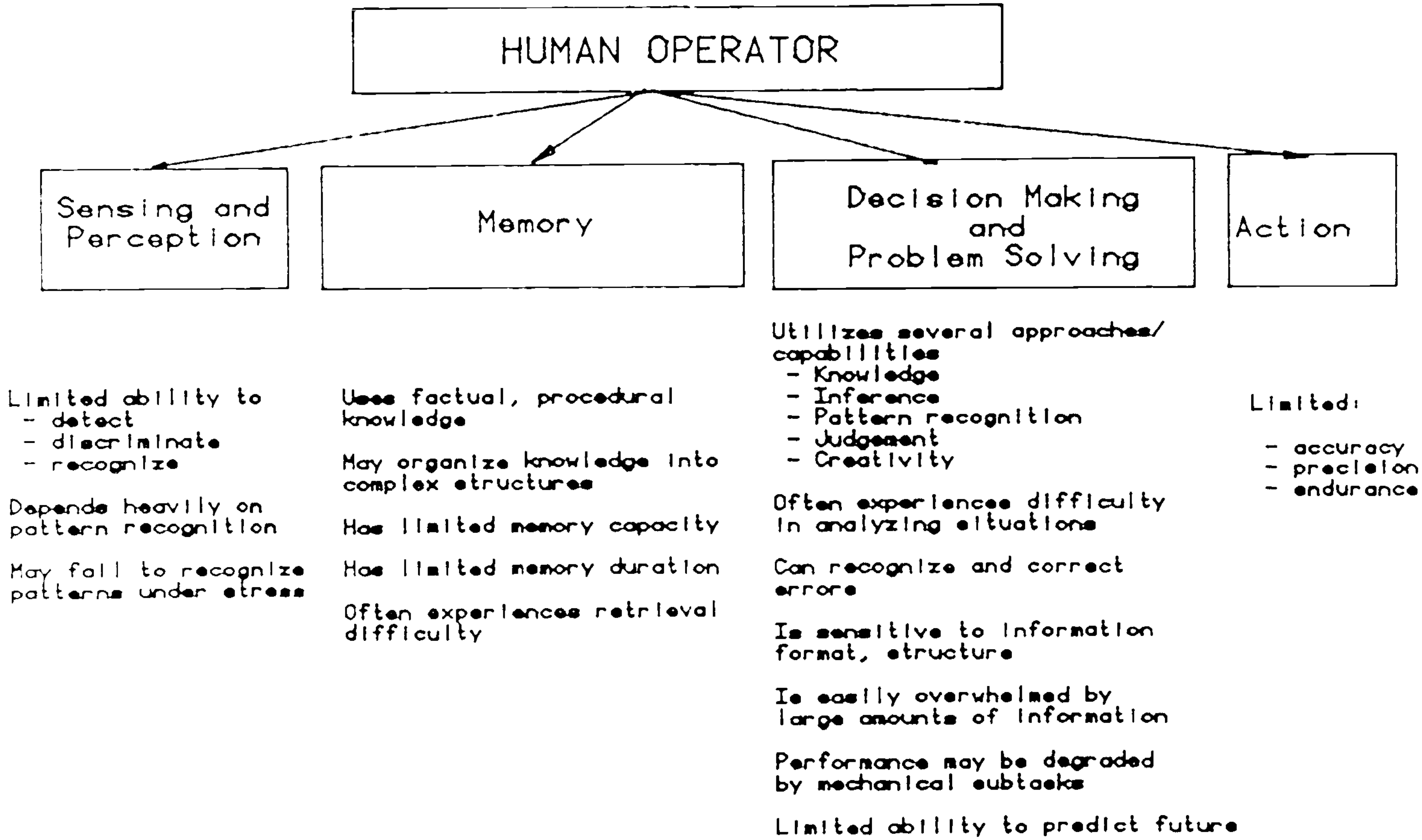


Figure 2-2 Characteristics of the human operator.

[1986] as illustrated in Figure 2-2. Among these characteristics, this thesis addresses memory and problem solving skills.

Since human performance on the MTII task is poor, we are interested in how many objects a human operator can keep track of at a time. Human memory can be augmented by proper coding methods, but it is limited in capacity and duration [Miller, 1956]. Greitzer et al [1982] conducted an experimental task which was similar to the MTII task to examine human memory limitations in multi-object tracking. They showed that an unaided operator can keep track of about seven tracks. When the number of tracks increased, the performance decreased. Thus, in order to elaborate the operator's memory, the elements representing information in an association task need to be meaningful to the operator.

From the results of Greitzer et al and Smith, it can be seen that one of the problems associated with human performance in the association task is the density of targets. Also, there is another serious problem which restricts human performance significantly, i.e., the sampling rate. A low sampling rate can result in an unresolved situation which will make the association task infeasible.

PROBABILITY POLYGONS

Now let us consider an example of an association task. In this example, two platforms (ships) are under ownship surveillance. Assume the bearing and range information of these two platforms are as listed in Table 2-1 but unknown to ownship, and the true movement patterns of these platforms are shown in Figure 2-3 (also unknown to ownship). During the surveillance period, ownship receives pieces of information about these platforms from sensor systems. Suppose the reports received on board contain only the information at time 0, 6, 10, and 12, then the best surveillance picture is as shown in Figure 2-4 where squares represent the platforms. Subsequently it becomes difficult to associate the new information received at time = 10 with the existing tracks because of the low sampling rate. Hence there is much uncertainty associated with this situation.

This association problem can be solved by either increasing sampling rate or having an improved representation of the information, which are two ways to reduce the uncertainty in the decision process.

The question of interest is what the decision will be if we show not only the range and bearing information but also the estimated direction of movement in the representation. The representation in Figure 2-5 shows

time	Whale		Bass	
	bearing	range	bearing	range
0	45	21	315	32
2	30	22	330	34
4	15	23	334	34
6	7	29	5	38
8	9	31	10	47
10	27	32	5	47
12	40	34	343	49

Table 2-1 The bearing and range information of the two platforms under surveillance.

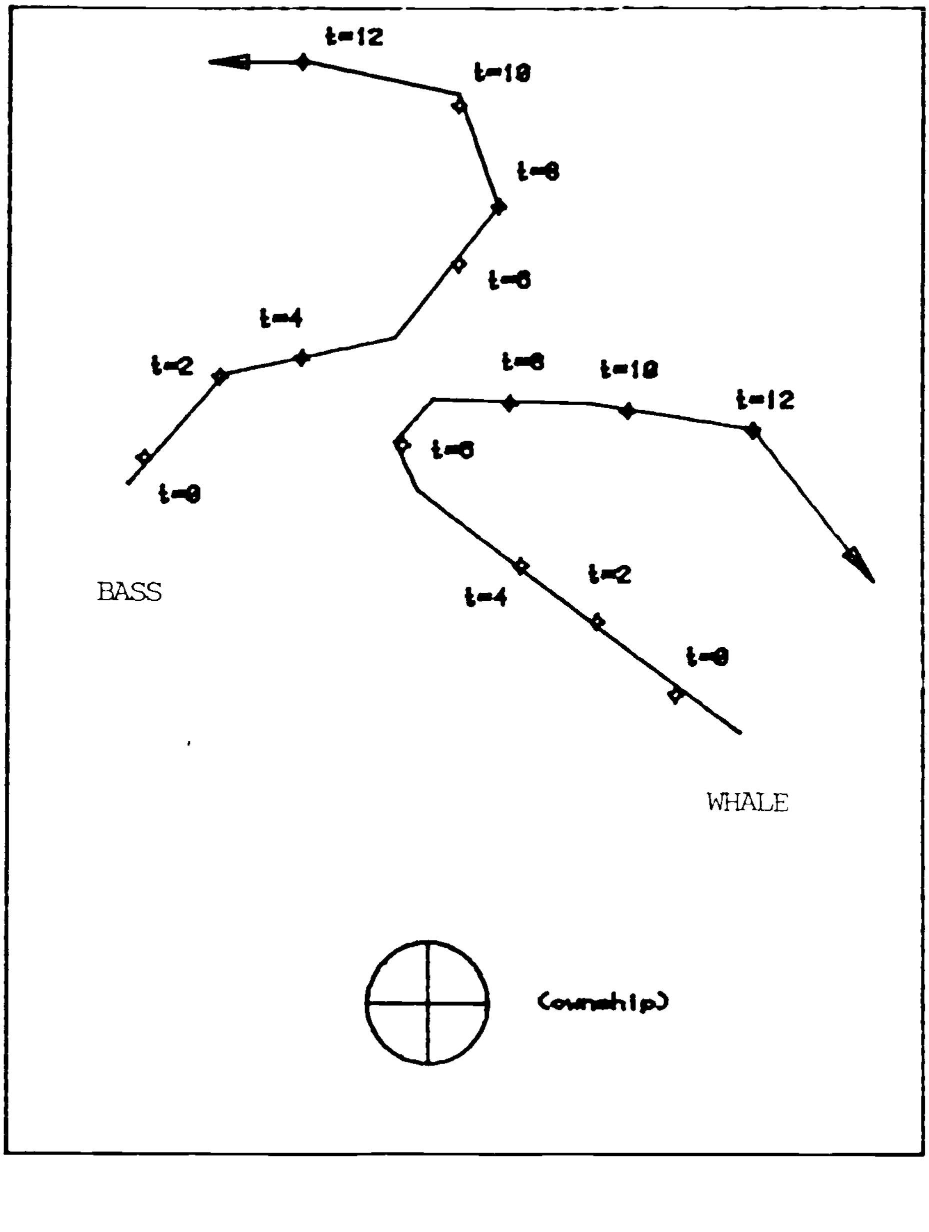


Figure 2-3 True movement pattern of the platforms under ownship surveillance from time 0 to 12.

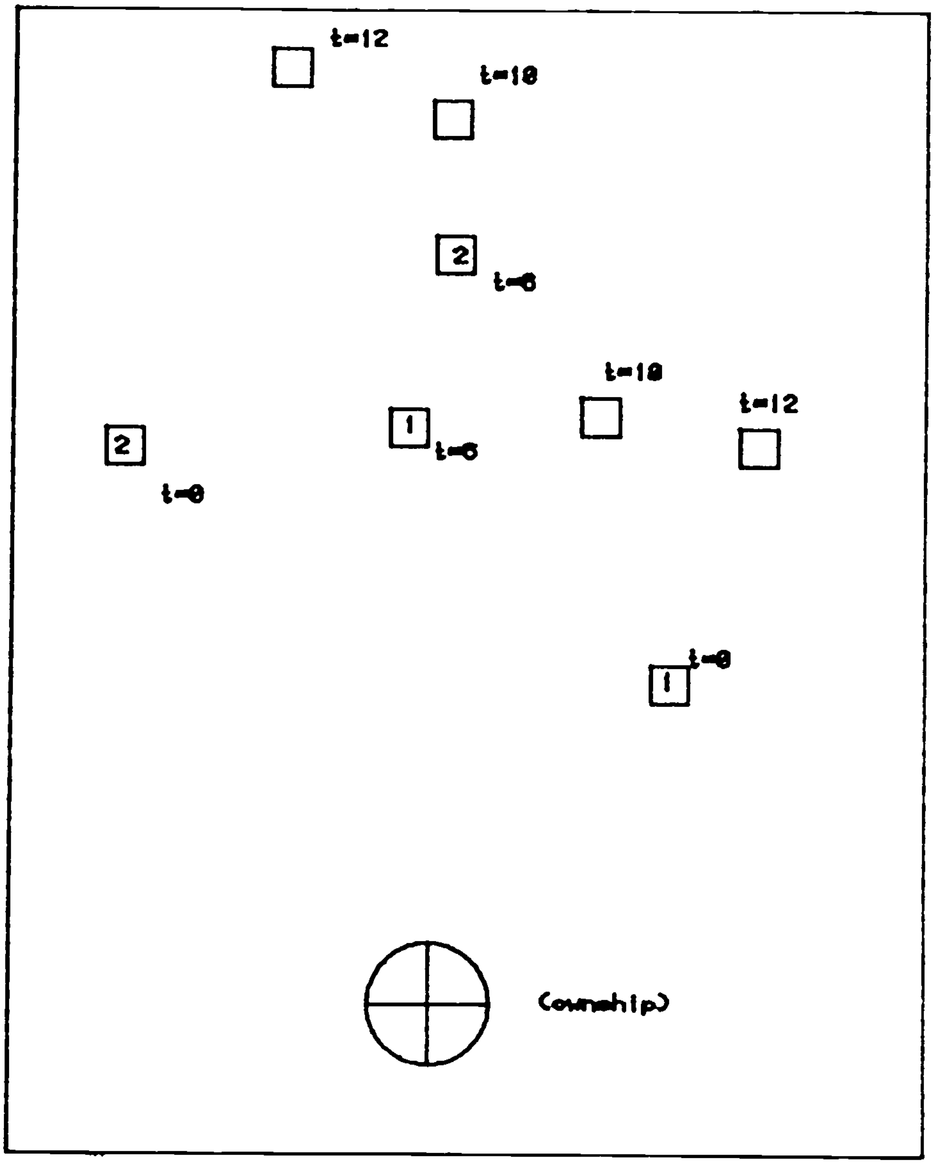


Figure 2-4 The best surveillance picture for the two platforms when information is received only at time 0, 6, 10, and 12.

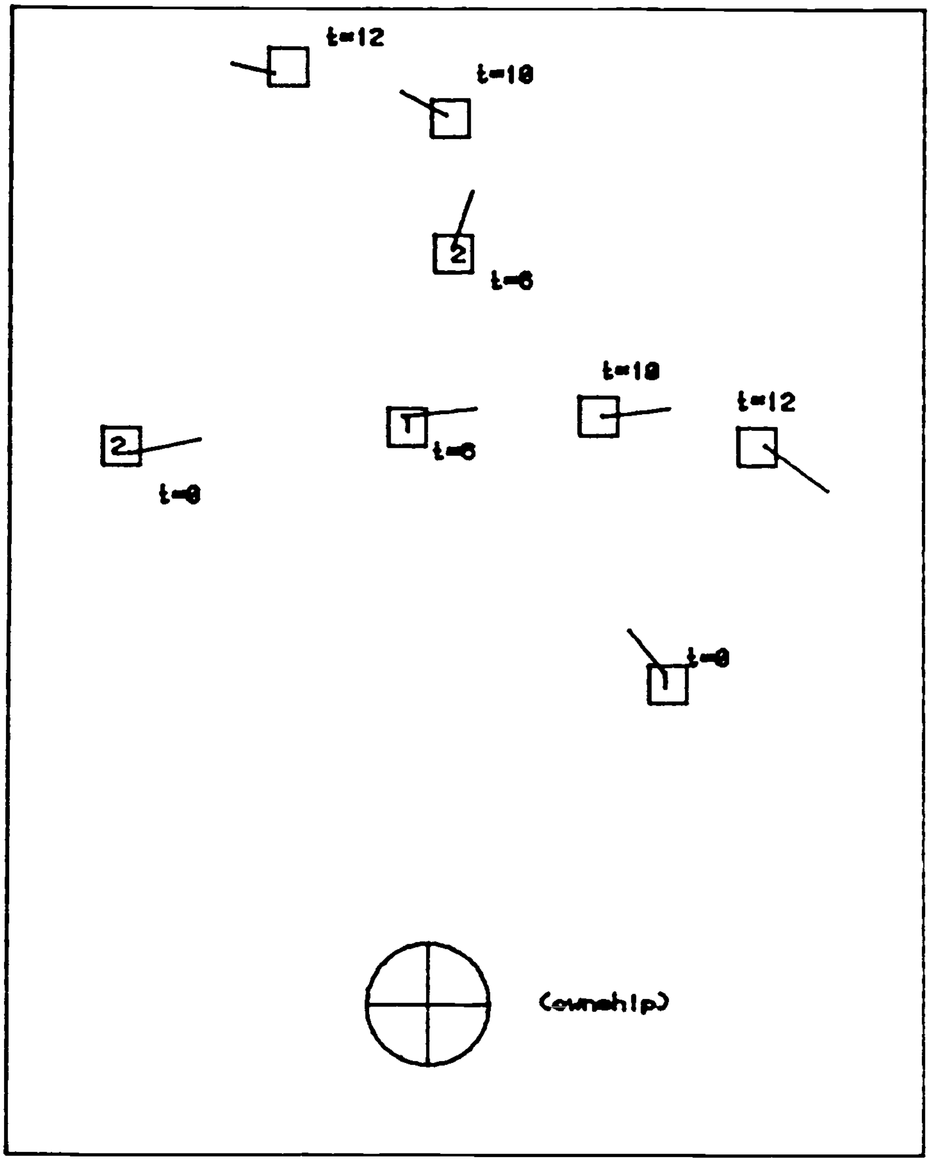


Figure 2-5 Illustration of bearing and range information and estimated moving direction in the representation.

such a situation for this example. This representation makes it easier to associate the updating information with the previous data.

Suppose that in this example the two platforms under surveillance use their radars over some period of time. Our goal is to detect such radar emissions and decide which platform each emission comes from. Such a situation is shown in Figure 2-6. Under this representation, most people will intuitively decide that radar emission B is from track 2. What will happen if the previous position is shown for each platform? Figure 2-7 shows the situation in which the last updates for those two platforms are also shown in the representation; the decision may be changed. [Funk, 1986]

Therefore, the decisions are strongly influenced by the representation of the situation when uncertainty exists in the information. And this raises the idea of explicitly showing uncertainty in the representation of information.

The representations of platforms are all point estimates of the true positions of the platforms, and are obtained by means of sampling. It is also possible to form a confidence region for a platform under surveillance by the method of sampling. The problem arises when such representation is presented to the human operator, i.e.,

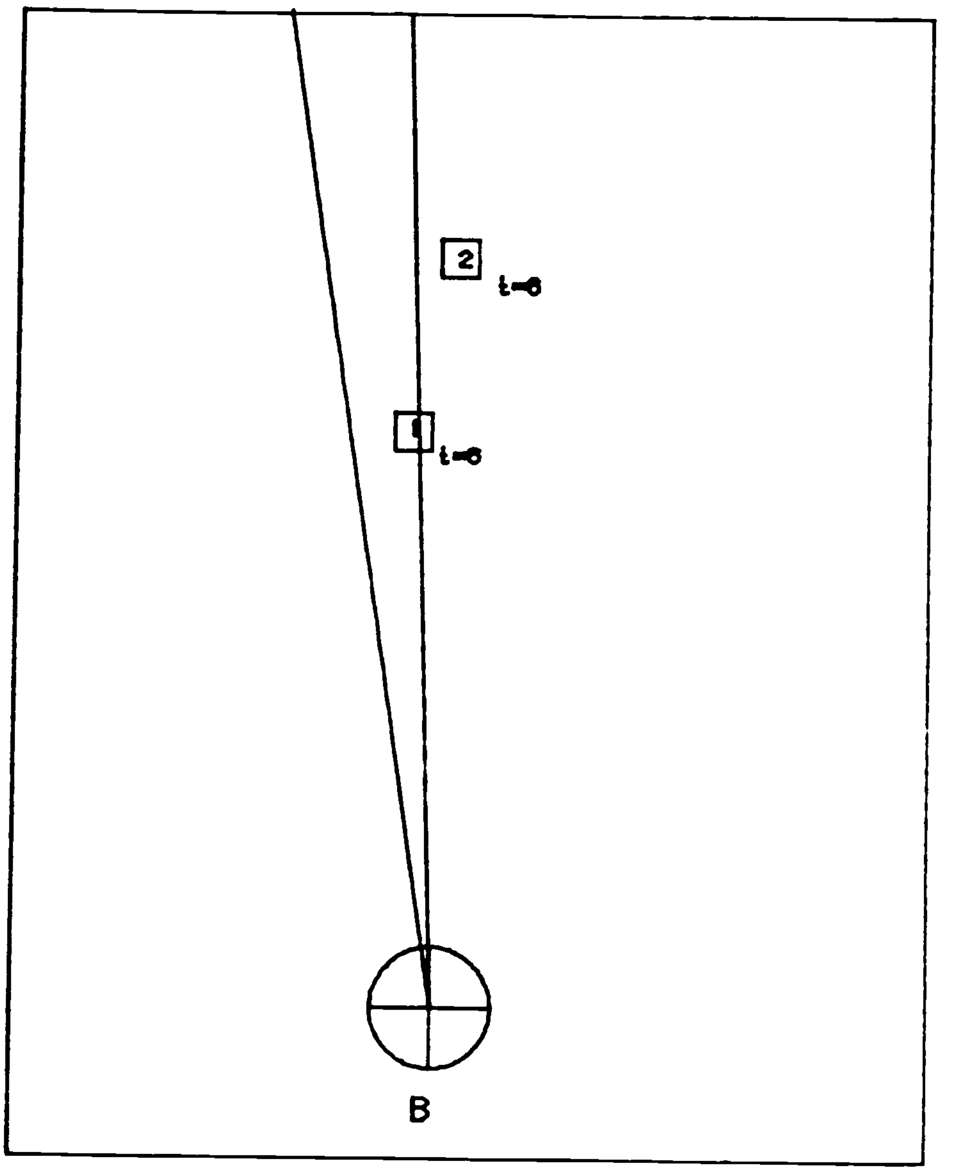


Figure 2-6 A radar emission B was detected at time = 7.

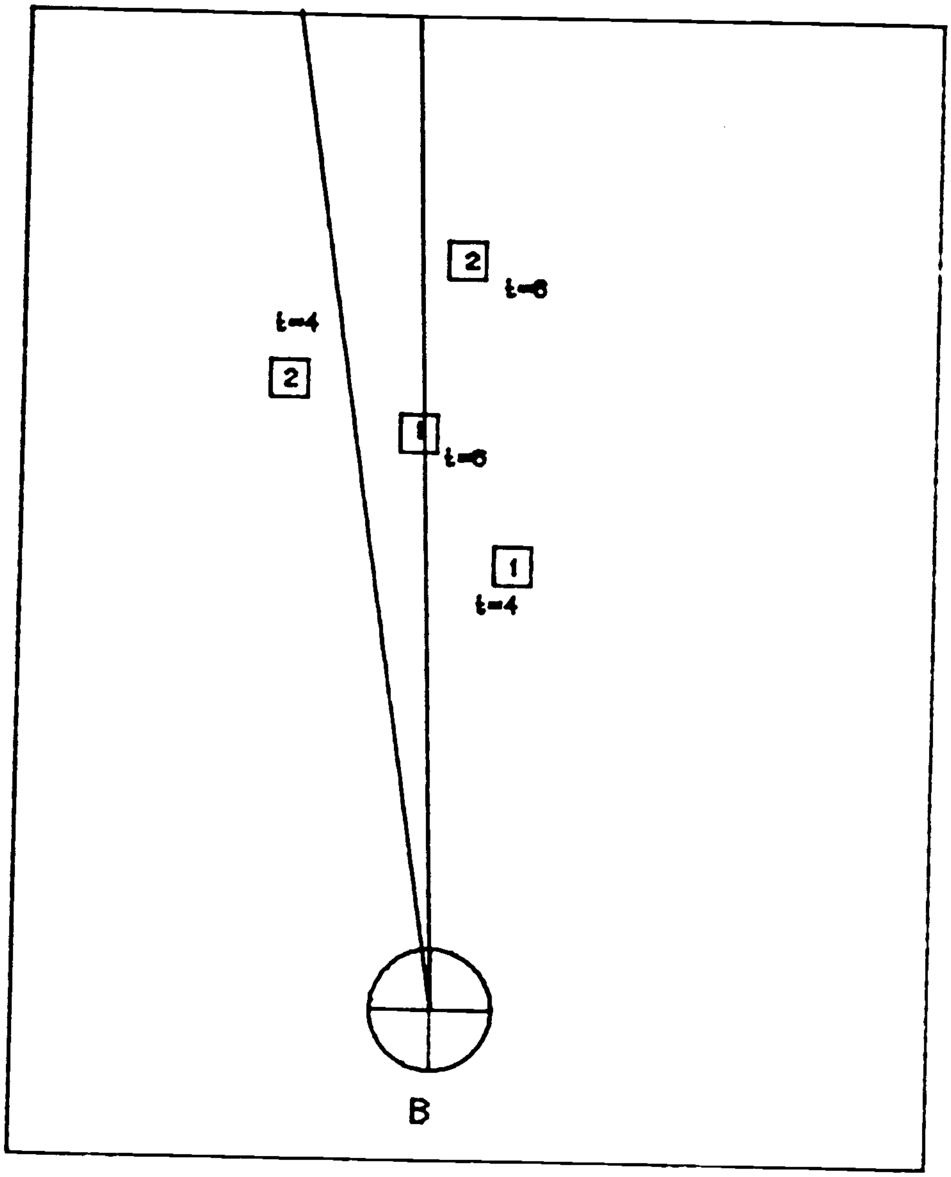


Figure 2-7 Representation which shows the last updates of the platforms.

can such a representation be a decision aid to the operator? This idea can be realized by providing the human operator with some extra information which is based on the same data collected for the association task.

Novak and Walker [1978] developed the concept of using probability polygons in representing information for the association task. As illustrated in Figure 2-8, the first example shows a polygon representing a 95% confidence ellipse from a shore-based sensor. If there is a heading/speed hypothesis for the object, the polygon can be updated according to this hypothesis for the object, allowing for uncertainty in the course and speed. The second example shows how bearing-only information from a sensor such as sonar could be represented as a wedge-shaped polygon. The angular width of the wedge represents the sensor's uncertainty of the bearing.

The ellipse shape representation can explicitly show the direction of movement of the platform, which is based on the estimated heading information of the platform. The area that an ellipse covers represent a confidence region of the platform that can move around in a certain amount of time, which is based on the estimated speed information of the platform. Although this information is based on estimates, the representation does explicitly show the uncertainty. Novak and Walker reasoned that this might help the human operator in an association task.

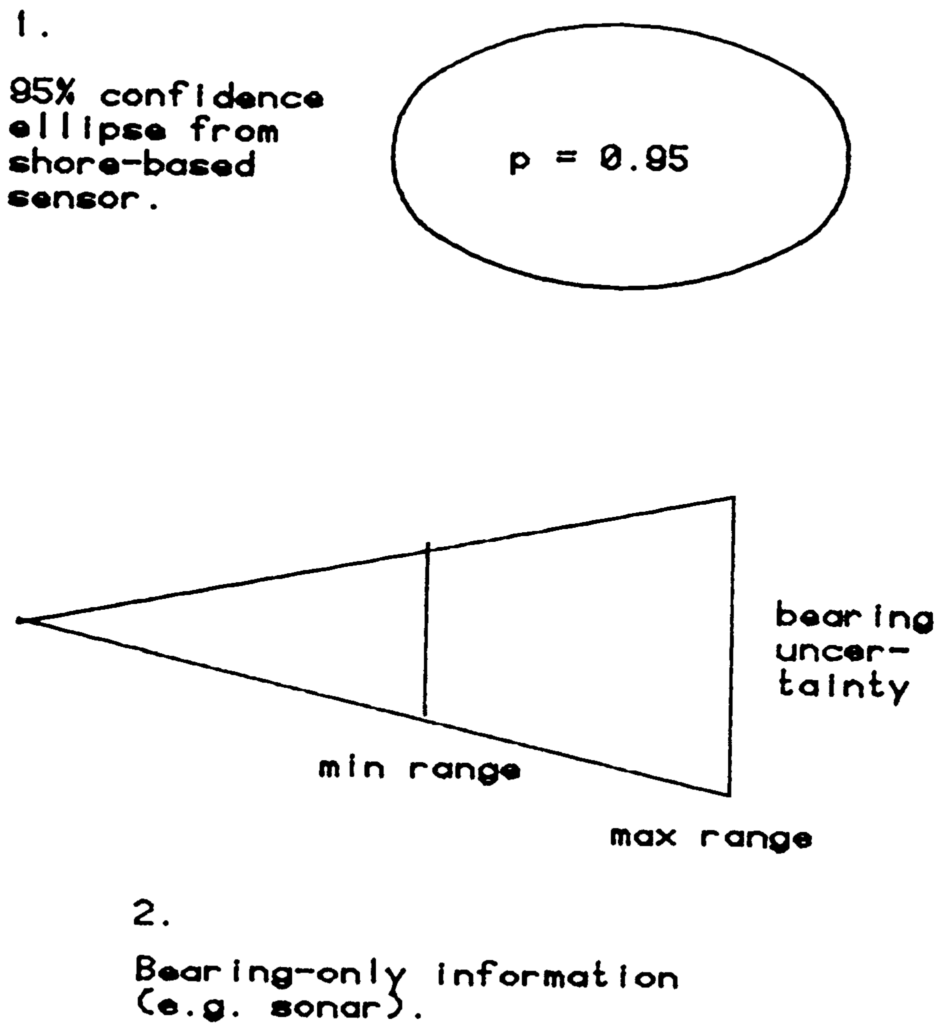


Figure 2-8 Examples of representation using probability polygons.

Let us reconsider the example in Figure 2-7. In Figure 2-9, the sensor reports provide information that locates targets within ellipses of specified orientation and axis lengths; typically, these ellipses are taken to represent a high confidence area of a bi-variate normal distribution. The question that arises from this discussion is: "Will association performance be improved using polygons?".

The above examples show that the probability polygon concept allows many types of information to be represented and used for an interface between human operators and the association task. As noted by Novak and Walker, the concepts of probability polygons offer a good possibility for improvement beyond what existing systems can do, and that these methods, when incorporated in a well-designed man/machine system, could help significantly with the operational problems of MTII. Furthermore, such a system could be used as a testbed to determine the effects of improvements in sensor systems in terms of the end-product of the multisensor integration process. Such a facility could be very valuable, because possible improvements or new sensor systems could be tested before major investments were made in their development. This research is aimed at evaluating human performance under such information representation methods.

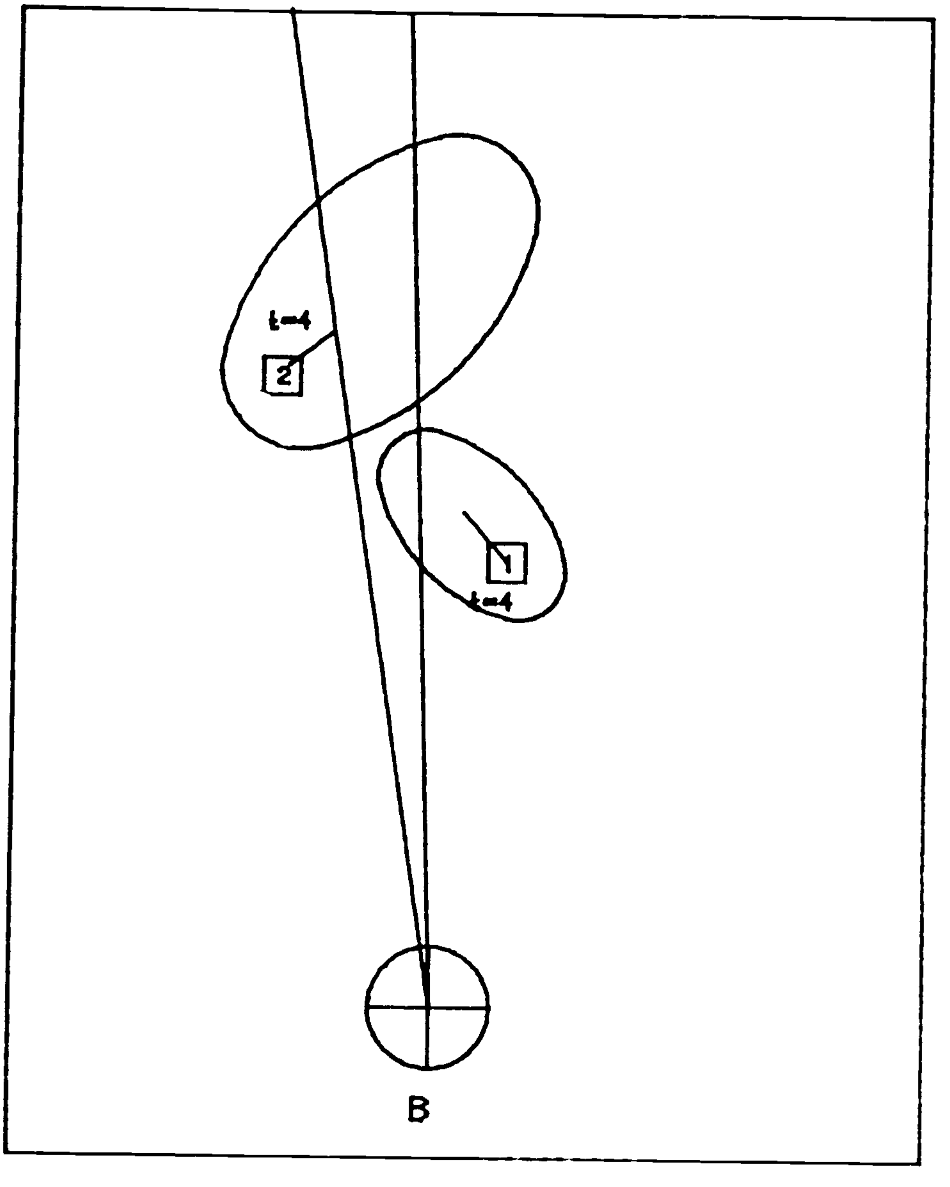


Figure 2-9 Another representation shows the confidence region of a platform association with the detected radar emission B.

The approach used was to perform an experiment and compare the performance of human operators under two different types of displays in which the displays were varied in the manner of representing information in an association task.

RESEARCH HYPOTHESIS

The baseline display for comparison with the probability polygon display was a display which did not explicitly show uncertainties (Figure 2-10). In this display, the sensor report contained estimated bearing information only.

The experiment was an attempt to compare the two types of displays: the probability polygon display and the baseline display. The former will be called the "Polygon type display", and the later will be called the "Line type display". The main objective was to see if one type of information representation would produce better operator performance than the other. In order to answer the questions being posed in the previous paragraphs, the following hypothesis can be stated: H_0: Display type (Polygon or Line) does not affect operator performance.

The experiment is described in the next chapter.

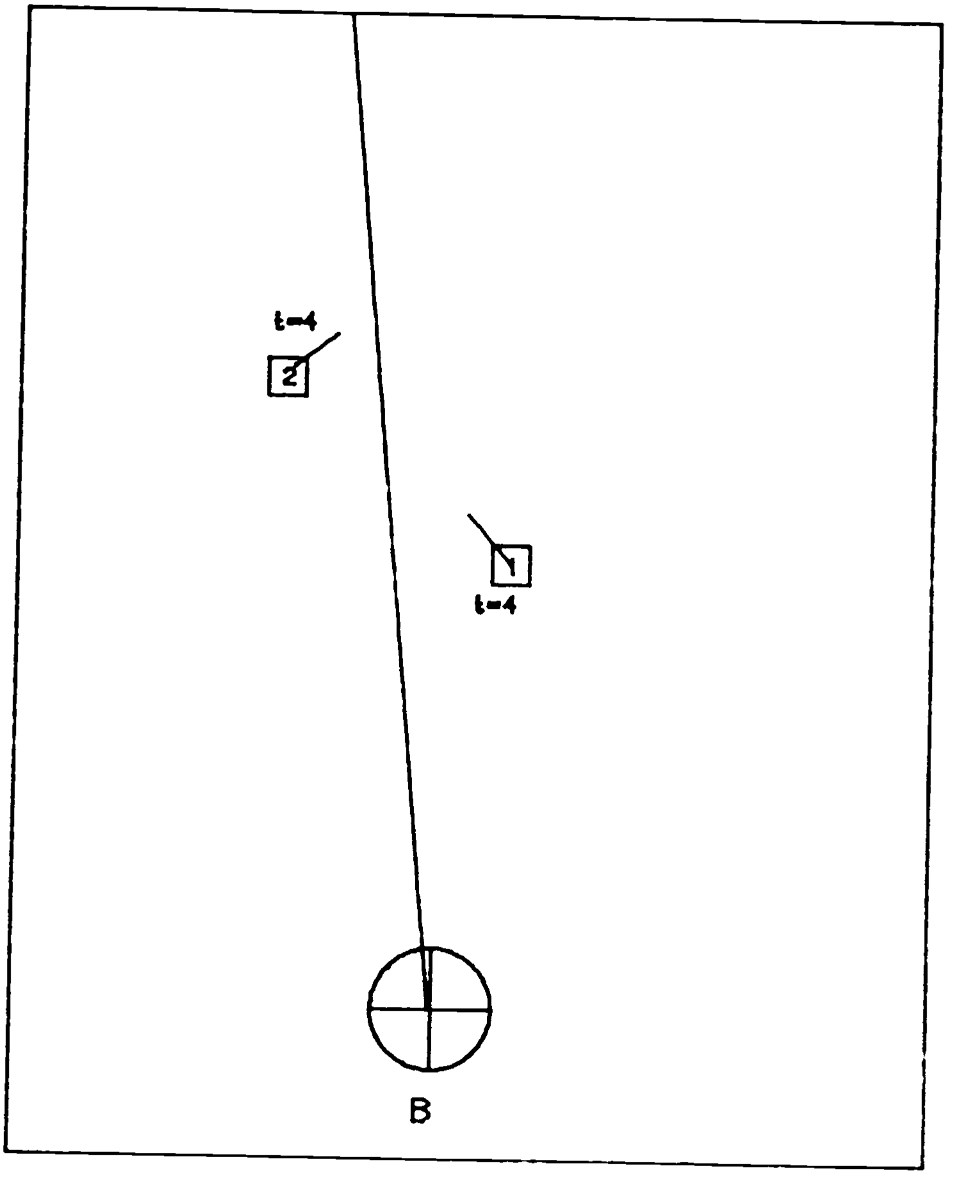


Figure 2-10 Baseline display for comparison: Line type display.

CHAPTER 3

METHODOLOGY

An experiment was designed to make inferences about human performance using the two types of displays described in the previous chapter: the Polygon type display and the Line type display. Each subject in this experiment utilized one of the displays in performing a simulated, ocean surveillance task.

In the task, the subject viewed a screen which represented the display of sensor information in the vicinity of ownship. When a sensor report appeared on the screen, the subject was required to associate radar emissions with the platform which he thought was the most likely one to have generated the emissions. For example, one of the Line type displays might be as shown in Figure 3-1(a) when a sensor-report event occured. The subject had to judge if radar emission B came from platform 3, 4, or 5. The same situation for a Polygon type display is shown in Figure 3-1(b).

INDEPENDENT VARIABLES AND DEPENDENT VARIABLES

Three independent variables were defined. They were:

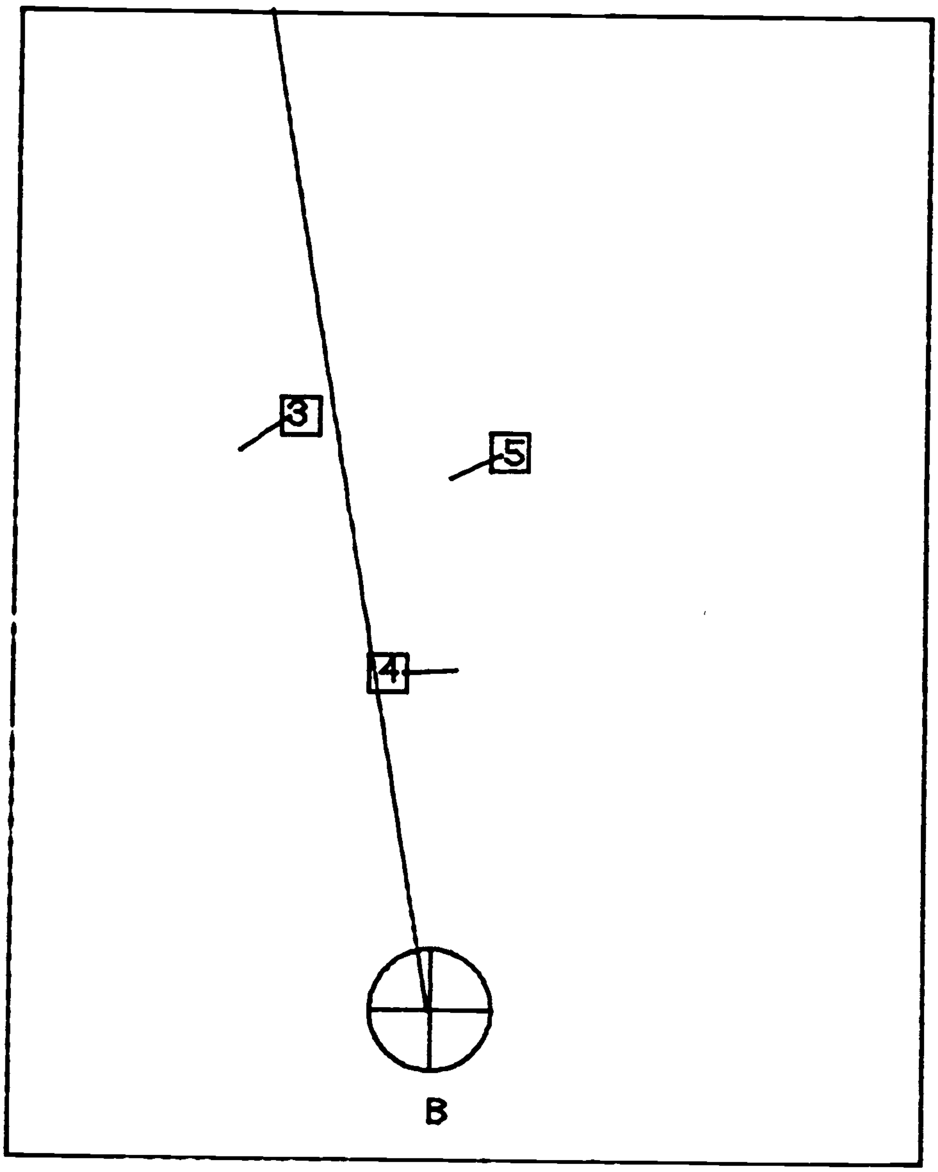


Figure 3-1a One possible configuration of the Line type display when a radar emission of the frequency B band was detected.

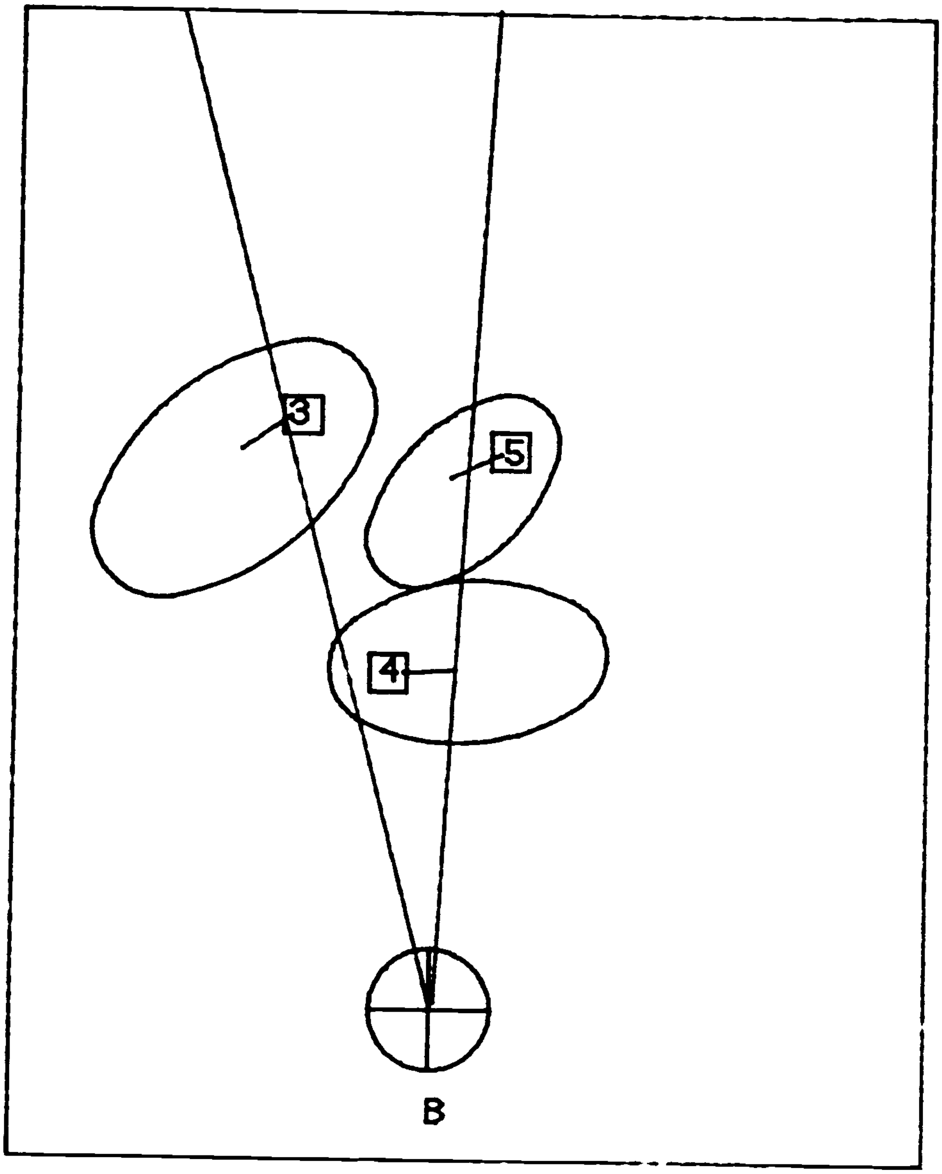


Figure 3-1b The configuration of the Polygon type display under the same conditions as in Figure 3-1a.

1) Display type (Polygon type and Line type).

2) Data rate (15 emissions and 30 emissions per experimental session.)

3) Ship density (4 ships per screen and 8 ships per screen).

Two dependent variables were selected to measure the performance:

1) Percent correct (accuracy).

2) Response time (time in seconds to respond to a sensor report).

DESIGN

The design was a completely randomized one in which there were 8 treatments. Each treatment contained 2 levels of each factor. The factors were the independent variables described in the previous section (Table 3-1).

Following Yate's notation [1933], capital letters are used to refer to factors; combinations of lower case letters and numerical subscripts, or simply the subscripts, are used to denote treatment combinations. Thus, the 8 treatments were as shown in Table 3-2. Two levels of display type, two levels of data rate, and two levels of ship density completed a 2^3 completely randomized factorial (CRF) design.

Factor	Abbreviated form	level 0	level 1
Display type	T	Polygon	Line
Data rate	R	15	30
Ship density	D	4	8

Table 3-1 Three factors, abbreviated forms, and levels.

treatment level factor	1	2	3	4	5	6	7	8
T	0	0	0	0	1	1	1	1
R	0	0	1	1	0	0	1	1
D	0	1	0	1	0	1	0	1

Table 3-2 An exhibit of the 8 treatment.

It was assumed that the experimental units were homogeneous; i.e., the variability among subjects was small.

The CRF design was chosen since, according to Steel and Torrie [1980], it is valuable in explanatory work where little is known concerning the optimum levels of the factors, or even which ones are important. When considerable information is available, the best approach may be to compare a very limited number of combinations of several factors at specific levels.

SUBJECTS

Thirty-two subjects participated voluntarily in this experiment. All subjects were from graduate and undergraduate schools at Oregon State University. Five of these subjects were Americans and twenty-seven were foreign students. Twenty-eight of the thirty-two were males. Two of the thirty-two reported that they had prior experience with radar systems.

EQUIPMENT AND MATERIALS

The equipment and materials used in this experiment included the following:

1) Hardware:

The experimental task was implemented on a Tektronix 4107 color graphics terminal driven by a Tektronix 4170 microcomputer. An HP-86 microcomputer with a real-time clock was interfaced to the Tektronix 4170 to pace the experiment. Figure 3-2 illustrates the configuration of this hardware.

2) Software:

Five programs were designed for this experiment to simulate the association task. Four of them were written in FORTRAN and one in BASIC (Table 3-3). The functions of each program are described in the following paragraphs. Listings of these program are contained in Appendix B.

Program	Computer language	Purpose
SCENARIO	FORTRAN	Generate a scenario file which included all information.
ESTIMATE	FORTRAN	Simulate the ownship sensor behavior.
POLYGON	FORTRAN	Draw Polygon type display.
LINE	FORTRAN	Draw Line type display.
READY	BASIC	Pace the performance of the experiment.

Table 3-3 Programs and their purposes.

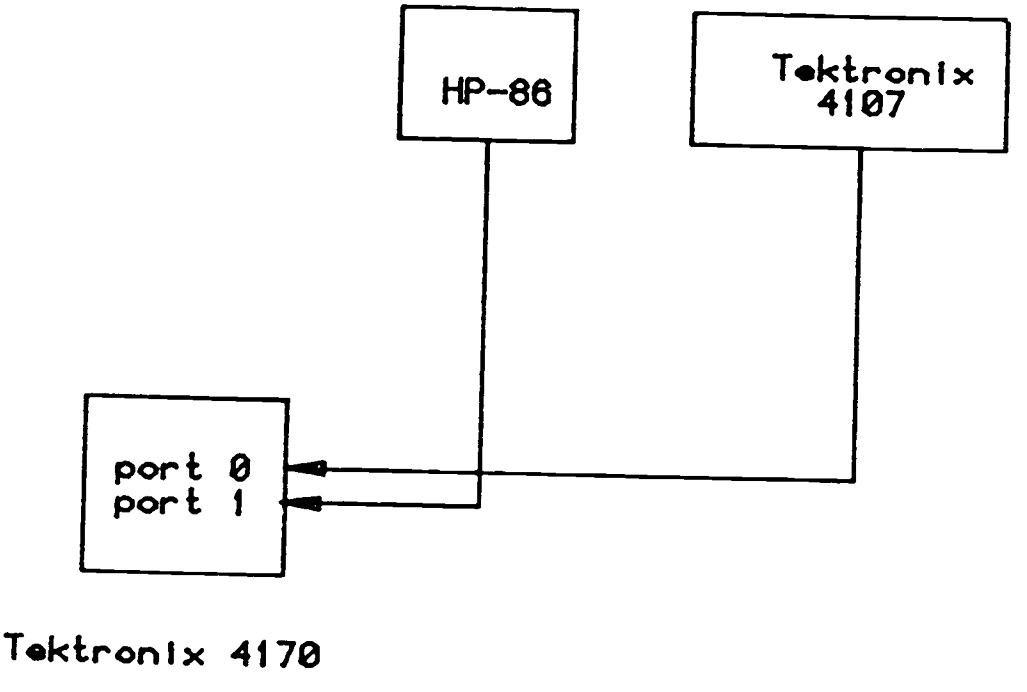


Figure 3-2 Hardware configuration.

a) SCENARIO: This program was used to generate true information about the platforms in the experiment, which included speed, direction of movement, radars on board, time between radar emissions, identity, and location. Several rules for the movement of platforms were derived by the experimenter from Warship Commander [1979], a commercially available naval war game which provides very realistic information about naval warfare. The program employed a simulation program developed by Law and Kelton [1982] called SIMLIB. SIMLIB is a discrete-event-oriented, FORTRAN-based simulation language, which offers simple file manipulation, statistics collection, and a time-advance routine for the programmer.

b) ESTIMATE: This program simulated ownship sensor behaviors. When there was no radar emission present, the program updated the position of each platform. When a radar emission was detected, the program generated estimated bearing information for the source of that radar emission. The main function of this program was to generate an "estimated position" for each platform through Monte Carlo simulation. It took the output file from SCENARIO, which included all true information required, and generated the "detected position" by the procedures provided by Law and Kelton [Page 258, 1982].

A sensor on board a ship can detect positions of platforms precisely only within its effective range. When the distance between the detected platform and ownship exceeds this range, precision decreases. For simplicity in this experiment, sensor accuracy was simulated by adding noise to the true position of the platform. The estimated location of a platform was computed as its true location (known from SCENARIO) plus a random term. The random term was a function of the distance between ownship and the detected ship. It was assumed that the estimated location of a platform followed a bivariate normal distribution [Law and Kelton, 1982].

c) POLYGON and LINE: These two programs utilized the Interactive Graphic Library (IGL) software support of Tektronix 4170 to facilitate the graphical display of position information on the Tektronix 4107. Symbols such as squares, lines, ellipses, and characters can be easily generated through this support. POLYGON took the output file from ESTIMATE and drew the relevant symbols for the Polygon type display, which contained ellipses, squares, lines and a circle. The size of the ellipse represented the region that a platform would be in. When a radar emission was detected, POLYGON took the information from ESTIMATE to draw a region for the source of the radar emission. The program LINE did basically what POLYGON did except it did not draw an ellipse and when a radar emission was detected, it drew a line passing through the mean estimated position of the emission source instead of a region.

d) READY: This program, which ran on the HP-86, was used to pace the experiment and to record the associations made by the subject. The pacing was performed by transferring data between the Tektronix 4170 and the HP-86. When there was no sensor-report event, the Tektronix 4170 simply sent an arbitray datum to the HP-86. Then the HP-86 responded with the simulation time. When sensor-report events occured, the Tektronix 4170 sent the request for subject input to the HP-86. Next, the HP-86 asked the subject to enter his choice. After the subject entered his decision, the HP-86 returned the answer to the Tektronix 4170. All associations made by the subject were shown as a table on the monitor of HP-86 for the subject's reference (Figure 3-3).

3) Written materials:

Written materials included two sets of instructions to the subject, a data sheet, a test sheet, and a trial sheet. A copy of these materials can be found in Appendix A.

ASSOCIATION INFORMATION

platform/association

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
1							C																							
2		G		G																										
3																														
4	A				B	B																								
5																														
6								E		I																				
7			F																											
8									G																					

Figure 3-3 Display of the HP-86.

STIMULI AND RESPONSE

The stimuli used in this experiment were a series of sensor-report events. For example, under the Polygon type of display, the presentation of a sensor-report event was shown as in Figure 3-4. In Figure 3-4(a), the display represents the state prior to a sensor-report event. After a period of time, a radar emission of type B was detected from either platform 3 or 1. Hence, a sensor-report event occured. This is represented in Figure 3-4(b). For the Line type display given the same scenario as in Figure 3-4, the situation is represented in Figure 3-5.

The subject was required to make one association when a sensor-report event occured. During the other time, the screen displayed the locations of all platforms. The positions of platforms were updated according to the time elapsed between each update.

PROCEDURE

The subjects were tested individually, each of them performing one treatment. Four subjects participated in each treatment.

The subject's task was to associate each radar emission given by a sensor report with one of the

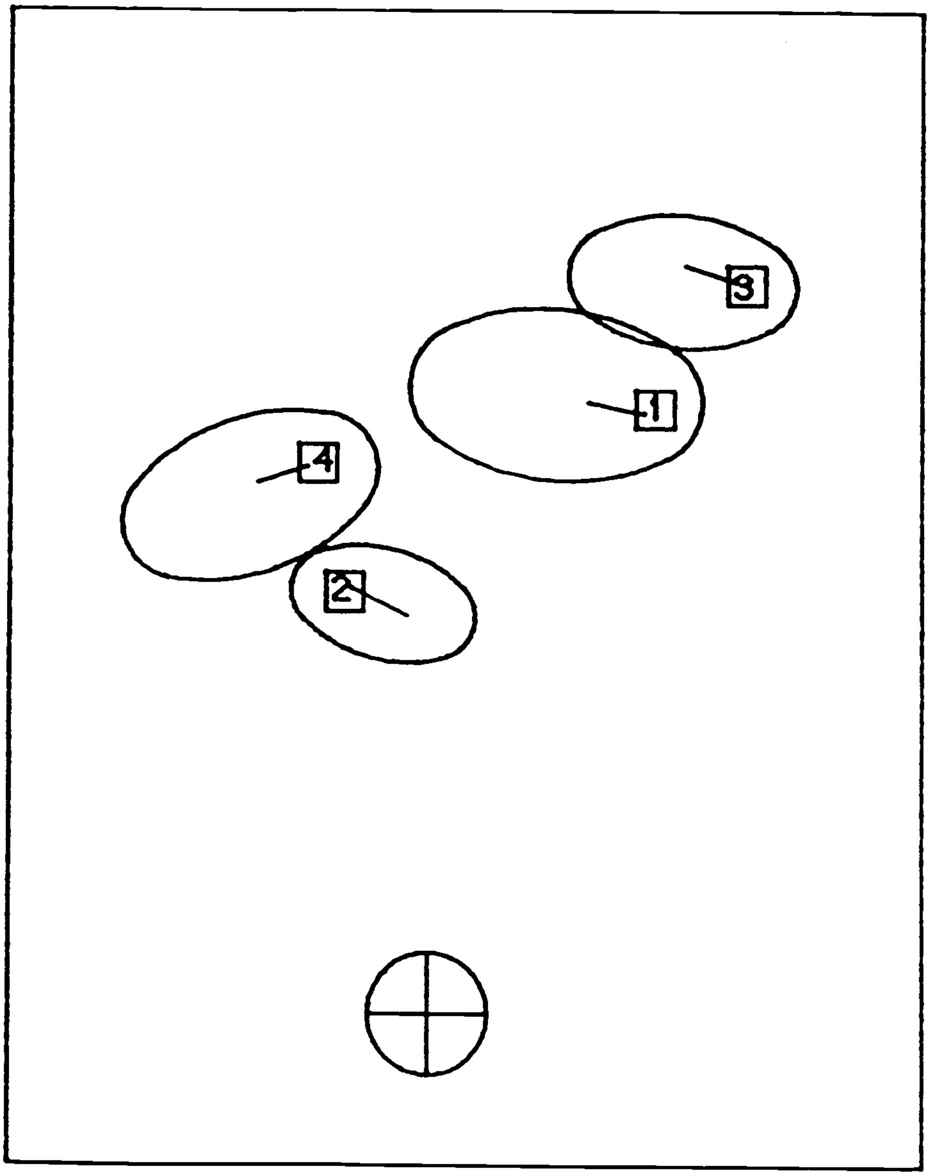


Figure 3-4a A state prior to a sensor-report event under the Polygon type display.

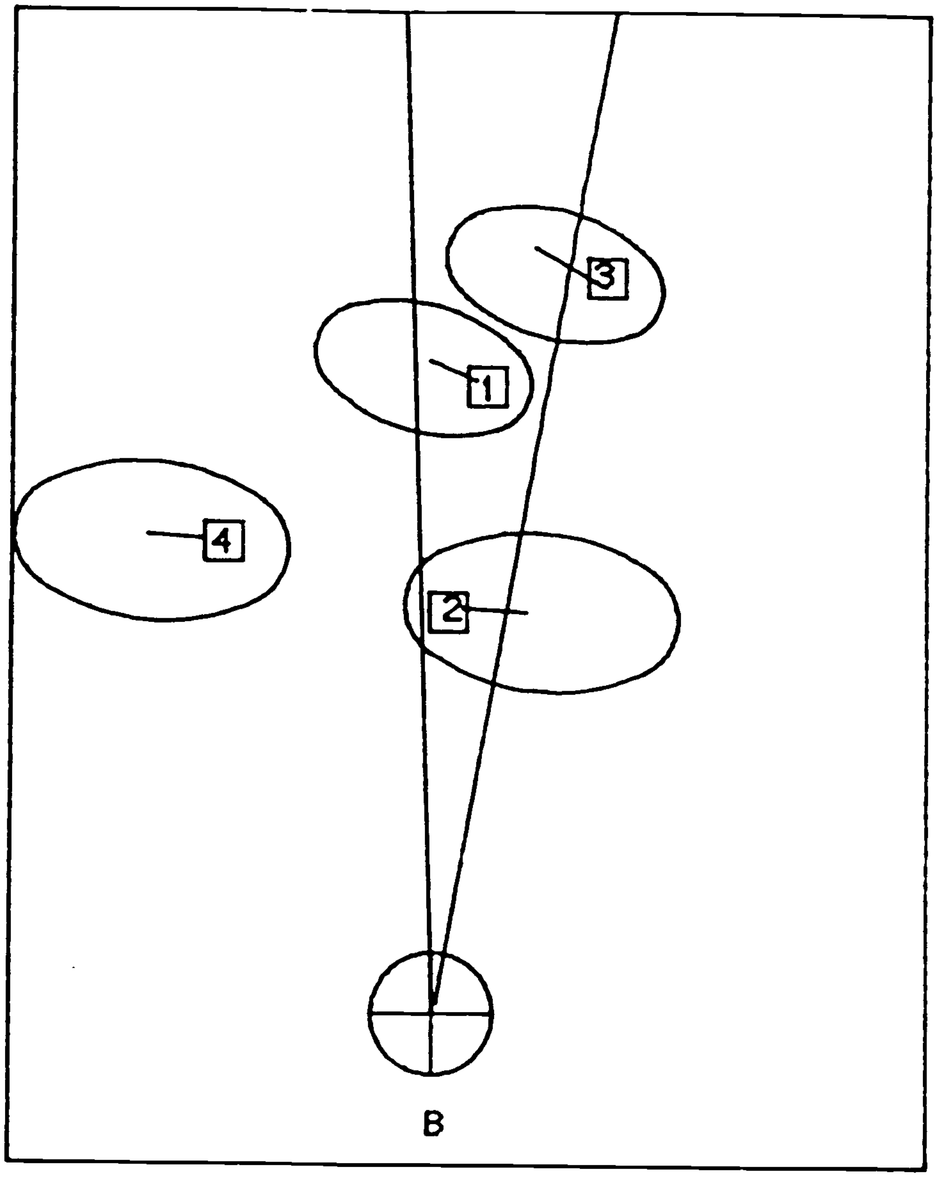


Figure 3-4b A state following the sensor-report event under the Polygon type display; the detected emission was of the B frequence band.

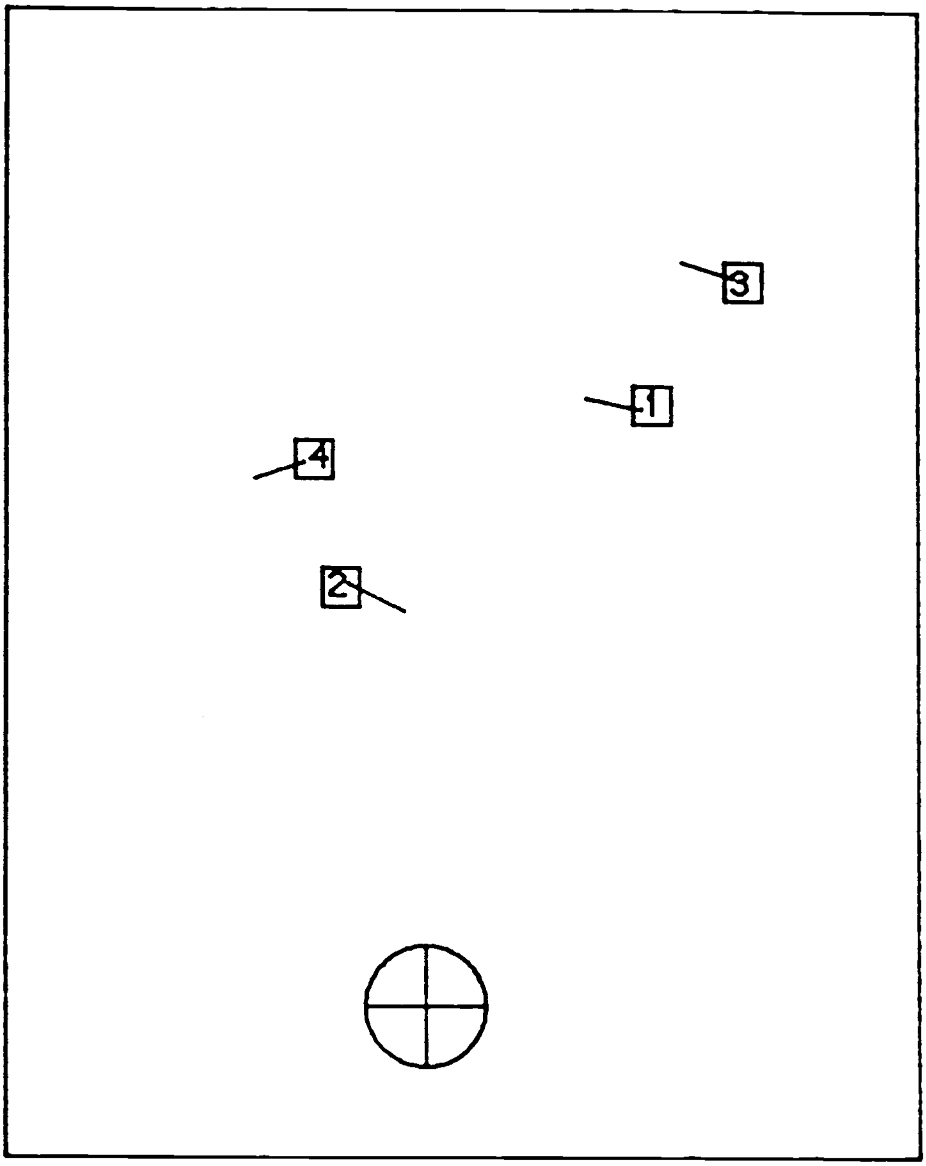


Figure 3-5a A state prior to a sensor-report event under the Line type display.

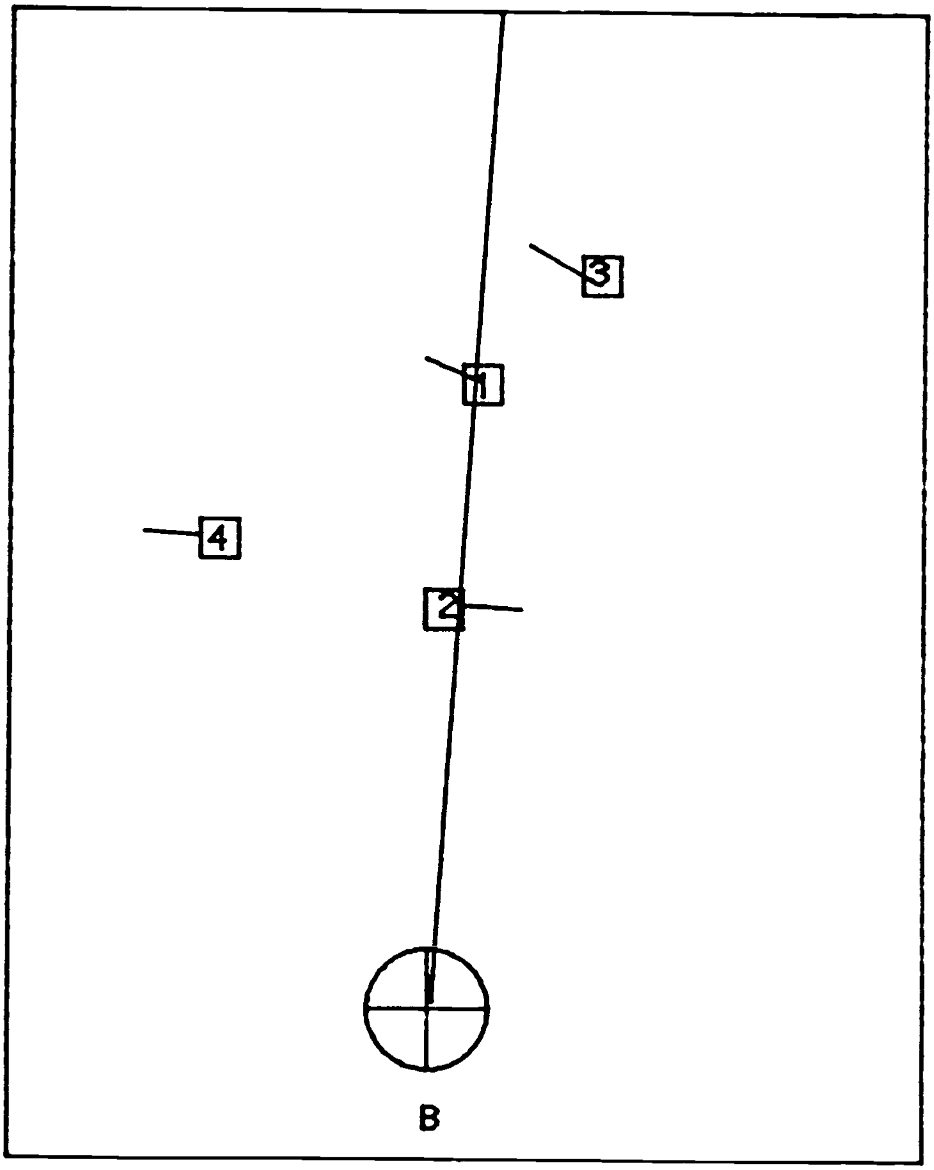


Figure 3-5b A state following the sensor-report event under the Line type display; the detected emission was of the B frequency band.

platforms. This required the subject to monitor the display and respond when a sensor-report event occurred. When this happened, the subject responded by hitting a key designating the identity of the platform which he thought was the most likely one responsible for generating the emission.

Testing was conducted in a small quiet room. The subject was first seated at a desk 29 inches in height with the center point of the Tektronix 4170 screen at a height of 12 inches above the desk. The HP-86 monitor was on the left hand side of the subject at the same height as the Tektronix 4107. Viewing distance was 22 inches (Figure 3-6).

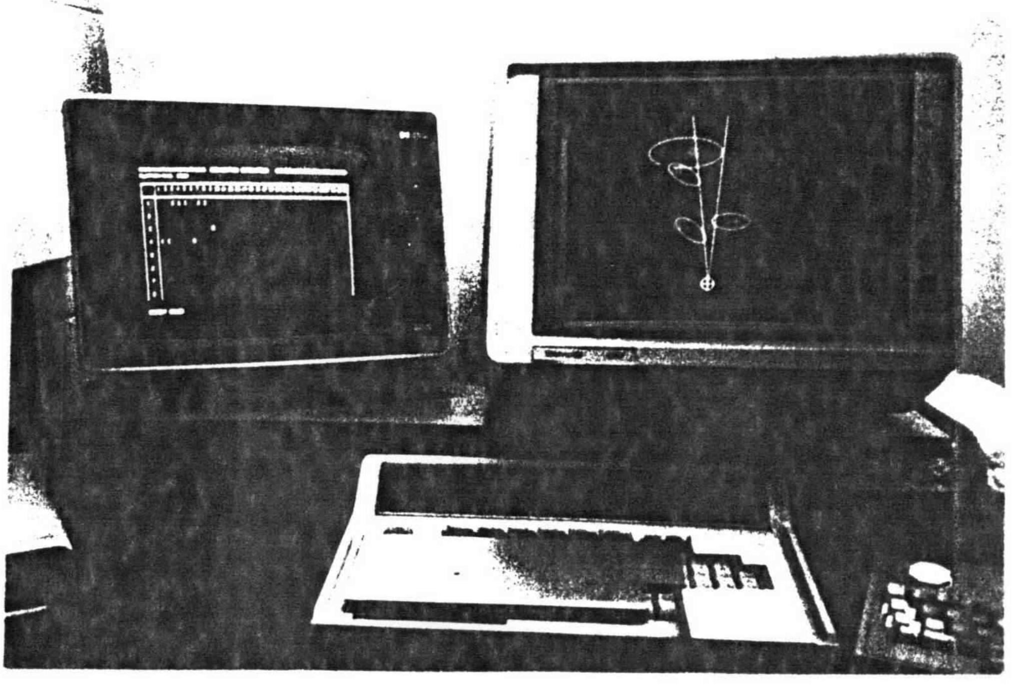


Figure 3-6 Experimental set up.

Next, the subject sat at the station, and detailed written instructions were provided to familarize him with the task. Two sets of instructions were used in the experiment, but each subject read only one of them according to the display assigned. Then he was given a written test to be qualified for the task. This was done to insure that the subject understood the database of ships and radars in the experiment. When the subject passed the written test and indicated that he understood the task procedures, 10 trials with feedback were provided for him as warmup. A trial sheet was provided to the subject during the 10 trials. It was required for the subject to fill out the trial sheet for these trials.

After the detailed instructions and warm up trials, the test began. Upon completion of the experiment the subject was given a questionnarie to fill out. The purpose of this questionnaire was to gather opinions and strategies used by the subjects. Two questionnaires were designed for the two types of displays; each subject received the one which was related to his treatment. Responses were used to perform subjective evaluation of the display on which he was tested, and to discover the strategies used in this experiment. A copy of the questionnarie can be found in Appendix A. Results of the questionnaires are discussed in Chapter 5.

SOURCES OF VARIATION

A characteristic of all experiments is variation. The sources of variation in this experiment are discussed below.

1) Scenario: The design of a scenario was based on two levels of data rate and two levels of ship density. There were altogether four different scenarios. The main objective of this experiment was to compare display types; therefore, when data rate and ship density were fixed, subjects with different displays still viewed the same scenario. For example, the subjects in treatment 1 (Polygon 15 emissions 4 ships), and treatment 5 (Line 15 emissions 4 ships), viewed the same scenario but under different displays. The same was true for subjects in treatment 2 and 7, 3 and 6, and 4 and 8.

2) Time of day: No early morning or late night sessions were conducted. All experiments were conducted between 9 A.M. and 5 P.M..

3) None of the subjects were color-blind, which might have influenced interpretation of the color coding.

The results of this experiment are presented in the following chapter.

CHAPTER 4

RESULTS

This section presents the statistical analysis and results of the experiment. The experiment was a completely randomized 2^3 factorial design, in which the factors were T (display type), R (data rate), and D (ship density). Performance measures of the experiment were percent correct and response time.

Two major statistical analyses are performed, analysis of variance (ANOVA) and comparison of means. The assumptions underlying these analyses are that the samples were independent, identical samples from normally distributed random variables. The process of analysis for each explanatory variable started with ANOVA, and then t-tests were made to compare the means of interest.

The ANOVA calculations were performed by a computer program developed for the IBM PC based on a statistical model adopted from Miller and Freund [1977]. A copy of this program can be found in Appendix C. Calculations for t-tests were done using the SIPS (Statistical Interactive Porgramming System) developed at Oregon State University. All statistical analyses employed the 0.05 level of significance.

DATA TRANSFORMATION

Cox [1970] described that the most useful way to deal with binary data is to perform a logistic transformation for fullfilling the assumption of normality. Percent correct of each subject was obtained by computing the sum of successful associations divided by total number of associations. Therefore, the raw data of percent correct were transformed according to the logistic transformation:

$$u_i = \ln (p_i/1 - p_i) \qquad i = 1 \text{ to } 32,$$

where p_i is the percent correct for the i^{th} subject, and u_i is the natural log of the success/failure ratio. For example, if pl = 0.9333, then $u_1 = \ln(0.9333/0.06667)$, which is 2.639, and so on. The original data and their transformations can be found in Appendix D.

PERCENT CORRECT

The analysis concerning percent correct was performed on the transformed data.

(A) ANOVA

The ANOVA table for percent correct is presented in Table 4-1. The effect of display type was not significant, which was not expected. [$F(1,21) = .1516$]. This will be discussed in the "Discussion" section. However, data rate had a very significant effect [$F(1,21) = 112.2655$]; as expected, the higher the data rate, the more information that had to be processed by the subject and the higher the probability of making wrong associations. The effect of D was also significant [$F(1,21) = 156.0246$]; in general, more ships meant that there were more choices and hence it was more difficult to decide. Figure 4-1 shows the effects of R and D. It can be easily seen that there is a trend to a lower performance as the rate and density increase.

By the ANOVA table, the effect of R interacted with that of D [$F(1,21) = 41.3891$, $p < .01$], which was expected, too. The display type T neither interacted with R nor with D. The three factor interaction was significant, too.

B) t-test

Inferences about mean performance under each factor were drawn. Before performing the tests, F tests were

Source of variation	Degree of freedom	Sum of squares	Mean squares	F value
Replicates	3	0.0191	0.0064	0.0673
Main effects				
T	1	0.0143	0.0143	0.1516
R	1	10.6127	10.6127	112.2655**
D	1	14.7664	14.7664	156.2046**
Two-factor interactions				
TR	1	0.2102	0.2102	2.2233
TD	1	0.3126	0.3126	3.3067
RD	1	3.9126	3.9126	41.3891**
Three-factor interaction				
TRD	1	2.0746	2.0746	21.9461*
Error	21	1.9852	0.0945	
Total	31	33.9076		

Table 4-1 ANOVA table for performance accuracy.

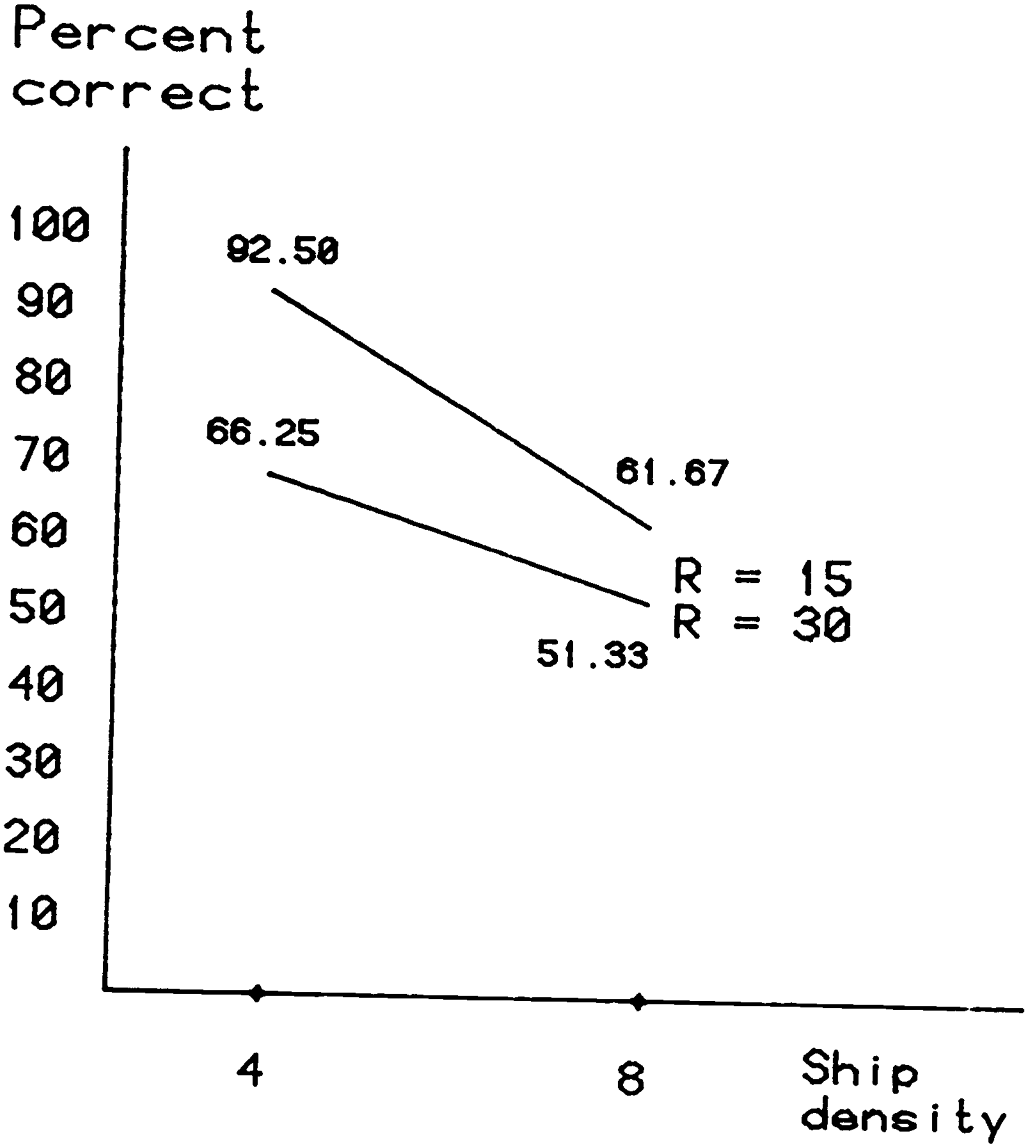


Figure 4-1 Percent correct as a function of Ship density and Data rate.

done to support the assumption that the two samples under comparision had a common population variance. For factor T, the display type, three treatment comparisons showed significant differences. Figure 4-2 and Figure 4-3 illustrate the descriptive results under this factor.

There was no difference between the display type when R = 15 and D = 4. However, the other three comparisons were all significant. For performance under R = 30 and D = 4, and R = 15 and D = 8, the polygon type of display was better than the Line type display. Performance under R = 30 and D= 8, the Line type display was better. The t values for each comparison are summarized in Table 4-2. The results were not clear enough to draw the conclusion that which type of display was better, or there was a difference between the displays.

Hypothesis: H_o: $\mu_A=\mu_B$ vs H_1^+: $\mu_A>\mu_B$ or H_1^-: $\mu_A<\mu_B$

Comparison between treatment	Calculated t value	Decision
1 and 5	-1.00	Accept H_o
2 and 6	2.9422	Accept H_1^+
3 and 7	3.1692	Accept H_1^+
4 and 8	-3.1538	Accept H_1^-

Table 4-2 t values for treatment comparisons; comparisons between display types.

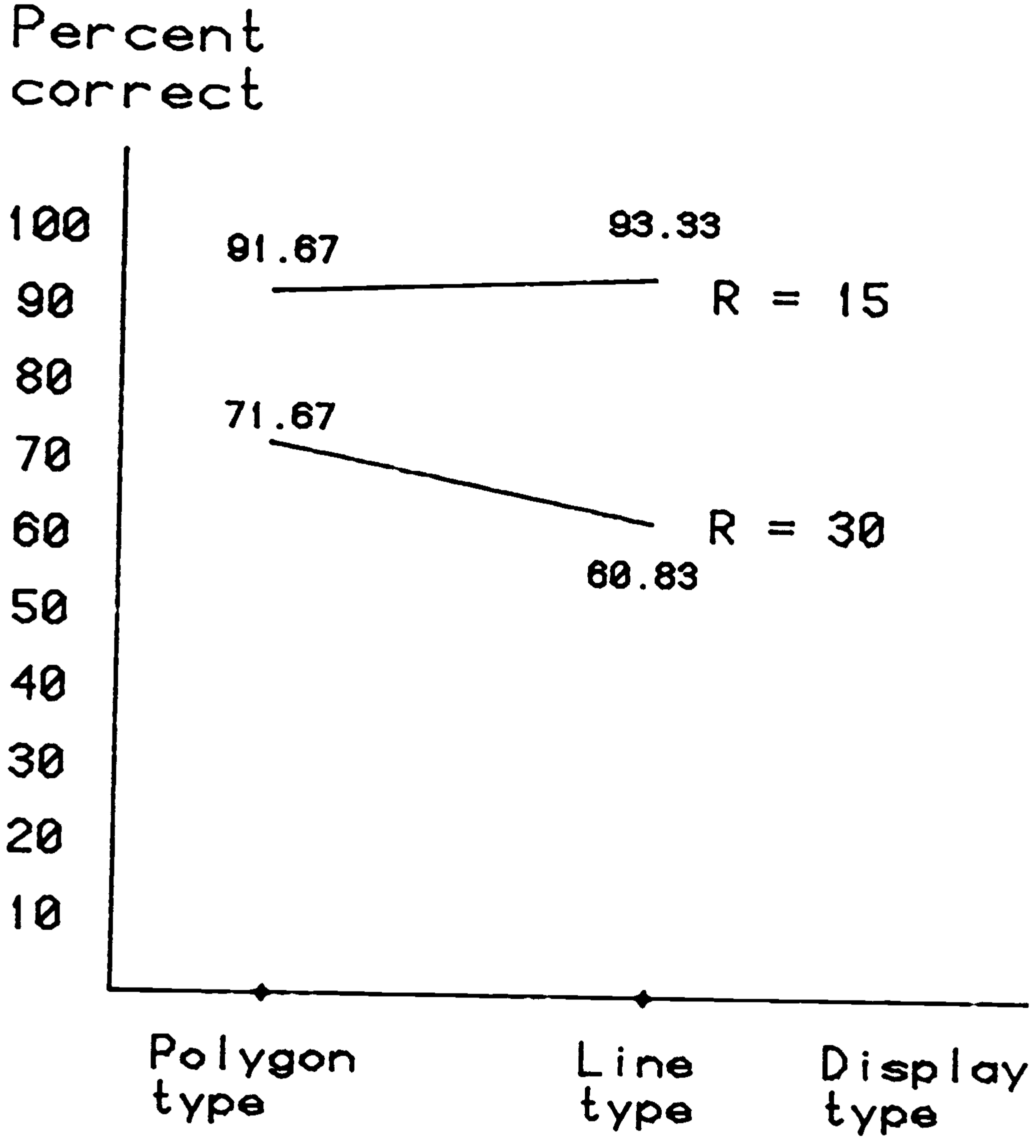


Figure 4-2 Percent correct as a function of Display type. Case 1: Ship density = 4.

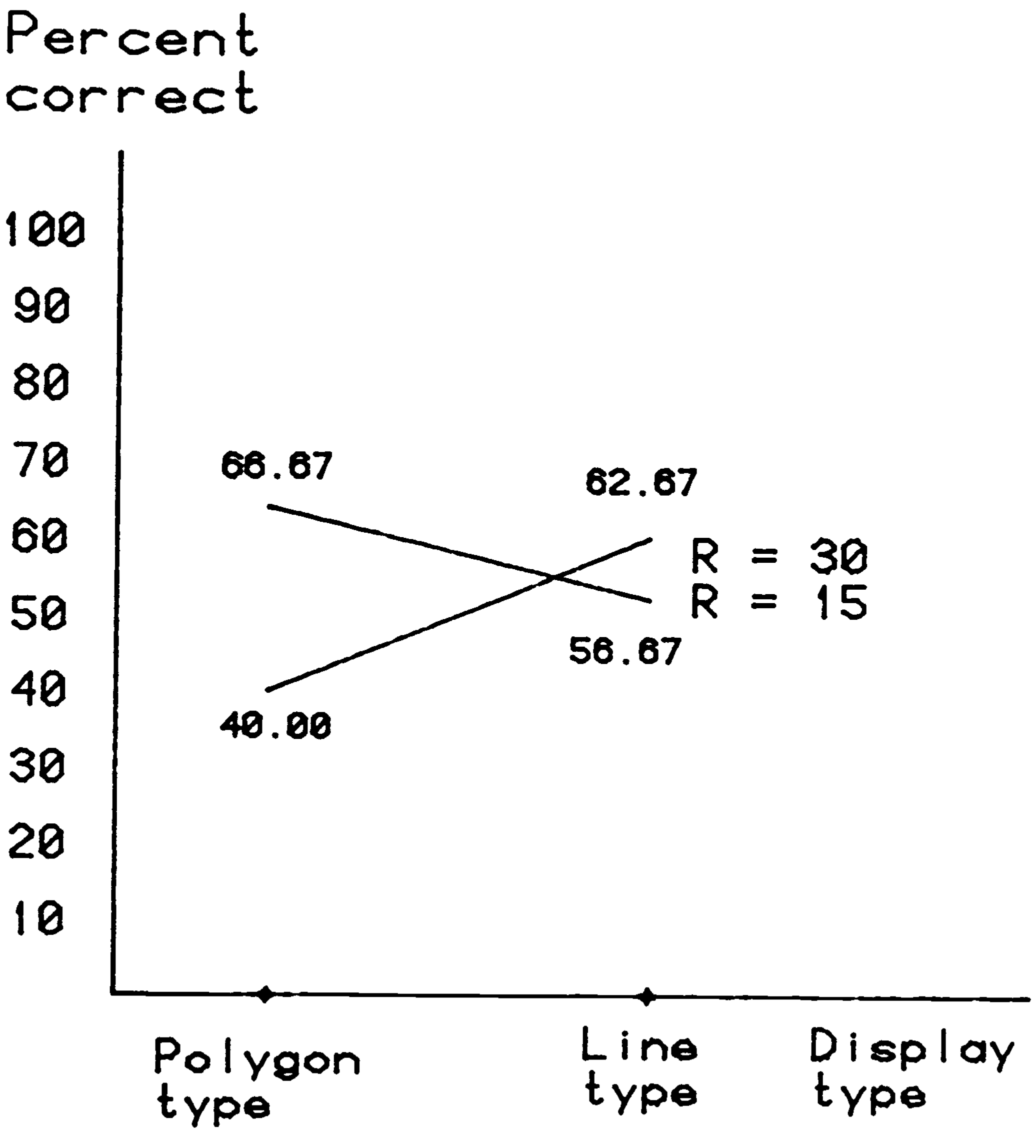


Figure 4-3 Percent correct as a function of Display type. Case 2: Ship density = 8.

Because T was not significant in the ANOVA, the results from the two types of displays were collapsed for further analyses. Figure 4-4 shows the performance as function of data rate. As the result of this analysis (Table 4-3), it is clear that the performance was better under the lower data rate condition.

Comparison between treatment	Calculated t value	Decision
1+5 and 3+7	12.0417**	Accept H_1^+
2+6 and 4+8	1.8658	Accept H_o

Table 4-3 t values for treatment comparisons; comparisons between data rate.

Figure 4-5 shows the performance as a function of ship density. The t values are listed in Table 4-4. It can be seen that although at significance level = 0.01 at data rate [D] = 30, 2.6489 < 2.97; but it would be significant if the significant level was 0.05. This means that the performance was better under ship density = 4. The t values for the above comparisons are contained in the SIPS output form and can be found in Appendix E.

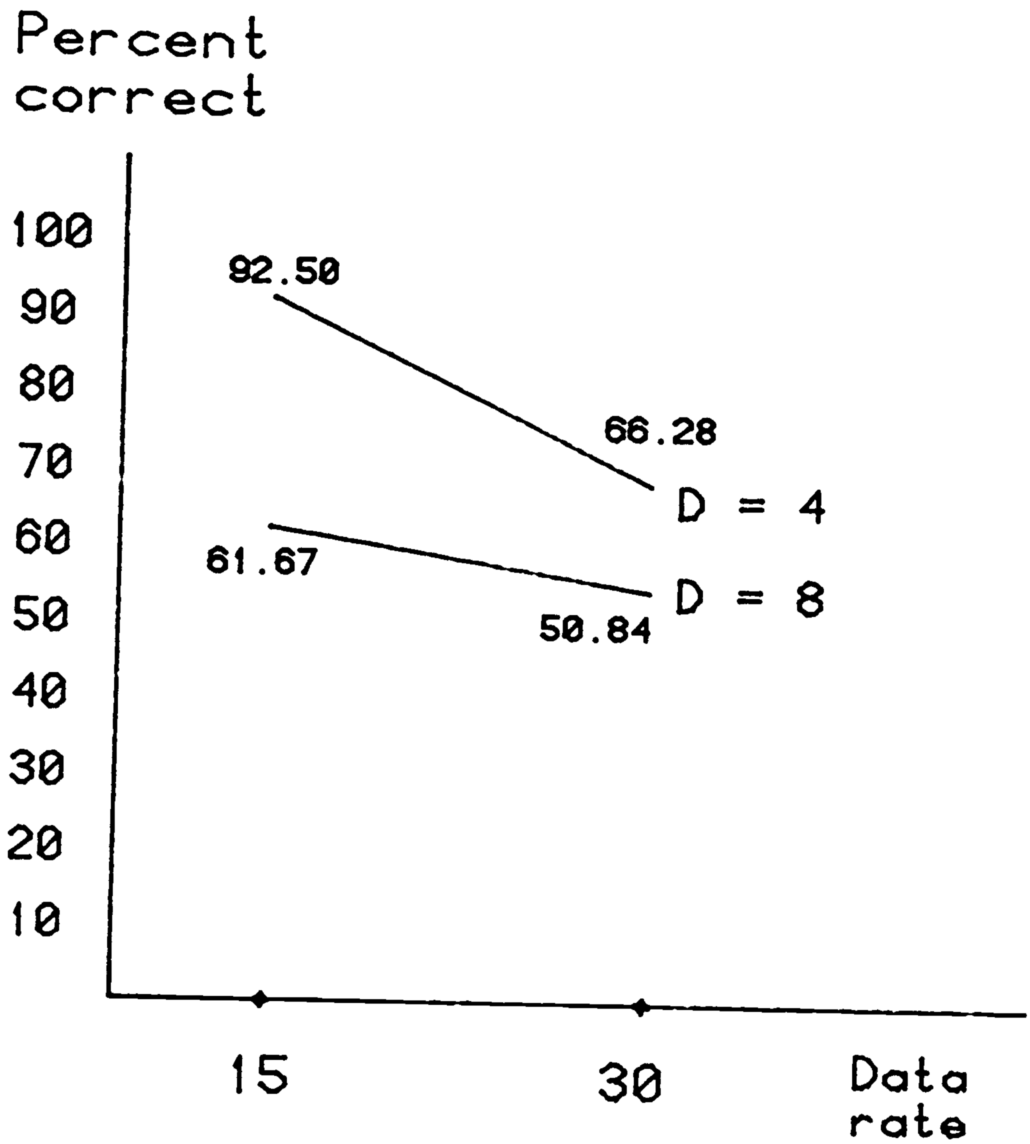


Figure 4-4 Performance as a function of Data rate.

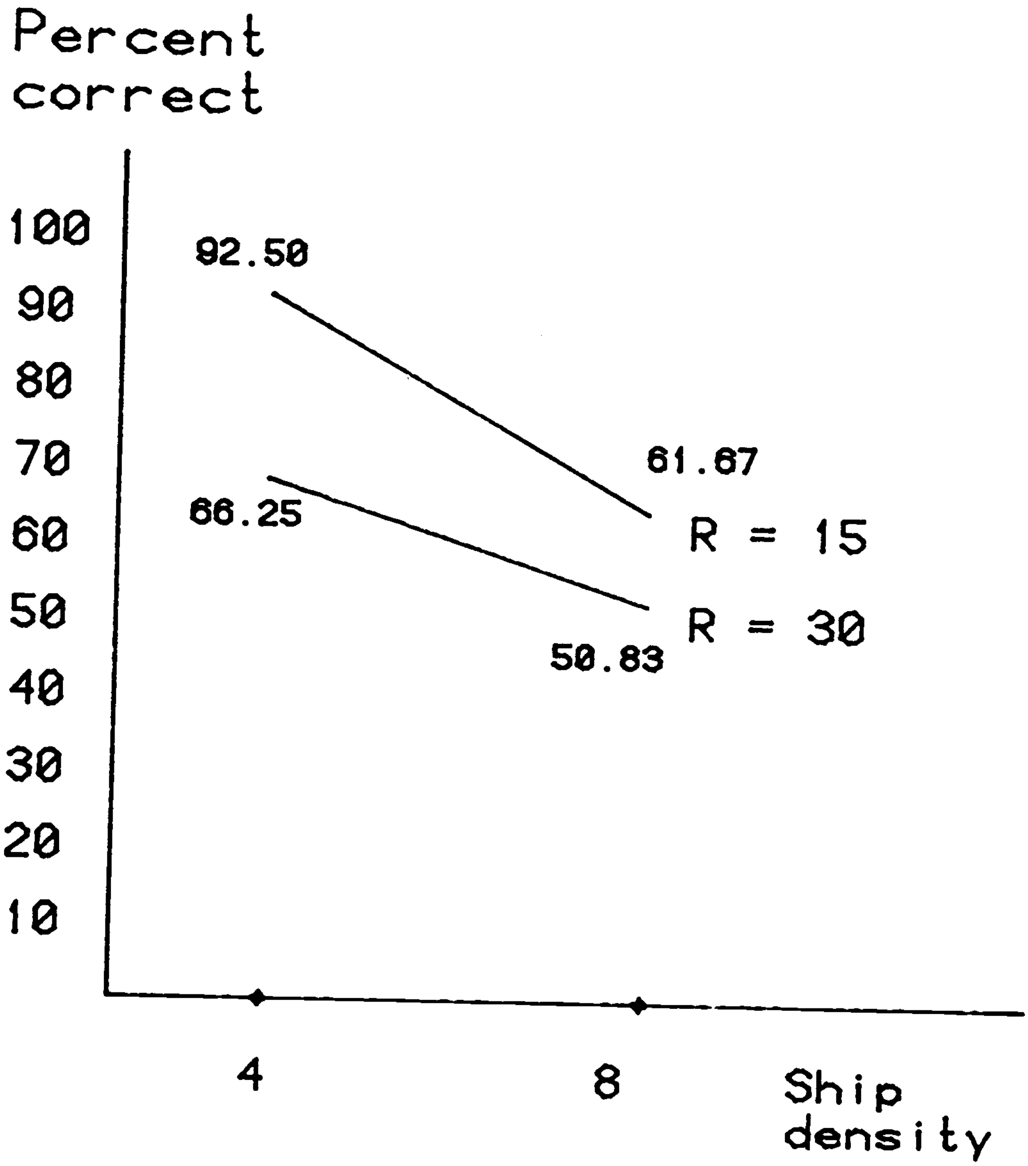


Figure 4-5 Performance as a function of Ship density.

Comparison between treatment	Calculated t value	Decision
1+5 and 2+6	14.3911**	Accept H_1^+
3+7 and 4+8	2.6489	Accept H_o

Table 4-4 t values for treatment comparisons; comparisons between data density.

RESPONSE TIME

A) ANOVA

Table 4-5 shows the ANOVA table for response time. The results showed that neither the main effects nor the interactions were significant. The reasons for these results could be due to the design of this experiment, which will be discussed in the next chapter.

B) t-test

Before comparing the mean performances between each factor, the data were combined. For example, the performances under different data rate and ship density were combined to compare the performance of two display types. To compare the performance under the two data rates, data from different display types and ship densities were combined, and so on.

Source of variation	Degree of freedom	Sum of squares	Mean squares	F value
Replicates	3	1144.3570	381.4523	2.4677
Main effects				
T	1	19.4376	19.4376	0.1257
R	1	464.4963	464.4963	3.0050
D	1	61.1291	61.1291	0.3955
Two-factor interactions				
TR	1	8.6930	8.6930	0.0562
TD	1	0.3894	0.3894	0.0025
RD	1	197.1007	197.1007	1.2751
Three-factor interaction				
TRD	1	53.2377	53.2377	0.3444
Error	21	3246.0880	154.5756	
Total	31	5194.9290		

Table 4-5 ANOVA table for performance of response time.

Figure 4-6 shows the mean response time for the two display types. It can be seen that there was no statistical difference. The t value was .3357, which was less than the table value of 2.0423 (degrees of freedom = 30).

The performance under different data rates was not significant, either. Even though Figure 4-7 seems to indicate that response time was longer for data rate = 30, this difference was not statistically significant (t value = -1.7163, table t value = -2.0423)

As above, the effect of ship density on response time was not significant. The t value was -1.1549 and the mean response time is shown in Figure 4-8.

The t values for the above three comparisons are contained in the SIPS output form and can be found in Appendix E.

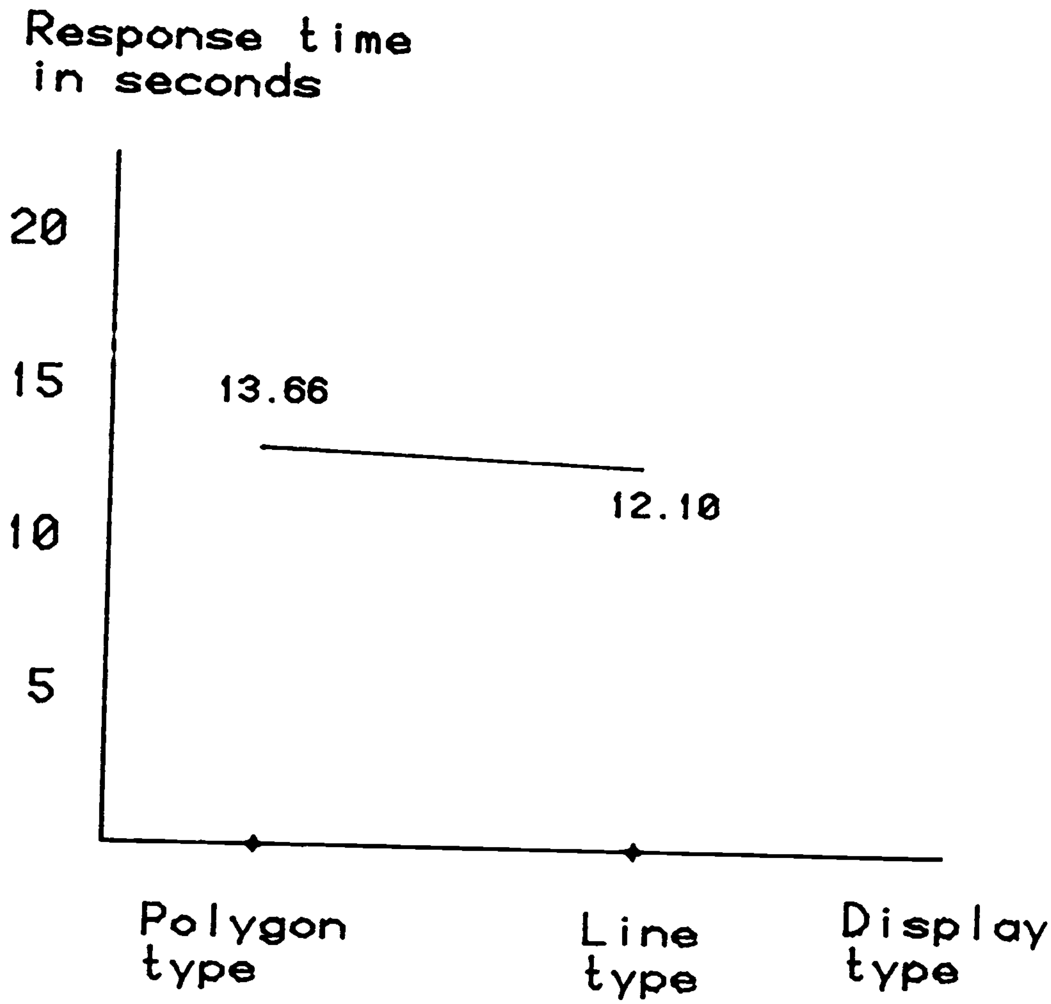


Figure 4-6 Response time as a function of Display type.

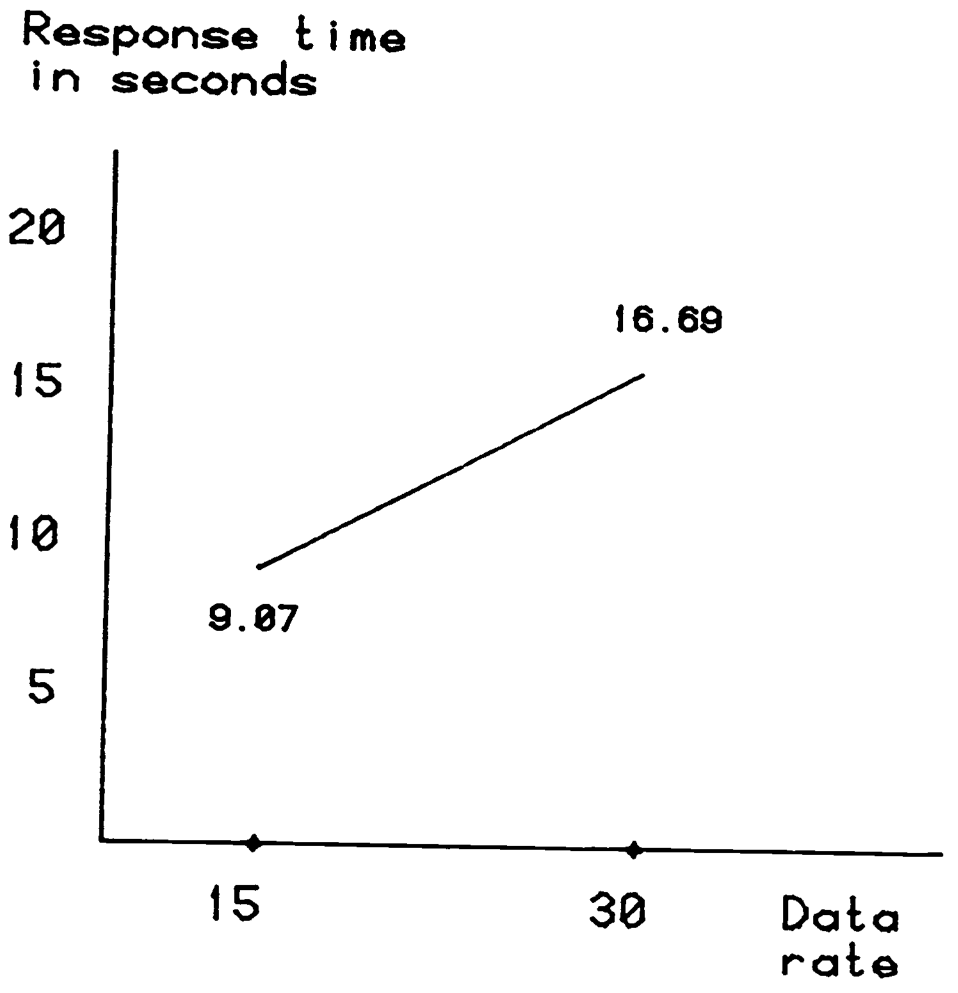


Figure 4-7 Response time as a function of Data rate.

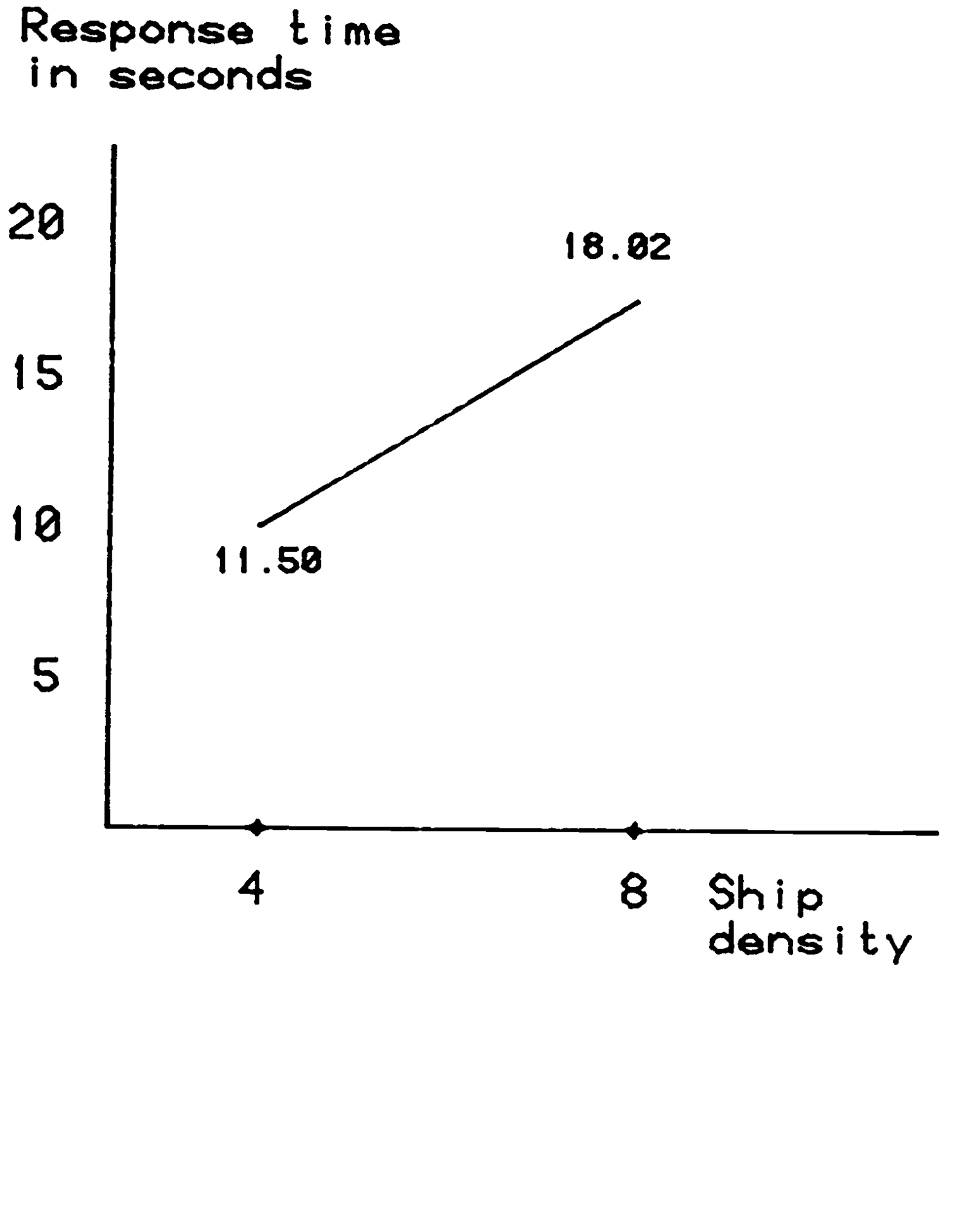


Figure 4-8 Performance as a function of Ship density.

SUMMARY

Table 4-6 presents the statistical significance for the ANOVA for all levels of treatment in all categories considered for this experiment.

The summary for all t tests under both performance measurements is presented in Table 4-7. For the comparison of display types, "1 vs 5" stands for "treatment 1 vs treatment 5", and so on. The table shows comparison criteria, calculated t values for each test, table t values under given test and degrees of freedom, test decision, and the factor which resulted in higher (better) performance.

Source of variation	Percent correct	Response time
Replicates	NS	NS
Display type (T)	NS	NS
Data rate (R)	S	NS
Ship density (D)	S	NS
T * R	NS	NS
T * D	NS	NS
R * D	S	NS
T * R * D	S	NS

TABLE 4-6 Summary of Significance.

Comparison factor: Display type

Performance measurement	Comparison criteria	Calculated t statistic	Table t value	Test decision	Higher performance criteria
Percent correct	1 vs 5	-1.00	t=2.4469 (d.f.=6)	not significant	no difference
	2 vs 6	2.9421	t=2.4469 (d.f.=6)	significant	Polygon type display
	3 vs 7	3.1692	t=2.4469 (d.f.=6)	significant	Polygon type display
	4 vs 8	-3.1538	t=2.4469 (d.f.=6)	significant	Line type display
Response time		0.3357	t=2.0423 (d.f.=30)	not significant	no difference

TABLE 4-7a Summary of the t tests (significant level = .05).

Comparison factor: Data rate

Performance measurement	Comparison criteria	Calculated t statistic	Table t value	Test decision	Higher performance criteria
Percent correct	case 1: test under ship density = 4.	12.0417	t=2.1450 (d.f. = 14)	significant	Low data rate (15 associations per session)
	case 2: test under ship density = 8.	1.8658	t=2.1450 (d.f. = 14)	not significant	no difference
Response time		-1.7163	t=2.0423 (d.f. = 30)	not significant	no difference

TABLE 4-7b Summary of the t tests (significant level = .05).

Comparison factor: Ship density

Performance measurement	Comparison criteria	Calculated t statistic	Table t value	Test decision	Higher performance criteria
Percent correct	case 1: test under data rate = 15	14.3911	t=2.1450 (d.f. = 14)	significant	Low ship density
	case 2: test under data rate = 30	2.6489	t=2.1450 (d.f. = 14)	significant	Low ship density
Response time		-1.1549	t=0423 (d.f. = 30)	not significant	no difference

TABLE 4-7c Summary of the t tests (significant level = .05).

CHAPTER 5

DISCUSSION

The two measurements of human performance used in this experiment were accuracy and response time. They were applied for two different types of displays (i.e., Polygon and Line) under a set of combinations of two levels of independent variables (i.e., data rate and data density). This section presents the discussion of the results from the experiment.

ACCURACY

There was no significant difference in overall accuracy between the two displays [$F(1,21) = .1516$, p-value = .10]. While there were some differences between the display types over different treatments, these differences were inconsistent. For example, the effect of display type was significant when comparing the two sample means under the same data rate and same data density; but neither display was consistently superior to the other. This can be explained by the possibility that some effects were canceled out by other effects in the treatments. Also, this may be related to factors such as the design of the displays, training of subjects, and the scenarios being used.

1) Design of the displays:

The design of the displays was the major concern in this research project. We were to test if the polygon type display could be an aid to human operators while performing the MTII task. One way to investigate this qualitative question is to look at the subjective responses from the post-experiment questionnaries.

The strategies employed by the subjects were obtained from the questionnaries as shown in Table 5-1. The strategies employed by the Polygon type subjects are listed in Table 5-1 (a); the strategies of the Line type subjects are listed in Table 5-1 (b).

Under the polygon type display, 31.25% of the subjects chose the platform which was closest to the middle of the sensor report (Center of the report), another 31.25% employed the largest intersection coverage of the uncertain region and the sensor report as their strategy (Intersection coverage), and 31.25% of the subjects relied on the direction of movement of the platform (direction of platform) when there was more than 1 candidate for the target platform. 6.25% of the subjects (i.e., one subject out of 16) used previous associations as a reference. This subject indicated that he made detailed reference to the HP-86 monitor. For example, as illustrated in Figure 5-1, the first 13

Polygon type display Strategies employed	Percent
Center of the report	31.25%
Intersection coverage	31.25%
Direction of platform	31.25%
Previous associations	6.25%

Table 5-1 (a) Strategies used in Polygon display.

Line type display Strategies employed	Percent
Closeness to report	75%
Distance to ownship	25%

Table 5-1 (b) Strategies used in Line display.

ASSOCIATION INFORMATION

platform/association

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
1							C	C		A			A-																	
2		G	F	G					G																					
3												D																		
4	A				B	B					B																			
5																														
6																														
7																														
8																														

Figure 5-1 This information reveals that platform 4 possesses radar type B.

associations show a tendency that platform 4 (track 4) emits radar B all the time. Hence, it makes more sense to select 4 again on the 14th association given the display on Figure 5-2. Under the Line type display, the strategies were narrowed down to closeness and distance: 75% of the subjects chose the platform which was nearest to the line (Closeness to report), and 25% of the subjects chose the platform which was nearest to ownship. On the quantity of strategies, the subjects using Polygon display employed more strategies than those who using Line type display. The mean accuracy under the two displays differ by a very small margin. Most of the subjects in this experiment turned out to be performing an absolute judgement task instead of thinking about the meaning behind the abstract symbols. This leads to the discussion of training.

2) Training

There was only one subject in the experiment who did a detailed trace on his previous associations. This strategy was good only if his associations of a specific platform were accurate, otherwise it could be even worse upon the completion of the experiment.

The other subjects did not pay much attention to the data sheet provided to them as the database of platforms and radar information in this experiment. One important

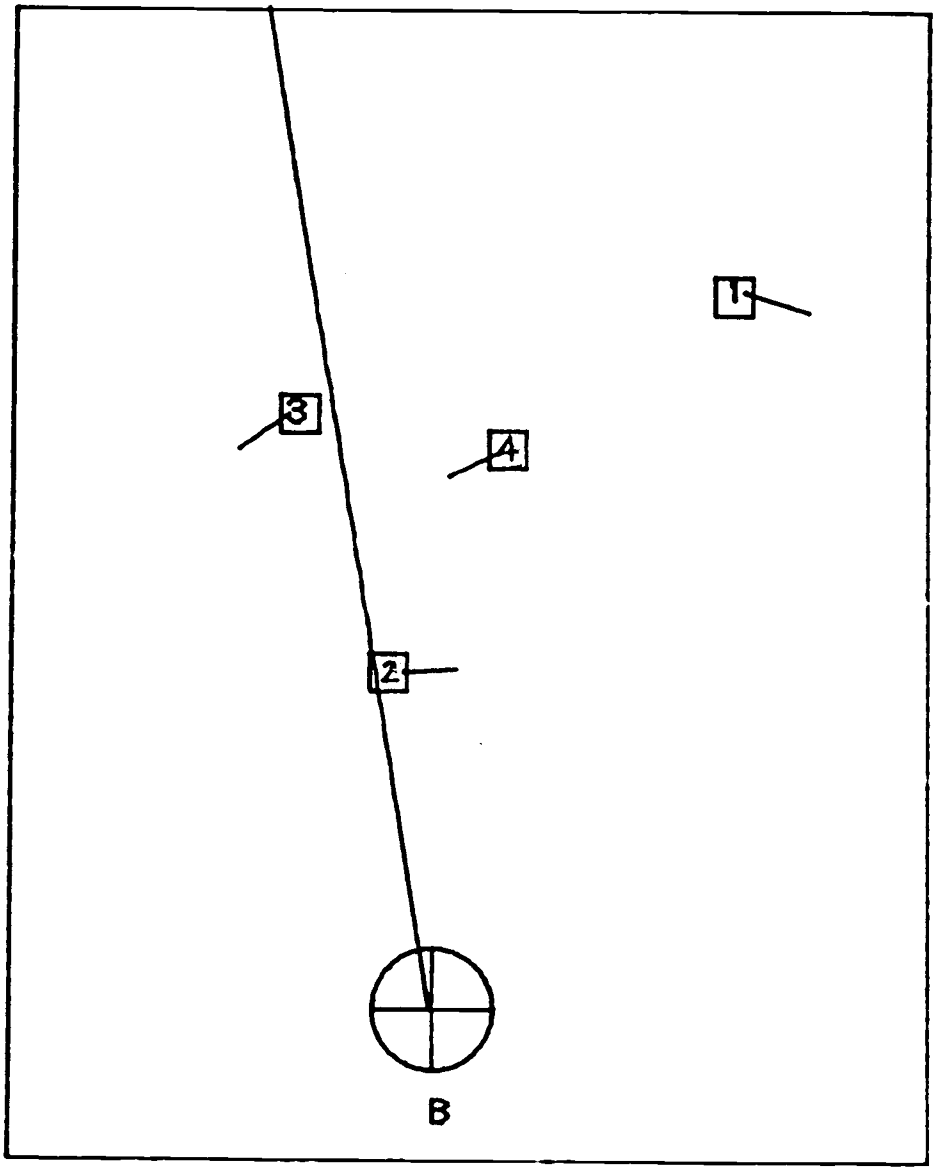


Figure 5-2 The display for the 14th association.

drawback of the design of this data sheet was that two of the platforms (i.e., WHALE and TROUT) had exactly the same type of radars (i.e., they both possessed radar types B, H, and I). This made the first question on the post-experiment questionnaire very difficult to answer if indeed WHALE and TROUT were engaged in the scenario. It would be much better to have only a few different type radars in different platforms in the data sheet.

For most of the subjects a 10-minute warm up trial was not enough to let them understand the task. First, the association task was complicated to perform. Next, it required more elaboration to encode abstract symbols into memory under such an experimental situation. The subjects failed to recognize the meaning behind those symbols. Mills [1973] performed an experiment to evaluate operator capabilities in performing a surveillance task of aircraft track initiation and maintainance. The subjects in his experiment received at least 50 hours of training in order to perform the task. Greitzer et al [1982] pointed out that rehearsal serves as a memory maintenance function that refreshes the encoded representations that would otherwise be lost from primary memory. Because of the lack of experience, it would be considered that the subjects in this study didn't even have enough training to encode enough knowledge to permit successful task performance. Therefore, the subjects possibly did not understand the task well enough. This could be resolved in two different ways. One would be to make the task easier. For example, make the experimental task into a simple stimulus/response task such that little cognitive ability is required. The other one is to prolong the training until each subject receives a specific amount of time. It is quite possible that, since rehearsal is a way to imptove the performance on a task, more training would result in different results for this experiment.

3) Scenario

Important factors affecting performance were the scenarios in this experiment. These scenarios were pseudo war games, which were designed by the experimenter. Figure 5-3 displays a scenario with 4 platforms on the sea. The symbol * represents the time for radar emission of each platform, and the positions of each platform are denoted by showing the current time beside its route. For example, the platform "WHALE" has the highest speed, and it emits its radar at time = 10, 20, and 40 units. Consider at time = 20, it would be easy for the subject to decide where radar E is from (in this example it is from platform "BASS"). But at time=45, while we can see from the scenario that radar E is from platform BASS, it is difficult for the subject to decide while he is viewing the display in real time. Therefore, the degree of

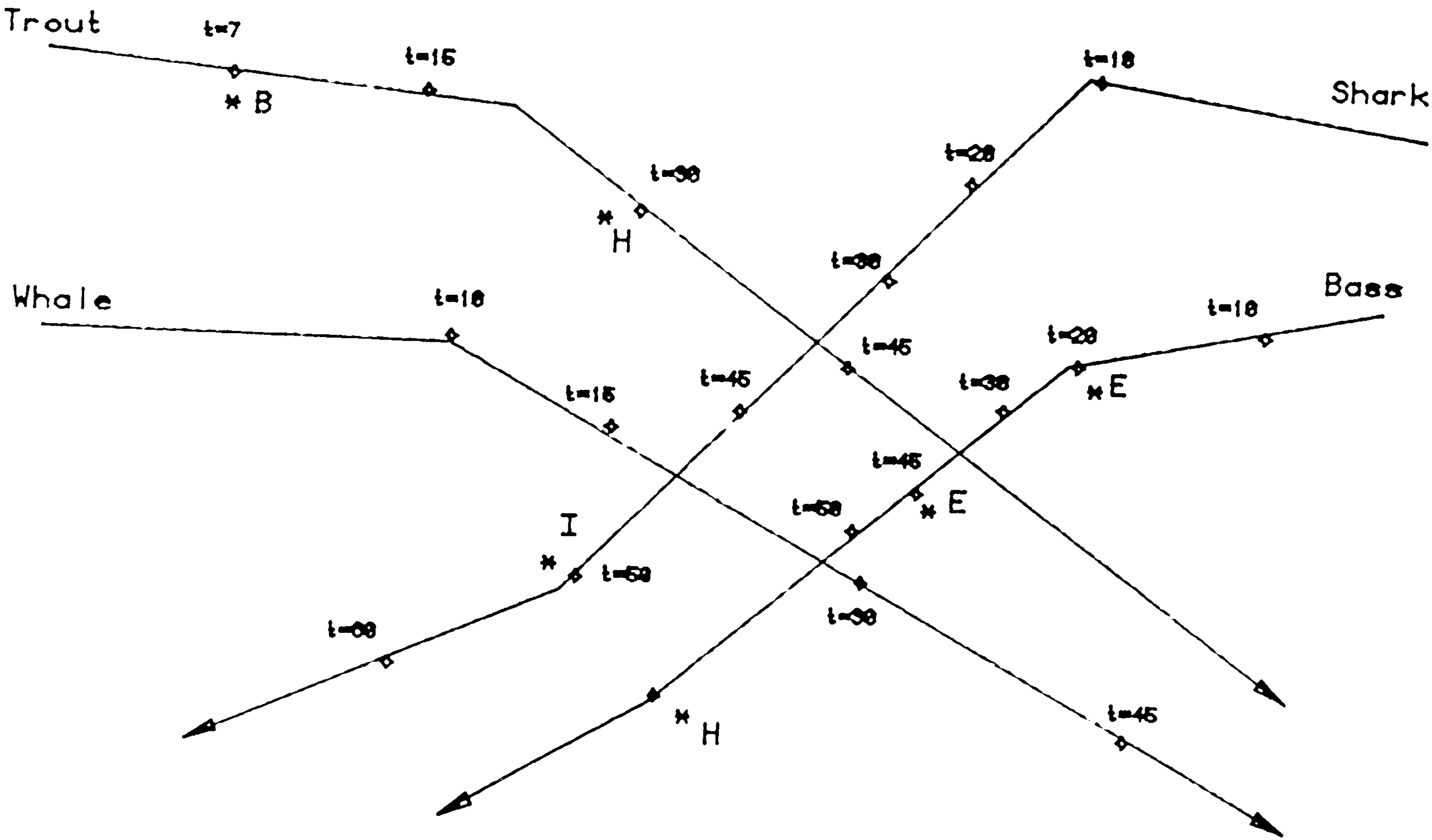


Figure 5-3 An illustration of the scenario.

difficulty varies over a large range. When the scenarios were applied using the two displays, the same scenario could be perceived differently by the same subject. For example, one of the scenes in this experimental task occured as shown in Figure 5-4. Radar emission B was truly from platform 2, but under the Line type display, the subjects employing the closeness strategy would choose platform 1 as their answer (which was a wrong association), but 75% of the subjects used such a strategy.

DATA RATE AND DATA DENSITY

The results revealed a significant difference in accuracy under different data rates [$F(1,21) = 112.2655$, p-value = 0] or under different data density [$F(1,21) = 156.2046$, p-value = 0]. The results were in accordance with Greitzer et al [1982]. There is a strong evidence that decision making performance degrades substantially as track load increases beyond human capacity limits.

One drawback of this 2^3 factorial design is that there are only two levels of independent variables: this makes the prediction of performance for other data rates and densities more difficult.

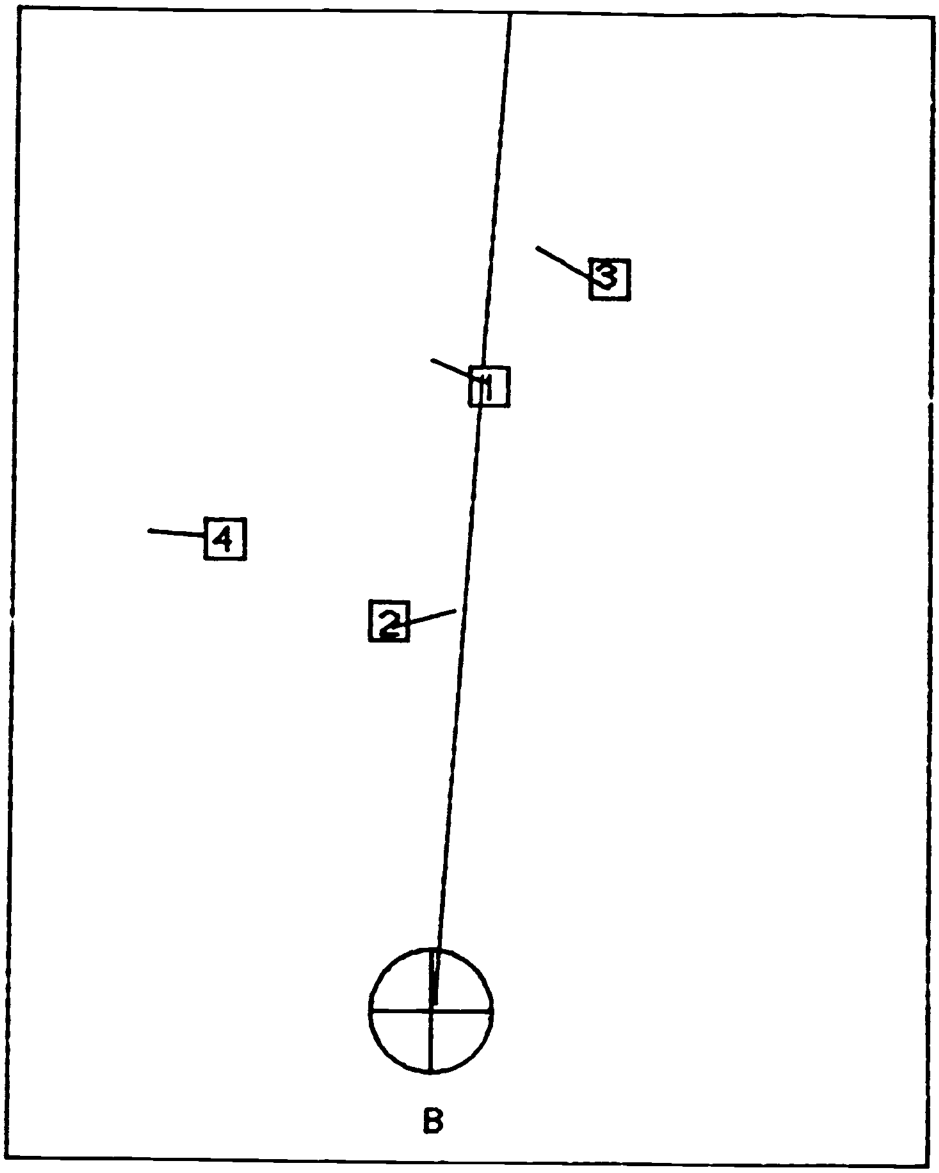


Figure 5-4 An illustration of a scene in which the strategies may cause low accuracy.

RESPONSE TIME

The results revealed there was no statistical significance for either the main effect or any interactions. In the simulated MTII task, there was no time limit for the subjects to respond. However, the situation is different in the real world such as in a threat condition where the human operator is required to produce a fast response. One consideration about designing this experiment was to select different time limits for subjects to respond in the experiment. For example, an association must be made in 3 , 5, or 10 seconds by the subject. Under such conditions, the factor of data rate would not be as consistent as it would be among the subjects, for the subjects might have missed some responses under a short response duration. It is also very difficult to simulate the task in the real world situation and produce crises to stress the subjects in the laboratory.

For the display types, the mean response time using the Polygon display was numerically greater than the Line type display. Since the number of visual elements used in the Polygon type display is more than the number in the Line type display, it is natural to expect such a result.

It would be expected that the performance would be inferior to the present results if time limits were set for the subjects to respond given the current experimental environment.

CHAPTER 6

CONCLUSIONS

The objective of this study was to examine the effect of uncertainty representation on information integration in a simulated ocean surveillance task. This objective was accomplished using an experimental task as data source for statical analysis. The conclusions are naturally limited to the design of this experiment.

The statistical analysis showed that there was no evidence to reject the null hypothesis in favor of the alternative hypothesis; i.e., we failed to reject the hypothesis that display type (Polygon or Line) does not affect operator performance.

This is a negative result with respect to what was expected before the experiment was conducted, because, intuitively, it seemed that explicitly showing uncertainty in the representation could produce better performance. One of the possible reasons that the experiment did not produce its expected results is that most subjects in this experiment failed to understand the true meaning behind the abstract symbols in both Polygon and Line displays. This observation was obtained from the post-experiment questionnaries, which indicates that lack of experience was likely an important factor in the experiment.

Also, although the experimental task was much simplified as compared to a real ocean surveillance task, it was still difficult for the subjects to perform. In the design of such an experimental task, it might be worthwhile to investigate some tasks simpler than the present one. For instance, the example illustrated in Chapter 2 contains many interesting representations for the information association task. Future work might be done on comparing some alternate displays instead of the present ones. It is also recommended that in future studies subjects take a longer training course instead of having only a 10-minute trial before performing the task.

The idea behind an association task should be introduced to the subjects clearly. And the training course should emphasize the association information on the HP-86, since most subjects did not pay attention to their previous associations. In order to increase the subjects' motivation, it might be helpful to pay the subjects instead of using volunteers.

Even though the present study did not produce the expected results, it does not mean that the study is without meaning. In the first place, it might actually be the case that there is no difference in human performance between the two types of displays. If this would be the situation, then it makes no sense to further develop polygon-type displays. However, there are clearly points where improvements could be made. On reflection on the study, we need to think about the population of experimental units where the sampling units come from with respect to the experience and training of subjects, the operations in the experimental task, and the treatments in the experiment in order to obtain positive results.

In summary, even though the study reached a negative conclusion, the work done for this thesis still contributes to the study of the MTII problem. It provides a basic model to approach the MTII problem for those who are interested in performing further research in the area. Furthermore, the work done so far has the flexibility to be modified. The computer programs are easy to understand, and the ideas presented in this thesis are clear enough for further research. We do feel that further studies are necessary for the MTII problem, especially in regard to human/machine interfaces.

BIBLIOGRAPHY

Anderson, J. R. Cognitive Psychology and Its Implications, W. H. Freemane & Company, 1980.

Cox, D. R. The Analysis of Binary Data, Chapman and Hall, London, 1980.

Funk, K., Personal interview, 1986.

Friedman, N. "C^3 War At Sea", U.S. Naval Institute Proceedings, Naval Review, 1977, 124-141.

Greitzer, F. L., Kelly, R. T., and Hershman, R. L. "Human Memory Limitations in Multi-Object Tracking", Navy Personnel Research and Development Center, San Diego, Ca,92152, 1982.

Kantowitz, B. H. and Sorkin, R. D. Human Factors: Understanding People-System Relationships, John Wiley & Sons, 1983.

Kullback, J. H. and Owens, M. E. B. "Multisensor Correlation for Ocean Surveillance: Problems and Limitations", Naval Research Laboratory (Washington D. C.), 1972.

Law, A. M. and Kelton, W. D. Simulation Modeling and Analysis, McGraw-Hill Book Company, 1982.

Lawson, J. S., "The Art and Science of Military Decision Making", Military Operations Research Society, 49th Symposium, 1982.

Miller, G. A., "The Magic Number Seven, Plus or Minus Two", Psychology Review, 1956, 63, 81-97.

Miller, I. and Freund J. E., Probability and Statistics for Engineering, 2ed, Prentice Hall, 1977.

Mills, R. G., "Performance Prediction in a Single Operator Simulated Surveillance System", Human Factors, August, 1973, 337-348.

Novak Jr, G. S. and Walker, D. E., "Multisource Tactical Information Integration", Final Technical Report, Naval Ocean Systems Center, 1978.

Simth, P. L., "Computer-Aided Ship Tracking", IEEE Trans on Systems, Man, and Cybernetic, vol, SMC-9, No. 1, January 1979.

Steel, R. G. D. and Torrie, J. H., *Principles And Procedures of Statistics*, a biometrical approach, 2ed, McGraw-Hill Book Company, 1980.

Warship Commander, published by Enola Games, P.O. Box 1900, Brooklyn, N.Y. 11201, 1979.

Yates, F., "The Principles of Orthoganality and Confounding in Replicated Experiments", *Journal of Agri. Science*, 23, 1933, 108-145.

APPENDIX

(This page is left blank intentionally.)

APPENDIX A

Written materials

Instruction to Subjects

A Simplified Ocean Surveillance Task

Polygon Type Display

Instruction to Subjects

You will be participating in a simplified simulated ocean surveillance task. Four basic components are in this task, i.e., platforms (ships), ownship, sensor reports, and ellipses. Imagine that you are in the ownship monitoring an information integration screen.

A platform is a ship, and can emit radar emissions according to its commander. Assume a platform can emit one type of radar at a time. The type of radar a platform can emit depends upon the radar systems in it. For example, platform "TROUT" can never emit radar D if it has only radars B, H, and I on board. In this experiment, a platform is represented by a square and a bar which shows its direction of movement (Figure 1). Each platform has a unique number.

Figure 1 Representations of platforms. Platforms 3 and 2 are moving toward east; platform 1 is moving toward sourthwest.

Your ownship is defined as the shaped in Figure 2. The center of the crosshair is the position of your ship on the sea.

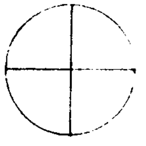


Figure 2 Representation of ownship.

In a military activity, own force always wants to obtain as much information as possible about the current situation. Suppose that your sensors on board can detect the type of radar which is emitted by one of the platforms and report to you. This is called a "sensor report". The sensor report in this task will provide not only estimated bearing and range information about the source of the emission but the type of radar. One important point for you to keep in mind is that the sensor report covers the platform which emitted the detected radar emission most of the time. The coverage of a sensor report varies from time to time; but the angle which represents the range information will be within 0° to 80° limit. Therefore, there exists some biased sensor reports in this experiment. See Figure 3 for shape of a sensor report (on the next page).

An ellipse represents the uncertainty area that a platform can be in. The geometry of the ellipse takes the direction of movement and speed of the platform into consideration, as shown in Figure 4, and can be enlarged auromatically by your system. The meaning of enlargement is to increase the degree of uncertainty about the position of that platform between each update of sensor report.

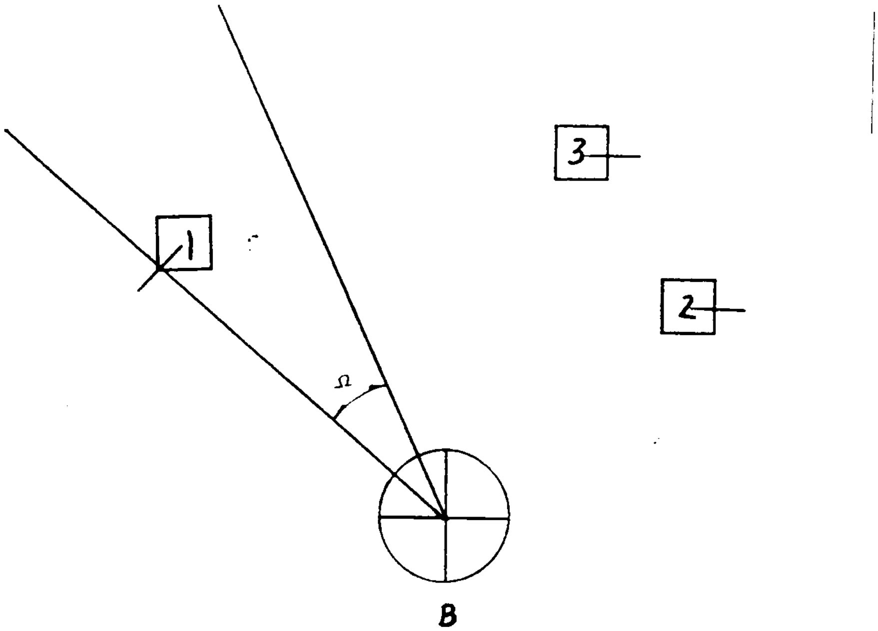


Figure 3 Representation of a sensor report. Note that the angle Ω ranges from 0° to 80°.

The enlargement rate of an ellipse is proportional to the speed of the platform and time between update of sensor reports.

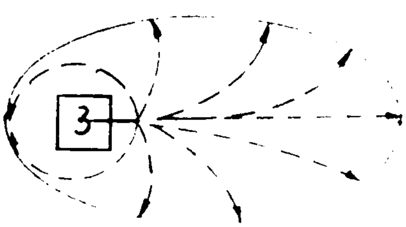


Figure 4 Representation of uncertainty area about platform 3 in a fixed amount of time.

Your task in this experiment is to associate each sensor report with one of the platforms, when the sensor appears on the screen. In other words, the stimulus is the sensor report. Your response is the identity of the platform which you think is the most likely source of the emission.

When a sensor report appears on the screen, you respond by hitting one of the numerical key which designates the identity of that platform once. For example, if the screen is as shown in Figure 5, then you need only hit the numerical key "1" (which I think is the most likely source of radar emission B) on the keyboard once.

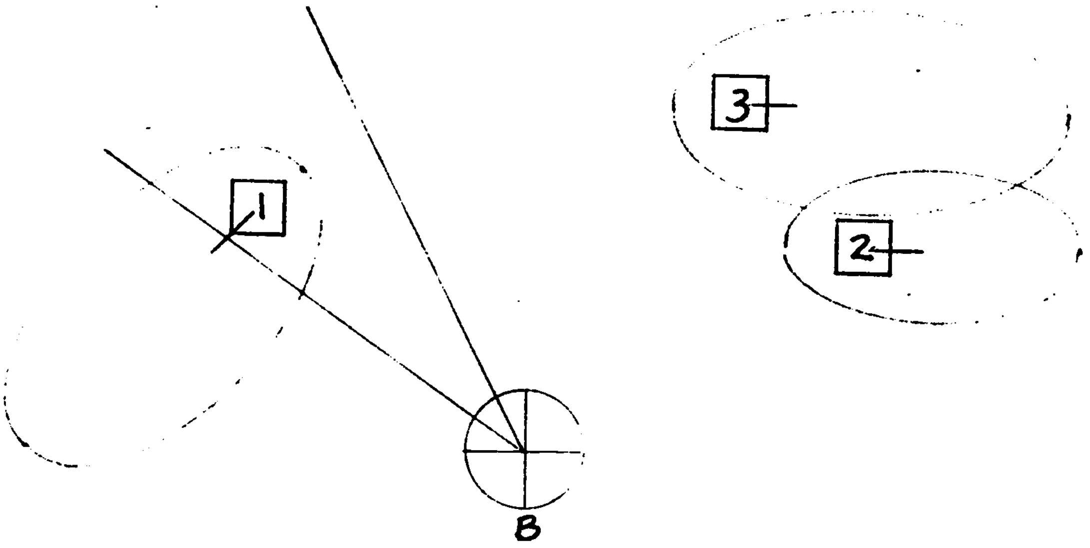


Figure 5 Full representation of the display when a response is needed to be made.

The associations you've made will be displayed on a seperate monitor. Reference to that information might help you in your decision.

start performing it.

Now begin to study the data sheet and tell the experimenter when you are ready to take the test.

STOP ..... Until you have finished the test.

You will start by practicing 10 trials to warm up. These results will not be recorded as your performance. Before you start the task, I want to remind you that your primary objective should be to choose the most likely platform. There is no time limit for you to respond; but the sooner the better.

Feedback of your association will be given to you in the 10 trials. You have enough time to make your decision, so think carefully before you decide. There will be 4 platforms for you to practice in the trials. The number of platforms remains constant throughout your experiment.

You can see your ownship on the screen now. When there is a sensor report, the computer will beep once. When you hit a corresponding key, the computer will beep to tell you that it has received your responce.

Now hit the "GO ON" key and wait for the sensor report. Don't respond; go to next page.

Answer the question:

What radar emission is reported by this sensor report?

Please put your answer on the other sheet of paper.

Now enter your decision by hitting a corresponding key and see the other monitor for your feedback. Go to next page.

Answer of association 1: platform 4

Now hit the "GO ON" key for the second trial. Answer the following question before you respond.

What radar emission is reported by this sensor report?

Please put your answer on the other sheet of paper.

Now enter your decision by hitting a corresponding key and see the other monitor for your information of association.

GO TO NEXT PAGE.....

Answer of association 2: platform 4

Now hit the "GO ON" key for the next trial. Answer the following question before you respond.

What radar emission is reported by this sensor report?

Please put your answer on the other sheet of paper.

Now enter your decision by hitting a corresponding key and see the other monitor for your information of association.

GO TO NEXT PAGE.....

Answer of association 3: platform 1

Now hit the "GO ON" key for the next trial. Answer the following question before you respond.

What radar emission is reported by this sensor report?

Please put your answer on the other sheet of paper.

Now enter your decision by hitting a corresponding key and see the other monitor for your information of association.

GO TO NEXT PAGE.....

Answer of association 4: platform 1

Now hit the "GO ON" key for the next trial. Answer the following question before you respond.

What radar emission is reported by this sensor report?

Please put your answer on the other sheet of paper.

Now enter your decision by hitting a corresponding key and see the other monitor for your information of association.

GO TO NEXT PAGE.....

Answer of association 6: platform 4

Now hit the "GO ON" key for the next trial. Answer the following question before you respond.

What radar emission is reported by this sensor report?

Please put your answer on the other sheet of paper.

Now enter your decision by hitting a corresponding key and see the other monitor for your information of association.

GO TO NEXT PAGE.....

Answer of association 5: platform 1

Now hit the "GO ON" key for the next trial. Answer the following question before you respond.

What radar emission is reported by this sensor report?

Please put your answer on the other sheet of paper.

Now enter your decision by hitting a corresponding key and see the other monitor for your information of association.

GO TO NEXT PAGE.....

Answer of association 7: platform 1

Now hit the "GO ON" key for the next trial. Answer the following question before you respond.

What radar emission is reported by this sensor report?

Please put your answer on the other sheet of paper.

Now enter your decision by hitting a corresponding key and see the other monitor for your information of association.

GO TO NEXT PAGE.....

Answer of association 8: platform 1

Now hit the "GO ON" key for the next trial. Answer the following question before you respond.

What radar emission is reported by this sensor report?

Please put your answer on the other sheet of paper.

Now enter your decision by hitting a corresponding key and see the other monitor for your information of association.

GO TO NEXT PAGE.....

Answer of association 9: platform 3

Now hit the "GO ON" key for the next trial. Answer the following question before you respond.

What radar emission is reported by this sensor report?

Please put your answer on the other sheet of paper.

Now enter your decision by hitting a corresponding key and see the other monitor for your information of association.

GO TO NEXT PAGE.....

Answer of association 10: platform 2

GO TO NEXT PAGE.....

Now the experimenter will set up the task for you to perform. A serial of associations without feedback will be presented to you. You don't need to hit the "GO ON" key anymore; just enter your response.

When you are ready, hit the "GO ON" key once to begin the task.

A Simplified Ocean Surveillance Task

Line Type Display

Instruction to Subjects

You will be participating in a simplified simulated ocean surveillance task. Three basic components are in this task, i.e., platforms (ships), ownship, and sensor report. Imagine that you are in the ownship monitoring an information integration screen.

A platform is a ship, and can emit radar emissions according to its commander. Assume a platform can emit one type of radar at a time. The type of radar a platform can emit depends upon the radar systems in it. For example, platform "TROUT" can never emit radar D if it has only radars B, H, and I on board. In this experiment, a platform is represented by a square and a bar which shows its direction of movement (Figure 1). Each platform has a unique number.

Figure 1 Representations of platforms. Platforms 3 and 2 are moving toward east; platform 1 is moving toward sourthwest.

Your ownship is defined as the shape in Figure 2. The center of the crosshair is the position of your ship on the sea.

In a military activity, own force always wants to obtain as

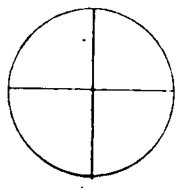


Figure 2 Representations of ownship.

much information as possible about the current situation. Suppose that your sensors on board can detect the type of radar which is emitted by one of the platforms and report to you. This is called a "sensor report". The sensor report in this task is a line, which provides <u>bearing information</u> and the <u>type of radar</u> from its source of emission.

The bearing information in this task will contain a bias <u>upto</u> <u>±40°</u>. For example, if the screen is shown as in Figure 3, then radar B is possibly emitted by anyone of platform 1, 2, and 3.

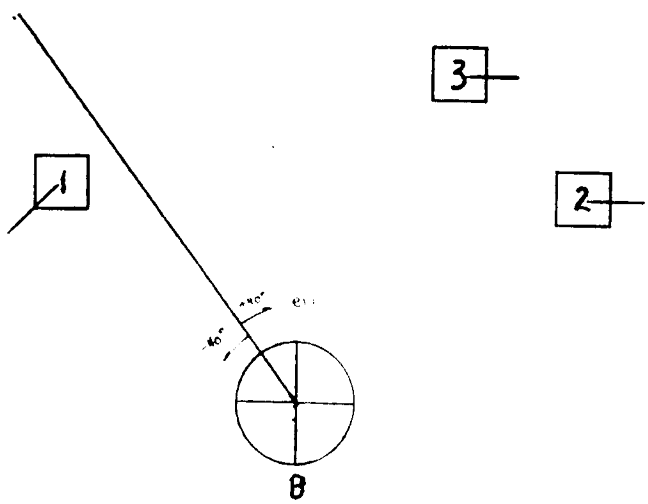


Figure 3 Representation of a sensor report with <u>radar type B emission</u>.

Your task in this experiment is to associate each sensor report with one of the platforms, when the sensor report appears on the screen. In other words, the stimulus is the sensor report. Your response is the identity of the platform which you think is the most likely source of the emission.

When a sensor report appears on the screen, you respond by hitting <u>one of the numerical key</u> which designates the identity of that platform <u>once</u>.

The associations you've made will be displayed on a <u>seperate monitor</u>. Reference to that information might help you in your decision.

The knowledge base about all possible platforms and their radars is shown on the <u>data sheet</u>. Please note once more that a platform can never emit a radar it does not have on board. And a radar emission can never be emitted from a platform which is not on the screen.

Take 3 to 5 minutes to distinguish the characteristics of each platform (e.g., "What type of radars does platform "PIKE" have?", or "Which platform has a type G radar?", etc.).

You need to take a written test in order to start the experiment. Don't hesitate to ask any questions about the experiment before you start performing it.

Now begin to study the data sheet and tell the experimenter when you are ready to take the test.

STOP ..... Until you have finished the test.

Data sheet

Data Sheet

Platform:	SHARK	WHALE	TROUT	BASS	SALMON	CATFISH	PIKE	BLUEGILL
						A	A	A
		B	B	B		B		
	C				C	C		C
	D			D		D		
Radars					E			
	F				F		F	
				G		G		G
	H	H	H			H		
		I	I				I	

Note:

1) Total number of possible radar emission is 9, i.e., from A to I.

2) Total number of possible platforms is 8.

Test sheet

Test Sheet

1. Which platform(s) has the least amount of radars among those 8 platforms ? How many does (do) it (they) have?
Answer:

2. Which platform has the most amount of radars? What are they?
Answer:

3. What are the common radar(s) of platform 2 and 8?
Answer:

4. Can platform 3 give radar emission of type A?
Answer:

5. What is the common radar among platforms 1, 4, 5, and 6?
Answer:

Subject #:_______

Trial sheet

TRIAL SHEET

Circle one radar emission for each association in the trial:

association 1 : A B C D E F G H I

association 2 : A B C D E F G H I

association 3 : A B C D E F G H I

association 4 : A B C D E F G H I

association 5 : A B C D E F G H I

association 6 : A B C D E F G H I

association 7 : A B C D E F G H I

association 8 : A B C D E F G H I

association 9 : A B C D E F G H I

association 10: A B C D E F G H I

Questionnaries

POST EXPERIMENT QUESTIONNAIRE

SUBJECT #:__________ AGE:__________ SEX: F M

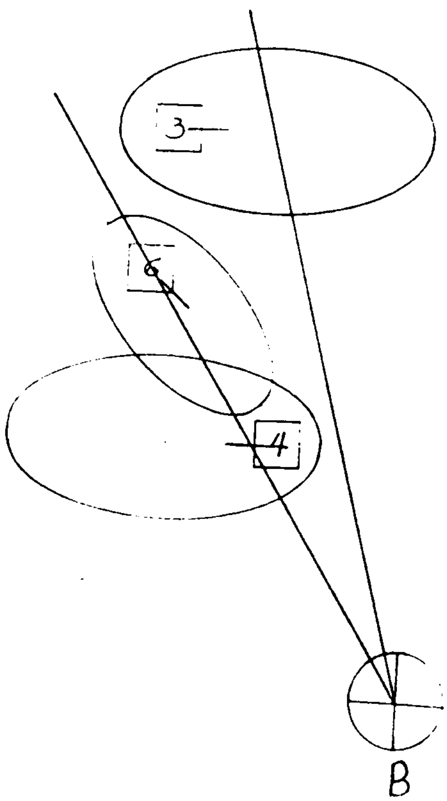
WORK EXPERIENCE ON RADAR SYSTEM: Yes No

Please answer the following questions:

1) Did you use the data sheet as your reference in this experiment? If yes, could you give the names of platform 1, 2, 3, and 4 in this experiment?

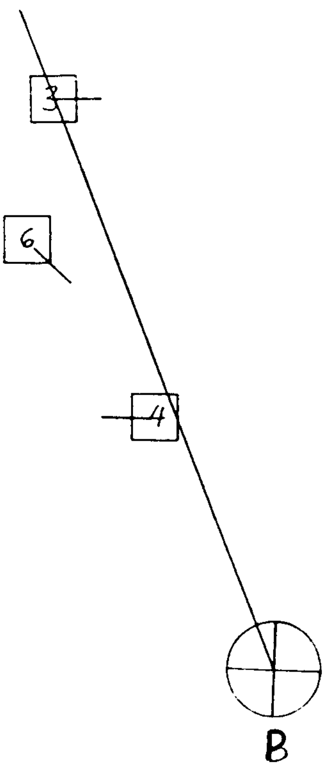
2) Do you think the representation of a sensor report helped you in your decision? How?

3) Whcih platform do you think radar B is from in the following figure? Why?



3) Whcih platform do you think radar B is from in the following figure?

Why?



APPENDIX B

Program listings

SCENARIO

```
      SUBROUTINE DECIDE
C
C
C
      INTEGER ID,IRAD(8),KSHIP(8),KOUNT(8),NEXT1(8),
     1TOTSNR,NEXT2(8),NOFPTS(8),NOFSHP
C
C
C
      LOGICAL TURN(8),DONE(8)
C
C
C
      REAL CTIME(8,10),DIFF,DIST(8),MARVL(8),MTSNR,RADAR(8,10),RDIST(8),
     1RTIME(8),SHIP(8,10,2),SPEED(8),TNOW,X1(8),X2(8),X3(8),Y3(8),
     2Y1(8),Y2(8),XTRUE(8),YTRUE(8),ANGLE(8),DIST12(8),DIST23(8)
C
C
C
      COMMON /MODEL/ CTIME,DIFF,DIST,ID,IRAD,KSHIP,KOUNT,MARVL,MTSNR,
     1NEXT1,NEXT2,NOFPTS,NOFSHP,RADAR,RDIST,RTIME,SHIP,SPEED,X1,X2,Y1,Y2
     2XTRUE,YTRUE,TOTSNR,ANGLE,DIST12,DIST23,TURN,DONE,X3,Y3,ROTDEG,ITEMP
C
C
C
      COMMON /SYSTEM/ LRANK(25),LSIZE(25),MAXATR,NEXT,TIME,TRNSFR(10)
C
C*************************************************
C
      K1 = 0
55    K1 = K1 + 1
      IF ( K1 .GT. NOFSHP ) GOTO 65
      ID = K1
C***
      PRINT 89,ID,RDIST(ID),KSHIP(ID)
89    FORMAT ('IN DECIDE - SHIP ',I3,' REAL DIS: ',F8.3,' KSHIP: ',I3)
         IF ( ( KSHIP(ID) .GE. (NOFPTS(ID)-2) )  .OR. DONE(ID) ) GOTO 55
         IF ( RDIST(ID) .EQ. 0.) THEN
              X1(ID) = SHIP(ID,NEXT1(ID),1)
              Y1(ID) = SHIP(ID,NEXT1(ID),2)
              NEXT1(ID) = NEXT1(ID) + 1
              X2(ID) = SHIP(ID,NEXT1(ID),1)
              Y2(ID) = SHIP(ID,NEXT1(ID),2)
              NEXT2(ID) = NEXT2(ID) + 1
              X3(ID) = SHIP(ID,NEXT2(ID),1)
              Y3(ID) = SHIP(ID,NEXT2(ID),2)
C
C
C
              CALL DATA1
              RTIME(ID) = TIME
              CALL VALUES
              KSHIP(ID) = KSHIP(ID) + 1
         ELSEIF ((RDIST(ID) .LE. DIST12(ID)).AND.(RDIST(ID) .GT. 0.)) THEN
              RTIME(ID) = RTIME(ID) + DIFF
              CALL VALUES
         ENDIF
      GOTO 55
65    CONTINUE
      RETURN
```

```
      END
C
C
C
      SUBROUTINE VALUES
C
C
C
      INTEGER ID,IRAD(8),KSHIP(8),KOUNT(8),NEXT1(8),
     1TOTSNR,NEXT2(8),NOFPTS(8),NOFSHP
C
C
C
      LOGICAL TURN(8),DONE(8)
C
C
C
      REAL CTIME(8,10),DIFF,DIST(8),MARVL(8),MTSNR,RADAR(8,10),RDIST(8),
     1RTIME(8),SHIP(8,10,2),SPEED(8),TNOW,X1(8),X2(8),X3(8),Y3(8),
     2Y1(8),Y2(8),XTRUE(8),YTRUE(8),ANGLE(8),DIST12(8),DIST23(8)
C
C
C
      REAL FACTOR,ROTDEG,XTEMP,YTEMP,A1,A2,A3,A4,A5,A6,A7,A8
      REAL TSIMLT
C
C
C
      COMMON /MODEL/ CTIME,DIFF,DIST,ID,IRAD,KSHIP,KOUNT,MARVL,MTSNR,
     1NEXT1,NEXT2,NOFPTS,NOFSHP,RADAR,RDIST,RTIME,SHIP,SPEED,X1,X2,Y1,Y2
     2XTRUE,YTRUE,TOTSNR,ANGLE,DIST12,DIST23,TURN,DONE,X3,Y3,ROTDEG,ITEM
C
C
C
      COMMON /SYSTEM/ LRANK(25),LSIZE(25),MAXATR,NEXT,TIME,TRNSFR(10)
      COMMON /TSTRT/ TSIMLT
C
C****************************************************
C
      CALL DRADAR
C
C     Calculate the true location first, then compare the real
C  distace and the pre-specified distance to determine if there is
C  a turning action or not.
C
      FACTOR = RTIME(ID)*SPEED(ID)/DIST12(ID)
      XTRUE(ID) = X1(ID) + FACTOR*(X2(ID)-X1(ID))
      YTRUE(ID) = Y1(ID) + FACTOR*(Y2(ID)-Y1(ID))
      RDIST(ID) = SQRT((XTRUE(ID)-X1(ID))**2 + (YTRUE(ID)-Y1(ID))**2)
      IF ( RDIST(ID) .GE. DIST12(ID) ) THEN
C
C     There is a turning action here.
C     The time for this movement will depend on DIFF.
C
          RTIME(ID) = DIFF
C
C     Recalculate the factor for moving in such a short period of
C  time, and calculate the short distance of this moving.
C
          FACTOR = RTIME(ID)*SPEED(ID)/DIST12(ID)
```

```
            XTEMP   = X1(ID) + FACTOR*(X2(ID)-X1(ID))
            YTEMP   = Y1(ID) + FACTOR*(Y2(ID)-Y1(ID))
            RMOVE   = SQRT ((XTEMP-X1(ID))**2 + (YTEMP-Y1(ID))**2)
C
C     Update those point coordinates accordingly.
C
            X1(ID) = X2(ID)
            X2(ID) = X3(ID)
            NEXT2(ID) = NEXT2(ID) + 1
            X3(ID) = SHIP(ID,NEXT2(ID),1)
            Y1(ID) = Y2(ID)
            Y2(ID) = Y3(ID)
            Y3(ID) = SHIP(ID,NEXT2(ID),2)
            EXCESS = RDIST(ID) - DIST12(ID)
            FACTOR = (EXCESS/RMOVE)*RTIME(ID)*SPEED(ID)/DIST23(ID)
C
C     Test if the turning angle is greater than 90 degrees.
C     Rules:     1) Greater than 90: Reduce the speed by 2/3.
C                2) No greater than 90: Reduce speed only by 1/3.
C
           IF ( ANGLE(ID) .GT. 90.) THEN
               FACTOR = FACTOR*2./3.
           ELSE
               FACTOR = FACTOR/3.
           ENDIF
C
C     Testing if this is the last point for this ship.
C
           IF ( (X3(ID) .EQ. -1000.00) .AND. (Y3(ID) .EQ. -1000.00) ) THEN
               DONE(ID) = .TRUE.
           ELSE
               DONE(ID) = .FALSE.
           ENDIF
C
C     Set an indicator for swithing the starting point for the next
C    segment.
C
           IND = 1
C
C     Add one to the ship which has being processed.
C
           KSHIP(ID) = KSHIP(ID) + 1
        ENDIF
C
C     Calculate once more the true location for the situation of
C    turning.
C
        XTRUE(ID) = X1(ID) + FACTOR*(X2(ID)-X1(ID))
        YTRUE(ID) = Y1(ID) + FACTOR*(Y2(ID)-Y1(ID))
        RDIST(ID) = SQRT((XTRUE(ID)-X1(ID))**2 + (YTRUE(ID)-Y1(ID))**2 )
C
C     The following is the degree of rotation for drawing the ellipse.
C
        ROTDEG = ATAN2 ((YTRUE(ID)-Y1(ID)),(XTRUE(ID)-X1(ID)))
        ROTDEG = (ROTDEG/3.1415926)*180.
C
C     Switching the new starting point for the next segment.
C
        IF ( IND .EQ. 1) THEN
            X1(ID) = XTRUE(ID)
```

```
              Y1(ID) = YTRUE(ID)
              IF ( DONE(ID) ) THEN
                  DIST12(ID) = SQRT((X2(ID)-X1(ID))**2 + (Y2(ID)-Y1(ID))**2)
              ELSE
                  CALL DATA1
              ENDIF
              IND = 0
          ENDIF
          A1 = TIME
          A2 = FLOAT(ID)
          A3 = XTRUE(ID)
          A4 = YTRUE(ID)
          A5 = ROTDEG
          A6 = RADAR(ID,KOUNT(ID))
          A7 = SPEED(ID)
          A8 = FLOAT(ITEMP)
          IF ( TIME .GE. TSIMLT) THEN
          WRITE(16,100) A1,A2,A3,A4,A5,A6,A7,A8
100       FORMAT (F9.3,1X,F3.1,1X,6(F8.2,1X))
          ENDIF
C
C
C
          RETURN
          END
C
C
C
          SUBROUTINE DRADAR
C
C
C
          INTEGER ID,IRAD(8),KSHIP(8),KOUNT(8),NEXT1(8),
         1TOTSNR,NEXT2(8),NOFPTS(8),NOFSHP
C
C
C
          LOGICAL TURN(8),DONE(8)
C
C
C
          REAL CTIME(8,10),DIFF,DIST(8),MARVL(8),MTSNR,RADAR(8,10),RDIST(8),
         1RTIME(8),SHIP(8,10,2),SPEED(8),TNOW,X1(8),X2(8),X3(8),Y3(8),
         2Y1(8),Y2(8),XTRUE(8),YTRUE(8),ANGLE(8),DIST12(8),DIST23(8)
C
C
C
          COMMON /MODEL/ CTIME,DIFF,DIST,ID,IRAD,KSHIP,KOUNT,MARVL,MTSNR,
         1NEXT1,NEXT2,NOFPTS,NOFSHP,RADAR,RDIST,RTIME,SHIP,SPEED,X1,X2,Y1,Y2
         2XTRUE,YTRUE,TOTSNR,ANGLE,DIST12,DIST23,TURN,DONE,X3,Y3,ROTDEG,ITEM
P
C
C
C
          COMMON /SYSTEM/ LRANK(25),LSIZE(25),MAXATR,NEXT,TIME,TRNSFR(10)
C
C*****************************************************
C
          DO 90 I1=1,NOFSHP
          DO 80  J=1,IRAD(I1)
C
```

```
          IF (TIME .LT. CTIME(I1,J) ) THEN
              KOUNT(I1) = J
              GOTO 90
          ENDIF
   80     CONTINUE
   90     CONTINUE
C
C
C
          RETURN
          END
C
C
C
C
C
C
          SUBROUTINE DATA1
C
C
C
          INTEGER ID,IRAD(8),KSHIP(8),KOUNT(8),NEXT1(8),
         1TOTSNR,NEXT2(8),NOFPTS(8),NOFSHP
C
C
C
          LOGICAL TURN(8),DONE(8)
C
C
C
          REAL CTIME(8,10),DIFF,DIST(8),MARVL(8),MTSNR,RADAR(8,10),RDIST(8),
         1RTIME(8),SHIP(8,10,2),SPEED(8),TNOW,X1(8),X2(8),X3(8),Y3(8),
         2Y1(8),Y2(8),XTRUE(8),YTRUE(8),ANGLE(8),DIST12(8),DIST23(8)
C
C
          REAL TEMPX1,TEMPX2,TEMPY1,TEMPY2,TEMP
C
C
C
          COMMON /MODEL/ CTIME,DIFF,DIST,ID,IRAD,KSHIP,KOUNT,MARVL,MTSNR,
         1NEXT1,NEXT2,NOFPTS,NOFSHP,RADAR,RDIST,RTIME,SHIP,SPEED,X1,X2,Y1,Y2
,
         2XTRUE,YTRUE,TOTSNR,ANGLE,DIST12,DIST23,TURN,DONE,X3,Y3,ROTDEG,ITEM
P
C
C
C
          COMMON /SYSTEM/ LRANK(25),LSIZE(25),MAXATR,NEXT,TIME,TRNSFR(10)
C
C
C*****************************************************
          DIST12(ID) = SQRT( (X2(ID)-X1(ID))**2 + (Y2(ID)-Y1(ID))**2 )
          DIST23(ID) = SQRT( (X3(ID)-X2(ID))**2 + (Y3(ID)-Y2(ID))**2 )
          PRINT 44,X1(ID),X2(ID),X3(ID),Y1(ID),Y2(ID),Y3(ID),ID
   44     FORMAT ('POSI. X1,X2,X3,Y1,Y2,Y3,ID: ',6F8.2,I5)
          TEMPX1 = X1(ID)-X2(ID)
          TEMPX2 = X3(ID)-X2(ID)
          TEMPY1 = Y1(ID)-Y2(ID)
          TEMPY2 = Y3(ID)-Y2(ID)
          TEMP = TEMPX1*TEMPX2 + TEMPY1*TEMPY2
C
C
```

```
C
      ANGLE(ID) = ACOS(TEMP/(DIST12(ID)*DIST23(ID)))
      ANGLE(ID) = (ANGLE(ID)/3.1415926)*180.
      PRINT 10,ANGLE(ID),ID
10    FORMAT ('SUB DATA1, ANGLE IS: ',F9.3,' FOR SHIP ',I2)
      RETURN
      END
C
C
C
      SUBROUTINE ENTER
C
C
C
       INTEGER ID,IRAD(8),KSHIP(8),KOUNT(8),NEXT1(8),
     1TOTSNR,NEXT2(8),NOFPTS(8),NOFSHP
C
C
C
       INTEGER*4 ISEED
C
C
C
       LOGICAL TURN(8),DONE(8)
C
C
C
       REAL CTIME(8,10),DIFF,DIST(8),MARVL(8),MTSNR,RADAR(8,10),RDIST(8),
     1RTIME(8),SHIP(8,10,2),SPEED(8),TNOW,X1(8),X2(8),X3(8),Y3(8),
     2Y1(8),Y2(8),XTRUE(8),YTRUE(8),ANGLE(8),DIST12(8),DIST23(8)
       REAL TSIMLT
C
C
C
       COMMON /MODEL/ CTIME,DIFF,DIST,ID,IRAD,KSHIP,KOUNT,MARVL,MTSNR,
     1NEXT1,NEXT2,NOFPTS,NOFSHP,RADAR,RDIST,RTIME,SHIP,SPEED,X1,X2,Y1,Y2,
     2XTRUE,YTRUE,TOTSNR,ANGLE,DIST12,DIST23,TURN,DONE,X3,Y3,ROTDEG,ITEMP
       COMMON /TSTRT/ TSIMLT
C
C
C
       COMMON /SYSTEM/ LRANK(25),LSIZE(25),MAXATR,NEXT,TIME,TRNSFR(10)
C
C
C
       COMMON /ABC/ ISEED
C
C*****************************************************
C
       READ (15,10,ERR = 1001) NOFSHP
       READ (15,10,ERR = 1002) (NOFPTS(I),I=1,NOFSHP)
       READ (15,10,ERR = 1003) (IRAD(I),I=1,NOFSHP)
       READ (15,10,ERR = 1004) TOTSNR
       READ (15,20,ERR = 1005) (((SHIP(I,J,K),K=1,2),J=1,NOFPTS(I)),I=1,NOFSHP)
       READ (15,30,ERR = 1006) (SPEED(I),I=1,NOFSHP)
       READ (15,30,ERR = 1007) (MARVL(I),I=1,NOFSHP)
       READ (15,30,ERR = 1008) MTSNR
       READ (15,30,ERR = 1009) ((CTIME(I,J),J=1,IRAD(I)),I=1,NOFSHP)
       READ (15,30,ERR = 1010) ((RADAR(I,J),J=1,IRAD(I)),I=1,NOFSHP)
       READ (15,40,ERR = 1011) ISEED
```

```
      READ (15,55,ERR = 2000) TSIMLT
C
C
C
10    FORMAT (I2)
20    FORMAT (2F8.2)
30    FORMAT (F6.2)
40    FORMAT (I8)
55    FORMAT (F9.3)
      RETURN
C
C
C
1001  PRINT 1101
1101  FORMAT (/////,'WRONG NOFSHP')
      STOP
1002  PRINT 1102,I
1102  FORMAT (/////,'WRONG NOFPTS AT THE ',I2,' VALUE.')
      STOP
1003  PRINT 1103,I
1103  FORMAT (/////,'WRONG IRAD AT THE ',I2,' VALUE.')
      STOP
1004  PRINT 1104
1104  FORMAT (/////,'WRONG TOTSNR.')
      STOP
1005  PRINT 1105
1105  FORMAT (/////,'WRONG SHIP')
      STOP
1006  PRINT 1106,I
1106  FORMAT (/////,'WRONG SPEED FOR THE ',I2,' SHIP.')
      STOP
1007  PRINT 1107,I
1107  FORMAT (/////,'WRONG MEAN TIME BETWEEN EMISSION OF THE ',I2,' SHIP')
      STOP
1008  PRINT 1108
1108  FORMAT (/////,'WRONG TIME INTERVAL BETWEEN UPDATING.')
      STOP
1009  PRINT 1109
1109  FORMAT (/////,'WRONG CTIME')
      STOP
1010  PRINT 1110,J,I
1110  FORMAT (/////,'WRONG RADAR OF THE ',I2,' RADAR OF THE ',I2,' SHIP')
      STOP
1011  PRINT 1111
1111  FORMAT (/////,'SEED VALUE WRONG.')
      STOP
2000  PRINT 2010
2010  FORMAT (/////,'TIME START TO SIMULATE WRONG.')
      RETURN
      END
C
C
C
      SUBROUTINE START
C
C
C
      INTEGER ID,IRAD(8),KSHIP(8),KOUNT(8),NEXT1(8),
     1TOTSNR,NEXT2(8),NOFPTS(8),NOFSHP
C
```

```
C
C
       LOGICAL TURN(8),DONE(8)
C
C
C
       REAL CTIME(8,10),DIFF,DIST(8),MARVL(8),MTSNR,RADAR(8,10),RDIST(8),
      1RTIME(8),SHIP(8,10,2),SPEED(8),TNOW,X1(8),X2(8),X3(8),Y3(8),
      2Y1(8),Y2(8),XTRUE(8),YTRUE(8),ANGLE(8),DIST12(8),DIST23(8)
C
C
C
       COMMON /MODEL/ CTIME,DIFF,DIST,ID,IRAD,KSHIP,KOUNT,MARVL,MTSNR,
      1NEXT1,NEXT2,NOFPTS,NOFSHP,RADAR,RDIST,RTIME,SHIP,SPEED,X1,X2,Y1,Y2,
      2XTRUE,YTRUE,TOTSNR,ANGLE,DIST12,DIST23,TURN,DONE,X3,Y3,ROTDEG,ITEMP
C
C
C
       COMMON /SYSTEM/ LRANK(25),LSIZE(25),MAXATR,NEXT,TIME,TRNSFR(10)
C
C**************************************************
C
       DO 100 ID=1,NOFSHP
       NEXT1(ID) = 1
       NEXT2(ID) = 2
       TURN(ID) = .FALSE.
       DONE(ID) = .FALSE.
       KOUNT(ID)  = 1
       KSHIP(ID)  = 0
       RDIST(ID) = 0.
100    CONTINUE
C
C
C
       RETURN
       END
C
C
C
       PROGRAM CONTROL
C
C
C
       INTEGER ID,IRAD(8),KSHIP(8),KOUNT(8),NEXT1(8),
      1TOTSNR,NEXT2(8),NOFPTS(8),NOFSHP
       INTEGER ITEMP,NUMSNR
C
C
C
       INTEGER*4 ISEED
C
C
C
       LOGICAL TURN(8),DONE(8)
C
C
C
       REAL CTIME(8,10),DIFF,DIST(8),MARVL(8),MTSNR,RADAR(8,10),RDIST(8),
      1RTIME(8),SHIP(8,10,2),SPEED(8),TNOW,X1(8),X2(8),X3(8),Y3(8),
      2Y1(8),Y2(8),XTRUE(8),YTRUE(8),ANGLE(8),DIST12(8),DIST23(8)
```

```
      REAL TSIMLT,RADNT
C
C
C
      COMMON /MODEL/ CTIME,DIFF,DIST,ID,IRAD,KSHIP,KOUNT,MARVL,MTSNR,
     1NEXT1,NEXT2,NOFPTS,NOFSHP,RADAR,RDIST,RTIME,SHIP,SPEED,X1,X2,Y1,Y2,
     2XTRUE,YTRUE,TOTSNR,ANGLE,DIST12,DIST23,TURN,DONE,X3,Y3,ROTDEG,ITEMP
C
C
C
      COMMON /SYSTEM/ LRANK(25),LSIZE(25),MAXATR,NEXT,TIME,TRNSFR(10)
C
C
C
      COMMON /ABC/ ISEED
      COMMON /TSTRT/ TSIMLT
C
C*****************************************************
C
      OPEN (UNIT=15)
      OPEN (UNIT=16)
      OPEN (UNIT=17)
C
C
C
C
      NUMSNR = 0
      WRITE (16,5)
5     FORMAT ('   TIME   SHIP  XTRUE    YTRUE     DEG.      RADAR     SPEED
EMITTER')
C
C
      CALL INITLK
      MAXATR = 3
      TNOW = 0.
C
C
C
      CALL ENTER
      CALL START
C
C        THERE IS ONE TYPE OF EVENT IN THIS SIMULATION: RADAR EMISSION
C     EVENT. THE STATISCAL RESULTS ABOUT SUCH EVENT ARE CALCULATED
C     THROUGH THE SAMPST UTILITY IN SIMLIB.
C
C        THE VALUE OF EACH PLATFORM IS ITS ID NUMBER.
C
      DO 100 ID=1,NOFSHP
      TRNSFR(1) = EXPON( MARVL (ID) )
      TRNSFR(2) = 1
      TRNSFR(3) = FLOAT(ID)
      TRNSFR(4) = TRNSFR(1)
      CALL FILE(3,25)
100   CONTINUE
      TRNSFR(1) = MTSNR
      TRNSFR(2) = 1
      TRNSFR(3) = 11.
      TRNSFR(4) = TRNSFR(1)
      CALL FILE(3,25)
C
C
```

```
C
110   CALL TIMING
      ID = IFIX( TRNSFR(3) )
C
C     COLLECT RELEVANT INFORMATION ABOUT RADAR EMISSIONS.
C
      CALL SAMPST (TRNSFR(4),ID)
      ITEMP = ID
      DIFF = TIME - TNOW
      TNOW = TIME
      IF ( ID .EQ. 11 ) THEN
      TRNSFR(1) = TIME + MTSNR
      TRNSFR(2) = 1.
      TRNSFR(3) = 11.
      TRNSFR(4) = MTSNR
      ELSE
      RADNT = EXPON( MARVL(ID) )
      TRNSFR(1) = TIME + RADNT
      TRNSFR(2) = 1.
      TRNSFR(3) = FLOAT(ID)
      TRNSFR(4) = RADNT
      ENDIF
      CALL FILE(3,25)
C
C
C
      CALL DECIDE
      IF ( (ITEMP .EQ. 11) .OR. DONE(ITEMP) ) GOTO 134
          IF (TIME .GE. TSIMLT) THEN
          NUMSNR = NUMSNR + 1
          RICK = -999.
          WRITE (16,133) RICK
133       FORMAT (F9.3)
          ENDIF
134   IF ( NUMSNR .LE. TOTSNR ) GOTO 110
C
C     SUMMARY THE REPORT.
C
      CALL REPORT
C
C
C
      CLOSE (15)
      CLOSE (16)
      CLOSE (17)
      STOP
      END
C
C
C
      FUNCTION EXPON(RMEAN)
      INTEGER*4 ISEED
      REAL RMEAN,U
      COMMON /ABC/ ISEED
      U = RAND(ISEED)
      EXPON = -RMEAN*ALOG(U)
      RETURN
      END
C
C
```

```
C
       FUNCTION RAND(IX)
       REAL*8 RAND
       INTEGER *4 A,P,IX,B15,B16,XHI,XALO,LEFTLO,FHI,K
       DATA A/16807/,B15/32768/,B16/65536/,P/2147483647/
       XHI=IX/B16
       XALO=(IX-XHI*B16)*A
       LEFTLO=XALO/B16
       FHI=XHI*A+LEFTLO
       K=FHI/B15
       IX=(((XALO-LEFTLO*B16)-P)+(FHI-K*B15)*B16)+K
       IF(IX.LT.0)IX=IX+P
       RAND = FLOAT(IX) * 4.656612875E-10
       RETURN
       END
C
C
C
      SUBROUTINE REPORT
      INTEGER ID,IRAD(8),KSHIP(8),KOUNT(8),NEXT1(8),
     1TOTSNR,NEXT2(8),NOFPTS(8),NOFSHP
      INTEGER ITEMP,NUMSNR
C
C
C
      LOGICAL TURN(8),DONE(8)
C
C
C
      REAL CTIME(8,10),DIFF,DIST(8),MARVL(8),MTSNR,RADAR(8,10),RDIST(8),
     1RTIME(8),SHIP(8,10,2),SPEED(8),TNOW,X1(8),X2(8),X3(8),Y3(8),
     2Y1(8),Y2(8),XTRUE(8),YTRUE(8),ANGLE(8),DIST12(8),DIST23(8)
      REAL TSIMLT
C
C
C
      COMMON /MODEL/ CTIME,DIFF,DIST,ID,IRAD,KSHIP,KOUNT,MARVL,MTSNR,
     1NEXT1,NEXT2,NOFPTS,NOFSHP,RADAR,RDIST,RTIME,SHIP,SPEED,X1,X2,Y1,Y2
,
     2XTRUE,YTRUE,TOTSNR,ANGLE,DIST12,DIST23,TURN,DONE,X3,Y3,ROTDEG,ITEM
P
C
C
C
      COMMON /SYSTEM/ LRANK(25),LSIZE(25),MAXATR,NEXT,TIME,TRNSFR(10)
C
C     HEADING FOR UNIT 17.
C
      WRITE (17,50)
50    FORMAT ('********** TIME INFORMATION FOR RADAR EMISSIONS *********
*')
      WRITE (17,51)
51    FORMAT (' PLATFORM    MTBE     # OF OBSERVATION   MINIMUM   MAXIMUM
')
      WRITE (17,52)
52    FORMAT ('
')
C
C
C
      DO 103 I1=1,NOFSHP
         CALL SAMPST (0.,-I1)
         WRITE (17,100) I1,TRNSFR(1),TRNSFR(2),TRNSFR(3),TRNSFR(4)
100      FORMAT (4X,I2,2X,F9.3,9X,I3,8X,F9.3,1X,F9.3)
103   CONTINUE
      RETURN
      END
```

ESTIMATE

```
C
C         This program is used to simulate the estimated locations
C     of ships and estimated sensor report shapes.
C
      PROGRAM ESTIMATE
      REAL TBASE(300,8),ROTDEG,TRNSFR(10)
      REAL XTRUE,YTRUE,XT,YT,XEST,YEST,X1,X2,X3,X4,Y1,Y2,XS1,XS2,XS3,
     1XS4,YS1,YS2
      INTEGER*4 ISEED
      CHARACTER*30 FLINE
      DATA TBASE /2400*0./,TRNSFR/10*0./
      OPEN (UNIT = 13)
      OPEN (UNIT = 15)
      OPEN (UNIT = 16)
      OPEN (UNIT = 22)
      OPEN (UNIT = 30)
C
C         Read the input values from a data file from unit13 named
C     COMING.DAT, which includes: r.n. seed value, # of records to
C     be read in, sample size of estimation, and t-value for such
C     df in a 95% confidence interval.
C
      READ (22,34) ISEED,NODATA,NSMPL,ACURCY,T95,T90
34    FORMAT (I10,5X,I5,5X,I5,5X,F6.2,5X,2(F6.3,5X))
C
C     FORMAT THE CORRESPONDING OUTPUT FILES:
C
C     UNIT13 = CONFIDENCE INTERVALS FOR ESTIMATIONS OF SENSOR LOCATIONS.
C     UNIT30 = CONFIDENCE INTERVALS FOR ESTIMATED PLATFORMS LOCATIONS.
C
      CALL HEAD(NSMPL,ACURCY,T95,T90)
C
C     PROCESS THE DATA.
C
      READ (15,108) FLINE
108   FORMAT (A30)
      READ (15,98) ((TBASE(I,J),J=1,8),I=1,NODATA)
98    FORMAT (F9.3,1X,F3.1,1X,F8.2,1X,F8.2,1X,F8.2,1X,F8.2,1X,F8.2,1X,F8
.2,1X)
C
C
C
      WRITE (16,499) NODATA
499   FORMAT (I5)
C
C*****************************************************************
C
150   DO 200 I=1,NODATA
       IF (TBASE(I,1) .EQ. 0.) GOTO 500
       IF ((TBASE(I,1).EQ.-999.).AND.(PREVS.GE.TBASE(I-1,1))) THEN
           WRITE(16,33) (TRNSFR(K1),K1=1,10)
33         FORMAT (F9.3,1X,F3.1,1X,F8.2,1X,F8.2,1X,F8.2,1X,F8.2,1X,F8.2,1
X,
      1F8.2,1X,F8.2,1X,F3.1)
           GOTO 199
       ENDIF
       XTRUE = TBASE(I,3)
       YTRUE = TBASE(I,4)
       CALL EST(XTRUE,YTRUE,XEST,YEST,ISEED,NSMPL,T95,TBASE(I,2))
       IF (TBASE(I,2) .EQ. TBASE(I,8)) THEN
           ROTDEG = ATAN2 ((YTRUE-13.),(XTRUE-65.))
           ROTDEG = (ROTDEG/3.1415926)*180.
```

```
            CALL ESTSNR (XEST,YEST,X1,X2,X3,X4,Y1,Y2,ISEED,NSMPL,ACURCY,
     *T95,T90,TBASE(I,2))
            TRNSFR(1) = TBASE(I,1)
            TRNSFR(2) = TBASE(I,2)
            TRNSFR(3) = X1
            TRNSFR(4) = X2
            TRNSFR(5) = X3
            TRNSFR(6) = X4
            TRNSFR(7) = Y1
            TRNSFR(8) = Y2
            TRNSFR(9) = ROTDEG
            TRNSFR(10) = TBASE(I,6)
            PREVS = TRNSFR(1)
      ENDIF
      TBASE(I,3) = XEST
      TBASE(I,4) = YEST
      IF ( TBASE(I,1) .NE. -999. ) THEN
      WRITE (16,110) (TBASE(I,L),L=1,8)
110   FORMAT (F9.3,1X,F3.1,1X,6(F8.2,1X))
      ENDIF
199   CONTINUE
200   CONTINUE
500   CONTINUE
      CLOSE(13)
      CLOSE(15)
      CLOSE(16)
      CLOSE(22)
      CLOSE(30)
      STOP
      END
C
C
C
C
C
C
        FUNCTION RAND(IX)
        REAL*8 RAND
        INTEGER *4 A,P,IX,B15,B16,XHI,XALO,LEFTLO,FHI,K
        DATA A/16807/,B15/32768/,B16/65536/,P/2147483647/
        XHI=IX/B16
        XALO=(IX-XHI*B16)*A
        LEFTLO=XALO/B16
        FHI=XHI*A+LEFTLO
        K=FHI/B15
        IX=(((XALO-LEFTLO*B16)-P)+(FHI-K*B15)*B16)+K
        IF(IX.LT.0)IX=IX+P
        RAND = FLOAT(IX) * 4.656612875E-10
        RETURN
        END
C
C
C
C
C
C
C
       SUBROUTINE EST(XA,YA,XMEAN,YMEAN,IVAL,ISMPL,T95,SHIPID)
       REAL X(50),Y(50),XMEAN,YMEAN
       REAL A,B,C,S,XE,YE,XA,YA,XSUM,YSUM,DENUM,E,XUP,XLOW,YUP,YLOW
```

```
      INTEGER*4 IVAL
      DATA X/50*0./,Y/50*0./
      XMEAN = 0.
      YMEAN = 0.
      XSUM = 0.
      YSUM = 0.
      DO 30 I=1,ISMPL
10    A = RAND(IVAL)
      A = 2.*A - 1.
      B = RAND(IVAL)
      B = 2.*B - 1.
C
C
C
      IF ( (A*A+B*B) .GT. 1.) GOTO 10
      DENUM = A*A + B*B
      C = (A*A-B*B)/DENUM
      S = (2.*A*B)/DENUM
      E = SQRT(-ALOG(RAND(IVAL)))
      XE = E*C*SQRT(2.)
      YE = E*S*SQRT(2.)
      X(I) = XE + XA
      Y(I) = YE + YA
30    CONTINUE
C
C
C
      DO 40 I=1,ISMPL
         XMEAN = X(I) + XMEAN
         YMEAN = Y(I) + YMEAN
40    CONTINUE
      XMEAN = XMEAN/ISMPL
      YMEAN = YMEAN/ISMPL
      DO 50 I=1,ISMPL
         XSUM = XSUM + (X(I)-XMEAN)*(X(I)-XMEAN)
         YSUM = YSUM + (Y(I)-YMEAN)*(Y(I)-YMEAN)
50    CONTINUE
      XSUM = SQRT( (XSUM/FLOAT(ISMPL-1) )/ISMPL)
      YSUM = SQRT( (YSUM/FLOAT(ISMPL-1) )/ISMPL)
C
C
C
      XUP = XMEAN+ T95 * XSUM
      XLOW = XMEAN- T95 * XSUM
      YUP = YMEAN+ T95 * YSUM
      YLOW = YMEAN- T95 * YSUM
C
C     OUTPUT THE ACCORDING DATA FOR PLATFORM INFROMATION (UNIT30).
C
      ISHIPD = IFIX(SHIPID)
      INS1 = 0
      INS2 = 0
      IF ( (XA .GE. XLOW) .AND. (XA .LE. XUP) ) INS1 = 1
      IF ( (YA .GE. YLOW) .AND. (YA .LE. YUP) ) INS2 = 1
      WRITE (30,91) ISHIPD,XA,YA,XMEAN,YMEAN,XLOW,XUP,INS1,YLOW,YUP,INS2
91    FORMAT (4X,I2,4X,'( ',F6.2,' , ',F6.2,' )  ( ',F6.2,' , ',F6.2,' )
( ',
     $F6.2,'  , ',F6.2,' ) ',3X,I1,3X,'( ',F6.2,' , ',F6.2,' ) ',3X,I1)
      RETURN
      END
C
```

```
C
C
C
C
C
C
      SUBROUTINE ESTSNR (XT,YT,XS1,XS2,XS3,XS4,YS1,YS2,IXAL,ISMPL,PCISN,
     *T95,T90,SHIPID)
      REAL X(50),Y(50),XMEAN,YMEAN,XEST1,YEST1,XS1,XS2,XS3,XS4,YS1,YS2
      REAL XSUM,YSUM,A,B,C,S,E,XE,YE,XA,YA
      INTEGER*4 IXAL
      DATA X/50*0./,Y/50*0./
      XMEAN = 0.
      YMEAN = 0.
      XSUM = 0.
      YSUM = 0.
C
C     SET AN INDECATOR FOR GENERATING A DIRECTION OF NOISING.
C
            IRV = 1
            IF ( RAND(IXAL) .GE. 0.5 ) IRV = -1
            IF ( RAND(IXAL) .GE. (1.-PCISN) ) IRV = 0
C
C
C
      DIS = SQRT ((XT-65.)**2+(YT-13.)**2)
      FATR = DIS/100.
      IF (DIS .LT. 100.) STD = 21.
      IF (DIS .LT.  90.) STD = 18.
      IF (DIS .LT.  80.) STD = 15.
      IF (DIS .LT.  70.) STD = 12.
      IF (DIS .LT.  60.) STD = 9.
      IF (DIS .LT.  20.) STD = 1.5
C
C     Senser accuracy is accomplished by the following calculations.
C
      NOFOUT = ISMPL - IFIX(ISMPL*PCISN)
C
C
C
      DO 30 I=1,ISMPL-NOFOUT
 10   A = RAND(IXAL)
      A = 2.*A-1.
      B = RAND(IXAL)
      B = 2.*B-1.
      IF ( (A*A+B*B) .GT. 1.) GOTO 10
      DENUM = A*A + B*B
      C = (A*A-B*B)/DENUM
      S = 2.*A*B/DENUM
      E = SQRT (-ALOG(RAND(IXAL)))
      XEST1 = E*C*SQRT(2.)
      YEST1 = E*S*SQRT(2.)
      X(I) = XEST1*STD+XT
      Y(I) = YEST1*STD+YT
 30   CONTINUE
      DO 35 I=(ISMPL-NOFOUT+1),ISMPL
 36   A = RAND(IXAL)
      A = 2.*A-1.
      B = RAND(IXAL)
      B = 2.*B-1.
```

```
      IF ( (A*A+B*B) .GT. 1) GOTO 36
      DENUM = A*A + B*B
      C = (A*A-B*B)/DENUM
      S = 2.*A*B/DENUM
      E = SQRT (-ALOG(RAND(IXAL)))
      XEST1 = E*C*SQRT(2.)*STD+XT
      YEST1 = E*S*SQRT(2.)*STD+XT
C
C     PUT SOME RANDOM NOISE TO THE POSITION.
C
      X(I) = XEST1 + XEST1*FATR*RAND(IXAL)*IRV
      Y(I) = YEST1 + YEST1*FATR*RAND(IXAL)*IRV
 35   CONTINUE
      DO 40 I=1,ISMPL
      XMEAN = X(I) + XMEAN
      YMEAN = Y(I) + YMEAN
 40   CONTINUE
      XMEAN = XMEAN/ISMPL
      YMEAN = YMEAN/ISMPL
      DO 50 I=1,ISMPL
         XSUM = XSUM + (X(I)-XMEAN)*(X(I)-XMEAN)
         YSUM = YSUM + (Y(I)-YMEAN)*(Y(I)-YMEAN)
 50   CONTINUE
      XSUM = SQRT ( (XSUM/FLOAT(ISMPL-1))/ISMPL )
      YSUM = SQRT ( (YSUM/FLOAT(ISMPL-1))/ISMPL )
C
C
C
      XS1 = XMEAN + T95* XSUM
      XS2 = XMEAN + T90* XSUM
      XS3 = XMEAN - T95* XSUM
      XS4 = XMEAN - T90* XSUM
      XTEMP1 = XS1
      XTEMP3 = XS3
C
C
C
      YS1 = YMEAN + T95* YSUM
      YS2 = YMEAN - T95* YSUM
C
C
C
C
C     WRITE SENSOR INFORMATION TO UNIT 13.
C
      ISHIPD = IFIX(SHIPID)
      INDRC = 0
      IF ( (XS1 .GE. XT) .AND. (XS3 .LE. XT) ) INDRC = 1
      WRITE (13,104) ISHIPD,XT,YT,XMEAN,YMEAN,XS3,XS1,YS2,YS1,XS4,XS2,IR
V,INDRC
 104  FORMAT (2X,I2,3X,1H(,F6.2,1H,,F6.2,') (',F6.2,1H,,F6.2,') (',F6.2,
     $1H,,F6.2,') (',F6.2,1H,,F6.2,') (',F6.2,1H,,F6.2,') ',30X,I1,4X,I1
)
C
C
C
         XS1 = 10.*(XS1-65.) + 65.
         XS3 = 10.*(XS3-65.) + 65.
         YS1 = 10.*(YMEAN-13) + 13.
         YS2 = YS1
         XS2 = (XS1+XS3)/2.
      RETURN
```

```
          END
    C
    C
    C
          SUBROUTINE HEAD(NSMPL,ACURCY,T95,T90)
    C
    C
    C
    C
    C     HEADING FOR UNIT 13.
    C
          WRITE (13,495)
    495   FORMAT ('***************************************** ',
         $'RELEVANT DATA FOR SENSOR INFORMATION ****************************
*',
         $'**************')
          WRITE (13,560) NSMPL,ACURCY,T95,T90
    560   FORMAT (25X,'SAMPLE POINT = ',I5,2X,', % ACCURACY = ',F6.2,2X,
         $'; T-VAL = ',F6.3,3X,F6.3)
          WRITE (13,505)
    505   FORMAT ('_________________________________________________',
         $'_________________________________________________',
         $'______________________')
          WRITE (13,801)
          WRITE (13,802)
    801   FORMAT ('PLAT- ',6X,'EST.',9X,'EST. -2ND',7X,'95% CONFIDENCE INTER
VALS',
         $9X,'90% C.I.',5X,'              ',6X,'          IRV  IN')
    802   FORMAT ('FORM      LOCATION',8X,'LOCATION',7X,'X-POSITION',6X,
         $'Y-POSITION',6X,'X-POSITION    DIREC.  DIREC.  EST. DIREC.')
          WRITE (13,505)
    C
    C     HEADING FOR UNIT 30.
    C
          WRITE (30,510)
    510   FORMAT ('************************************ ',
         $'RELEVANT DATA FOR PLATFORM INFORMATION ***************************
****',
         $'******')
          WRITE (30,560) NSMPL,ACURCY,T95,T90
          WRITE (30,995)
    995   FORMAT ('________________________________________________________
',
---',
         $'___________________________________________________')
          WRITE (30,811)
    811   FORMAT ('PLATFORM',5X,'TRUE LOCATION',5X,'ESTIMATED LOCATION',
         $3X,'95% C.I. X-POSITION   IN   95% C.I. Y-POSITION    IN')
          WRITE (30,995)
          RETURN
          END
```

POLYGON

```
C
C
C      This program is composed of the following routines:
C
C      1. CONTROL (MAIN PROGRAM).
C      2. CRETRK:
C           This subroutine is to draw a track for a ship given the
C           estimated position and direction.
C      3. CRELPS:
C           This is used for the drawing of a polygon (ellipse),
C           according to the given position and direction of the
C           track.
C
C
C      This is the main program which controls:
C
C      1. Input/Output routines.
C      2. Creating and deleting polygons routines.
C         In this section, the program also calls routines which
C      add words to radar types, ID number onto tracks.  Deleting
C      routine is to erase those words and polygons by puting a
C      panel in the vicinity of a word and by drawing the same
C      polygon by using a black line color.
C
C      3. Timing routines (with 4051).
C
C      Variables:
C          TBASE = Data base used for testing this program. (*** needs to be modified.)
C          REPLY = A logical variable used to determine the occurance
C                  of an event.  If true, the event is scheduled.
C        SCAL(I) = The original scale factor of polygon for track I.
C       OSCAL(I) = After calling a timing routine, the scale factor
C                  changes to this factor.
C         INDSNR = Indicator variable of sensor report, 0 means there
C                  is no sensor report so far; after the first sensor
C                  report has been processed, it is given the value 1.
C
       PROGRAM CONTROL
       LOGICAL REPLY
       REAL SCAL(11),OSCAL(11)
       REAL TBASE(300,10),TRNSFR(11,10)
       INTEGER INDSNR,ID,IRADAR.IFROM,NOFTRK
       DATA IDUM /0/
       DATA INDSNR /0/,NOFTRK /0/
       DATA SCAL /11*0./
       DATA TRNSFR /110*0.0/
C
C      Open some logical devices.
C
       OPEN (UNIT = 15)
       OPEN (UNIT=20,FILE='AXO:')
       OPEN (UNIT=30,FILE='AXI:')
       OPEN (UNIT = 16)
C
C      NODATA stands for "Number Of DATA".
C
       READ (15,7) NODATA,ISEED,NSMPL,ACURCY
 7     FORMAT (I5,5X,I10,5X,I5,5X.F6.2)
C
C      Heading of the output file.
```

```
C
      CALL HEAD (ISEED,NSMPL,ACURCY)
      READ (15,10) ((TBASE(I,J),J=1,10),I=1,NODATA)
10    FORMAT (F9.3,1X,F3.1,1X,F8.2,1X,F8.2,1X,F8.2,1X,F8.2,1X,F8.2,1X,F8
.2,1X,
     1F8.2,1X,F3.1)
C
C     The record of this data file is arranged as following:
C
C     time-of-event, irrelevant, ID (if track),
C     x coordinate, y coordinate, degree of rotation,
C     radar type, and speed.
C
C
C
      CALL GRSTRT(4105,1)
C
C     First define colors and panel pattern.
C
      CALL COLOR
C
C     Now define some functions for the terminal.
C
      CALL DEFINE
      CALL NEWPAG
      CALL PROMPT
      WRITE (6,1111)
      WRITE (6,1111)
      WRITE (6,1111)
1111  FORMAT ('                ')
C
C     Process the data sequentially.
C
      DO 160 I=1, NODATA
C
C     Set the third element of the record to be an ID number for
C     it (a track, or a sensor report; sensor report always has
C     an ID of 11).
C
      ID = IFIX (TBASE(I,2))
C
C
100   CALL TIMING (TBASE(I,1),TBASE(I,7),REPLY,SCAL(ID))
      IF (REPLY) THEN
           IF (TBASE(I,10) .GT. 0.) THEN
                IRADAR = IFIX (TBASE(I,10))
                IFROM = IFIX (TBASE(I,2))
                CALL CMCLOS
                CALL CMOPEN
                CALL LINCLR (2)
                CALL CRESNR (TBASE(I,3),TBASE(I,5),TBASE(I,7),TBASE(I,8)
)
                CALL PROMPT
                CALL RADAR (IRADAR)
                INDSNR = 1
C
C    Count for the number of tracks which are on the screen, and then
C    prompt for the operator to input his dicision.
C
                DO 118 KOUNT=1,10
                IF (TRNSFR(KOUNT,3) .NE. 0.0) THEN
                    NOFTRK = NOFTRK + 1
```

```
                        ENDIF
 118                    CONTINUE
                        CALL DECIDE (TBASE(I,1),IFROM,NOFTRK,IRADAR)
                        CALL CMCLOS
                        CALL CMOPEN
                        CALL LINCLR (1)
                        CALL CRESNR (TBASE(I,3),TBASE(I,5),TBASE(I,7),TBASE(I,8)
)
                        CALL ERADAR
                        CALL PROMPT
                   ELSE IF (TRNSFR(ID,2) .EQ. 0.0) THEN
                        GOTO 120
                   ELSE
                        CALL LINCLR (1)
                        CALL CRETRK (TRNSFR(ID,3),TRNSFR(ID,4),TRNSFR(ID,5))
                        CALL ERASE (TRNSFR(ID,3),TRNSFR(ID,4))
                        CALL CRELPS (TRNSFR(ID,3),TRNSFR(ID,4),TRNSFR(ID,5),OSCA
L(ID))
 120                    CALL LINCLR (4)
                        CALL CRETRK (TBASE(I,3),TBASE(I,4),TBASE(I,5))
                        CALL WORD (TBASE(I,3),TBASE(I,4),ID)
                        CALL LINCLR (3)
                        CALL CRELPS (TBASE(I,3),TBASE(I,4),TBASE(I,5),SCAL(ID))
                        DO 125 K=1,8
                        TRNSFR(ID,K) =  TBASE(I,K)
 125                    CONTINUE
                        OSCAL(ID) = SCAL(ID)
                   ENDIF
              ELSE
                   DO 130 K=1,10
                   CALL CMCLOS
                   WRITE (20,159)
 159               FORMAT ('ENLRGE')
                   READ (30,169) TIME
 169               FORMAT (F9.3)
                   CALL CMOPEN
                   IF (TRNSFR(K,2) .EQ. 0.0) THEN
                        GOTO 130
                   ELSE
              CALL TMXFTR (TRNSFR(K,1),TRNSFR(K,7),OSCAL(K),SCAL(K))
                        CALL LINCLR (3)
                        CALL CRELPS (TRNSFR(K,3),TRNSFR(K,4),TRNSFR(K,5),SCAL(K)
)
                        CALL LINCLR (1)
                        CALL CRELPS (TRNSFR(K,3),TRNSFR(K,4),TRNSFR(K,5),OSCAL(K
))
                        OSCAL(K) = SCAL(K)
                   ENDIF
 130               CONTINUE
                   GOTO 100
              ENDIF
 160          CONTINUE
              CALL FINE (NOFTRK)
              CALL GRSTOP
              CLOSE (9)
              CLOSE (20)
              CLOSE (30)
              CLOSE (40)
              STOP
              END
 C
 C              THIS SUBROUTINE IS TO GENERATE THE TRACK OF A SHIP UNDER THE
 C            HYPOTHESIS OF KNOWN MOVING DIRECTION.
 C
```

```
C   VARIABLES:
C        XTRK = X COORDINATE OF THE TRACK.
C        YTRK = Y COORDINATE OF THE TRACK.
C         DEG = ANGLE OF DIRECTION.
C           X = ARRAY TO STORE THE X COORDINATES OF THE POSITION OF THE TRACK.
C           Y = ARRAY TO STORE THE Y COORDINATES OF THE POSITION OF THE TRACK.
C
      SUBROUTINE CRETRK (XTRK,YTRK,DEG)
      REAL     X(4),    Y(4)
C
C  A DIMOND SHAPE POLYGON FOR THE TRACK.
C
      DATA X/-4.,0.,4.,0./
      DATA Y/0.,-4.,0.,4./
C
C  FIRST DRAW THE TRACK, THEN PIVOT THE CENTER POINT OF THIS TRACK FOR ROTATION.
C
      CALL MOVE(XTRK+2.,YTRK+2.)
      CALL VECREL
      CALL POLY(4,    X,     Y)
      CALL VECABS
      CALL PIVOT(XTRK,YTRK)
      CALL ROTATE(DEG,DEG)
      CALL MOVE(XTRK+1.,YTRK)
C
C  DRAW THE DIRECTION OF THE TRACK, THEN ROTATE THE COORDINATION BACK.
C
      CALL DRAW(XTRK+4.,YTRK)
      CALL ROTATE(-DEG,-DEG)
      RETURN
      END
C
C     This subroutine is to draw an ellipse.
C

      SUBROUTINE CRELPS (XLPS,YLPS,DEGLPS,COELPS)
      REAL X(30),Y(30)
C
C
C
      IF (COELPS .GE. 4.) COELPS = 4.
      DO 10 I=1,29
      ANGL = FLOAT(I-1)/29. * 2. * 3.1415926
      X(I) = XLPS + 4.5 + 9. * COS(ANGL) * COELPS
      Y(I) = YLPS + 4. * SIN(ANGL) * COELPS
10    CONTINUE
      X(30) = X(1)
      Y(30) = Y(1)
C
C
C
      CALL PIVOT (XLPS,YLPS)
      CALL ROTATE (DEGLPS,DEGLPS)
C
C
C
      CALL SKIP
      CALL POLY (30,X,Y)
C
C
```

```
C
      CALL ROTATE (-DEGLPS,-DEGLPS)
      RETURN
      END
C
C
C
C
      SUBROUTINE CRESNR (XR,XL,YL,YR)
      REAL XR,XL,YL,YR
C
C
C
      CALL SKIP
C
C
C
      CALL MOVE (65.,13.)
      CALL DRAW (XR,YR)
      CALL MOVE (65.,13.)
      CALL DRAW (XL,YL)
      RETURN
      END
C
C
C
C
C
C     This subroutine defines the colors of pens used in this program.
C
      SUBROUTINE COLOR
C
C           BLACK = IGL COLOR MAP, PEN #1.
C            BLUE = IGL COLOR MAP, PEN #2.
C          YELLOW = IGL COLOR MAP, PEN #3.
C             RED = IGL COLOR MAP, PEN #4.
C           GREEN = IGL COLOR MAP, PEN #5.
C           WHITE = IGL COLOR MAP, PEN #6.
C
      REAL BLACK(3),BLUE(3),YELLOW(3),RED(3),GREEN(3),WHITE(3)
      DATA BLUE/300.0,50.0,100.0/
      DATA YELLOW /180.0,50.0,100.0/
      DATA BLACK /3*0.0/
      DATA RED /100.0,50.0,100.0/
      DATA GREEN /240.,50.,100./
      DATA WHITE /0.,100.,100./
C
C     DEFINE THE COLORS USED IN THIS PROGRAM.
C
      CALL CLRMAP (1,3,BLACK)
      CALL CLRMAP (2,3,BLUE)
      CALL CLRMAP (3,3,YELLOW)
      CALL CLRMAP (4,3,RED)
      CALL CLRMAP (5,3,GREEN)
      CALL CLRMAP (6,3,WHITE)
C
C     FILPAN: pattern 17 is black, FALSE means no boundry specified.
C
      CALL FILPAN (17,.FALSE.)
      CALL TXTCLR (5)
```

```
      RETURN
      END
C
C
C
C
      SUBROUTINE TIMING (OCTIME,SPDFTR,ACTING,COEFF)
      LOGICAL ACTING
      REAL TIME
      DATA IABC /0/
C
C
C
      CALL CMCLOS
      WRITE (20,100)
100   FORMAT ('TIMING')
      READ (30,200) TIME
200   FORMAT (F9.3)
      IF (TIME .GE. OCTIME) THEN
           ACTING = .TRUE.
           IF ( IFIX(SPDFTR*5) .EQ. 0.) THEN
                COEFF = 1.
           ELSE
                COEFF = IFIX (SPDFTR*5)
           ENDIF
           CALL CMOPEN
           RETURN
      ELSE
           ACTING = .FALSE.
           COEFF = 1.
           CALL CMOPEN
      ENDIF
      RETURN
      END
C
C
C
      SUBROUTINE ERASE (XASE,YASE)
      REAL X(4),Y(4)
      DATA X/-4.,-4.,4.,4./
      DATA Y/4.,-4.,-4.,4./
      CALL MOVE(XASE+1.5,YASE)
      CALL VECREL
      CALL PANEL(4,X,Y)
      CALL VECABS
      RETURN
      END
C
C
C
      SUBROUTINE WORD (XORD,YORD,IORD)
      CHARACTER*1 ABC
      CHARACTER*2 EFG
      INTEGER IORD,J
      J = IORD + 48
      CALL MOVE (XORD-1.,YORD-1.)
      IF (IORD .EQ. 10) THEN
           EFG = CHAR (J)
           CALL TEXT (2,EFG)
      ELSE
```

```
               ABC = CHAR (J)
               CALL TEXT (1,ABC)
          ENDIF
          CALL SKIP
          RETURN
          END
C
C
C
C
          SUBROUTINE TMXFTR (HAPTME,SPDFTR,OLDFTR.WENFTR)
          REAL TIME,ENLRGE,OLDFTR,WENFTR
          INTEGER IS
          CALL CMCLOS
          WRITE (20,100)
  100     FORMAT ('TMXFTR')
          READ (30,200) TIME
  200     FORMAT (F9.3)
          ENLRGE = (TIME - HAPTME) * .002
          IS = IFIX (SPDFTR*5)
          GOTO (300,400,500,550) IS
  300     WENFTR = OLDFTR + ENLRGE
          GOTO 600
  400     WENFTR = OLDFTR + ENLRGE * 2.
          GOTO 600
  500     WENFTR = OLDFTR + ENLRGE * 3.
          GOTO 600
  550     WENFTR = OLDFTR + ENLRGE * 4.
  600     CALL CMOPEN
          RETURN
          END
C
C
C
C
          SUBROUTINE ERADAR
          REAL X(4),Y(4)
          DATA X /8.,0.,-8.,0./
          DATA Y /0.,8.,0.,-8./
          CALL MOVE (63.,6.)
          CALL VECREL
          CALL PANEL (4,X,Y)
          CALL VECABS
          CALL SKIP
          RETURN
          END
C
C
C
C
          SUBROUTINE RADAR (IDAR)
          CHARACTER*1 ABC
          INTEGER IDAR,J
          J = IDAR + 64
          ABC = CHAR (J)
          CALL MOVE (64.5,7.)
          CALL TEXT (1,ABC)
          CALL SKIP
          RETURN
          END
```

```
C
C
C
C
C
        SUBROUTINE DECIDE (OCTIME,IOFTRK,NOSUM,ITYPE)
        INTEGER IOFTRK,ANSWER,NOSUM,ITYPE
        REAL OCTIME
        CHARACTER*1 ABC
        CALL CMCLOS
        KNSWER = 1
        J = ITYPE + 64
        ABC = CHAR(J)
        WRITE (20,100)
  100   FORMAT ('DECIDE')
        READ (30,105) TIME
  105   FORMAT (F9.3)
        CALL CMOPEN
        CALL CMCLOS
        WRITE (20,110) IFROM
        CALL CMOPEN
        CALL CMCLOS
        WRITE (20,110) ITYPE
  110   FORMAT (I3)
        READ (30,120) ANSWER
  120   FORMAT (I2)
        IF (ANSWER .NE. ITYPE) KNSWER=0
        WRITE (16,160) OCTIME,ABC,IOFTRK,ANSWER,TIME,KNSWER
  160   FORMAT (F9.3,5X,A1,10X,I1,9X,I1,5X,F9.3,5X,I1)
        CALL CMOPEN
        NOSUM=0
        RETURN
        END
C
C       This subroutine defines some setting for the terminal 4107.
C
        SUBROUTINE DEFINE
        CALL BEGOS ('L','L',10,IBUF)
        CALL CHROS ('30',IBUF)
        CALL ENDOS (IBUF,.TRUE.)
        CALL BEGOS ('L','I',10,JBUF)
        CALL INTOS (1,JBUF)
        CALL ENDOS (JBUF,.TRUE.)
        CALL BEGOS ('L','V',10,KBUF)
        CALL INTOS (0,KBUF)
        CALL ENDOS (KBUF,.TRUE.)
        WRITE (6,10)
   10   FORMAT (A1)
C
C
C
        RETURN
        END
C

C
C
        SUBROUTINE PROMPT
        CALL LINCLR (6)
        CALL MOVE (62.,13.)
```

```
      CALL DRAW (68.,13.)
      CALL MOVE (65.,16.)
      CALL DRAW (65.,10.)
      CALL MOVE (62.,13.)
      CALL ARC3PT (65.,16.,68.,13.)
      CALL ARC3PT (65.,10.,62.,13.)
      RETURN
      END
C
C
C
      SUBROUTINE FINE(IOFTRK)
      CALL BEG05 ('L','I',10,KBUF)
      CALL INT05 (1,KBUF)
      CALL INT05 (4,KBUF)
      CALL INT05 (4,KBUF)
      CALL END05 (KBUF,.TRUE.)
      CALL BEG05 ('L','V',10,JBUF)
      CALL INT05 (1,JBUF)
      CALL END05 (JBUF,.TRUE.)
      CALL CMCLOS
      WRITE (16,10)
10    FORMAT ('________________________________________________________________
__')
      WRITE (16,15) IOFTRK
15    FORMAT ('***** DISPLAY TYPE: POLYGONS *** SHIP DENSITY: ',I3,'****
*')
      CALL CMOPEN
      RETURN
      END
C
C     Heading of the output file.
C
      SUBROUTINE HEAD (IXAL,ISMPL,PCISN)
      WRITE (16,10)
      WRITE (16,15) PCISN,ISMPL,IXAL
10    FORMAT ('***************** RESULTS OF ASSOCIATION ****************
**')
15    FORMAT ('  S. ACCURCY: ',F6.2,', SAMPLE SIZE: ',I5,', SEED: ',I10)
      WRITE (16,20)
20    FORMAT ('________________________________________________________________
__')
      WRITE (16,25)
25    FORMAT (3X,'TIME',5X,'RADAR',4X,'PLATFORM',2X,'RESPONSE',2X,'RES.
TIME',
     *3X,'POINT')
      WRITE (16,20)
      RETURN
      END
```

LINE

```
      PROGRAM CONTROL
      LOGICAL REPLY
      REAL TBASE(300,10),TRNSFR(11,10)
      INTEGER INDSNR,ID,IRADAR,IFROM,NOFTRK
      DATA IDUM /0/
      DATA INDSNR /0/,NOFTRK /0/
      DATA TRNSFR /110*0.0/
C
C     Open some logical devices.
C
      OPEN (UNIT = 15)
      OPEN (UNIT=20,FILE='AXO:')
      OPEN (UNIT=30,FILE='AXI:')
      OPEN (UNIT = 16)
C
C     NODATA stands for "Number Of DATA".
C
      READ (15,7) NODATA,ISEED,NSMPL,ACURCY
 7    FORMAT (I5,5X,I10,5X,I5,5X,F6.2)
      READ (15,10) ((TBASE(I,J),J=1,10),I=1,NODATA)
 10   FORMAT (F9.3,1X,F3.1,1X,F8.2,1X,F8.2,1X,F8.2,1X,F8.2,1X,F8.2,1X,F8.2,1X,
     1F8.2,1X,F3.1)
C
      CALL HEAD(ISEED,NSMPL,ACURCY)
C
C
      CALL GRSTRT(4105,1)
C
C     First define colors and panel pattern.
C
      CALL COLOR
C
C     Now define some functions for the terminal.
C
      CALL DEFINE
      CALL NEWPAG
      CALL PROMPT
C
C     Process the data sequentially.
C
      DO 160 I=1, NODATA
C
C     Set the third element of the record to be an ID number for
C     it (a track, or a sensor report; sensor report always has
C     an ID of 11).
C
      ID = IFIX (TBASE(I,2))
C
C
 100  CALL TIMING (TBASE(I,1),REPLY)
      IF (REPLY) THEN
           IF (TBASE(I,10) .GT. 0.) THEN
                IRADAR = IFIX (TBASE(I,10))
                IFROM = IFIX (TBASE(I,2))
                CALL PROMPT
                CALL CMCLOS
                CALL CMOPEN
                CALL LINCLR (2)
                CALL CRESNR (TBASE(I,4),TBASE(I,7))
                CALL PROMPT
```

```
                     CALL RADAR (IRADAR)
                     DO 115 K=1,10
                     TRNSFR(11,K)=TBASE(I,K)
      115            CONTINUE
      C
      C     Count for the number of tracks which are on the screen, and then
      C     prompt for the operator to input his dicision.
      C
                     DO 118 KOUNT=1,10
                     IF (TRNSFR(KOUNT,3) .NE. 0.0) THEN
                        NOFTRK = NOFTRK + 1
                     ENDIF
      118            CONTINUE
                     CALL DECIDE (TBASE(I,1),IFROM,NOFTRK,IRADAR)
                     CALL LINCLR (1)
                     CALL CRESNR (TBASE(I,4),TBASE(I,7))
                     CALL ERADAR
                     CALL PROMPT
                ELSE IF (TRNSFR(ID,2) .EQ. 0.0) THEN
                     GOTO 120
                ELSE
                     CALL LINCLR (1)
                     CALL CRETRK (TRNSFR(ID,3),TRNSFR(ID,4),TRNSFR(ID,5))
                     CALL ERASE (TRNSFR(ID,3),TRNSFR(ID,4))
      120            CALL LINCLR (4)
                     CALL CRETRK (TBASE(I,3),TBASE(I,4),TBASE(I,5))
                     CALL WORD (TBASE(I,3),TBASE(I,4),ID)
                     CALL LINCLR (3)
                     DO 125 K=1,8
                     TRNSFR(ID,K) =  TBASE(I,K)
      125            CONTINUE
                ENDIF
            ELSE
                GOTO 100
            ENDIF
      160   CONTINUE
            CALL FINE(NOFTRK)
            CALL GRSTOP
            CLOSE (9)
            CLOSE (20)
            CLOSE (30)
            CLOSE (40)
            STOP
            END
      C
      C     THIS SUBROUTINE IS TO GENERATE THE TRACK OF A SHIP UNDER THE
      C    HYPOTHESIS OF KNOWN MOVING DIRECTION.
      C
      C    VARIABLES:
      C         XTRK = X COORDINATE OF THE TRACK.
      C         YTRK = Y COORDINATE OF THE TRACK.
      C          DEG = ANGLE OF DIRECTION.
      C            X = ARRAY TO STORE THE X COORDINATES OF THE POSITION OF THE
TRACK.
      C            Y = ARRAY TO STORE THE Y COORDINATES OF THE POSITION OF THE
TRACK.
      C
            SUBROUTINE CRETRK (XTRK,YTRK,DEG)
            REAL     X(4),   Y(4)
      C
      C   A SQUARE SHAPE POLYGON FOR THE TRACK.
      C
```

```
         DATA X/-4.,0.,4.,0./
         DATA Y/0.,-4.,0.,4./
C
C  FIRST DRAW THE TRACK, THEN PIVOT THE CENTER POINT OF THIS TRACK FOR ROTATION.
C
         CALL MOVE(XTRK+2.,YTRK+2.)
         CALL VECREL
         CALL POLY(4,   X,    Y)
         CALL VECABS
         CALL PIVOT(XTRK,YTRK)
         CALL ROTATE(DEG,DEG)
         CALL MOVE(XTRK+1.,YTRK)
C
C  DRAW THE DIRECTION OF THE TRACK, THEN ROTATE THE COORDINATION BACK.
C
         CALL DRAW(XTRK+4.,YTRK)
         CALL ROTATE(-DEG,-DEG)
         RETURN
         END
C
C
C
C
         SUBROUTINE CRESNR (XP,YP)
         REAL XP,YP
C
C
C
         CALL SKIP
         CALL MOVE (65.,13.)
         CALL DRAW (XP,YP)
C
C
         RETURN
         END
C
C
C
C
C
C        This subroutine defines the colors of pens used in this program.
C
         SUBROUTINE COLOR
C
C             BLACK = IGL COLOR MAP, PEN #1.
C              BLUE = IGL COLOR MAP, PEN #2.
C            YELLOW = IGL COLOR MAP, PEN #3.
C               RED = IGL COLOR MAP, PEN #4.
C             GREEN = IGL COLOR MAP, PEN #5.
C             WHITE = IGL COLOR MAP, PEN #6.
C
         REAL BLACK(3),BLUE(3),YELLOW(3),RED(3),GREEN(3),WHITE(3)
         DATA BLUE/300.,50.0,100.0/
         DATA YELLOW /180.0,50.0,100.0/
         DATA BLACK /3*0.0/
         DATA RED /100.0,50.0,100.0/
         DATA GREEN /240.,50.,100./
         DATA WHITE /0.,100.,100./
C
C        DEFINE THE COLORS USED IN THIS PROGRAM.
```

```
C
      CALL CLRMAP (1,3,BLACK)
      CALL CLRMAP (2,3,BLUE)
      CALL CLRMAP (3,3,YELLOW)
      CALL CLRMAP (4,3,RED)
      CALL CLRMAP (5,3,GREEN)
      CALL CLRMAP (6,3,WHITE)
C
C     FILPAN: pattern 17 is black, FALSE means no boundry specifi
C
      CALL FILPAN (17,.FALSE.)
      CALL TXTCLR (5)
      RETURN
      END
C
C
C
C
C
C
C
C
C
      SUBROUTINE TIMING (OCTIME,ACTING)
      LOGICAL ACTING
      REAL TIME
      DATA IABC /0/
C
C
C
      CALL CMCLOS
      WRITE (20,100)
100   FORMAT ('TIMING')
      READ (30,200) TIME
200   FORMAT (F9.3)
        ACTING = .FALSE.
        IF ( TIME .GE. OCTIME ) ACTING = .TRUE.
      CALL CMOPEN
      RETURN
      END
C
C
C
      SUBROUTINE ERASE (XASE,YASE)
      REAL X(4),Y(4)
      DATA X/-4.,-4.,4.,4./
      DATA Y/4.,-4.,-4.,4./
      CALL MOVE(XASE+1.5.YASE)
      CALL VECREL
      CALL PANEL(4.X.Y)
      CALL VECABS
      RETURN
      END
C
C
C
      SUBROUTINE WORD (XORD.YORD.IORD)
      CHARACTER*1 ABC
      CHARACTER*2 EFG
      INTEGER IORD.J
```

```
      J = IORD + 48
      CALL MOVE (XORD-1.,YORD-1.)
      IF (IORD .EQ. 10) THEN
           EFG = CHAR (J)
           CALL TEXT (2,EFG)
      ELSE
           ABC = CHAR (J)
           CALL TEXT (1,ABC)
      ENDIF
      CALL SKIP
      RETURN
      END
C
C
C
C
C
C
C
      SUBROUTINE ERADAR
      REAL X(4),Y(4)
      DATA X /8.,0.,-8.,0./
      DATA Y /0.,8.,0.,-8./
      CALL MOVE (63.,6.)
      CALL VECREL
      CALL PANEL (4,X,Y)
      CALL VECABS
      CALL SKIP
      RETURN
      END
C
C
C
C
      SUBROUTINE RADAR (IDAR)
      CHARACTER*1 ABC
      INTEGER IDAR,J
      J = IDAR + 64
      ABC = CHAR (J)
      CALL MOVE (64.5,7.)
      CALL TEXT (1,ABC)
      CALL SKIP
      RETURN
      END
C
C
C
C
C
      SUBROUTINE DECIDE (OCTIME,IOFTRK,NOSUM,ITYPE)
      INTEGER IOFTRK,ANSWER,NOSUM,ITYPE
      REAL OCTIME
      CHARACTER*1 ABC
      CALL CMCLOS
      J = ITYPE + 64
      ABC = CHAR(J)
      WRITE (20,100)
100   FORMAT ('DECIDE')
      CALL CMOPEN
      CALL CMCLOS
```

```
      WRITE (20,110) IOFTRK
      READ (30,105) TIME
105   FORMAT (F9.3)
      CALL CMOPEN
      CALL CMCLOS
      WRITE (20,110) ITYPE
110   FORMAT (I3)
      KANSWER = 1
      IF (ANSWER .NE. IOFTRK) KANSWER=0
      READ (30,120) ANSWER
120   FORMAT (I2)
      WRITE (16,160) OCTIME,ABC,IOFTRK,ANSWER,TIME,KNSWER
160   FORMAT (F9.3,5X,A1,10X,I1,9X,I1,5X,F9.3,5X,I1)
      CALL CMOPEN
      NOSUM=0
      RETURN
      END
C
C     This subroutine defines some setting for the terminal 4107.
C
      SUBROUTINE DEFINE
      CALL BEGO5 ('L','L',10,IBUF)
      CALL CHRO5 ('30',IBUF)
      CALL ENDO5 (IBUF,.TRUE.)
      CALL BEGO5 ('L','I',10,JBUF)
      CALL INTO5 (1,JBUF)
      CALL ENDO5 (JBUF,.TRUE.)
      CALL BEGO5 ('L','V',10,KBUF)
      CALL INTO5 (0,KBUF)
      CALL ENDO5 (KBUF,.TRUE.)
      WRITE (6,10)
10    FORMAT (A1)
C
C
C
      RETURN
      END
C

C
C
      SUBROUTINE PROMPT
      CALL LINCLR (6)
      CALL MOVE (62.,13.)
      CALL DRAW (68.,13.)
      CALL MOVE (65.,16.)
      CALL DRAW (65.,10.)
      CALL MOVE (62.,13.)
      CALL ARC3PT (65.,16.,68.,13.)
      CALL ARC3PT (65.,10.,62.,13.)
      RETURN
      END
C
C
C
      SUBROUTINE FINE(IOFTRK)
      CALL BEGO5 ('L','I',10,KBUF)
      CALL INTO5 (1,KBUF)
      CALL INTO5 (4,KBUF)
      CALL INTO5 (4,KBUF)
```

```
      CALL ENDOS (KBUF,.TRUE.)
      CALL BEGOS ('L','V',10,JBUF)
      CALL INTOS (1,JBUF)
      CALL ENDOS (JBUF,.TRUE.)
      CALL CMCLOS
      WRITE (16,10)
10    FORMAT ('______________________________________________________________
__')
      WRITE (16,15) IOFTRK
15    FORMAT ('***** DISPLAY TYPE: NON-POLG *** SHIP DENSITY: ',I3,'****
*')
      CALL CMOPEN
      RETURN
      END
C
C     HEADING OF OUTPUT FILE.
C
      SUBROUTINE HEAD (IXAL,ISMPL,PCISN)
      WRITE (16,10)
      WRITE (16,15) PCISN,ISMPL,IXAL
10    FORMAT ('***************** RESULTS OF ASSOCIATION ****************
**')
15    FORMAT ('  S. ACCURCY: ',F6.2,', SAMPLE SIZE: ',I5,', SEED: ',I10)
      WRITE (16,20)
20    FORMAT ('______________________________________________________________
__')
      WRITE (16,25)
25    FORMAT (3X,'TIME',5X,'RADAR',4X,'PLATFORM',2X,'RESPONSE',2X,'RES.
TIME',
     $3X,'POINT')
      WRITE (16,20)
      RETURN
      END
```

READY

```
10 REM **********************************************
20 REM *                   READY4S                  *
30 REM **********************************************
40 REM This program facilitates the communication between the HP-86
50 REM and the Tektronix 4170.  The first part is to draw a figure
60 REM in which the operator can see what associations he/she made.
70 REM The second part is to do the communicaion.
80 REM **********   First part.    **********
90 DIM ORDER$(30)
100 DATA "1","2"."3","4","5","6","7","8","9","10","11","12","13","14","15","16",
"17"
110 DATA "18","19","20","21","22","23","24","25","26","27","28","29","30"
120 FOR K=1 TO 30 @ READ ORDER$(K)@ NEXT K
130 GRAPHALL
140 GCLEAR
150 CSIZE 4
160 SCALE 0,160,0,120
170 MOVE 0,110 @ LABEL "*****************************  ASSOCIATION INFORMATION  **
********************************"
180 CSIZE 4
190 MOVE 0,105 @ LABEL "PLATFORM/ASSO. ORDER"
200 MOVE 0,100 @ DRAW 160,100
210 FOR I=0 TO 10 STEP 9
220 MOVE 0,I*10
230 DRAW 160,I*10
240 NEXT I
250 MOVE 10,0 @ DRAW 10,100
260 FOR J=0 TO 33 STEP 32
270 MOVE J*5,0
280 DRAW J*5,100
290 NEXT J
300 FOR K=1 TO 30
310 K1=7.5+5*K
320 IF K>= 10 THEN K1=K1-1.1
330 MOVE K1,95
340 LABEL ORDER$(K)
350 NEXT K
360 CSIZE 4
370 FOR K=1 TO 8
380 K1=94-K*10
390 MOVE 4,K1
400 LABEL ORDER$(K)
410 NEXT K
420 REM **********   Second part.   **********
430 ON KEY# 1 GOTO 450
440 GOTO 440
450 OFF KEY# 1
460 X1=12.5
470 I=99
480 SETTIME 144,0
490 ENTER 10 USING "#,K  " ; A$
500 IF A$="DECIDE" THEN 520 ELSE 590
510 END
520 KIND=TIME
530 BEEP 20,180
540 GOSUB 610
550 MOVE X1,94-I*10 @ LABEL RADAR$
560 X1=X1+5
570 OUTPUT 10 USING "#,K" : I
580 GOTO 490
590 OUTPUT 10 USING "DDDDD.DDD" : TIME
600 GOTO 490
610 ON KYBD P1,"12345678" GOTO 630
620 GOTO 620
630 I=P1-48
640 P1=0
650 OFF KYBD "12345678"
660 KIND=TIME -KIND
670 BEEP 10,180
680 OUTPUT 10 USING "DDDDD.DDD" ; KIND
690 ENTER 10 USING "#,K" ; KEY
700 ENTER 10 USING "#,K" ; N
710 N=N+64
720 RADAR$=CHR$ (N)
730 RETURN
```

APPENDIX C

Program listing for ANOVA

```
10 REM Program for analyzing the data from my experiment.
20 REM A four dimensional array is set for data storage.
30 DFT=1
40 OPTION BASE 0
50 REM First element:= factor A
60 REM Second element:= factor B
70 REM Third eleent:= factor C
80 REM Fourth element:= replicates
90 DIM THE.DATA(2,2,2,10),REPLICATIONS(10)
100 REM now enter the data
110 INPUT "Enter the number of replications:";NODATA
120 FOR I=0 TO 1
130 FOR J=0 TO 1
140 FOR K=0 TO 1
150 FOR L=0 TO NODATA-1
160 READ THE.DATA(I,J,K,L)
170 THE.DATA(I,J,K,L)=LOG(THE.DATA(I,J,K,L)/(1-THE.DATA(I,J,K,L)))
180 NEXT L
190 NEXT K
200 NEXT J
210 NEXT I
220 DATA .93333,.86666,.93333,.93333
230 DATA .66666,.66666,.73333,.6
240 DATA .63333,.76666,.7,.76666
250 DATA .5,.46666,.26666,.36666
260 DATA .93333,.93333,.93333,.93333
270 DATA .53333,.53333,.6,.6
280 DATA .6,.6,.63333,.6
290 DATA .56666,.73333,.56666,.6
300 FOR L=0 TO NODATA-1
310 ONE=THE.DATA(0,0,0,L)+ONE
320 A=THE.DATA(1,0,0,L)+A
330 B=THE.DATA(0,1,0,L)+B
340 C=THE.DATA(0,0,1,L)+C
350 AB=THE.DATA(1,1,0,L)+AB
360 AC=THE.DATA(1,0,1,L)+AC
370 BC=THE.DATA(0,1,1,L)+BC
380 ABC=THE.DATA(1,1,1,L)+ABC
390 NEXT L
400 REM Now calculate the sum of squares.
410 EFFECT.I = ONE+A+B+AB+C+AC+BC+ABC
420 EFFECT.A = -ONE+A-B+AB-C+AC-BC+ABC
430 EFFECT.B = -ONE-A+B+AB-C-AC+BC+ABC
440 EFFECT.AB = ONE-A-B+AB+C-AC-BC+ABC
450 EFFECT.C = -ONE-A-B-AB+C+AC+BC+ABC
460 EFFECT.AC = ONE-A+B-AB-C+AC-BC+ABC
470 EFFECT.BC = ONE+A-B-AB-C-AC+BC+ABC
480 EFFECT.ABC = -ONE+A+B-AB+C-AC-BC+ABC
490 REM Now do some relevant calculations.
500 FOR I=0 TO 1
510 FOR J=0 TO 1
520 FOR K=0 TO 1
530 FOR L=0 TO NODATA-1
540 SUM=SUM+THE.DATA(I,J,K,L)
550 TREATMENT.SQUARE.SUM=TREATMENT.SQUARE.SUM+THE.DATA(I,J,K,L)^2
```

```
560 NEXT L
570 NEXT F
580 NEXT J
590 NEXT I
600 FOR COUNT=0 TO NODATA-1
610 FOR I=0 TO 1
620 FOR J=0 TO 1
630 FOR K=0 TO 1
640 REPLICATIONS(COUNT)=REPLICATIONS(COUNT)+THE.DATA(I,J,K,COUNT)
650 NEXT K
660 NEXT J
670 NEXT I
680 NEXT COUNT
690 CORRECTION.TERM=SUM*SUM/(8*NODATA)
700 SUM.OF.SQUARE.T=TREATMENT.SQUARE.SUM-CORRECTION.TERM
710 SUM.OF.SQUARE.TREATMENT=(1/NODATA)*((ONE^2)+(A^2)+(B^2)+(AB^2)+(C^2)+(AC^2)+
(BC^2)+(ABC^2))-CORRECTION.TERM
720 FOR COUNT=0 TO NODATA-1
730 SUM.OF.SQUARE.R=SUM.OF.SQUARE.R+(REPLICATIONS(COUNT)^2)
740 NEXT COUNT
750 SUM.OF.SQUARE.R=SUM.OF.SQUARE.R/8-CORRECTION.TERM
760 SUM.OF.SQUARE.ERROR=SUM.OF.SQUARE.T-SUM.OF.SQUARE.TREATMENT-SUM.OF.SQUARE.ER
ROR
770 REM Now calculate sum of squares for each effect.
780 DENUM=NODATA*8
790 SSA = EFFECT.A^2/DENUM
800 SSB = EFFECT.B^2/DENUM
810 SSAB = EFFECT.AB^2/DENUM
820 SSC = EFFECT.C^2/DENUM
830 SSBC = EFFECT.BC^2/DENUM
840 SSAC = EFFECT.AC^2/DENUM
850 SSABC = EFFECT.ABC^2/DENUM
860 PRINT "SSA=";SSA
870 PRINT "SSB=";SSB
880 PRINT "SSC=";SSC
890 PRINT "SSAB=";SSAB
900 PRINT "SSBC=";SSBC
910 PRINT "SSAC=";SSAC
920 PRINT "SSABC=";SSABC
930 PRINT:PRINT:PRINT:PRINT:PRINT:PRINT
940 PRINT "SUM.OF.SQUARE.R=";SUM.OF.SQUARE.R
950 PRINT "SUM.OF.SQUARE.TREATMENT=";SUM.OF.SQUARE.TREATMENT
960 PRINT "SUM.OF.SQUARE.T=";SUM.OF.SQUARE.T
970 PRINT "CORRECTION TERM=";CORRECTION.TERM
980 PRINT "SUM = ";SUM
990 SUM.OF.SQUARE.ERROR=SUM.OF.SQUARE.T-SUM.OF.SQUARE.TREATMENT-SUM.OF.SQUARE.R
1000 PRINT "sum of square of error =";SUM.OF.SQUARE.ERROR
1010 DFR=NODATA-1
1020 DFTOTAL=NODATA*8-1
1030 DFERROR=DFTOTAL-DFR-7
1040 MEAN.SSE=SUM.OF.SQUARE.ERROR/DFERROR
1050 REM This section prints out the ANOVA table.
1060 PRINT "   Source of       Degree of     Sum of          Mean                F
"
1070 PRINT "   variation        freedom      squares         squares"
```

```
1080 PRINT
1090 PRINT "   Replicates";TAB(24);DFR;TAB(34)
1100 PRINT USING "##.####";SUM.OF.SQUARE.R;:PRINT TAB(50) : PRINT USING "##.####
";SUM.OF.SQUARE.R/DFR;
1110 PRINT TAB(66) : PRINT USING "##.####";SUM.OF.SQUARE.R/(DFR*MEAN.SSE)
1120 PRINT
1130 PRINT "   Main effects"
1140 PRINT "         T ";:PRINT TAB(24) : PRINT DFT; :PRINT TAB(34) :PRINT USING
####.####";SSA;:PRINT TAB(50) : PRINT USING "####.####";SSA;:PRINT TAB(66) : PRI
NT USING "####.####";SSA/MEAN.SSE
1150 PRINT "         R ";: PRINT TAB(24) : PRINT DFT; : PRINT TAB(34) : PRINT USIN
G "####.####";SSB;: PRINT TAB(50) : PRINT USING "####.####";SSB;: PRINT TAB(66)
: PRINT USING "####.####";SSB/MEAN.SSE
1160 PRINT "         D ";: PRINT TAB(24) : PRINT DFT; : PRINT TAB(34) : PRINT USIN
G "####.####";SSC;: PRINT TAB(50) : PRINT USING "####.####";SSC;: PRINT TAB(66)
: PRINT USING "####.####";SSC/MEAN.SSE
1170 PRINT
1180 PRINT "   Two-factor"
1190 PRINT "    interactions"
1200 PRINT "        TR ";:PRINT TAB(24) : PRINT DFT; :PRINT TAB(34) :PRINT USING "
####.####";SSAB;:PRINT TAB(50) : PRINT USING "####.####";SSAB;:PRINT TAB(66) : P
RINT USING "####.####";SSAB/MEAN.SSE
1210 PRINT "        TD ";: PRINT TAB(24) : PRINT DFT; : PRINT TAB(34) : PRINT USIN
G "####.####";SSAC;: PRINT TAB(50) : PRINT USING "####.####";SSAC;: PRINT TAB(66
) : PRINT USING "####.####";SSAC/MEAN.SSE
1220  PRINT "       RD ";: PRINT TAB(24) : PRINT DFT; : PRINT TAB(34) : PRINT USI
NG "####.####";SSBC;: PRINT TAB(50) : PRINT USING "####.####";SSBC;: PRINT TAB(6
6) : PRINT USING "####.####";SSBC/MEAN.SSE
1230 PRINT
1240 PRINT "   Three-factor"
1250 PRINT "    interaction"
1260 PRINT "        TRD";:PRINT TAB(24) : PRINT DFT; :PRINT TAB(34) :PRINT USING "
####.####";SSABC;:PRINT TAB(50) : PRINT USING "####.####";SSABC;:PRINT TAB(66) :
 PRINT USING "####.####";SSABC/MEAN.SSE
1270 PRINT
1280 PRINT "   Error ";:PRINT TAB(24) : PRINT DFERROR; :PRINT TAB(34) :PRINT USI
NG "####.####";SUM.OF.SQUARE.ERROR;:PRINT TAB(50) : PRINT USING "####.####";MEAN
.SSE
1290 PRINT
1300 PRINT "   Total ";:PRINT TAB(24) : PRINT DFTOTAL; :PRINT TAB(34) :PRINT USI
NG "####.####";SUM.OF.SQUARE.T
1310 LPRINT CHR$(15)
1320 LPRINT "_________________________________________________________________
             "
-------------
1330 LPRINT "   Source of        Degree of      Sum of          Mean
s"
1340 LPRINT "   variation         freedom       squares         squares"
1350 LPRINT "_________________________________________________________________
             "
-------------
1360 LPRINT
1370 LPRINT "   Replicates";TAB(24);DFR;TAB(34)
1380 LPRINT USING "##.####";SUM.OF.SQUARE.R;:LPRINT TAB(50) :LPRINT USING "##.##
##";SUM.OF.SQUARE.R/DFR;
1390 LPRINT TAB(66) :LPRINT USING "##.####";SUM.OF.SQUARE.R/(DFR*MEAN.SSE)
1400 LPRINT "_________________________________________________________________
             "
-------------
```

```
1410 LPRINT
1420 LPRINT "   Main effects"
1430 LPRINT "        T ";:LPRINT TAB(24) :LPRINT DFT; :LPRINT TAB(34) :LPRINT USI
NG "####.####";SSA;:LPRINT TAB(50) :LPRINT USING "####.####";SSA;:LPRINT TAB(66)
 :LPRINT USING "####.####";SSA/MEAN.SSE
1440 LPRINT "        R ";:LPRINT TAB(24) :LPRINT DFT; :LPRINT TAB(34) :LPRINT USI
NG "####.####";SSB;:LPRINT TAB(50) :LPRINT USING "####.####";SSB;:LPRINT TAB(66)
 :LPRINT USING "####.####";SSB/MEAN.SSE
1450 LPRINT "        D ";:LPRINT TAB(24) :LPRINT DFT; :LPRINT TAB(34) :LPRINT USI
NG "####.####";SSC;:LPRINT TAB(50) :LPRINT USING "####.####";SSC;:LPRINT TAB(66)
 :LPRINT USING "####.####";SSC/MEAN.SSE
1460 LPRINT "-------------------------------------------------------------------
------------"
1470 LPRINT
1480 LPRINT "   Two-factor"
1490 LPRINT "    interactions"
1500 LPRINT "        TR ";:LPRINT TAB(24) :LPRINT DFT; :LPRINT TAB(34) :LPRINT USI
NG "####.####";SSAB;:LPRINT TAB(50) :LPRINT USING "####.####";SSAB;:LPRINT TAB(6
6) :LPRINT USING "####.####";SSAB/MEAN.SSE
1510 LPRINT "        TD ";:LPRINT TAB(24) :LPRINT DFT; :LPRINT TAB(34) :LPRINT USI
NG "####.####";SSAC;:LPRINT TAB(50) :LPRINT USING "####.####";SSAC;:LPRINT TAB(6
6) :LPRINT USING "####.####";SSAC/MEAN.SSE
1520 LPRINT "        RD ";:LPRINT TAB(24) :LPRINT DFT; :LPRINT TAB(34) :LPRINT USI
NG "####.####";SSBC;:LPRINT TAB(50) :LPRINT USING "####.####";SSBC;:LPRINT TAB(6
6) :LPRINT USING "####.####";SSBC/MEAN.SSE
1530 LPRINT "-------------------------------------------------------------------
------------"
1540 LPRINT
1550 LPRINT "   Three-factor"
1560 LPRINT "    interaction"
1570 LPRINT "        TRD";:LPRINT TAB(24) : LPRINT DFT; :LPRINT TAB(34) :LPRINT US
ING "####.####";SSABC;:LPRINT TAB(50) : LPRINT USING "####.####";SSABC;:LPRINT T
AB(66) : LPRINT USING "####.####";SSABC/MEAN.SSE
1580 LPRINT "-------------------------------------------------------------------
------------"
1590 LPRINT
1600 LPRINT "   Error ";:LPRINT TAB(24) : LPRINT DFERROR; :LPRINT TAB(34) :LPRIN
T USING "####.####";SUM.OF.SQUARE.ERROR;:LPRINT TAB(50) : LPRINT USING "####.###
#";MEAN.SSE
1610 LPRINT "-------------------------------------------------------------------
------------"
1620 LPRINT
1630 LPRINT "   Total ";:LPRINT TAB(24) :LPRINT DFTOTAL; :LPRINT TAB(34) :LPRINT
 USING "####.####";SUM.OF.SQUARE.T
1640 LPRINT "-------------------------------------------------------------------
------------"
1650 END
```

APPENDIX D

Original data and their transformations

1. Original data

a. Proportion correct (accuracy)

treat-ment 1	2	3	4	5	6	7	8
.9333	.6667	.6333	.5000	.9333	.5333	.6000	.5667
.8667	.6667	.7667	.4667	.9333	.5333	.6000	.7333
.9333	.7333	.7000	.2667	.9333	.6000	.6333	.5667
.9333	.6000	.7667	.3667	.9333	.6000	.6000	.6000

b. Response time

treat-ment 1	2	3	4	5	6	7	8
9.2212	6.9885	8.3034	11.5154	1.5827	8.5861	2.5991	4.1378
14.2562	12.8275	41.8292	21.7393	36.1521	1.3113	27.8834	9.9906
9.9605	10.2221	4.0822	39.8057	3.5398	7.2864	3.5463	53.3504
2.6984	7.4985	7.0311	9.6616	3.9941	8.1041	7.3763	14.2741

2. Transformed data for proportion correct:

treatment

1	2	3	4	5	6	7	8
2.6390	.6931	.5465	.0000	2.6390	.1335	.4055	.2682
1.8717	.6931	1.1895	-.1336	2.6390	.1335	.4055	1.0116
2.6390	1.0116	.8473	-1.0116	2.6390	.4055	.5465	.2682
2.6390	.4055	1.1895	-.5466	2.6390	.4055	.4055	.4055

APPENDIX E

SIPS output

```
$READ DATA1,1-8

* THIS COMPARES TREATMENT 1 AND 5 - POLYGON TO LINE
$COMPARE,1,5

                     VARIABLES      1               5
SAMPLE SIZE                             4               4
MEAN                              2.44718         2.63900
STD. ERR OF MEAN                   .19183          .00000
VARIANCE                           .14719          .00000
STANDARD DEVIATION                 .38365          .00000
RANGE                              .76730          .00000

T-VALUE                  =      -1.000000
DEGREES OF FREEDOM       =              6

T-TABLE VALUE AT (.95) =         2.446866
T-TABLE VALUE AT (.99) =         3.706605

* THIS COMPARES TREATMENT 2 AND 5 - POLYGON TO LINE
$COMPARE,2,6

                     VARIABLES      2               6
SAMPLE SIZE                             4               4
MEAN                               .70083          .26950
STD. ERR OF MEAN                   .12380          .07852
VARIANCE                           .06131          .02466
STANDARD DEVIATION                 .24760          .15704
RANGE                              .60610          .27200

T-VALUE                  =       2.942175
DEGREES OF FREEDOM       =              6

T-TABLE VALUE AT (.95) =         2.446866
T-TABLE VALUE AT (.99) =         3.706605

* THIS COMPARES TREATMENT 3 AND 6 - POLYGON TO LINE
$COMPARE,3,7

                     VARIABLES      3               7
SAMPLE SIZE                             4               4
MEAN                               .94363          .44075
STD. ERR OF MEAN                   .15471          .03525
VARIANCE                           .09574          .00497
STANDARD DEVIATION                 .30942          .07050
RANGE                              .64300          .14100

T-VALUE                  =       3.169167
DEGREES OF FREEDOM       =              6

T-TABLE VALUE AT (.95) =         2.446866
T-TABLE VALUE AT (.99) =         3.706605

* THIS COMPARES TREATMENT 4 AND 8 - POLYGON TO LINE
$COMPARE,4,8

                     VARIABLES      4               8
SAMPLE SIZE                             4               4
MEAN                              -.42295          .48838
STD. ERR OF MEAN                   .22811          .17739
VARIANCE                           .20814          .12586
STANDARD DEVIATION                 .45622          .35477
RANGE                             1.01160          .74340

T-VALUE                  =      -3.153778
DEGREES OF FREEDOM       =              6

T-TABLE VALUE AT (.95) =         2.446866
T-TABLE VALUE AT (.99) =         3.706605
```

```
$READ,DATA3,1-4
* THIS COMPARES SHIP DENSITY - UNDER DATA RATE = 15
$COMPARE,1,2

                        VARIABLES    1              2
SAMPLE SIZE                              8              8
MEAN                               2.54309         .48516
STD. ERR OF MEAN                    .09591         .10606

VARIANCE                            .07359         .09000
STANDARD DEVIATION                  .27128         .30000
RANGE                               .76730         .87810

T-VALUE                  =      14.391113
DEGREES OF FREEDOM       =             14

T-TABLE VALUE AT (.95) =         2.144830
T-TABLE VALUE AT (.99) =         2.977075

* THIS COMPARES SHIP DENSTY - UNDER DATA RATE = 30
$COMPARE,3,4

                        VARIABLES    3              4
SAMPLE SIZE                              8              8
MEAN                                .69219         .03271
STD. ERR OF MEAN                    .12011         .21807
VARIANCE                            .11542         .38043
STANDARD DEVIATION                  .33973         .61679
RANGE                               .78400        2.02320

T-VALUE                  =       2.648924
DEGREES OF FREEDOM       =             14

T-TABLE VALUE AT (.95) =         2.144830
T-TABLE VALUE AT (.99) =         2.977075

* THIS COMPARES DATA RATE - UNDER SHIP DENSITY = 4
$COMPARE,1,3

                        VARIABLES    1              3
SAMPLE SIZE                              8              8
MEAN                               2.54309         .69219
STD. ERR OF MEAN                    .09591         .12011
VARIANCE                            .07359         .11542
STANDARD DEVIATION                  .27128         .33973
RANGE                               .76730         .78400

T-VALUE                  =      12.041665
DEGREES OF FREEDOM       =             14

T-TABLE VALUE AT (.95) =         2.144830
T-TABLE VALUE AT (.99) =         2.977075

* THIS COMPARES DATA RATE - UNDER SHIP DENSITY = 8
$COMPARE,2,4

                        VARIABLES    2              4
SAMPLE SIZE                              8              8
MEAN                                .48516         .03271
STD. ERR OF MEAN                    .10606         .21807
VARIANCE                            .09000         .38043
STANDARD DEVIATION                  .30000         .61679
RANGE                               .87810        2.02320

T-VALUE                  =       1.865814
DEGREES OF FREEDOM       =             14

T-TABLE VALUE AT (.95) =         2.144830
T-TABLE VALUE AT (.99) =         2.977075
```

```
$READ,TIMET,1-2
* THIS COMPARES DISPLAY TYPE - RESPONSE TIME
$COMPARE,1,2
                         VARIABLES     1              2
SAMPLE SIZE                           16             16
MEAN                            13.66473       12.10604
STD. ERR OF MEAN                 2.86520        3.65444
VARIANCE                       131.35008      213.67898
STANDARD DEVIATION              11.46081       14.61776
RANGE                           39.12980       52.03910

T-VALUE                  =         .335654
DEGREES OF FREEDOM       =              30

T-TABLE VALUE AT (.95) =          2.042312
T-TABLE VALUE AT (.99) =          2.750199

$READ,TIMER,1-2
* THIS COMPARES DATA RATE - RESPONSE TIME
$COMPARE,1,2



                         VARIABLES     1              2
SAMPLE SIZE                           16             16
MEAN                             9.07549       16.69527
STD. ERR OF MEAN                 2.04561        3.94023
VARIANCE                        66.95217      248.40677
STANDARD DEVIATION               8.18243       15.76093
RANGE                           34.84080       50.75130

T-VALUE                  =       -1.716325
DEGREES OF FREEDOM       =              30

T-TABLE VALUE AT (.95) =          2.042312
T-TABLE VALUE AT (.99) =          2.750199

$READ,TIMED,1-2
* THIS COMPARES SHIP DENSITY - RESPONSE TIME
$COMPARE,1,2
                         VARIABLES     1              2
SAMPLE SIZE                           16             16
MEAN                            11.50324       18.01755
STD. ERR OF MEAN                 3.13005        4.69217
VARIANCE                       156.75537      352.26284
STANDARD DEVIATION              12.52020       18.76867
RANGE                           40.24550       65.67720

T-VALUE                  =       -1.154947
DEGREES OF FREEDOM       =              30

T-TABLE VALUE AT (.95) =          2.042312
T-TABLE VALUE AT (.99) =          2.750199
```