NAVAL POSTGRADUATE SCHOOL

Monterey, California

THESIS

INVESTIGATION OF INITIAL DETECTION MODELS IN THE SEARCH AND LOCALIZATION TACTICAL DECISION AID (SALT)

by

Vance S. Tisdale

September, 1990

Thesis Advisor: Alan R. Washburn

Approved for public release; distribution is unlimited

REPORT DOCUMENTATION PAGE

1a Report Security Classification Unclassified	1b Restrictive Markings
2a Security Classification Authority	3 Distribution Availability of Report Approved for public release; distribution is unlimited.
2b Declassification Downgrading Schedule	
4 Performing Organization Report Number(s)	5 Monitoring Organization Report Number(s)
6a Name of Performing Organization Naval Postgraduate School — 6b Office Symbol *(if applicable)* OR	7a Name of Monitoring Organization Naval Postgraduate School
6c Address *(city, state, and ZIP code)* Monterey, CA 93943-5000	7b Address *(city, state, and ZIP code)* Monterey, CA 93943-5000
8a Name of Funding Sponsoring Organization — 8b Office Symbol *(if applicable)*	9 Procurement Instrument Identification Number
8c Address *(city, state, and ZIP code)*	10 Source of Funding Numbers Program Element No | Project No | Task No | Work Unit Accession No

11 Title *(include security classification)* INVESTIGATION OF INITIAL DETECTION MODELS IN THE SEARCH AND LOCALIZATION TACTICAL DECISION AID (SALT)

12 Personal Author(s) Vance S. Tisdale

13a Type of Report	13b Time Covered	14 Date of Report *(year, month, day)*	15 Page Count
Master's Thesis	From To	September 1990	53

16 Supplementary Notation The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

17 Cosati Codes			18 Subject Terms *(continue on reverse if necessary and identify by block number)*
Field	Group	Subgroup	Computer Assisted Search, Antisubmarine Warfare, Tactical Decision Aids

19 Abstract *(continue on reverse if necessary and identify by block number)*

The goal of this thesis is to investigate the initial search planning phase of the Search and Localization Tactical Decision Aid (SALT) developed by METRON, Incorporated of McLean, Virginia. SALT is a Computer Assited Search (CAS) program intended for use by P3 UPDATE IV crews to assist them in optimal deployment of a sonobuoy field to prosecute a submarine threat.

The initial search planning phase of SALT takes as user inputs environmental data, an initial elliptical Search Probability Area (SPA), an assumed target motion model, and the duration of the search. Outputs include a recommended sonobuoy pattern and the probability of detection of this pattern. The investigation of this phase of the algorithm is conducted in two parts. First, a series of simulation routines is used to ensure that the probability of detection of the sonobuoy patterns generated by SALT is mathematically correct. Second, these same simulation routines are used to determine if there are alternate sonobuoy patterns that result in higher probabilities of detection.

20 Distribution Availability of Abstract ☒ unclassified unlimited ☐ same as report ☐ DTIC users	21 Abstract Security Classification Unclassified
22a Name of Responsible Individual Alan R. Washburn	22b Telephone *(include Area code)* (408) 646-3127 — 22c Office Symbol OR WS

DD FORM 1473,84 MAR — 83 APR edition may be used until exhausted. All other editions are obsolete

Approved for public release; distribution is unlimited.

Investigation of Initial Detection Models in the Search and Localization Tactical Decision Aid (SALT)

by

Vance S. Tisdale
Lieutenant, United States Navy
B.S., United States Naval Academy, 1983

Submitted in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE IN OPERATIONS RESEARCH

from the

NAVAL POSTGRADUATE SCHOOL
September 1990

Author: ______________________
Vance S. Tisdale

Approved by: ______________________
Alan R. Washburn, Thesis Advisor

____
James N. Eagle, Second Reader

____
Peter Purdue, Chairman,
Department of Operations Research

ABSTRACT

The goal of this thesis is to investigate the initial search planning phase of the Search and Localization Tactical Decision Aid (SALT) developed by METRON, Incorporated of McLean, Virginia. SALT is a Computer Assisted Search (CAS) program intended for use by P3 UPDATE IV crews to assist them in optimal deployment of a sonobuoy field to prosecute a submarine threat.

The initial search planning phase of SALT takes as user inputs environmental data, an initial elliptical Search Probability Area (SPA), an assumed target motion model, and the duration of the search. Outputs include a recommended sonobuoy pattern and the probability of detection of this pattern. The investigation of this phase of the algorithm is conducted in two parts. First, a series of simulation routines is used to ensure that the probability of detection of the sonobuoy patterns generated by SALT is mathematically correct. Second, these same simulation routines are used to determine if there are alternate sonobuoy patterns that result in higher probabilities of detection.

THESIS DISCLAIMER

The reader is cautioned that computer programs developed in this research may not have been exercised for all cases of interest. While every effort has been made, within the time available, to ensure that the programs are free of computational and logic errors, they cannot be considered validated. Any application of these programs without additional verification is at the risk of the user.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

I. INTRODUCTION

Tactical Decision Aids (TDA's) have been developed and implemented on computers in various forms to assist users in making decisions on a wide variety of topics. One prominent area where TDA's have proven to be of value is in the area of Antisubmarine Warfare (ASW). In the search theory phase of the ASW problem, in particular, there is an extremely large amount of data which must be assimilated and presented to the user in a manner conducive to rapid evaluation so that a sound decision can be made and acted upon. Several TDA's have been developed to aid the user in the area of search theory. These TDA's have been grouped together under the title Computer Assisted Search (CAS). A common thread which runs through all of the CAS programs is the concept of a "probability map". It is this probability map which allows all of the input data, mathematical calculations, and time step updates to be output as a 2-dimensional graphic display that is easily understood by the user in terms of relative probabilities on the location of the target of interest.

Many CAS programs represent the stochastic aspect of target motion by a Monte Carlo simulation of a large number of possible target tracks. As information, either positive (detection) or negative (no detection after a period of time), is obtained from the search, the probabilities of the individual tracks are updated and displayed as a probability map at the user requested time. The most prominent TDA's of this type are VPCAS and PACSEARCH. This type of TDA possesses an inherent simplicity which promotes ease of understanding of the underlying concepts. The use of the Theorem of Total Probability and Bayes' Theorem are easily applied to small scale problems which can be worked by hand and the results naturally generalize to large scale Computer Assisted Search problems. The difficulty in this type of approach is that the specific tracks selected for the probability map do not cover all of the possibilities of target motion. [Ref. 1: pp. II-1,II-2]

In an attempt to negate the above difficulty with the Monte Carlo approach to target motion, a new algorithm entitled *Search and Localization Tactical Decision Aid* (SALT) has been developed by METRON, Incorporated of McLean, Virginia. SALT is a CAS algorithm intended for use by P3 UPDATE IV crews to assist them in the optimal deployment of a sonobuoy field to prosecute a submarine threat. The SALT algorithm uses a discrete time and space Markov Chain to simulate target motion. The state space of this Markov Chain is a five element vector (position, velocity, depth). The number of discrete cells in the state space is 21 entries each for latitude and longitude, eight entries each for north-south and east-west velocity components, and two entries for depth. Thus the current state of the target is represented probabilistically as 56,448 five-dimensional cells. This analytic approach to solving the problem of unknown target motion allows for many more possibilities of target motion than the simplistic track technique used in previous CAS models. [Ref. 1: pp. II-46, II-47]

The initial search planning phase of SALT takes as user inputs environmental data, an elliptical Search Probability Area (SPA), an assumed target motion model, and duration of search. The algorithm then outputs a recommended sonobuoy deployment pattern along with the probability of detection of this pattern. [Ref. 2: p. 1] This initial search planning phase is the topic which will be investigated here. The investigation consists of two parts. First, the sonobuoy patterns recommended by SALT will be tested using FORTRAN simulations to ensure that the probability of detection for each pattern as output by SALT is mathematically accurate. Second, alternative sonobuoy patterns will be explored with the goal of developing patterns that improve on the SALT probability of detection.

We shall begin in Chapter II by discussing the three types of motion models used by SALT to develop initial probability maps. Chapter III follows with a presentation of how buoy patterns are developed and evaluated based on the input data for each type of target motion. Chapter IV provides the documentation of the FORTRAN simulations that are used to verify the probability of detection for the SALT generated patterns. In Chapter V, the results obtained from the FORTRAN simulation will be presented and compared to the output from the SALT algorithm. Chapter VI explores the use of alternative sonobuoy patterns with the goal of developing patterns which will improve the probability of detection of those patterns currently used in the SALT program. Chapter VII is a summary of the results. The inputs to the SALT algorithm used to obtain the sonobuoy patterns for testing are listed in Appendix A. The source code for examples of the FORTRAN simulation routines is presented in Appendix B.

II. SALT MOTION MODELS

There are three different models of target motion built in to the SALT algorithm. These models are a patrolling target, a transiting target, and a fleeing datum target. The specifics of each type of target motion are discussed below. In general, each scenario begins with an elliptical SPA which is input by the user. To specify a SPA requires seven input parameters: date-time-group, latitude and longitude of the SPA center, semi-major axis (nm), semi-minor axis (nm), orientation (deg N), and containment percent. This SPA represents a Bivariate Normal distribution for the target's location that has been obtained from an external source. [Ref. 3: pp. 12, 21]

A. PATROLLING TARGET MODEL

In the patrolling target model, the target is assumed to be randomly patrolling in the vicinity of the SPA. The target selects an initial course from a Uniform (0, 360) distribution and an initial speed from a truncated triangular distribution. This triangular distribution takes as input the average target speed. The distribution ranges from half of the input speed to two times the input speed with the average speed being eight times as likely as either of the endpoints. These distributions are shown in Figure 1 below.

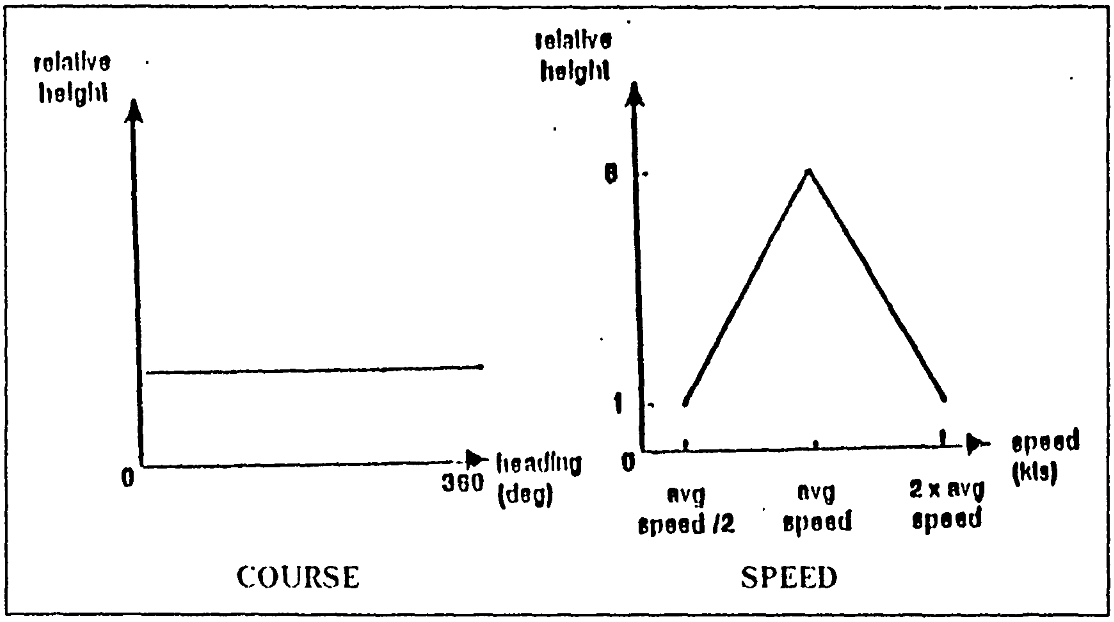

Figure 1. Patrolling Target Course and Speed Distributions

The target is assumed to continue on this course and speed for an amount of time determined from an exponential distribution with mean time between course changes being input by the user. At the completion of this leg in the target's track, an independent course, speed, and time on leg are selected from the same distributions and the process continues throughout the scenario. [Ref. 3: pp. 10-11, 20]

The values of the input parameters for a patrolling target were set as follows to obtain SALT output for testing:

- Average target speed (knots) 4
- Average time between course changes (hours) 3

B. TRANSITING TARGET MODEL

The transiting target motion model assumes that the target has a preferred course and speed of advance (SOA) to travel. In this model, the user is required to input five parameters. The parameters are average target course (deg N), maximum variation in target course, average SOA (kts), maximum variation in SOA, and average time between course changes. For each leg, the target selects a course from a truncated triangular distribution ranging over the values average target course ± maximum variation in target course with the average target course in the center being eight times as likely as either of the endpoints. The same type of distribution is used for selecting the target's speed of advance in the direction of the average target course on each leg. These distributions are shown in Figure 2 below.

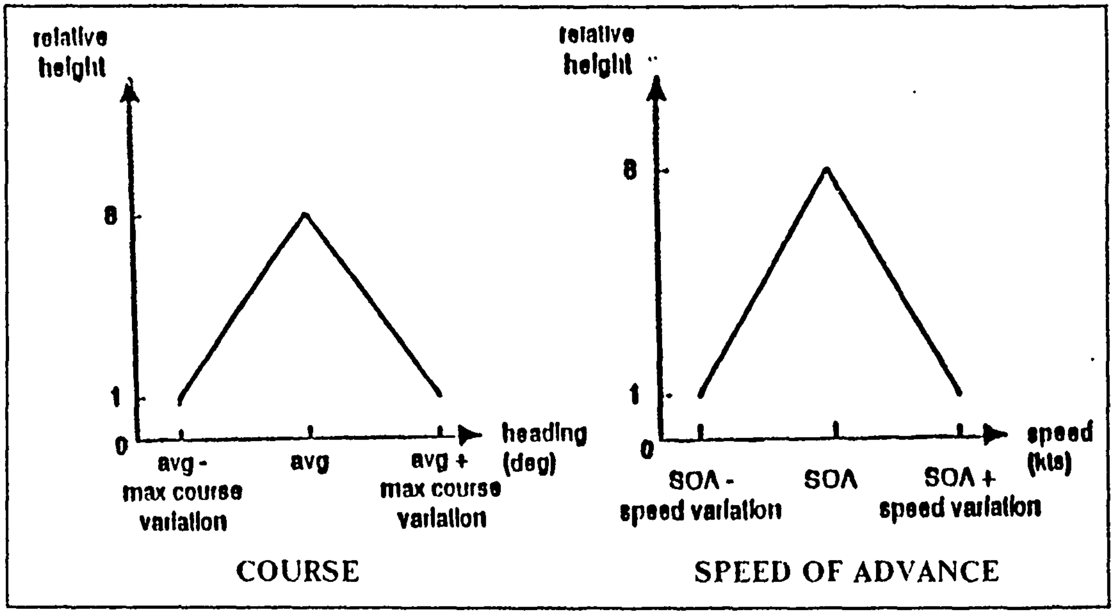

Figure 2. Transiting Target Course and Speed of Advance Distributions

Each leg is assumed to be maintained for an exponential amount of time as discussed in the Patrolling Target Model section above. [Ref. 3: pp. 10-11, 20]

The values of the input parameters for a transiting target were set as follows to obtain SALT output for testing:

• Average SOA (knots)	12
• Maximum variation in SOA (knots)	4
• Average target course (deg N)	045
• Maximum variation in target course (deg)	30
• Average time between course changes (hours)	6

C. FLEEING DATUM TARGET MODEL

The model for fleeing datum target motion assumes that the target is clearing datum from an area where it suspects it has been detected. To this end, the target selects a course and speed from the specified distributions and maintains that course and speed throughout the encounter. The target course is a draw from a Uniform (0, 360) distribution. The target speed is selected from a truncated triangular distribution with four input parameters. The parameters to be input are average speed, minimum speed, maximum speed, and average speed likelihood factor. The first three parameters are self-explanatory, and the average speed likelihood factor determines how many times more likely the average speed is over the endpoint speeds. These distributions are shown in Figure 3 on page 6.

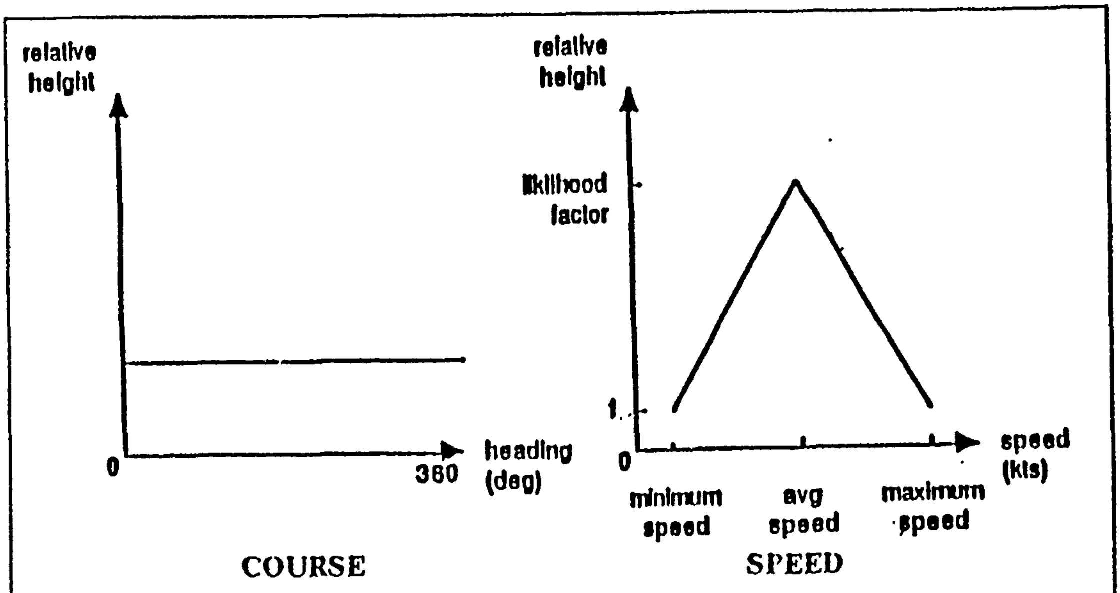

Figure 3. Fleeing Datum Target Course and Speed Distributions

Since the target maintains course and speed the entire time, there is no requirement for a mean time between course changes parameter. [Ref. 3: pp. 10-11, 20]

The values of the input parameters for a fleeing datum target were set as follows to obtain SALT output for testing:

• Average target speed (knots)	12
• Maximum speed (knots)	15
• Minimum speed (knots)	9
• Average speed likelihood factor	8

This completes the discussion of the three types of target motion models available for user selection in the SALT algorithm. The next chapter will discuss the development and evaluation of sonobuoy patterns to be deployed against each specific type of target motion.

III. SALT SONOBUOY PATTERNS

The sonobuoy patterns recommended by SALT for each type of target motion model are presented here in brief overview form. For each type of target motion, both the pattern development and evaluation of probability of detection of that pattern are discussed. The overall objective of the sonobuoy patterns developed by the initial search planning phase of SALT is to maximize the probability of detecting the target.

A. PATROLLING TARGET PATTERNS

A target which is patrolling in an area as discussed in the previous chapter will have an average velocity of zero and will tend to remain in the vicinity of the initial Search Probability Area (SPA). The optimal search density along each of the axes of the bivariate normal distribution would be parabolic. The optimization of search effort would require solving a very complicated nonlinear function and the density of search effort required would be very difficult to obtain with sonobuoys. In order to overcome the difficulties associated with attempting to optimize the search effort, the SALT algorithm uses an approximation to the optimal search effort. The size of a rectangle which most closely approximates the bivariate normal distribution of target location is first determined. The available search effort is then distributed uniformly over this rectangle. This process is described in more detail below. [Ref. 2: p 36-37]

As the search for the target progresses without any detections, the elliptical area of uncertainty of the initial SPA continues to expand. To capture a static value for the size of the SPA to use in determining and evaluating the sonobuoy deployment pattern, the initial SPA is first expanded to correspond to the time which is half way through the input search duration. This expansion gives larger values for σ_1 and σ_2 which represent the average growth in uncertainty in the targets position over the duration of the search. [Ref. 2: p. 20-22]

The bivariate normal distribution determined by the SPA, expanded as described above, can be expressed as the product of two independent univariate normal distributions defined on the principal axes of the SPA. Let σ_1 and σ_2 be the standard deviation of the target distribution along the first and second axes. The lengths of the sides of the rectangle parallel to these axes are $2K\sigma_1$ and $2K\sigma_2$ respectively for some constant K. These values determine a family of rectangles with the parameter K determining the size of the rectangle as illustrated in Figure 4 below. [Ref. 2: pp. 37-38]

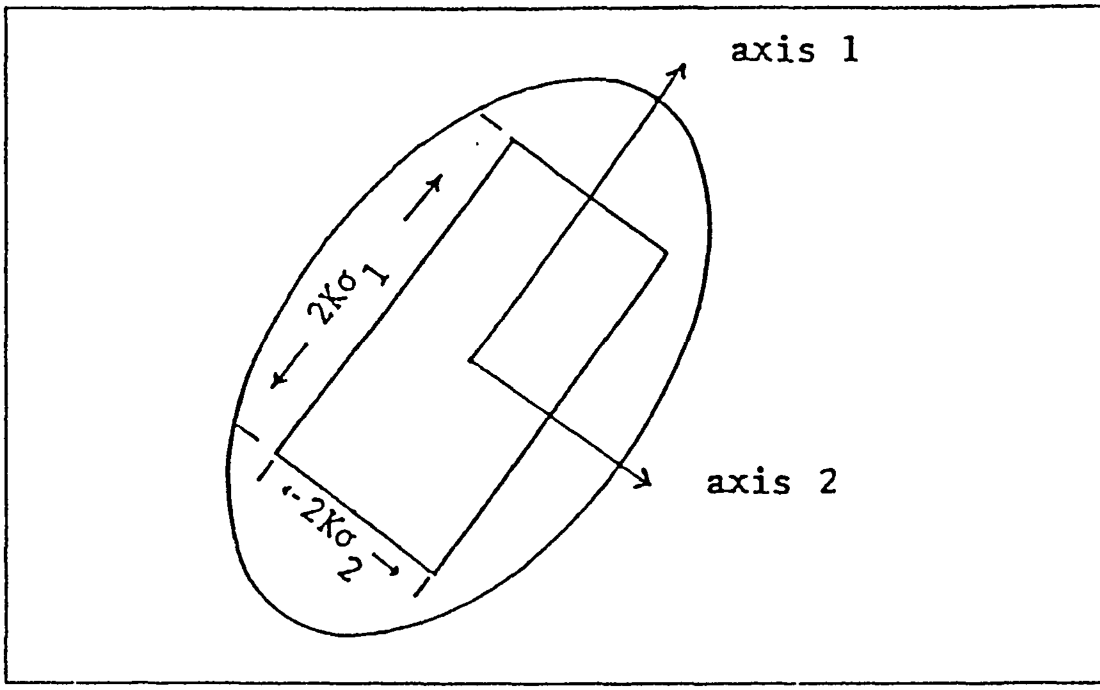

Figure 4. Family of Rectangle Plans

To determine the probability of detection for a rectangular pattern of a given size, it is necessary to multiply the probability that the target is located in the rectangle by the conditional probability that the target is detected given that it is in the rectangle. This conditional probability is approximated by an exponential detection function which is based on the pessimistic assumptions of a random search [Ref. 4: p. 28]. This idea of removing the conditioning is used by the initial search planning phase of SALT to determine that value of K for which the probability of detection calculated by the following equation is maximized.

$$P_D(K) = [\Phi(K) - \Phi(-K)]^2[1 - \exp(\frac{-nWVT}{4K^2\sigma_1\sigma_2})] \tag{3.1}$$

where:

n = the number of sonobuoys

W = the sweep width of each sonobuoy in nautical miles

V = average speed of target as input by the user

T = search duration in hours

This maximum value is output by SALT as the probability of detection for the recommended sonobuoy pattern. [Ref. 2: pp. 39-42]

It should be noted here that the term nWVT in equation (3.1) can be thought of as the total area covered by the pattern during time period T due to target motion through the sonobuoy field. We will later argue that a more accurate probability of detection can be calculated by accounting for the possibility of detecting the target at the beginning of the search.

To develop this operationally feasible sonobuoy pattern, SALT uses the optimally sized rectangle calculated above and distributes the number of sonobuoys available over that rectangle to approximate a uniform distribution of search effort. To accomplish this distribution of effort, the sonobuoys are positioned in equally spaced rows with the spacing between the buoys in a row being equal. If the number of sonobuoys is not divisible by the number of rows, remaining buoys are allotted to rows beginning with the center row and working outward. The sonobuoy pattern is output to the screen graphically along with the estimate of probability of detection. [Ref. 2: pp. 43-50]

An example of the graphical sonobuoy pattern output for a scenario involving 16 sonobuoys deployed against a patrolling target is presented in Figure 5 on page 10. The inner ellipse represents the initial SPA and the outer ellipse represents the midsearch duration ellipse used to perform the above calculations. The optimally sized rectangle is also displayed in the pattern.

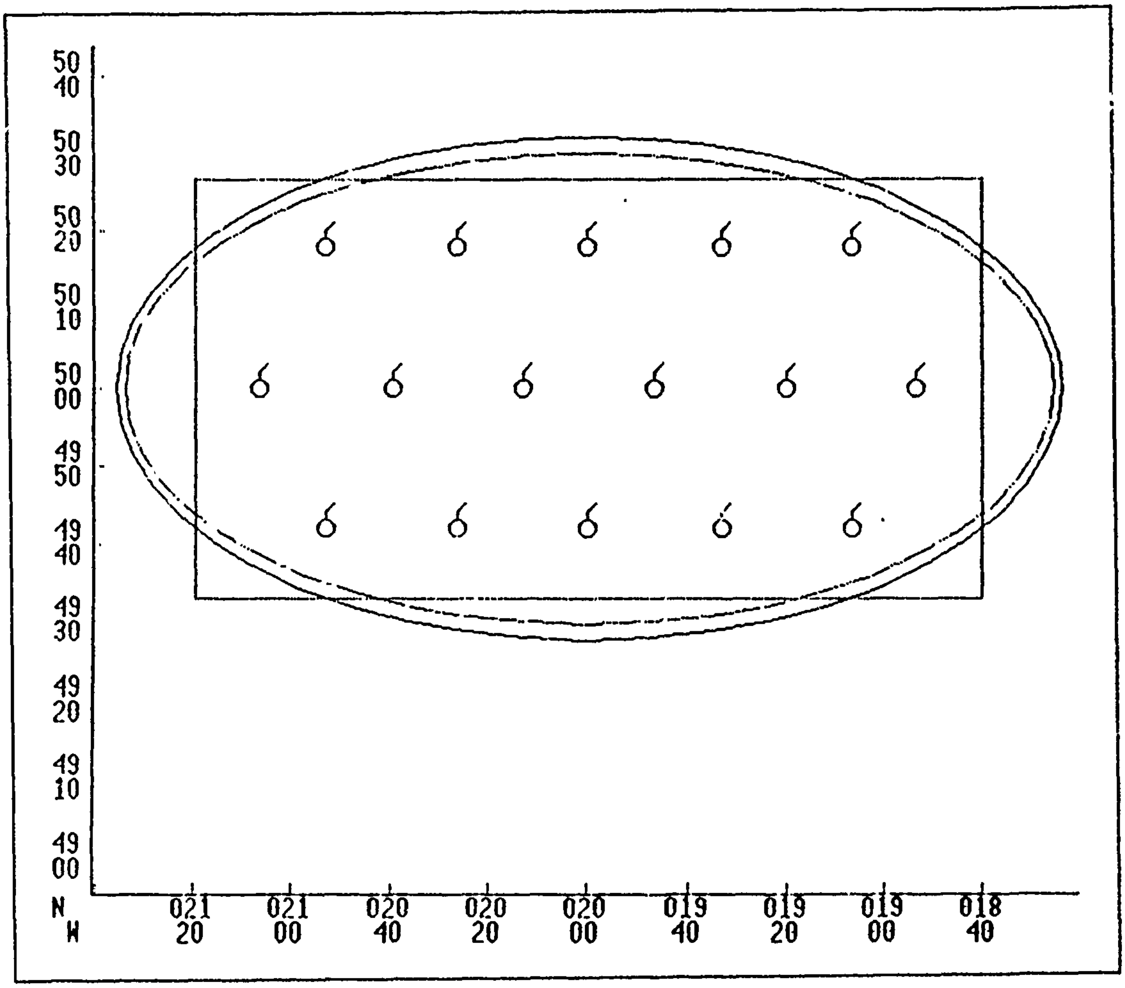

Figure 5. Recommended Pattern for 16 Sonobuoys vs. a Patrolling Target

B. TRANSITING TARGET PATTERNS

A target which is assumed to be transisting has a preferred direction of motion which can be exploited by the search planning algorithm by using barriers which "walk" against the target motion. This fact makes it possible for a line of sonobuoys to sweep out a large rectangular area. The size of the area swept depends on target speed in the direction of the barrier. The development and evaluation of sonobuoy patterns for deployment against a transiting target is similar to that discussed in the previous section for patrolling targets. The major difference in the two developments is that the transiting target search plans are constructed in the relative motion space of the mean velocity of the target and then translated to geographic space. [Ref. 2: pp. 50-59]

An example of the graphical output of the recommended deployment pattern for 16 sonobuoys against a transiting target with input parameters as discussed in Chapter II is shown in Figure 6 below. The ellipse displayed in the figure represents the initial SPA. The ellipse which is updated to the midsearch duration for use in performing calculations is not shown.

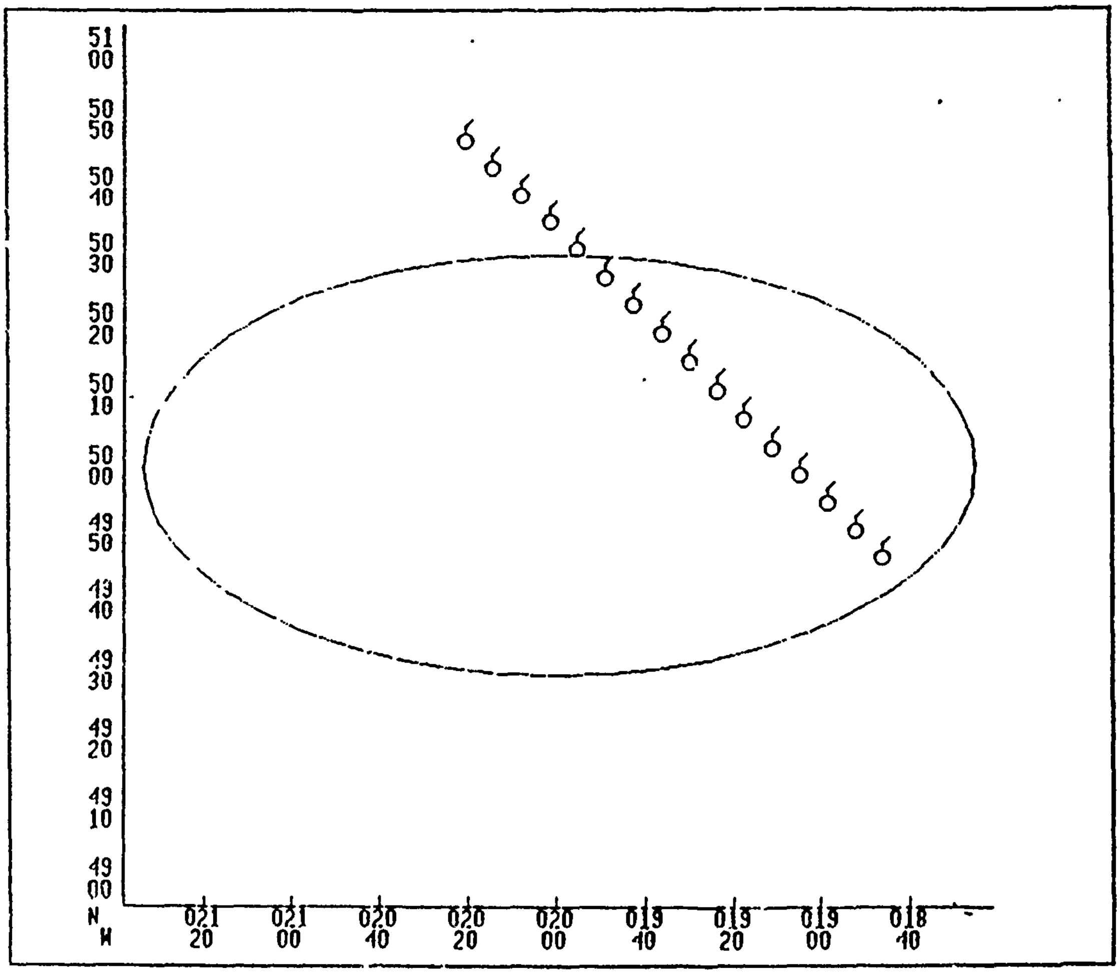

Figure 6. Recommended Pattern for 16 Sonobuoys vs. a Transiting Target

C. FLEEING DATUM TARGET PATTERNS

The development of a sonobuoy pattern to be deployed against a fleeing datum target begins by updating the initial SPA to the mid-search time. This update is accomplished using a discrete time and space Markov chain. As the search progresses, the distribution of target location spreads radially outward eventually becoming donut shaped with the mean, or center, of the SPA becoming a low probability area. [Ref. 2: p. 62]

In order to accommodate the distribution for a fleeing datum target, SALT uses a class of search plans referred to as "two rectangle plans". These plans are defined by an inner and an outer rectangle oriented concentrically such that the sides of the rectangles are parallel to the major and minor axes of the SPA. The total search effort is spread uniformly over the annular region between these two rectangles. The stated objective of SALT is to select the sizes of these rectangles so as to maximize the probability of detecting the target. [Ref. 2: p. 62]

The probability of detection for these fleeing datum sonobuoy patterns is computed in a manner similar to the calculation discussed above for a patrolling target. The probability that the target is contained in the two rectangle annulus is multiplied by the conditional probability of detecting the target given that it is contained in this region. This conditional probability is determined by an exponential detection function as was done in the patrolling target case. [Ref. 2: p. 65]

An example of this type of pattern for 16 sonobuoys as output by SALT is shown in Figure 7 on page 13. The sonobuoys in this pattern are located so far from the initial SPA due to the adjustment of an input parameter to the SALT scenario known as time-late which is the elapsed time between when the SPA was obtained and the time at which the search begins. In this scenario, time-late has been set to four hours.

In performing the above calculations of probability of detection for the sonobuoy patterns to be deployed against any of the three types of assumed target motion, SALT assumes that the sonobuoys are all dropped in place instantly at the beginning of the search. This assumption does not adversely affect the results of either the patrolling target or the transiting target. In the case of the patrolling target, the target tends to remain in the search area for the duration of the search thus being exposed to the entire pattern as if it were laid instantaneously. The transiting target sonobuoy patterns are generally short, straight line patterns that do not take a long time to lay. This fact combined with the time required for the target to intercept the pattern tend to make this assumption plausible. The sonobuoy patterns deployed against the fleeing datum target, however, are usually laid a large distance from the initial target SPA with large spacing between the sonobuoys. Due to the length of time required to lay these patterns and the higher speeds associated with fleeing datum targets, the assumption of having the sonobuoys in place initially is faulty. During the data collection runs of the fleeing datum target scenarios, the target was sometimes seen to be beyond the reach of the pat-

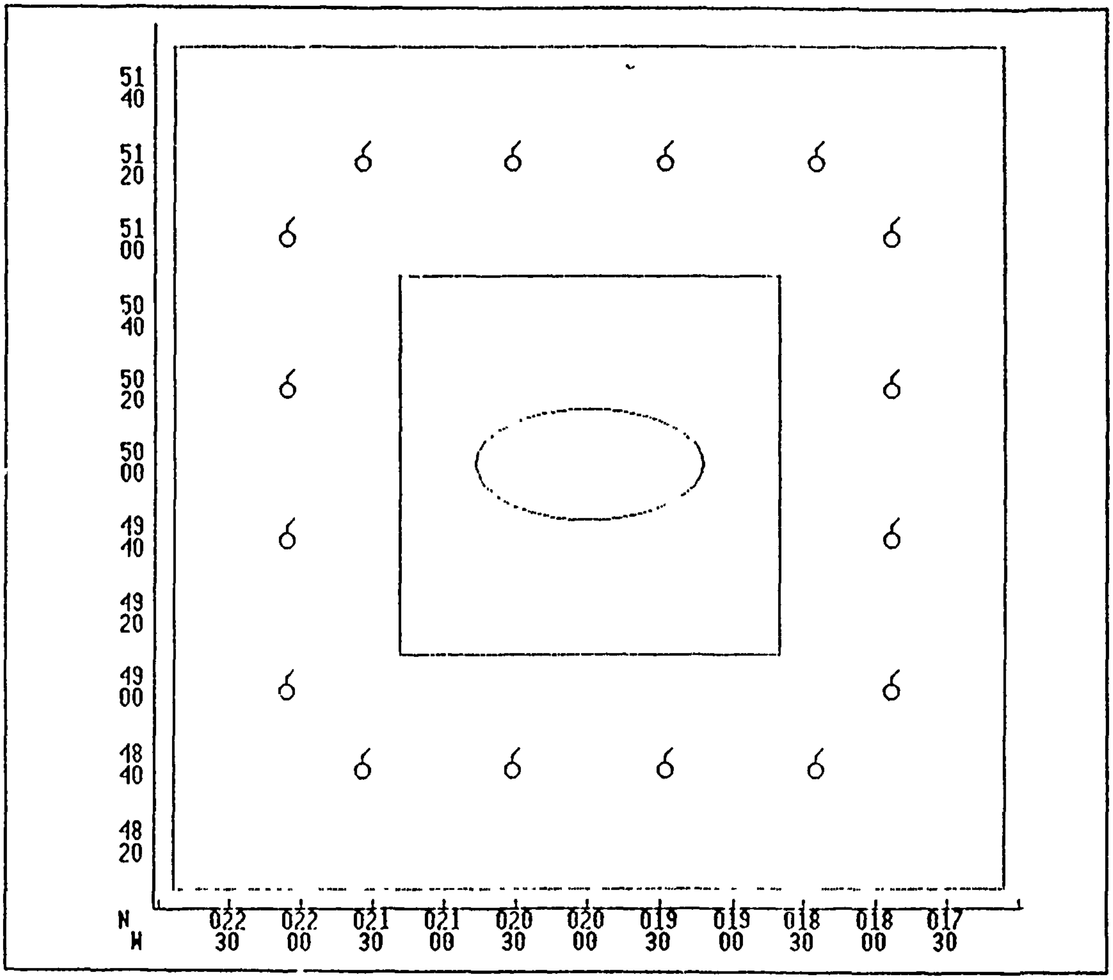

Figure 7. Recommended Pattern for 16 Sonobuoys vs. a Fleeing Datum Target

tern by the time the critical portion of the pattern was put in place. Since the goal of this thesis is only to test the validity of the probability of detection as output by SALT, this same assumption has been built into the simulations discussed in the following chapter.

IV. SIMULATION DOCUMENTATION

Two examples of the FORTRAN simulation routines used to test the probability of detection of the sonobuoy patterns recommended by SALT as well as to determine if better patterns could be developed are presented in Appendix B. Seven FORTRAN programs are required to accomplish the desired testing. For each target motion model there are two separate programs. The first of these programs is written to be used with Definite Range Law sonobuoys, and the second program simulates sonobuoys with detection ranges that vary according to a Lambda-Sigma Jump model. (SALT will handle either case.) The remaining program is a generic detection subroutine (DTCT) which is used by all of the simulations. Each of these programs is discussed below. All of the simulation routines are developed using the SALT assumption that each sonobuoy is placed in its exact location at the beginning of the search.

A series of calibration runs has been executed for each of the target motion simulation programs discussed below to ensure proper operation. This testing was accomplished by having the program write the endpoints of each track segment to an output file. This output was then graphed and compared to the respective target motion model. Each simulation showed target motion consistent with the assumptions for that type of target.

A. PATROLLING TARGET SIMULATIONS

The simulation routine for a patrolling target with Definite Range Law sonobuoys is named "PATROL". The first step in the program is to read in the input parameters from a data file. The eight parameters needed by the simulation are as follows: σ_X and σ_Y , the standard deviations of the normal distributions required to simulate the initial Search Probability Area (SPA) and its containment percent, search duration in hours, target's course change rate in inverse hours, detection range of the sonobuoys in nautical miles, an integer seed for use in the random number generator, average target speed in knots, and finally, the Cartesian coordinates of the location of each of the sonobuoys in the pattern.

The target parameters are simulated as follows. The initial target position is determined by random draws from two independent normal distributions both with mean zero and standard deviations of σ_X and σ_Y. The target course, in radians, is selected from a Uniform (0, 2π) distribution. The triangular distribution required for the target speed is simulated by converting a Uniform (0, 1) random number into a speed using the inverse CDF technique. The speeds generated by this technique are displayed as a histogram in Figure 8 on page 15. It can be seen from this figure that the speeds for patrolling targets in this simulation are in fact representative of a triangular distribution.

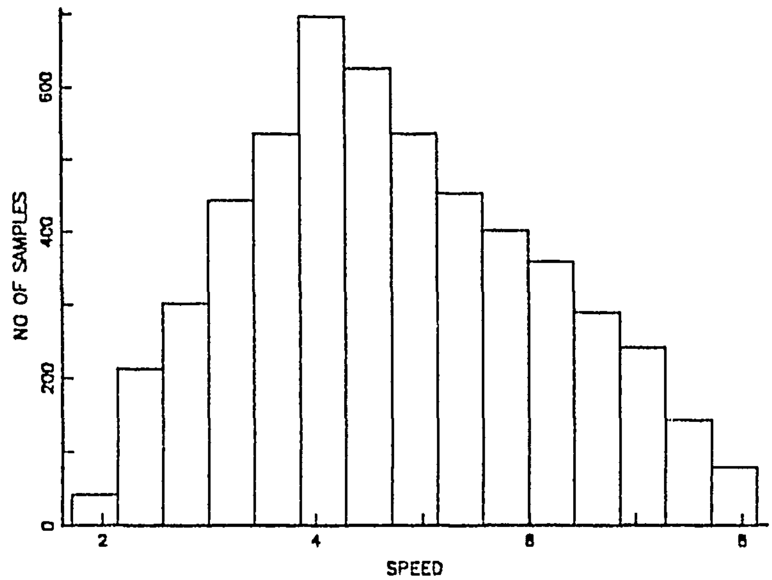

Figure 8. Patrolling Target Speed Distribution

The time that the target will spend on the chosen course at the chosen speed is determined by a random draw from an exponential distribution with mean (course change rate)$^{-1}$. These three parameters, course, speed, and time, determine a line segment which represents a leg of the target track. The final target position for this leg of the track is calculated using the initial position, course, speed, and time on leg determined as discussed above.

The coordinates of a sonobuoy in the pattern, the detection range of that sonobuoy, and the endpoints of the current leg of the target track are used in a call to the DTCT subroutine to determine whether or not a detection has occurred. This process is repeated until all sonobuoys in the pattern under investigation have been checked for possible detections. If a detection has occurred, the time of the detection is calculated, and the number of detections is incremented by one. The detection time of that sonobuoy which detects the target first in the time sequence is written to an output file for use in further analysis, and this run of the simulation is terminated. If no detections take place on this leg, the final position of this leg is reset to be the initial position on the next leg and the time that the search has been underway is incremented. If this time is less than the search duration input by the user, a new leg and final position are selected as above and the process is repeated.

The program which simulates a patrolling target in a sonobuoy field in which detection range is fluctuating according to a Lambda-Sigma Jump model is entitled "LSPAT". The input parameters for this program are the same as those discussed above with one exception. Instead of a constant value of sonobuoy detection range being input, a Figure of Merit (FOM) is input by the user in dB.

The target parameters, course, speed, and time on leg, are calculated as described above. However, a new environmental parameter is introduced into this model. This parameter is the acoustic fluctuation in dB which is represented by a random draw from a Normal distribution with mean zero and standard deviation, sigma = 8 dB. This acoustic fluctuation remains constant for a period of time determined from an exponential distribution with mean, lambda = 1 hour.

The acoustic fluctuations simulated as discussed in the previous paragraph are considered to be ambient acoustic fluctuations which apply uniformly to all sonobuoys in the pattern. The latest acoustic fluctuation is added to the input FOM to determine the current FOM. The current FOM is used to determine the detection range of the sonobuoys by linear interpolation between the two nearest points on the transmission loss curve which is input as a separate data file. The detection range calculated in this way applies to all of the sonobuoys in the pattern.

Calls to the DTCT subroutine for each sonobuoy to determine detection/non-detection are treated as discussed above. The only difference is that the current segment of the target track has its final position either at the end of the acoustic fluctuation time or at the end of the time on the current leg whichever occurs next in the time sequence. If the completion of acoustic fluctuation time occurs first, then time on leg is decremented by the acoustic fluctuation time and total elapsed search time is incremented by the same amount of time. A new acoustic fluctuation and time for the fluctuation are then drawn . If the completion of the time on leg occurs first, then the acoustic fluctuation time is decremented by leg time and total time is incremented by it. New values of course, speed, and time on leg are then computed.

If detection occurs, the number of detections is incremented by one, the time of the detection is written to an output file, and this run of the simulation is finished. If no detections have occurred and the total elapsed search time is less than the search duration input by the user, the process continues.

B. TRANSITING TARGET SIMULATION

The program which simulates a transiting target versus a field of Definite Range Law sonobuoys functions in a manner similar to that described in the above section on Patrolling Target Simulations. The input parameters for this program are the same as those discussed above except that the minimum and maximum course are input in radians and the minimum and maximum speed are input in knots.

The transiting target parameters are simulated as follows. The initial position is determined the same way as described under the Patrolling Target Simulations section above. Both target course and speed of advance (SOA) are simulated from triangular distributions using the inverse CDF technique. These values of course and SOA are then converted to target speed by dividing the SOA by the cosine of the angle between average target course and the target course determined for this track segment. These triangular distributions have been verified by histogram as in the patrolling target case, but the figure is not reproduced here. The remainder of the simulation proceeds exactly as described above for patrolling targets.

The program which models the Lambda-Sigma Jump sonobuoys generalizes from the transiting target Definite Range Law sonobuoy program in the same manner as for patrolling targets.

C. FLEEING DATUM TARGET SIMULATION

The next program to be discussed is the one used to simulate fleeing datum targets and Definite Range Law sonobuoys. The input parameters are the same as those discussed in the previous sections. The target course is drawn from a Uniform $(0, 2\pi)$ distribution and target speed is drawn from a triangular distribution determined by the input parameters of minimum and maximum target speed. The course and speed need only be drawn one time for each simulation run since the target is assumed to maintain that course and speed throughout the engagement. The initial position is aged by the time-late value of four hours using the selected course and speed. The detection/non-detection event is then determined as described in the Patrolling Target Simulations section. This simulation routine generalizes to the Lambda-Sigma Jump model for sonobuoy detection range in the same manner as previously discussed.

D. DETECTION SUBROUTINE

The detection subroutine entitled "DTCT" is used by all six of the main programs to determine whether or not a target detection occurs. If a detection does take place, the subroutine returns the distance from the beginning of the current segment of the target track to the point where the earliest detection occurs.

The inputs to the DTCT subroutine are the detection range of the sonobuoys, the cartesian coordinates of the endpoints of the current track segment, and the location of the current sonobuoy being tested for detection criterion. These seven parameters are passed into the subroutine at each call to it.

The first step of this subroutine is to calculate the perpendicular distance between the line determined by the target's track segment and the location of the sonobuoy. If this value is greater than the detection range of the sonobuoy, no detection takes place. However, if this perpendicular distance is less than the detection range, there are three cases to consider to determine whether or not detection occurs. If the track segment contains the point on the line that is closest to the buoy center, then a detection occurs. If the track segment does not contain that point, but either endpoint of the track segment falls within the range of detection of the buoy a detection will also take place. If neither of these conditions holds, then there is no detection for this track segment-sonobuoy combination. Each of these four cases is illustrated in Figure 9 on page 19.

In order to ensure the proper operation of this subroutine, examples of the four cases described above were developed using simple graphical techniques. These examples were then input into the subroutine for testing. The outputs from the subroutine confirm that it performs its function properly as described here.

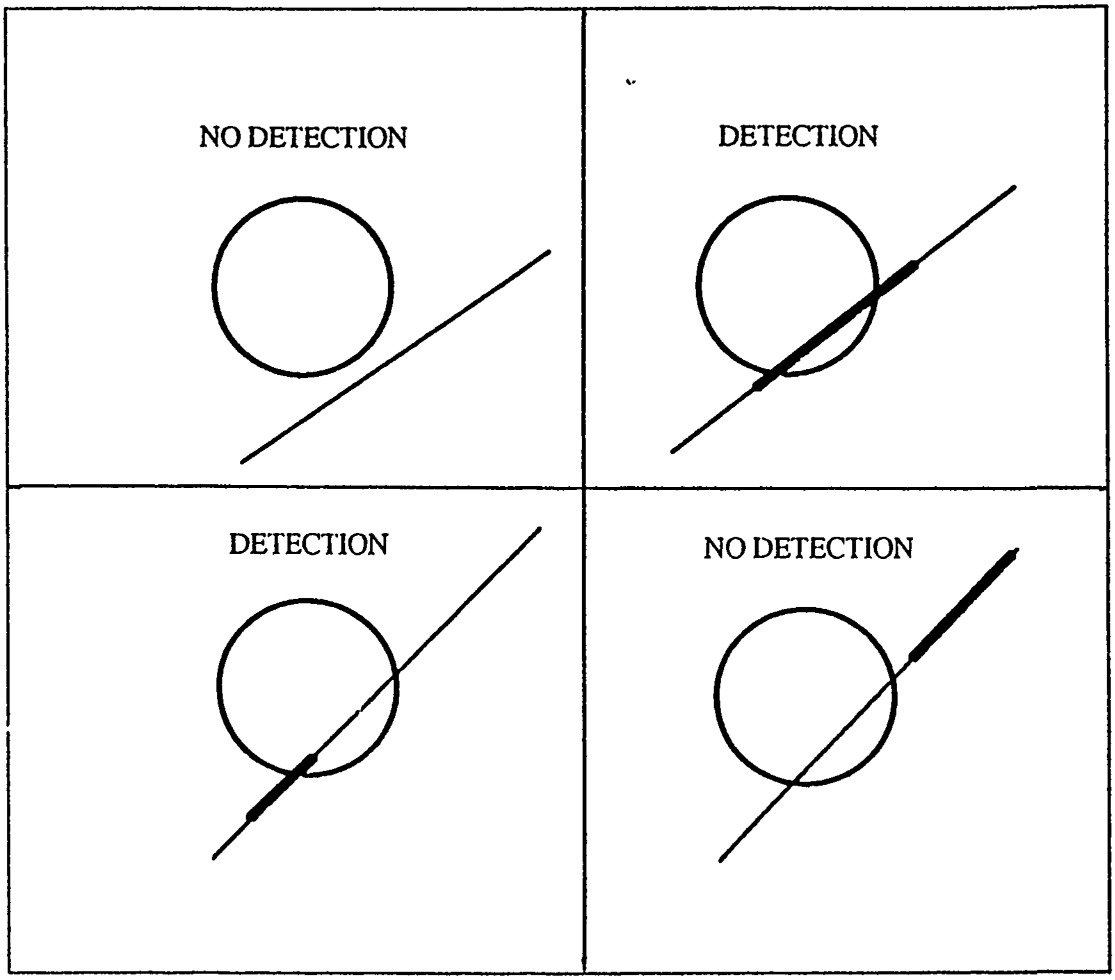

Figure 9. Sonobuoy Detection/Non-detection Possibilities

V. SIMULATION RESULTS OF SALT SONOBUOY PATTERNS

The sonobuoy patterns generated by the SALT algorithm for each specific type of target motion were tested using the respective simulation routine discussed in the previous chapter to determine if the probability of detection for that pattern as output by SALT is accurate. The simulation results are output as a number of detections which can be easily converted to a probability of detection if the number of detections is divided by the number of runs of the simulation which is 10,000 runs in each case. Each proportion, $\hat{p}$, which represents the probability of detection can be expanded to a two-sided $(1-\alpha)\%$ Confidence Interval by use of the following formula:

$$\hat{p} \pm Z_{1-\frac{\alpha}{2}} \sqrt{\frac{\hat{p}(1-\hat{p})}{n}} \qquad (5.1)$$

where $Z_{1-\frac{\alpha}{2}}$ represents the proper percentage point of a Standard Normal cumulative distribution.

The SALT algorithm was run three times for each type of target motion using sonobuoys that have a Definite Range Law detection function. The parameters within the SALT program which determine the acoustic fluctuations were set to their minimum values to accomplish this (see Appendix A). For each run of a specific target motion type, all inputs remain the same with the exception of the number of sonobuoys. The number of sonobuoys used for each run are 8, 16, and 24. These numbers of sonobuoys allowed for testing over a range of values of probability of detection. The same nine runs of the SALT algorithm were repeated to obtain data for sonobuoys which behave according to a Lambda-Sigma Jump model. The parameters for this phase of testing are an ambient acoustic fluctuation rate of one per hour and standard deviation of ambient acoustic fluctuations of eight dB. The probability of detection as output by SALT will now be compared to the simulation results for each type of target motion.

A. PATROLLING TARGET

The probability of detection for the Definite Range Law sonobuoy patterns recommended by SALT for a patrolling target are displayed in Table 1 on page 21. Both the SALT calculated probability of detection and the simulation routine output probability of detection are listed for the number of sonobuoys used in each pattern. A 95% Confidence Interval, calculated as described above, is also displayed for each simulation probability of detection.

Table 1. PATROLLING TARGET COMPARISONS (DEFINITE RANGE)

NUMBER OF BUOYS	SALT P_D	SIMULATION P_D	95% CONFIDENCE INTERVAL
8	0.49	0.8848	(0.8785, 0.8911)
16	0.67	0.9402	(0.9378, 0.9426)
24	0.76	0.9773	(0.9758, 0.9788)

It can be seen from this table that the SALT algorithm tends to understate the probability of detection for a target in this case. The low probabilities of detection output by SALT have the potential to be very misleading. For example, the SALT recommended pattern for eight sonobuoys lists as its probability of detection a value of 0.49. This low probability of detection may encourage a P-3C crew to opt instead for a sonobuoy pattern with more buoys which would be a waste of resources considering the actual probability of detection for the target using an eight buoy pattern is 0.88.

The *Technical Documentation of SALT* claims that the exponential detection function used in equation (3.1) on page 9 gives a lower bound on the probability of detection that can be achieved by a systematic uniform search of a region [Ref. 2: p. 26]. Due to this fact, one would expect to get resulting probabilities of detection that are a little higher than the SALT output values. However, the disparity in the values given here is too high to be so easily explained away. In order to gain insight into why the simulation probability of detection is so different from the SALT probability of detection, the detection times as output by the simulation program are graphed into a Cumulative Distribution Function in Figure 10 on page 22.

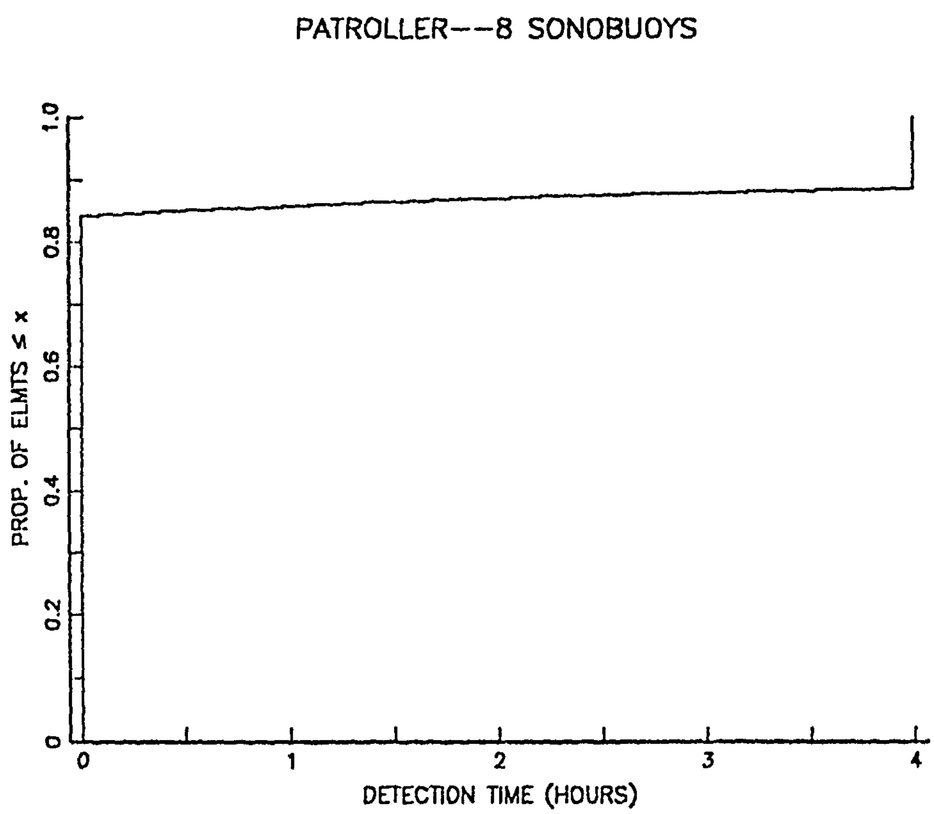

Figure 10. Cumulative Distribution of Detection Times

It can be seen from this CDF that a large percentage of the targets which get detected are initial detections. This is not too surprising considering the tight packing of the sonobuoys into the 86% containment ellipse. The sonobuoy detection circles are shown plotted over this 86% containment ellipse in Figure 11 on page 23. The ellipse shown in this figure has a semi-major axis of 60 nautical miles and a semi-minor axis of 30 nautical miles. The detection range circles are 18.2 nautical miles in radius and are placed in the locations specified by SALT.

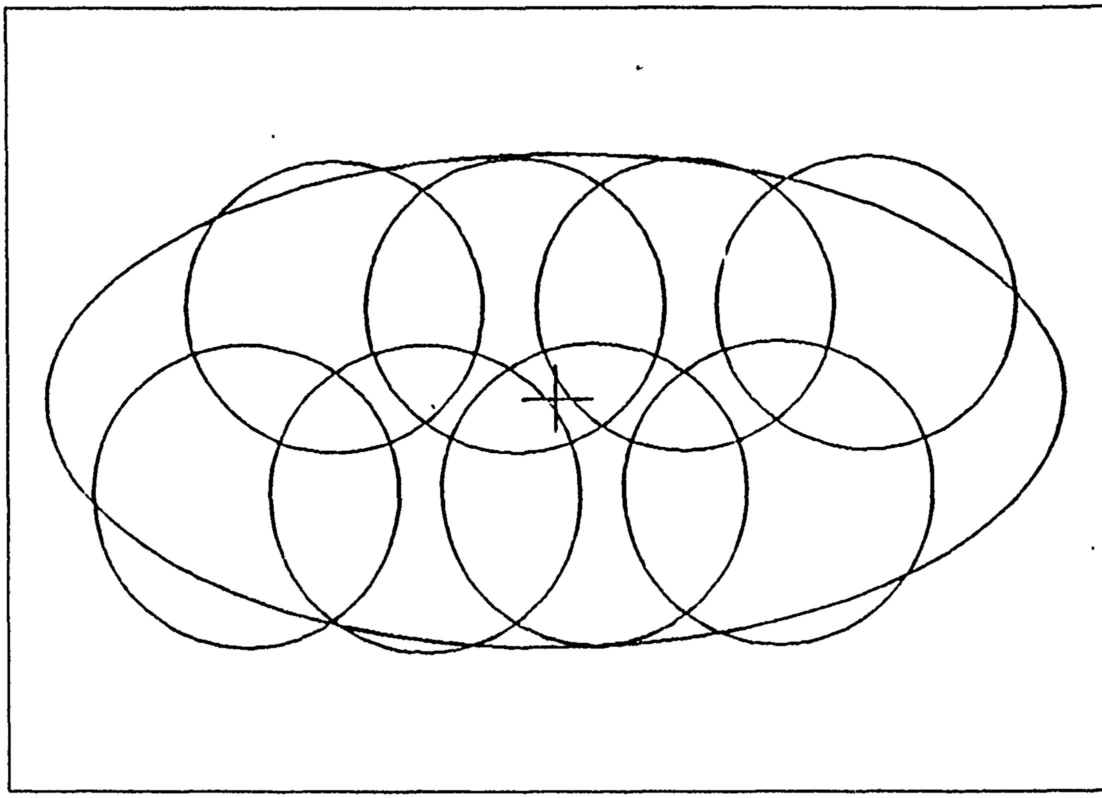

Figure 11. Detection Ranges and 86% Containment Ellipse for an 8 Sonobuoy Pattern

To account for this initial probability of detection in the patrolling target sonobuoy patterns, equation (3.1) on page 9 can be modified slightly to read as follows:

$$P_D(K) = [\Phi(K) - \Phi(-K)]^2[1 - \exp(-\frac{nWVT + n\pi R^2}{4K^2\sigma_1\sigma_2})] \qquad (5.2)$$

where:

R = the detection range of the sonobuoys in nautical miles

and all other variables are defined as they are in equation (3.1).

The above modification to the calculation of the probability of detection takes into account the detection area of each sonobuoy at the initiation of search. The value of the product nWVT is clearly time dependent. This means that in equation (3.1) there is zero probability of an initial detection. The detection times shown in Figure 10 on page 22 clearly show the need to account for this initial probability of detection.

Using equation (5.2) to calculate the probability of detection for the three Definite Range Law sonobuoy patterns specified in Table 1 on page 21 results in probabilities of detection of 0.74, 0.89, 0.93 for 8, 16, and 24 sonobuoy patterns respectively. While these values do not exactly reflect the results of the simulation runs, they do provide the decision maker with a much more realistic assessment of the probability of detection for the recommended pattern.

The Lambda-Sigma Jump model comparisons for the similar patterns are given in Table 2. The results for the 8 and 16 sonobuoy patterns show the same phenomenon discussed above for Definite Range Law sonobuoys. In these cases, equation (5.2) again provides a more realistic value of probability of detection.

Table 2. PATROLLING TARGET COMPARISONS (LAMBDA-SIGMA)

NUMBER OF BUOYS	SALT P_D	SIMULATION P_D	95% CONFIDENCE INTERVAL
8	0.59	0.8664	(0.8630, 0.8698)
16	0.76	0.9028	(0.8994, 0.9062)
24	0.84	0.8120	(0.8081, 0.8159)

The SALT output for 24 sonobuoys in this scenario was a pattern which was noticeably different from the five other sonobuoy patterns recommended for deployment against a patrolling target. The SALT generated pattern is a BRUSHTAC pattern where all of the buoys are placed in a narrowly spaced line along the major axis of the initial SPA. In a telephone conversation with Lawrence D. Stone, the president of Metron, Inc., it has been determined that an error in the SALT code resulted in an optimal target depth being selected which would give the sonobuoys a convergence zone detection. This selection was made by the algorithm in spite of the fact that the probability of the target being at that depth was input as zero. Due to this unfortunate fact, the results displayed in Table 2 for a 24 sonobuoy pattern have no meaning.

B. TRANSITING TARGET

The comparison of the SALT output probability of detection for transiting target sonobuoy patterns to the probability of detection for the same patterns as output by the simulation routine is presented in Table 3 on page 25.

Table 3. TRANSITING TARGET COMPARISONS (DEFINITE RANGE)

NUMBER OF BUOYS	SALT P_D	SIMULATION P_D	95% CONFIDENCE INTERVAL
8	0.72	0.8857	(0.8825, 0.8889)
16	0.81	0.9085	(0.9056, 0.9114)
24	0.82	0.9156	(0.9128, 0.9184)

The probabilities of detection displayed above do not exhibit the same magnitude of difference that those of the patrolling target exhibit in the previous section as seen in Table 1 on page 21. However, there is still a slight difference in these probabilities of detection that requires consideration. The CDF plot for transiting target detection times reveals that only about 35% of the detections obtained are due to initial detection. The reason for this smaller number of initial detections is that the transiting target search plans are constructed in the relative motion space of the mean velocity of the target as discussed in the Transiting Target Patterns section of Chapter III. The length of the side of the rectangular area swept out which is parallel to the mean velocity vector is determined by the following equation: [Ref. 2: p. 52]

$$\text{Length} = VT + 2R \tag{5.3}$$

where:

V = average SOA of the target in knots

T = search duration in hours

R = the detection range of the sonobuoys in nautical miles

When this relative motion space is translated into geographic space, only the 2R portion of the length as determined by equation (5.3) has the possibility for initial detections. In order to account for these initial detections in the calculation of probability of detection, this fraction of the total length should be used.

Due to the same depth selection error discussed at the end of the previous section, the SALT output for transiting targets using the Lambda-Sigma Jump model is only valid for the eight sonobuoy case. The SALT output probability of detection for this scenario is 0.81 while the simulation output for the identical pattern is 0.7996. The 95%

Confidence Interval about this value is (0.7956, 0.8036). In this more realistic case, the initial probability of detection does not seem to be an important factor.

C. FLEEING DATUM TARGET

The results of the simulation analysis of Definite Range Law sonobuoy patterns deployed against a fleeing datum target are shown in Table 4.

Table 4. **FLEEING DATUM TARGET COMPARISONS (DEFINITE RANGE)**

NUMBER OF BUOYS	SALT P_D	SIMULATION P_D	95% CONFIDENCE INTERVAL
8	0.47	0.4829	(0.4779, 0.4879)
16	0.69	0.9579	(0.9559, 0.9599)
24	0.81	0.9985	(0.9981, 0.9989)

The probability of detection as output by both SALT and the simulation of this scenario appear to be in close agreement for the eight sonobuoy case while the other two patterns show a highly significant difference. This difference cannot be explained by initial detections as is done in the cases of patrolling and transiting targets above. Less than 10% of the detections were achieved initially when these scenarios were run using the simulation routines. An alternate reason for this phenomenon can be found by determining the fraction of the perimeter of these patterns that is covered by sonobuoy detection radii. In the eight sonobuoy pattern, 48% of this perimeter is covered by such detection ranges. Similar calculations for the 16 sonobuoy pattern reveal that 94% of the perimeter is so covered, and the percentage of coverage for the 24 sonobuoy case is 100%. This fact accurately reflects that when the sonobuoys are assumed to be placed in the water immediately upon commencement of search, it is highly unlikely that a target will avoid having to pass through the annular region where the sonobuoys are placed.

As discussed in the Fleeing Datum Target Patterns section of Chapter III, the probability of detection in these scenarios is calculated within the SALT algorithm by multiplying the probability of detecting the target given that it is contained in the two rectangle annular region by the probability that the target is contained in this region. By performing this simplified calculation, SALT fails to take into account the fact that a target which is inside of the inner rectangle at the commencement of search must pass through this annular region to evade detection. The SALT algorithm must be modified in some way to account for this possibility in order to more realistically evaluate the probability of detection for fleeing datum targets.

The results for the same target scenarios versus Lambda-Sigma Jump model sonobuoys are presented in Table 5. These results show a disparity similar to that discussed above for Definite Range Law sonobuoys. Thus the corrective action specified above is valid for both types of sonobuoy detection models.

Table 5. FLEEING DATUM TARGET COMPARISONS (LAMBDA-SIGMA)

NUMBER OF BUOYS	SALT P_D	SIMULATION P_D	95% CONFIDENCE INTERVAL
8	0.59	0.8623	(0.8589, 0.8657)
16	0.80	0.9138	(0.9110, 0.9166)
24	0.88	0.9413	(0.9389, 0.9437)

It can be seen from these analyses that in all cases of target motion models the SALT code requires modification to be able to more accurately reflect the probability of detection so that sound decisions can be made by the user of this tactical decision aid.

VI. ALTERNATIVE SONOBUOY PATTERN DEVELOPMENT

The second phase of this research effort involves attempting to determine whether or not the sonobuoy patterns output by the SALT algorithm can be improved. The measure of effectiveness of a sonobuoy pattern will be the probability of detection as output by the same simulation routines used to verify the probability of detection. The improvements considered and the results are presented for each target motion model in the following sections.

A. PATROLLING TARGET MODEL

As was shown previously in Figure 11 on page 23, the pattern for eight sonobuoys deployed against a patrolling target consists of excessive overlap of the buoy detection regions. The sonobuoy spacing in this pattern is 22.3 nautical miles between the two rows and 21.2 nautical miles between the buoys within a row. The probability of detection of this pattern is 0.88 as determined from the simulation program.

The SALT assumption for deploying a sonobuoy pattern against a target of this type is that the search effort should be distributed uniformly over the region in which the target is most likely to be located. Under this assumption it can plainly be seen that the overlap of the buoy detection ranges is excessive. By increasing the spacing between the buoys in this pattern, some gains in probability of detection can be realized.

The best spacing to use between the sonobuoys in the pattern with eight Definite Range Law sonobuoys has been determined by trail and error. After many simulation runs on patterns with various spacing, it has been determined that the best effort can be achieved by making the spacing between rows 31.2 nautical miles, and setting the within row spacing to 36.0 nautical miles. The probability of detection for this pattern was output by the simulation as 0.96. Further spreading of the sonobuoys results in too much effort being allocated to outlying areas. The probability that the target is located in these areas is very low.

A graphical representation of the detection ranges of the sonobuoys superimposed on an 86% probability of containment ellipse is shown in Figure 12 on page 29. The ellipse shown in the figure has a semi-major axis of 60 nautical miles and a semi-minor axis of 30 nautical miles. The detection range circles are 18.2 nautical miles in radius, and overlap only a slight amount.

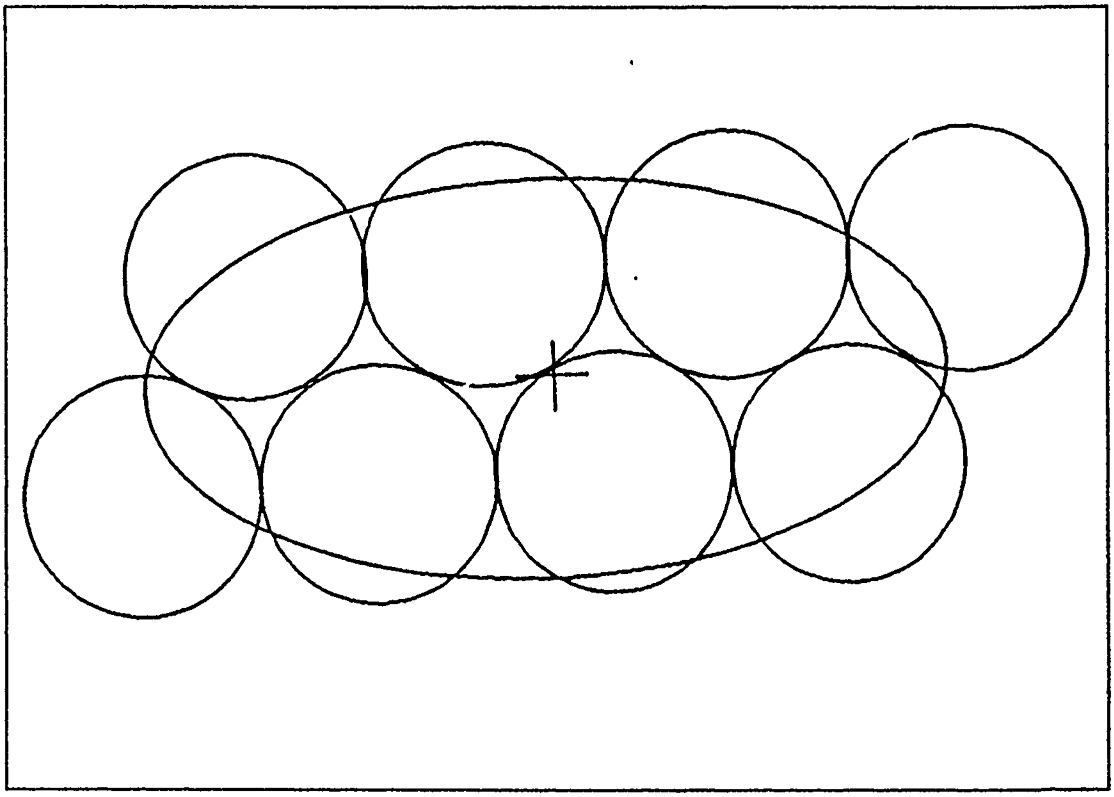

Figure 12. Detection Ranges and 86% Containment Ellipse for an 8 Sonobuoy Pattern with 36 Nautical Mile Spacing

Similar tests were run using the Lambda-Sigma Jump model sonobuoys, but altering the spacing does not affect the results as much since there are changes in sonobuoy detection range caused by acoustic fluctuations. The SALT output spacing for the eight sonobuoy Lambda-Sigma Jump pattern is 24.7 nautical miles between rows and 23.5 nautical miles between buoys within each row. Altering the spacing between the sonobuoys in this scenario provides no significant increases in the probability of detection.

Another type of pattern that was tested using Lambda-Sigma Jump model sonobuoys against a patrolling target is a 2 x 4 x 2 pattern. This pattern consists of 4 sonobuoys placed along the major axis of the 86% containment ellipse with 2 sonobuoys on each side of this line located near the minor axis. No notable increase in probability of detection is realized using this pattern.

Based on the results of the testing described above, it seems that the sonobuoy patterns recommended by SALT for use in the more realistic environmental setting modeled by the Lambda-Sigma Jump process are consistent with the stated objectives of the SALT algorithm.

B. TRANSITING TARGET MODEL

In attempting to improve the sonobuoy patterns recommended by SALT for use against a transiting target, only the more realistic case of the Lambda-Sigma Jump model sonobuoys is considered. The SALT pattern for this type of target motion model is a straight line of buoys placed perpendicular to the mean target course. In looking at this pattern of eight sonobuoys graphed along with the 86% containment ellipse, it appears that the line of sonobuoys is located too close to the center of the SPA. Moving the line of sonobuoys away from the center in the direction of the mean target course results in the probability of detection going down. This results from the fact that although fewer targets will be located on the wrong side of this line, more targets will be located further from the pattern and will not reach it prior to termination of the search. The next aspect of the pattern to be considered is the spacing between the sonobuoys in the line. The spacing between the sonobuoys in the SALT generated pattern is only 12.8 nautical miles, even though the average detection range is 26.3 nautical miles. The sonobuoy spacing was increased to 20 nautical miles. The simulation run for this new pattern showed no notable increase in probability of detection. This result is not difficult to understand in light of the fact that as the spacing between the sonobuoys is increased, the buoys near the ends of the line are being moved to areas with extremely low probability of achieving a detection.

The next logical step in the development of alternative patterns is to depart from the requirement to lay the sonobuoys in a straight line. By increasing the spacing between the sonobuoys to overcome the excessive overlap and then bringing the endpoints of the line closer to the SPA, it seems that the probability of detection should increase. Two patterns were tested using this theory.

The first pattern tested in which the sonobuoys are not placed in a straight line is a semi-circular pattern with spacing between the sonobuoys of 20 nautical miles. This pattern covers 90 degrees of arc on each side of the mean target course line. The pattern is positioned so that all sonobuoys are located at a distance midway between the lengths of the semi-major and semi-minor axes from the center of the SPA.

The second pattern tested is a chevron pattern. The sonobuoys in this pattern are divided equally between the two sides of the chevron, and the wedge angle of the chevron is set at 90 degrees. The point of the chevron is located along the mean target course line at a distance half way between the lengths of the semi-major and semi-minor axes. The spacing between the sonobuoys was again set to 20 nautical miles.

Neither of these patterns results in a significant increase in the probability of detection for the transiting target modeled here. Based on the test results described here, the sonobuoy patterns generated by SALT for deployment against transiting targets appear to adequately meet the objectives of SALT.

C. FLEEING DATUM TARGET MODEL

The first sonobuoy pattern output by SALT that is considered for improvement is the pattern with eight Definite Range Law sonobuoys. The analysis of the detection times output by the simulation routine for this pattern shows only 42 initial detections out of the 4829 detections obtained. This result leads to the suspicion that this pattern is located too far from the initial SPA. Each side of the square pattern generated by SALT is located 88 nautical miles from the center of the SPA. Several runs of the simulation have been conducted moving the sonobuoy pattern nearer the SPA each time and scaling it accordingly. These runs show that a pattern in which the sides are located 50 nautical miles from the center increases the probability of detection to 0.78.

The square shape of the sonobuoy patterns output by SALT for attempting to detect a fleeing datum target naturally suggests that a better use of available search effort would be to arrange the sonobuoys in a circle so that the buoys would be equidistant from the SPA center. Such a pattern was tested using the eight sonobuoy Lambda-Sigma Jump model scenario. The sonobuoys in this pattern were placed 82.5 nautical miles from the center of the SPA and arranged so as to uniformly cover the perimeter of the circle. This distance from the center was arrived at by averaging the semi-major and semi-minor axes of the initial SPA and then adding a distance to account for the time-late of the beginning of the search which is four hours. In this way, all sonobuoys in the pattern are located closer to the center than in the SALT pattern. Each edge of the square SALT pattern is located at 84.6 nautical miles from the center.

A similar circular pattern has also been tested in which the sonobuoys are placed 60 nautical miles from the center. This distance is based on the improved pattern for Definite Range Law sonobuoys discussed above. This pattern only increases the probability of detection by 0.03 which is not significant.

The above results of attempting to improve the sonobuoy patterns recommended by SALT again show that the Definite Range Law scenario patterns can be improved greatly while the Lambda-Sigma Jump scenario patterns do appear to be consistent with the SALT objectives. These results should not be weighted too heavily as they are obtained by remaining within the constraints of the SALT assumptions, most notably that all of the sonobuoys in a pattern deployed against a fleeing datum target are put in place at the commencement of the search. This assumption has been shown to be faulty at the end of Chapter III.

VII. CONCLUSIONS

It has been established at the beginning of this thesis that the goal of the research effort is to investigate the initial search planning phase of the *Search and Localization Tactical Decision Aid.* This investigation consists of first attempting to ensure that the probability of detection of various sonobuoy patterns recommended by SALT is correct under the stated assumptions and subsequently of testing alternative sonobuoy patterns which could lead to a better patterns than those recommended by SALT.

The probability of detection output by SALT for its own patterns is significantly understated for both the patrolling target model and the fleeing datum target model. Possible corrections to these errors have been presented here. The probability of detection for the transiting target model appears to be correct especially in the more realistic case of the Lambda-Sigma Jump model for sonobuoy detection ranges.

The sonobuoy patterns generated by SALT for Definite Range Law sonobuoys deployed against patrolling and fleeing datum targets have been shown to be inadequate for accomplishing the goal of maximizing probability of detection. The recommended patterns for all three motion models when the acoustic fluctuations which influence sonobuoy detection range are modeled by a Lambda-Sigma Jump process are consistent with the stated objectives of SALT. This fact is especially encouraging since the patterns output by the SALT algorithm are operationally easy to deploy from a P-3C aircraft. One must be reminded here that the testing performed on the sonobuoy patterns output by SALT has been conducted within the constraints of the assumptions made in developing the SALT algorithm.

One aspect of the fleeing datum target scenario which warrants further investigation is the assumption that all sonobuoys are put in place simultaneously at the commencement of the search. Some method of calculating the probability of detection for this type of target needs to be developed which allows for the fact that a target may be outside of the sonobuoy pattern by the time that the critical portion of these large patterns is put in place. A more useful alternative to altering the calculation of the probability of detection would be to develop a means of accounting for the required flight time in recommending the sonobuoy pattern. This change in the algorithm would open up the possibility of deploying patterns which spiral out from the center as time elapses and the target continues its assumed motion away from the SPA center. [Ref. 4: pp. 115-118]

The SALT algorithm is rated as effective in its initial search planning phase for patrolling targets with only minor modifications necessary to improve the calculations of probability of detection. The SALT recommendations from the initial search planning phase for transiting targets also appear to be realistic. Further development is required in the case of fleeing datum targets as discussed above in order to make SALT ready for operational testing.

APPENDIX A. SALT INPUTS FOR TEST CASES

The purpose of this appendix is to provide a list of the inputs to the SALT algorithm which were used to generate the sonobuoy patterns required for the testing performed here. Only those parameters which are relavent to the scenarios are presented. The remaining inputs are left at their default values. The target motion model parameters are not presented here since they are given in Chapter II.

The following list shows input values used in the Definite Range Law sonobuoy cases for patrolling and transiting targets.

Setup	
Buoy acoustic fluctuation rate (per hr)	0.01
Standard deviation of buoy acoustic fluctuations (dB)	0.01
Ambient acoustic fluctuation rate (per hour)	0.01
Standard deviation of ambient acoustic fluctuations (dB)	0.01
Search duration (hrs)	4.0
Environmental Data	
Ocean	Pacific
Ambient noise level (dB)	75
Target Parameters	
Depth 1 (ft)	90
Probability of depth 1	1.0
Depth 2 (ft)	300
Probability of depth 2	0.0
Frequency 1 (hz)	50
Source level 1 (dB)	160
Frequency 2 (hz)	300
Source level 2 (dB)	75

SPA Data	
Latitude	50 00 N
Longitude	020 00 W
Semi-major axis (nm)	60.0
Semi-minor axis (nm)	30.0
Orientation (deg N)	90.0
Containment percent	86.0

The input parameters which are changed to generate output patterns for fleeing datum targets is presented in the next list.

Setup	
Search duration (hrs)	8.0
SPA Data	
Semi-major axis (nm)	30.0
Semi-minor axis (nm)	15.0

The two input parameters that are changed to simulate sonobuoys with detection ranges which vary according to the Lambda-Sigma Jump model are as follows:

Setup	
Ambient acoustic fluctuation rate (per hour)	1.0
Standard deviation of ambient acoustic fluctuations (dB)	8.0

APPENDIX B. SIMULATION PROGRAMS

This appendix presents listings of examples of the FORTRAN computer code used in the simulation of the sonobuoy patterns deployed against the three target motion models of SALT.

The two programs used to simulate patrolling target motion models are shown here. The first program shows the FORTRAN code which tests the SALT output patterns generated using Definite Range Law sonobuoys. The second program provides a model for sonobuoys which behave according to a Lambda-Sigma Jump process. The programs used to simulate the transiting target and the fleeing datum target are similar except in the required target input parameters.

Following the patrolling target simulation listings is the listing of the detection subroutine. This subroutine is used by all six of the simulation programs to determine whether or not the sonobuoy detection criterion is met.

```
      PROGRAM PATROL
      REAL SIGMAX, SIGMAY, SCHTM, LMBDA, DETRNG
      INTEGER SEED, DET, NUMDET, I
      REAL AVGSP, SPEED, U, COURSE, X1, Y1, X2, Y2, XC, YC
      REAL T, LEGTM, TOTTIM, DSTDET, DT, DETTIM
      CALL EXCMS('FILEDEF 01 DISK PATPRM DATA A')
      CALL EXCMS('FILEDEF 02 DISK PATBYS DATA A')
      READ(01,*) SIGMAX, SIGMAY, SCHTM, LMBDA, DETRNG, SEED, AVGSP
      NUMDET = 0
      DO 40 I = 1, 10000
        CALL LNORM(SEED, X1, 1, 1, 0)
        X1 = X1 * SIGMAX
        CALL LNORM(SEED, Y1, 1, 1, 0)
        Y1 = Y1 * SIGMAY
        TOTTIM = 0.0
        DETTIM = 10000.0
   10   CONTINUE
          CALL LRND(SEED, COURSE, 1, 1, 0)
          COURSE = COURSE * 6.283185
          CALL LRND(SEED, U, 1, 1, 0)
          IF (U .LE. 1.0/3.0) THEN
            SPEED = AVGSP*(24 + SQRT(3024*U + 16))/56
          ELSE
            SPEED = AVGSP*(60 - SQRT(1528 - 1512*U))/28
          END IF
          CALL LEXPN(SEED, T, 1, 1, 0)
          LEGTM = T/LMBDA
          X2 = X1 + SPEED * LEGTM * SIN(COURSE)
          Y2 = Y1 + SPEED * LEGTM * COS(COURSE)
   20     CONTINUE
            READ(02,*) XC, YC
          IF (XC .GT. 999.0) GO TO 30
            CALL DTCT(DETRNG, X1, Y1, X2, Y2, XC, YC, DET, DSTDET)
            IF (DET .EQ. 1) THEN
               DT = TOTTIM + DSTDET/SPEED
               IF (DT .LT. DETTIM) THEN
                  DETTIM = DT
               END IF
            END IF
          GO TO 20
   30     CONTINUE
          REWIND 02
          X1 = X2
          Y1 = Y2
          TOTTIM = TOTTIM + LEGTM
         IF (TOTTIM .LT. SCHTM .AND. DETTIM .GT. 9999.0) GOTO 10
         IF (DETTIM .LT. SCHTM) THEN
          NUMDET = NUMDET + 1
          WRITE(11,*) DETTIM
         END IF
   40 CONTINUE
      PRINT*, 'NUMBER OF DETECTIONS =', NUMDET
      STOP
      END
```

```
0             PROGRAM LSPAT
              REAL SIGMAX, SIGMAY, SCHTM, LMBDA, DETRNG, FOM
              REAL ACFL, AFTIM, R, R1, TL, TL1
              INTEGER SEED, DET, NUMDET, I
              REAL AVGSP, SPEED, U, COURSE, X1, Y1, X2, Y2, XC, YC
              REAL T, LEGTM, TOTTIM, DSTDET, DT, DETTIM
              CALL EXCMS('FILEDEF 01 DISK PATPRM DATA A')
              CALL EXCMS('FILEDEF 02 DISK PATBYS DATA A')
              CALL EXCMS('FILEDEF 03 DISK TLCRV DATA A')
              READ(01,*) SIGMAX, SIGMAY, SCHTM, LMBDA, FOM, SEED, AVGSP
              NUMDET = 0
              DO 110 I = 1, 10000
    1           CALL LNORM(SEED, X1, 1, 1, 0)
    1           X1 = X1 * SIGMAX
    1           CALL LNORM(SEED, Y1, 1, 1, 0)
    1           Y1 = Y1 * SIGMAY
    1           TOTTIM = 0.0
    1           DETTIM = 10000.0
    1           CALL LRND(SEED, COURSE, 1, 1, 0)
    1           COURSE = COURSE * 6.283185
    1           CALL LRND(SEED, U, 1, 1, 0)
    1           IF (U .LE. 1.0/3.0) THEN
  1 1             SPEED = AVGSP*(24 + SQRT(3024*U + 16))/56
  1 1           ELSE
  1 1             SPEED = AVGSP*(60 - SQRT(1528 - 1512*U))/28
  1 1           END IF
    1           CALL LEXPN(SEED, T, 1, 1, 0)
    1           LEGTM = T/LMBDA
    1           CALL LEXPN(SEED, AFTIM, 1, 1, 0)
    1     10    CALL LNORM(SEED, ACFL, 1, 1, 0)
    1           ACFL = ACFL * 8.0
    1           FOM = FOM + ACFL
    1           IF (FOM .LE. 66.0 .OR. FOM .GE. 100.66) GO TO 10
    1     20    CONTINUE
    1             READ(03,*) R, TL
    1             IF (FOM .GE. TL) THEN
  1 1               R1 = R
  1 1               TL1 = TL
  1 1             ELSE
  1 1               DETRNG = R1 + (FOM - TL1)/(TL - TL1)
  1 1               GO TO 30
  1 1             END IF
    1           GO TO 20
    1     30    CONTINUE
    1           REWIND 03
    1     35    CONTINUE
    1             IF (AFTIM .LT. LEGTM) THEN
  1 1               X2 = X1 + SPEED * AFTIM * SIN(COURSE)
  1 1               Y2 = Y1 + SPEED * AFTIM * COS(COURSE)
  1 1     40        CONTINUE
  1 1                 READ(02,*) XC, YC
  1 1               IF (XC .GT. 999.0) GO TO 50
  1 1                 CALL DTCT(DETRNG, X1, Y1, X2, Y2, XC, YC, DET, DSTDET)
  1 1                 IF (DET .EQ. 1) THEN
  2 1                    DT = TOTTIM + DSTDET/SPEED
  2 1                    IF (DT .LT. DETTIM) THEN
```

```
   3  1                DETTIM = DT
   3  1              END IF
   2  1            END IF
   1  1          GO TO 40
   1  1     50   CONTINUE
   1  1          REWIND 02
   1  1          X1 = X2
   1  1          Y1 = Y2
   1  1          LEGTM = LEGTM - AFTIM
   1  1          TOTTIM = TOTTIM + AFTIM
   1  1          CALL LEXPN(SEED, AFTIM, 1, 1, 0)
   1  1     60   CALL LNORM(SEED, ACFL, 1, 1, 0)
   1  1          ACFL = ACFL * 8.0
   1  1          FOM = FOM + ACFL
0  1  1          IF (FOM .LE. 66.0 .OR. FOM .GE. 100.66) GO TO 60
   1  1     70   CONTINUE
   1  1            READ(03,*) R, TL
   1  1            IF (FOM .GE. TL) THEN
   2  1              R1 = R
   2  1              TL1 = TL
   2  1            ELSE
   2  1              DETRNG = R1 + (FOM - TL1)/(TL - TL1)
   2  1              GO TO 80
   2  1            END IF
   1  1          GO TO 70
   1  1     80   CONTINUE
   1  1          REWIND 03
   1  1        ELSE
   1  1          X2 = X1 + SPEED * LEGTM * SIN(COURSE)
   1  1          Y2 = Y1 + SPEED * LEGTM * COS(COURSE)
   1  1     90   CONTINUE
   1  1            READ(02,*) XC, YC
   1  1          IF (XC .GT. 999.0) GO TO 100
   1  1            CALL DTCT(DETRNG, X1, Y1, X2, Y2, XC, YC, DET, DSTDET
   1  1            IF (DET .EQ. 1) THEN
   2  1               DT = TOTTIM + DSTDET/SPEED
   2  1               IF (DT .LT. DETTIM) THEN
   3  1                 DETTIM = DT
   3  1               END IF
   2  1            END IF
   1  1          GO TO 90
   1  1    100   CONTINUE
   1  1          REWIND 02
   1  1          X1 = X2
   1  1          Y1 = Y2
   1  1          AFTIM = AFTIM - LEGTM
   1  1          TOTTIM = TOTTIM + LEGTM
   1  1          CALL LRND(SEED, COURSE, 1, 1, 0)
   1  1          COURSE = COURSE * 6.283185
   1  1          CALL LRND(SEED, U, 1, 1, 0)
   1  1          IF (U .LE. 1.0/3.0) THEN
   2  1            SPEED = AVGSP*(24 + SQRT(3024*U + 16))/56
   2  1          ELSE
   2  1            SPEED = AVGSP*(60 - SQRT(1528 - 1512*U))/28
   2  1          END IF
   1  1          CALL LEXPN(SEED, T, 1, 1, 0)
```

```
 1 1                 LEGTM = T/LMBDA
 1 1               END IF
   1             IF (TOTTIM .LT. SCHTM .AND. DETTIM .GT. 9999.0) GO TO 35
   1             IF (DETTIM .LT. SCHTM) THEN
 1 1               NUMDET = NUMDET + 1
 1 1               WRITE(11,*) DETTIM
 1 1             END IF
   1     110 CONTINUE
             PRINT*, 'NUMBER OF DETECTIONS =', NUMDET
             STOP
             END
```

```
      SUBROUTINE DTCT(R, X1, Y1, X2, Y2, XC, YC, DET, DSTDET)
      REAL R, RSQ, X1, Y1, X2, Y2, XC, YC, XP, YP, M, DSQ
      REAL D12SQ, D1PSQ, D2PSQ, Z, DSTDET
      INTEGER DET
      RSQ = R * R
      IF (X1 .EQ. X2) THEN
         XP = X1
         YP = YC
      ELSE IF (Y1 .EQ. Y2) THEN
         XP = XC
         YP = Y1
      ELSE
         M = (Y2 - Y1)/(X2 - X1)
         XP = (XC/M + YC + M*X1 - Y1)/(M + 1/M)
         YP = M*XP - M*X1 + Y1
      END IF
      DSQ = (XC - XP)*(XC-XP) + (YC - YP)*(YC - YP)
      DET = 0
      IF (DSQ .LT. RSQ) THEN
         D12SQ = (X1 - X2)*(X1 - X2) + (Y1 - Y2)*(Y1 - Y2)
         D1PSQ = (X1 - XP)*(X1 - XP) + (Y1 - YP)*(Y1 - YP)
         D2PSQ = (XP - X2)*(XP - X2) + (YP - Y2)*(YP - Y2)
         IF ((XC - X1)*(XC - X1) + (YC - Y1)*(YC - Y1) .LT. RSQ) THE
            DET = 1
            DSTDET = 0.0
         ELSE IF ((D1PSQ .LT. D12SQ) .AND. (D2PSQ .LT. D12SQ)) THEN
            DET = 1
            Z = SQRT(RSQ - DSQ)
            DSTDET = SQRT(D1PSQ) - Z
         ELSE IF ((X2 - XC)*(X2 - XC) + (Y2 - YC)*(Y2 - YC) .LT. RSQ
     +   THEN
            DET = 1
            Z = SQRT(RSQ - DSQ)
            DSTDET = SQRT(D1PSQ) - Z
         END IF
      END IF
      RETURN
      END
```

LIST OF REFERENCES

1. Wagner, D.H., *Naval Tactical Decision Aids,* Naval Postgraduate School, CA, September 1989.

2. Stone, L.D., Trader, D.A., Davison, M.E., and Corwin, T.L., *Technical Documentation of SALT (Version 1.1)* METRON, Inc., VA, 4 February 1987.

3. *User's Manual for SALT Version 1.1,* METRON, Inc., VA, 27 October 1986.

4. Koopman, B.O., *Search and Screening,* Operations Evaluation Group Report No. 56, Center for Naval Analysis, Alexandria, VA, 1946.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Technical Information Center Cameron Station Alexandria, VA 22304-6145	2
2.	Library, Code 52 Naval Postgraduate School Monterey, CA 93943-5002	2
3.	Naval Ocean Systems Center Code 624 San Diego, CA 92152-5000	1
4.	Metron, Inc. 1481 Chain Bridge Road Suite 203 McLean, VA 22101	1
5.	Department of the Navy OPNAV-81 Washington, D.C. 20350	2
6.	Military Operations Research Society 101 Whiting Street Alexandria, VA 22304	2
7.	COMSUBDEVRON TWELVE U.S. Naval Submarine Base New London Groton, CT 06340	1
8.	Dave Samuelson SONALYSTS, Inc. 215 Parkway North P.O. Box 280 Waterford, CT 06385	1
9.	Daniel H. Wagner, Associates Hqtrs. and Paoli Office Station Square One Paoli, PA 19301	1
10.	LT. Vance S. Tisdale 3241 Rothschild Drive Pensacola, FL 32505	1