NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

SUBMARINE PERISCOPE DEPTH COURSE SELECTION TACTICAL DECISION AID

by

D. J. Danko

December, 1997

Thesis Advisor: Alan R. Washburn
Second Reader: James N. Eagle

Approved for public release; distribution is unlimited.

REPORT DOCUMENTATION PAGE

Form Approved OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE December, 1997	3. REPORT TYPE AND DATES COVERED Master's Thesis

4. TITLE AND SUBTITLE SUBMARINE PERISCOPE DEPTH COURSE SELECTION TACTICAL DECISION AID

5. FUNDING NUMBERS

6. AUTHOR(S)
Danko, D. J.

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Naval Postgraduate School
Monterey, CA 93943-5000

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

10. SPONSORING / MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES
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.

12a. DISTRIBUTION / AVAILABILITY STATEMENT
Approved for public release; distribution is unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT *(maximum 200 words)*
Coming to periscope depth is one of the most intensive of the routine submarine operations. Errors in Fire Control and Sonar System information serve to produce uncertain contact solutions that complicate the decision of selecting a safe course. The model developed in this thesis simulates a specified number of trials on each possible course, with the measure of effectiveness for each course being the probability of the course being acceptable with respect to specified minimum range criteria. The model outputs a geographic display and a graph of the measures of effectiveness versus course.

14. SUBJECT TERMS
Submarine Periscope Depth Operations, Course Selection, Tactical Decision Aid, Simulations using Visual Basic Programming

15. NUMBER OF PAGES 64

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFI- CATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	UL

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. 239-18

SUBMARINE PERISCOPE DEPTH COURSE SELECTION TACTICAL DECISION AID

D. J. Danko
Lieutenant, United States Navy
B.S.E., University of Washington, 1990

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN OPERATIONS RESEARCH

from the

NAVAL POSTGRADUATE SCHOOL
December 1997

Author: ______________________
D. J. Danko

Approved by: ______________________
Alan R. Washburn, Thesis Advisor

____
James N. Eagle, Second Reader

____
Richard E. Rosenthal, Chairman
Department of Operations Research

ABSTRACT

Coming to periscope depth is one of the most intensive of the routine submarine operations. Errors in Fire Control and Sonar System information serve to produce uncertain contact solutions that complicate the decision of selecting a safe course. The model developed in this thesis simulates a specified number of trials on each possible course, with the measure of effectiveness for each course being the probability of the course being acceptable with respect to specified minimum range criteria. The model outputs a geographic display and a graph of the measures of effectiveness versus course.

TABLE OF CONTENTS

EXECUTIVE SUMMARY

Coming to periscope depth is one of the most intensive of the routine submarine operations. Errors in Fire Control and Sonar System information serve to produce uncertain contact solutions that complicate the selection of a safe course. The model developed in this thesis simulates a specified number of trials on each possible course, with the measure of effectiveness for each course being the probability of the course being acceptable with respect to specified minimum range criteria. The model outputs a geographic display and a graph of the measures of effectiveness versus course.

Specifically:

1. The model determines all safe courses with respect to either of two user-specified range criteria using a simulation model. The model is in a readily available programming language, Excel, with minimal yet thorough manual inputs from available data sources on board. This course scoring includes constraints pertaining to the current and time-advanced contact situation, the errors associated with each term defining the contact situation, and the directions of seas. Data inputs are limited to realistic information available for a timely ascent.

2. The model formulates a standard tactical graphical presentation depicting the scoring of all available courses. This visual aid displays:

    a) range and bearing to each contact,

    b) range and bearing probability ellipse for each contact,

    c) the 30 minute dead reckon true track for each contact,

    d) the scoring of each course for the given scenario,

    e) a table format summary of the input information.

3. The model provides a visual aid displaying the measures of effectiveness for each course. The visual aid also displays the preferred course sector for the given direction of seas.

I. INTRODUCTION

One of the most perilous of the routine submarine shiphandling operations is proceeding to periscope depth. This operation is a prelude to several evolutions, most notably surfacing. A course must be decided upon that will ensure safe completion of the ascent. A safe course is any course with an acceptably low collision risk, essentially zero, with any surface contact. A safe course must keep all contacts in excess of some specified minimum range, and it must be selected expeditiously. Three main factors contribute to the level of danger:

1. inaccuracies of Fire Control solutions and Sonar information,
2. own ship's presence and intentions are unknown to surface contacts,
3. environmental conditions.

While several sources of information are available, both equipment and operator induced inaccuracies must be considered. Collection of further information with own ship's maneuvers for solution refinement may be limited due to a need for a timely ascent.[1] Perhaps most dangerous, environmental conditions may contribute to partial or complete masking of contacts and limit the range of feasible courses. In areas of very high contact density, a submarine may have as many as ten significant contacts to account for during the ascent. To relate the difficult nature of this maneuver, one could imagine landing a plane with no windows in an active parking lot full of vehicles unaware of the plane's descent. Additionally, each vehicle present will have the right of way with respect to the approaching aircraft.

The objective of this thesis is to develop a tactical decision aid to assist the Officer of the Deck in determining the acceptable courses upon which to proceed to periscope depth. An acceptable course will place own ship clear of any contacts or other navigational hazards. Additionally for enhanced ship control, courses in a $\pm 30°$ sector on either side of the approaching seas are preferred.

1. Emergency ascents will not be covered.

The next chapter provides background information relevant to the problem and the model development. Chapter III describes the methodology for the problem formulation including any pertinent assumptions. Chapter IV addresses the development of the model. Chapter V provides an overview of the analysis performed on the output data with respect to variations of the input information. The final chapter states the conclusions.

II. ASCENT TO PERISCOPE DEPTH

A. BACKGROUND

A submarine comes to periscope depth for a variety of reasons in addition to performing a visual search. Operations at periscope depth are necessary for receipt of communications, navigational fixes, ventilation of ship, and as a prelude to surfacing.

In all but an emergency ascent or surfacing, some preliminary maneuvering is required to search for yet ungained contacts concealed by the baffles or the thermal layer. The baffles are the bearings along which own ship's sonar equipment has reduced capabilities due to the location of the sonar sensor on own ship. For example, a forward mounted sonar array will not detect contacts in a given sector aft of own ship. The baffles are much like a driver's blind spot. Unlike checking the mirrors when changing lanes, a quick glance and some additional speed are not sufficient. The effects of a thermal layer will be covered in Section 2.a below.

Own ship's maneuvers are used to resolve the essential parameters which define the motion of the contact(s). The process by which these parameters are determined is known as Target Motion Analysis, TMA. By use of judicious maneuvers and the conservative assumption of a closing contact, the essential information upon which to base a periscope depth course decision can be obtained in a timely manner. As critical as this periscope depth evolution is, only a few basic parameters about each contact are required. While several techniques are available for refining the contact data to determine the greater details of its motion, these greater details are not required to determine an acceptable course upon which to proceed to periscope depth. Consistent with a submarine's need to remain undetected, only passive TMA techniques are employed.

Ship's speed is limited in the ascent to periscope depth due to the hydrodynamic force exerted on the extended periscope mast and fairing. This speed limitation also somewhat restricts own ship's maneuverability due to the reduced steerageway. This effect is due to the change in the amount of hydrodynamic force exerted on the same rudder area at lower ship's speeds. The reduced maneuverability serves to intensify the evolu[illegible]. It should be noted that preascent maneuvering does not have this same speed limitation.

Now that the basic problem has been stated, the factors affecting the periscope depth course will be addressed.

B. FACTORS

1. Contact Situation

a. Contact Motion Parameters

A contact is any detected underway vessel. When considering possible periscope depth courses, each contact's motion must be considered. The essential parameters are:

1. relative position, determined by range and bearing from own ship,
2. true motion, determined by course and speed of each contact.

Determination of these parameters is possible through Bearing-Only TMA or from the Fire Control and Sonar Systems.

b. Associated Errors

The major problem with using the information from these latter sources is the errors associated with the generation of the information. These errors are both intrinsic in the equipment, be it from physical design or analytical methods used, and operator-imposed through human error, interpretation, or level of experience. The method by which these errors will be accounted for will be through simulation modeling focused on determining the effects of such errors and a probabilistic scoring method to display the viability of any available courses in the presence of these uncertain errors.

2. Environmental Conditions

In the broad spectrum of environmental conditions that affect submarine operations, only two significantly affect the periscope depth course chosen - one directly and one indirectly.

a. Thermal Layer

The presence or absence of a thermal layer can impact the contact scenario by concealing ungained contacts until the final ascent. Due to refraction, sound waves traveling through distinct transitions in density of the given medium, in this case seawater, are deflected or bent away from the regions of lower density, or warmer water. Therefore a warmer layer of seawater below the surface will cause surface noise to bend back towards the surface and sounds generated below the layer to be deflected towards the ocean bottom. As a result, sensors below the layer will not detect surface noise. The stronger the thermal layer, the greater the potential for gaining new contacts on ascent. The error terms associated with such contacts are greater due to the lack of solution refinement.

b. State and Direction of Seas

The direction of seas is the direction from which the seas are approaching. This direction is determined by acoustic trace patterns on certain sonar displays. Similar aural techniques can be used to determine sea state, an important factor in deciding how much relative weighting should be given to the direction of seas. The higher the sea state the more prominently shiphandling will be affected. It is preferred to come to periscope depth with the direction of seas within 30 degrees of the bow. Courses in this sector aid in ship control by maximizing the relative speed between own ship and the local current. This not only assists in maximizing steerageway, but also limits the two undesirable effects described in the next two paragraphs.

If the chosen course is perpendicular or nearly perpendicular to the direction of seas, excessive roll may be experienced, potentially limiting ship control.

If the chosen course is along or nearly along the direction of seas, ship's control is affected due to reduced steerageway. More importantly, with the seas coming up the stern, the relative speed of the seas with respect to own ship's speed is reduced. Because of this, own ship is more susceptible to the wave motion. As the waves travel along the longitudinal axis of the boat, depth control becomes difficult due to the "porpoising" effect caused by the resulting undulating pitch motion.

These effects are amplified in higher sea states and are next to negligible in calm seas.

3. Geographic or Navigational Constraints

Geographic or navigational constraints will not be considered in this model, as they are easily accounted for in the course decision. The option does exist to allow entry of a navigational hazard as a contact with no speed yet still possessing the error terms associated with its range and bearing.

III. METHODOLOGY

The methodology will be addressed from the standpoint of a single simulation trial. The model simulates over all possible own ship's courses to determine the acceptability of each course with respect to each specified range criterion encountered over a given time on course.

A. INPUT DATA

The simulation requires manual entry of all parameters. This entry is via dialog boxes. Specifics of the entry methods will be addressed in Chapter IV.

1. Required Input

The required parameters are categorized according to their use within the model. The parameters are:

1. Scenario Parameters
    a) *NumberOfContacts*, Number of Contacts to be entered, maximum of five
    b) $Rh_{ACCMINRH}$, Acceptable Minimum Range to any contact during time on course, in yards
    c) Rh_{SAFETY}, Safety Range to any contact during time on course, in yards, always less than $Rh_{ACCMINRH}$
    d) *TimeOnCourse*, Time on Course, in minutes
    e) *DMho*, Own Ships Speed, in knots
2. Simulation Parameters
    a) *NumberOfTrials*, Number Of Trials per course to be run
    b) *StepSize*, analyze every n^{th} course, integer value from one to seven
3. Contact Motion Parameters (for each contact)

a) *Ct*, Target Course, in degrees

b) *DMht*, Target Speed, in knots

c) *Rh*, Range, in yards

d) *By*, True Bearing to Contact, in degrees

e) *CtSigma*, Course Error, in degrees

f) *DMhtSigma*, Speed Error, in knots

g) *RhSigma*, Range Error, in yards

h) *BySigma*, Bearing Error, in degrees

B. TARGET MOTION ANALYSIS

All Target Motion Analysis is performed in the relative frame of reference. This accommodates determination of the range and time of closest point of approach as well as direct use of error terms in the simulation. All TMA is performed with own ship and all contacts on constant course and speed.

1. Translation between Coordinate Systems

Translation from the polar coordinate parameters of range and bearing is performed to allow use of Cartesian coordinates both in the determination of time of closest point of approach, t_{CPA} and the minimum range encountered, Rh_{MIN}. All computations are performed using the standard mathematical axes versus the standard tactical axes. Doing so simplifies the required computations within the simulation. The translation to the standard tactical axes is performed only as required to generate the geographic display.

The translation from polar to Cartesian coordinates is:

$$X_{POS} = Rh \cdot \cos(By)$$
$$Y_{POS} = Rh \cdot \sin(By)$$

Similarly own ship's and target's course and speed are translated to Cartesian coordinates. *Co* is own ship's course.

$$X_O = DMho \cdot \cos(Co)$$
$$Y_O = DMho \cdot \sin(Co)$$
$$X_C = DMht \cdot \cos(Ct)$$
$$Y_C = DMht \cdot \sin(Ct)$$

These conversions to Cartesian coordinates also allow the direct computation of points necessary to produce the dead reckoned traces present in the Geographic Display Graph.

The input parameters of *DMho*, *Ct*, and *DMht* along with *Co* as generated by the simulation are used to determine the Cartesian components of relative motion, *xDMhr* and *yDMhr*.

$$xDMhr = \left(X_C - X_O\right)\text{kts} \cdot \left(33.75 \frac{\text{yds}}{\text{kts} \cdot \text{min}}\right)$$
$$yDMhr = \left(Y_C - Y_O\right)\text{kts} \cdot \left(33.75 \frac{\text{yds}}{\text{kts} \cdot \text{min}}\right)$$

2. Incorporating Error Terms into Target Motion Analysis Parameters

a. Error Term Distributions

Each of the four TMA parameters, *Rh*, *By*, *Ct*, and *DMht*; have an associated input error term, *RhSigma*, *BySigma*, *CtSigma*, and *DMhtSigma* respectively. Each error term is used as twice the value of a standard deviation in that parameter. The mean of each error distribution is assumed to be zero. The distributions of the error terms used by the simulation are independent Normal ($\mu = 0$, σ =*stated error/2*).

It is critical that the operator realize the mathematical transition of the stated error to the standard deviation of the error term distribution. The probability that a normal error falls within a stated ± interval varies as shown in Figure 1.

Stated Error =	Probability in ± Interval
σ	0.6826
2σ	0.9544
3σ	0.9974

Figure 1.

These probability values were found using the standard normal tables and the cumulative distribution function as follows:

$$p = P(X \geq k\sigma) - P(X \leq -k\sigma), \quad \text{for } k = 1,2,3.$$

These values demonstrate the judgmental subjectivity with manual assignment of the error values. One operator may state a ± interval gauged from the perspective that the interval will contain the contact 90% of the time whereas another operator may gauge that percentage differently. For the purposes of this thesis error values are to be expressed as two standard deviations. This is to say, the specified contact parameter will be within the error bounds with probability 0.9544. The input dialog includes a note above the error term entry boxes to this effect.

The normal distribution generates a probability density according to:

$$f(x) = \frac{1}{\sqrt{2\pi\sigma}} e^{\frac{-(x-\mu)^2}{2\sigma^2}}$$

The cumulative distribution is therefore:

$$F(x) = \frac{1}{\sqrt{2\pi\sigma}} \int_{-\infty}^{x} e^{\frac{-(x-\mu)^2}{2\sigma^2}} dx$$

As no closed form solution exists for the normal cumulative distribution, solving for x directly is not possible. Instead, a value of the cumulative probability, *F(x)*, is supplied and a numerical method is applied to attain a value of x. An iterative normal inversion function, NormInv(p = probability, μ = mean, σ = standard deviation), is available in Excel, but it is expensive in regards to computational time. As an alternative to calling NormInv(p, μ, σ) several times within each trial, the standard normal was discretized into 21 equally sized bins with the values of the standard normal curve being assigned to the elements of an array. In doing so, each bin has a value equal to the number of standard deviations from the mean for the equally sized steps of cumulative probability from 0 to 1 by increments of 0.05. The values of 0 and 1 are avoided to prevent faulty returns from the normal inversion function call. The values of 0.001 and 0.999 are used instead.

The curve, *F(x)*, used to generate the bin values is plotted below in Figure 2. The vertical axis is the uniform random probability. The horizontal axis is the number of standard deviations from the mean.

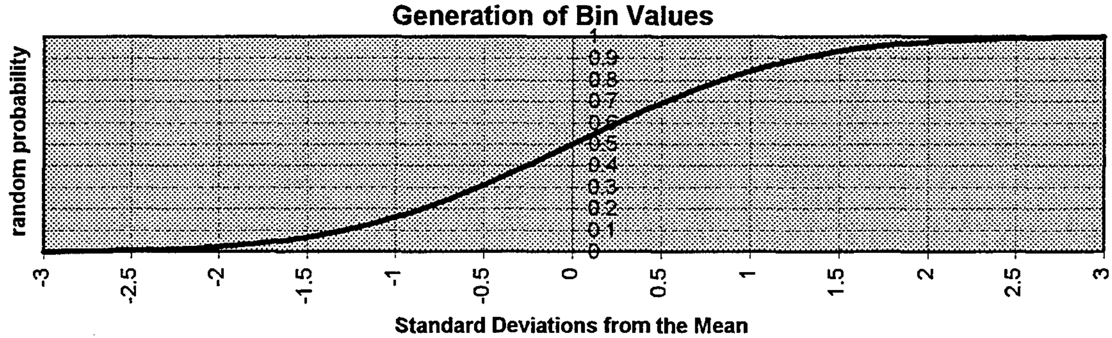


Figure 2.

A uniform random number is generated, multiplied by the number of bins, and truncated. This truncated value is the index of the array element to be used in determining the error term. The array element is multiplied by the standard deviation, σ.

$$p = \text{random number}$$
$$index = \text{int}(p * bins)$$
$$error = array(index) * \sigma$$

The resulting value is an approximately normal error term with a mean of zero and a standard deviation of the stated error, 2σ divided by 2.

The resulting error terms for *Rh, By, Ct,* and *DMht* are *RhError, ByError, CtError,* and *DMhtError* respectively. These values are regenerated with each trial and incorporated as follows:

$$trialX_P = (Rh + RhError) \cdot \cos(By + ByError)$$
$$trialY_P = (Rh + RhError) \cdot \sin(By + ByError)$$
$$trialX_C = (DMht + DMhtError) \cdot \cos(Ct + CtError)$$
$$trialY_C = (DMht + DMhtError) \cdot \sin(Ct + CtError)$$

The trial values of the parameters are sent to the subsequent function calls for determination of the closest point of approach.

b. The Independence Assumption

To justify the use of independence between the associated error terms, a dependent case example is presented as a sensitivity test. In passive TMA, contact bearing rate determines the contact speed across the line of sight rather accurately, [Ref. 2.]. This value is key to determining *Ct* and *Dmht*. Any *DMhtError* induces a corresponding *CtError* in the dependent case as *DMht* is used in attaining *Ct* or vice versa. As passive TMA may be the sole technique used by a submerged submarine, this case of *(Ct,DMht)* dependency is of interest. The dependent version of the model applies dependence between *Ct* and *DMht* as follows:

Let *xDMht* be contact speed across the line of sight, assumed known

Let *depDMht* be the dependant *DMht*

$$xDMht = DMht \cdot \sin(By - Ct)$$
$$depDMht = \frac{xDMht}{\sin(trialBy - trialCt)}$$

It should be noted that this case of dependence is one of many possible dependent relationships. This particular case is chosen for its straightforward nature. The results of the dependent version of the model will be addressed in Chapter V.

The input error terms for the simulation model are an instantaneous evaluation of the contact's solution accuracy as of the input time. The model makes no attempt to modify the magnitude of any error terms over the specified *TimeOnCourse* parameter. This is consistent in that the model does not account for any solution refinement or lack thereof during the time on course considered.

3. Minimum Range Determination

The time of closest point of approach is the time at which the relative motion vector is perpendicular to own ships position. To determine the time at which CPA occurs, the range function is minimized.

$$Rh(t) = \sqrt{(X_{POS} + xDMhr \cdot t)^2 + (Y_{POS} + yDMhr \cdot t)^2}$$

To find the time at which range is a minimum, the first derivative with respect to time is taken.

$$\frac{d}{dt}\left(Rh(t)\right) = \frac{d}{dt}\left(\sqrt{(X_{POS} + xDMhr \cdot t)^2 + (Y_{POS} + yDMhr \cdot t)^2}\right)$$

$$\frac{d}{dt}\left(Rh(t)\right) = \frac{2\left[xDMhr \cdot (X_{POS} + xDMhr \cdot t) + yDMhr \cdot (Y_{POS} + yDMhr \cdot t)\right]}{2\sqrt{(X_{POS} + xDMhr \cdot t)^2 + (Y_{POS} + yDMhr \cdot t)^2}}$$

By setting the first derivative of the range function to zero and solving for time.

$$xDMhr \cdot (X_{POS} + xDMhr \cdot t) + yDMhr \cdot (Y_{POS} + yDMhr \cdot t) = 0$$

$$xDMhr \cdot X_{POS} + xDMhr^2 \cdot t + yDMhr \cdot Y_{POS} + yDMhr^2 \cdot t = 0$$

$$t = t_{CPA} \equiv \frac{-xDMhr \cdot X_{POS} - yDMhr \cdot Y_{POS}}{xDMhr^2 + yDMhr^2}$$

The geometry of the problem dictates this value to be a minimum. Next, t_{CPA} is used to determine the minimum range experienced to the given contact over the *TimeOnCourse*.

If t_{CPA} is negative, or equivalently, before the time interval specified by *TimeOnCourse*, the contact is now opening in range; therefore the minimum squared range is:

$$Rh_{MIN}^2 = (Rh(0))^2$$

If t_{CPA} is within the time interval as specified by *TimeOnCourse*, then the minimum squared range is:

$$Rh_{MIN}^2 = \left(Rh\left(t_{CPA}\right)\right)^2$$

If t_{CPA} is greater than *TimeOnCourse*, then the contact closes in range until the end of the time interval, so

$$Rh_{MIN}^2 = (Rh(TimeOnCourse))^2$$

This illustrates an important consideration in using this model in that it does not account for contact ranges at times beyond that specified by the Time on Course parameter.

C. MEASURES OF EFFECTIVENESS

The measures of effectiveness, $MOE_{ACCMINRH}$ and MOE_{SAFETY}, are the percentage of trials for which all contacts remain at ranges in excess of the acceptable minimum range or safety range on any given course. Equivalently stated, each *MOE* is the probability that the course is acceptable as defined by not violating the applicable range constraint. The squared range is again used.

The first step in determining the *MOEs* is to take the minimum of the vector of minimum squared ranges (the use of squared ranges avoids the square root function) to all contacts as follows:

Let Rh_i^2 be the minimum squared range to contact i, and

$$z = \min\left\{Rh_1^2 \ldots Rh_{NumberOfContacts}^2\right\}$$

If z is greater than the acceptable minimum range squared, then n, a counter variable, is incremented. If the value of the minimum is greater than the safety range squared, then k, another counter variable, is also incremented. After all simulation trials have been run, the *MOEs* are calculated as follows:

$$\text{Let } n_f \text{ be final value of } n$$
$$\text{Let } k_f \text{ be final value of } k$$
$$MOE_{ACCMINRH} = \frac{n_f}{trials}$$
$$MOE_{SAFETY} = \frac{k_f}{trials}$$

$MOE_{ACCMINRH}$ is then the fraction of trials for which all contacts remain in excess of $Rh_{ACCMINRH}$, and MOE_{SAFETY} is the fraction of trials for which all contacts remain in excess of Rh_{SAFETY}. The value of each *MOE* is then stored in an internal array and displayed on the worksheet "Graphs".

D. OUTPUT GRAPHS

Two distinct output graphs are produced to provide not only a display of the *MOEs* of each course, but also to visually verify the results against a geographic display of the specified contact scenario.

1. Measures of Effectiveness Graph

Each of the *MOEs* is plotted as a scatter graph versus the entire range of courses, 0 to 359. The $MOE_{ACCMINRH}$ is plotted in front of the MOE_{SAFETY}. This is possible as the value of $MOE_{ACCMINRH}$ is always less than the value of MOE_{SAFETY}, given correct inputs.

The use of scatter plotting assists in showing the acceptable course sector(s) regardless of the specified *StepSize*. The graph also displays the courses allowed by the safety range constraint in the event that few or no courses meet the acceptable minimum range constraint. Additionally, spikes displaying the ± 30° preferred course sector with respect to the direction of seas are included. See Appendix A. Output #1, 2, and 3.

2. Geographic Display Graph

The geographic display was created to provide for internal verification of the model. The graphs provide an overview of the contact's position, motion, and the probability ellipse for *Rh* and *By* errors. First an ellipse with axes of *Rh* error and *By* error is created in Cartesian coordinates. The ellipse is then rotated by an angle equal to the contact's bearing to align the ellipse such that the *Rh* error axis is along the radius of a line extending from the center out in the direction of the contact's bearing. The ellipse is then translated out to the contact's *Rh* and *By* through a conversion to Cartesian coordinates.

In addition to the *(Rh, By)* probability ellipse, the contact's course, *Ct* and speed, *DMht* are reflected by dead reckoning each contact in two minute intervals out to 30 minutes. This permits a visual indication of the contact's motion. It does not include any visual representation of the *Ct* and *DMht* errors.

The Geographic Display Graph provides not only for viewing the contact scenario on a standard tactical display but also shows the scoring of all courses with respect to the acceptable minimum range criteria as a value proportioned to the maximum initial range to any contact. This method shows points created by the following:

$$Score = \max\left\{Rh_1(0), \ldots, Rh_{NumberOfContacts}(0)\right\} \cdot MOE_{ACCMINRH}$$

Using this relationship, scores at the maximum range of any contact have $MOE_{ACCMINRH} = 1$, at 50% of maximum range, $MOE_{ACCMINRH} = 0.5$, etc. Similar scoring points for safety range course scoring were excluded so as to not visually overload the display. See Appendix A, Examples #1, 2, and 3.

IV. SIMULATION MODEL

A. DESIGN

This section will address the various design elements of the model. The model consists of approximately 410 lines of code and one worksheet to support the graphing features.

1. Programming Language

The model is coded in Visual Basic for Applications, Windows Version 4.0, within Excel Version 7.0 of the Microsoft Office 95 Suite. This programming language was chosen due to extensive versatility with respect to mathematical, statistical, and graphical display capabilities.

The choice was also in consideration of availability of the underlying program in the event the model is deemed useful as either a tactical decision aid or a training aid. Use of this readily available program allows any user access to the model with no additional programs or system capabilities beyond those of an average PC, once a copy of the program has been provided. Appendices B, D, and E contain the coding for the input, simulation, and basic output. Recreation of the graphs is left to the user or will be provided upon request.

2. Modularity

The source code in separated into three modules, "Data Entry", "Simulation", and "Main".

a. Data Entry Module

The “Data Entry” module includes the procedures required to create the graphical user interfaces, GUIs, necessary to allow the user to enter the required parameters. This module also contains all the control procedures for the GUIs. The initialization procedure is also contained within this module. Lastly, the procedure that sends the input parameters to the “Graphs” worksheet is located in this module. The “Data Entry” module code is presented in Appendix B.

b. Simulation Module

The “Simulation” module includes the simulation procedure and all the functions called by it. Also included in this module is the procedure to send the *MOE* matrix to the “Graphs” worksheet. The “Simulation” module code is presented in Appendix D.

c. Main Module

The “Main” module contains only the main procedure. The main procedure calls the modules that control each major portion of the model; data entry, graph inputs, simulation, and graph output. The “Main” module code is presented in Appendix E.

B. IMPLEMENTATION

1. Control Flow Path

Control begins with the implementation of the “Main” procedure. “Main” calls “PrepareDialogs” which in turn calls “InitializeContacts” and also initializes the edit boxes of both input dialog boxes. The procedure then displays each dialog in turn retrieving the data input by the user. The input data is assigned to the applicable variable not by "PrepareDialogs" but instead by the individual control's "Change" procedure.

After this the control returns to "Main". Next "Main" calls "SendInputToGraphs", which sends the input to the applicable cells within the "Graphs" worksheet for use in generation of the graphs. Control now returns to "Main".

Next "Main" calls "Simulate" by assignment to the variable "Data". The inner workings of the "Simulate" function will be covered in section 3 below. After the simulation has ended control returns to "Main".

Finally "Main" calls "SendDataToGraphs", which sends the *MOE* matrix to the "Graphs" worksheet for use in displaying the applicable course scores on each output display. Control then returns to "Main" and the program is terminated.

2. Input Dialogs

The "PrepareDialogs" procedure displays each dialog as called. The first dialog is contained in the "GetInitialData" dialogsheet. This dialog collects the scenario and simulation parameters. See Appendix C, Figure 1.

The second dialog is contained in the "GetContactData" dialogsheet. It retrieves the contact motion and error parameters for each of the specified number of contacts. See Appendix C, Figure 2.

It is important to note that the edit boxes require strictly numeric entries. Any transition of the cursor between edit boxes must be performed by exclusive use of the Tab key. Movement within the contact number dropdown list of the second dialog is more flexible.

3. Simulation

The simulation begins by initializing the decision variable, *Co*. Next, the specified number of trials is performed. Within each trial, each contact motion parameter of the current contact is modified by an error. With these new contact motion parameters the

"GetTimeToCPA" function is called. The returned value of t_{CPA} is used in the function call to the "GetMinSquaredRange" function. The value returned from this call is the squared value of the minimum range achieved to the current contact. This value is assigned to a dynamic array of length commensurate to the specified number of contacts. This process is repeated within each trial for the specified number of contacts.

After all contact's minimum squared ranges have been determined, the absolute minimum squared range for the current trial is determined by taking the minimum of the minimum squared range array. The value of the absolute minimum squared range is now compared to the specified values of $Rh_{ACCMINRH}$ squared and Rh_{SAFETY} squared. If the absolute minimum squared range is greater than or equal to the applicable range squared, the appropriate counter variable is incremented. This entire process is repeated for the specified number of trials.

After all the trials have been executed for the current value of *Co*, each counter variable is divided by the specified number of trials then the value is assigned to the *MOE* matrix being positioned by column according to the applicable range criteria and by row according to the current value of *Co*.

The *Co* value is then incremented and the entire process repeated for all remaining courses. The final step is to assign the *MOE* matrix to the value of the function variable.

4. Output Graphs

The form and formatting of each output graph exists on the "Graph" worksheet. Each run of the model updates the graphs to depict the most recent execution of the model. This method reduces execution time by not recreating the graphing forms or formatting with each execution of the model. It is possible to view the visual updating of the graphs once the simulation portion of the model has been executed.

C. TRANSPORTABILITY

1. Add-In Conversion

The model could be transitioned into an Add-In format for internal inclusion in any version of Excel 7.0 and beyond. Add-Ins also afford greater security by allowing password protection of the program.

2. Platform Performance

The model is capable of reasonable execution times on an average PC. As a reference, the model can execute a five contact 100 trial simulation in approximately 10 minutes on a 120Mhz machine. This is clearly not within the range required to make an expeditious course decision on a rapid ascent, though is usable in other scenarios. This time will decrease, of course, on faster PC's.

It is clear through step-oriented call tracing that actual simulation takes about 85% of the total execution time. The remaining execution time is used in writing values to the "Graph" worksheet. Although graphs can be created directly from the source code, the execution time is increased in the creation of all the graph formatting. The process of writing values to worksheets seems to take an undue amount of time. The data transfer to the worksheet is an area to explore in attempting model improvement.

V. ANALYSIS

The simulation will be analyzed with the use of three examples. These examples are not very realistic in that the contacts are arranged symmetrically and it most cases are assigned similar motion parameters. This was done to assist in displaying the model's characteristics and in no way represents any limitation of the model. The outputs are provided in Appendix A.

The simulation model is verified through use of reference problems from the Maneuvering Board Manual, Pub 217, [Ref. 1.], and through variations of all input parameters to ensure compliance with anticipated effects and current tactical guidance. All verification of the model is internal. No external agencies have assisted in the verification.

A. VIABILITY OF MEASURES OF EFFECTIVENESS

In Examples 1,2, and 3, compliance is evident between the range of acceptable courses shown by the *MOE* graph and the intuitive course decision based on choosing a course which has the contacts on the left, drawing left, and contacts on the right, drawing right. Additional courses are shown to be acceptable by the *MOE* graph as the simulation course decision does not prevent use of courses that permit any contact to cross the bow of own ship. Use of courses that do not allow any contact to cross the bow of own ship are always preferred. These courses provide a greater safety margin to a worsening situation caused by contact maneuvers towards own ship. The additional courses shown by the *MOE* graph are useful in scenarios which have limited courses based on the bearing rate method alone.

B. ERROR VALUE INFLUENCES

The error values used for each scenario are displayed in the summary table of each example. The geographic display shows the *(Rh,By)* probability ellipse based on these values. While no visual representation of *Ct* or *DMht* errors is available, the influence of all the error terms is evident in both the scoring ring of the geographic display and on the *MOE* graph. Example #1 in Appendix A most clearly displays the effects of the magnitude of the error terms.

Contact 1, bearing 060°, has the lowest magnitude of error. The influence of the error terms is evident by inspection of the edges of the scoring ring in the geographic display, and in the edges of the scatter graphs of each *MOE* in the vicinity of each contact. The edges in both displays are steeper and have less variation along their slope nearer to Contact 1. The edges in the vicinity of Contacts 2 and 3 are much less steep and show greater variation in the scoring ring and the scatter graph slopes. Also the peaks on either side of lower error term contacts are higher and the valleys are lower. This verifies the correct inclusion of the error terms in that the greater the magnitude of the error terms, the less distinct the *MOE* graph results are for the given situation, as is evident due to the symmetry of the contact situation. Example #3 also clearly displays the effects of the magnitude of the stated error terms.

Example #2 is highly asymmetrical. The asymmetry is in support of the discussion in sections C and E below. The range of acceptable courses from 95° to 200° concurs with the author's intuitive course selection made through inspection of the contact scenario alone.

Example #3 displays the justification of the independence assumption of *Ct* and *DMht*. Contacts 1 and 2, shaded in the summary table, are formulated with the dependent version of the model coding, see Section B.2.b of Chapter III. Contacts 3 and 4 are formulated using the standard independent version. The *RhError* and *ByError* for all contacts are set to zero to isolate the effects of the *Ct* and *DMht* dependency. The asymmetry in the *MOEs* and scoring ring about the North-South axis displays the effects of the independence assumption. In the *MOE* graph, the dependent results have flatter peaks and valleys though steeper edges. The independent case is conservative with respect to the dependent in that a more narrow range of courses is acceptable for contacts with the same error terms. The use of independent error term distributions also simplifies computations within the inner-most loop of the simulation model, thereby reducing execution time.

C. SAFETY RANGE VS. ACCEPTABLE MINIMUM RANGE

The inclusion of the safety range *MOE* is valuable in that it allows visual representation of any additional acceptable courses with respect to the lower of the range criteria. This element of the model is also very useful in showing courses acceptable with respect to the safety range criterion when no courses meet the acceptable minimum range criterion. Example #2 demonstrates an instance of this case about the course of 265°.

D. CONTROL OF ACCURACY AND EXECUTION TIME

1. Accuracy

The values of the simulation parameters dictate the accuracy of the output. A higher *NumberOfTrials* per course results in greater accuracy in the *MOE*'s. A lower value of *StepSize* results in more thorough information by analyzing a greater portion of all available courses.

2. Execution Time

Execution time is roughly proportional to the values of *NumberOfTrials* and *NumberOfContacts,* and inversely proportional to *StepSize*. The greater the *NumberOfTrials* and/or the greater the *NumberOfContacts,* the greater the execution times. The greater the *StepSize*, the lower the execution times. The user can select values for *NumberOfTrials* and *StepSize*. The *NumberOfContacts* is limited to a maximum of 5 for execution time concerns. The user must consider the tradeoff between the exclusion of any contact and the lack of consideration, by the model, of that contact over the entire *TimeOnCourse*. If contacts are to be screened for entry, the contact must be deemed not a concern for the entire *TimeOnCourse* for any choice of own ships course. Reference values of execution times on a 120MHz PC are as follows:

NUMBEROFTRIALS	*STEPSIZE*	*NUMBEROFCONTACTS*	**EXECUTION TIME**
100	1	5	~10 min
100	3	5	~ 5 min
1000	6	5	~ 11 min

Example #2 demonstrates the difference in the *MOE* graph display for a *StepSize* other than 1.

E. DIRECTION OF SEAS

The model does not make any attempt to mathematically incorporate the direction of seas into the simulation. The relative weighting of direction of seas with respect to course selection is dependent on several factors, most notably the sea state. Due to the very subjective nature of the incorporation of the direction of seas into the course decision, the model only displays the preferred course sector with respect to the direction of seas. Example #2 displays a case in which several courses within the preferred sector are acceptable with respect to the Rh_{SAFETY} criteria, while none are acceptable with respect to the $Rh_{ACCMINRH}$ criteria.

VI. CONCLUSIONS

A. MODEL PURPOSE

The model provides useful tactical information given standard and readily available inputs. Reference problems and multiple challenging trial scenarios demonstrate compliance with the primary course selection criterion. Furthermore, the model clearly displays courses acceptable with respect to the specified range criteria in a single display, a valuable resource not currently available.

B. THESIS OBJECTIVES

The simulation model provides a viable base from which to continue development. The proposed tactical decision aid is useful in many scenarios in addition to the periscope depth scenario. The need for further development lies mainly in reducing execution times so as to allow use of up to 1000 trials per course thereby providing greater accuracy and reliability, very necessary qualities in data used towards this level of decision.

C. TACTICAL USE

The primary inhibitor in immediate tactical use is external validation. Accreditation of the model would need to follow. The attractive feature with respect to the development is that once these processes are complete, the transition to onboard use is expeditious as the model is developed for use on a PC.

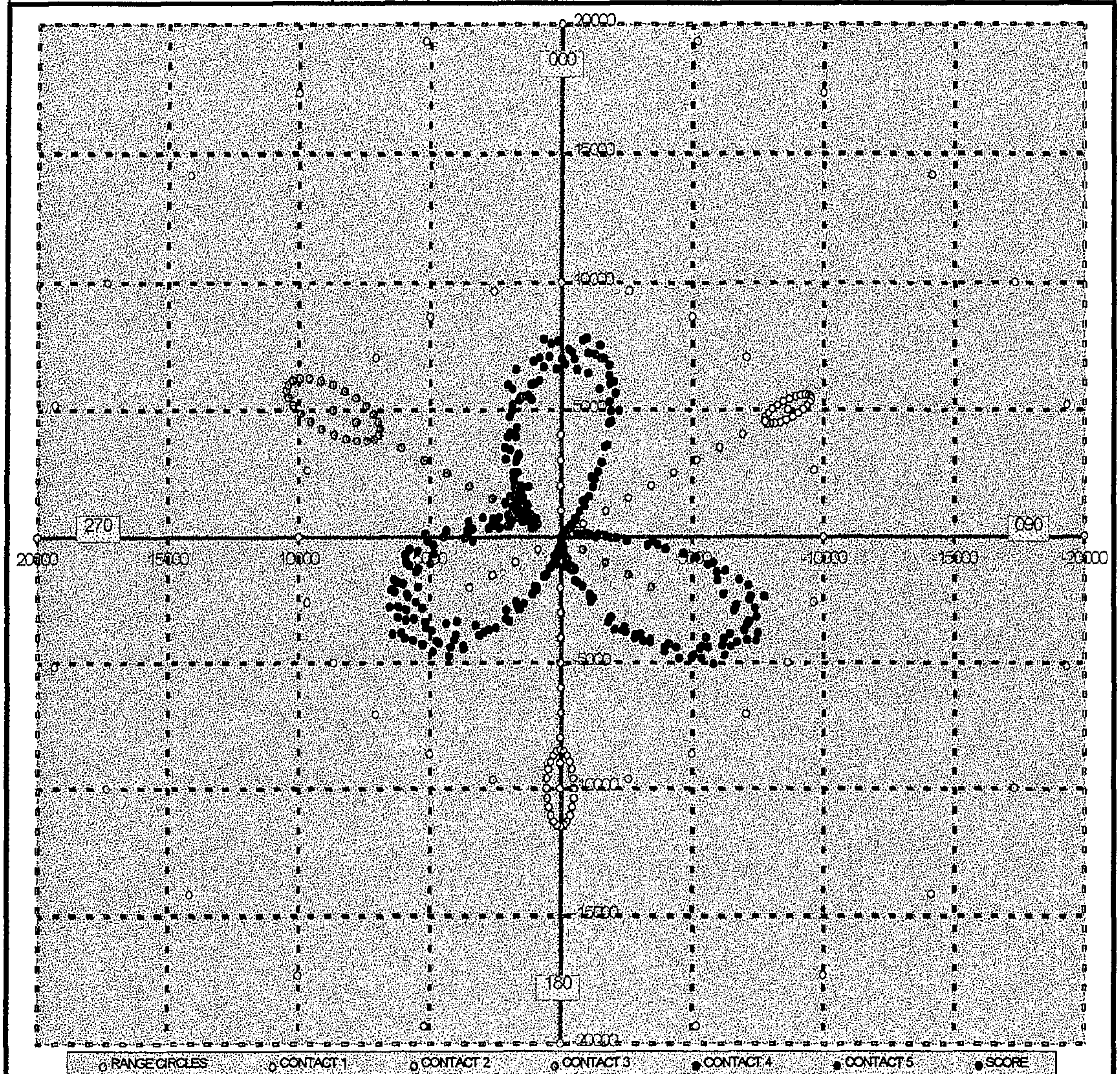
D. TRAINING USE

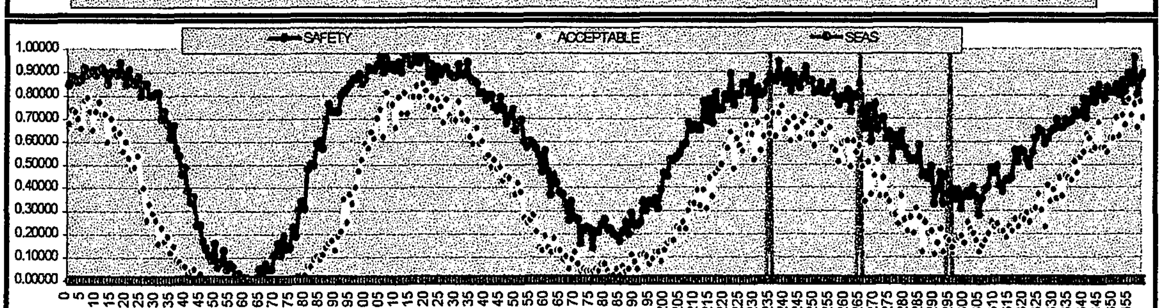
The model has potential for use as a training aid, where the problem of execution times is less restrictive. The development of a random contact generator would greatly increase the value as a training aid by allowing the user to run more scenarios in the same amount of time.

While the model was developed primarily to focus on the periscope depth issue, it is potentially useful in any surfaced or submerged course selection scenario.

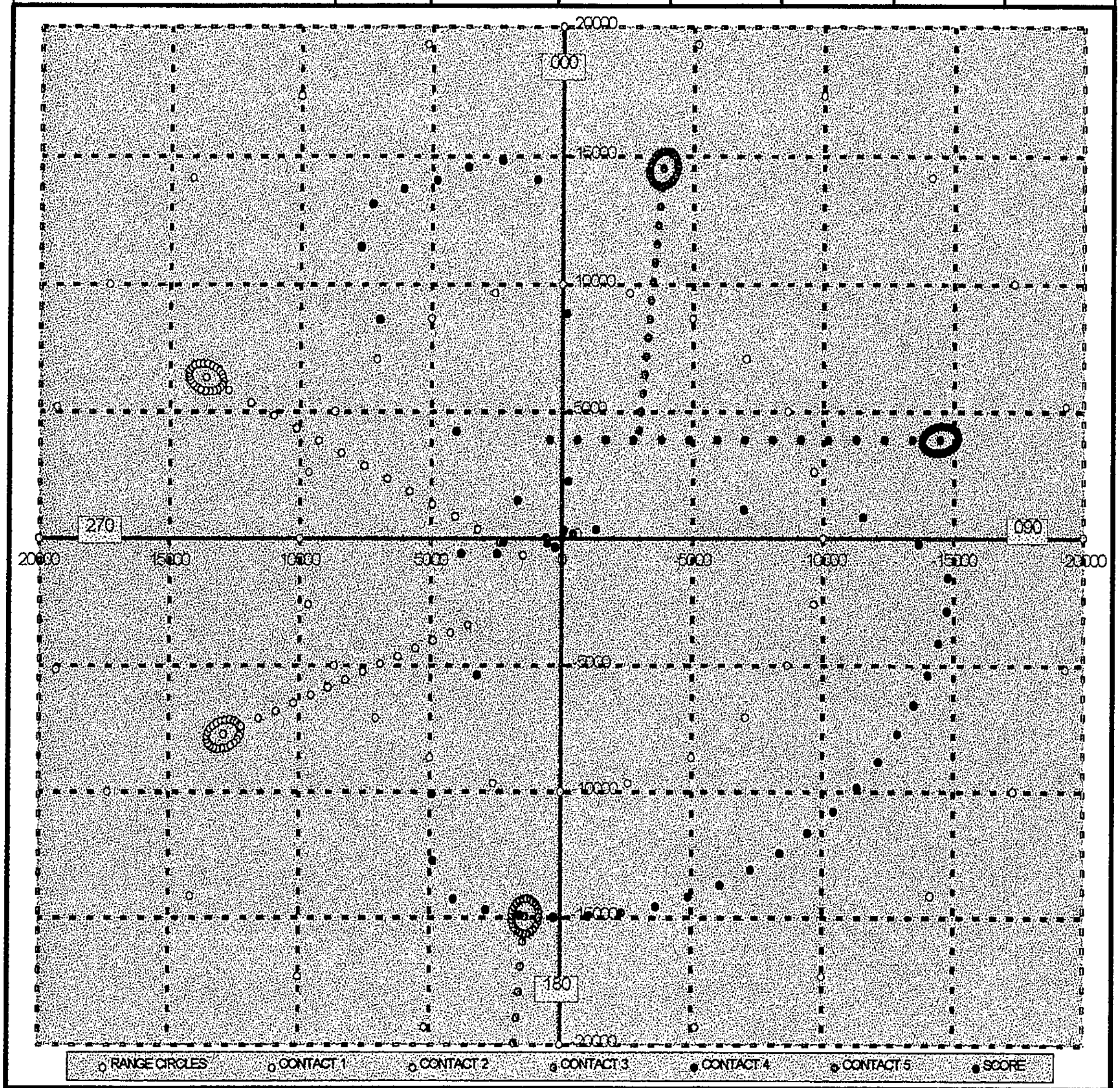
APPENDIX A. COMPOSITE OUTPUT DISPLAYS

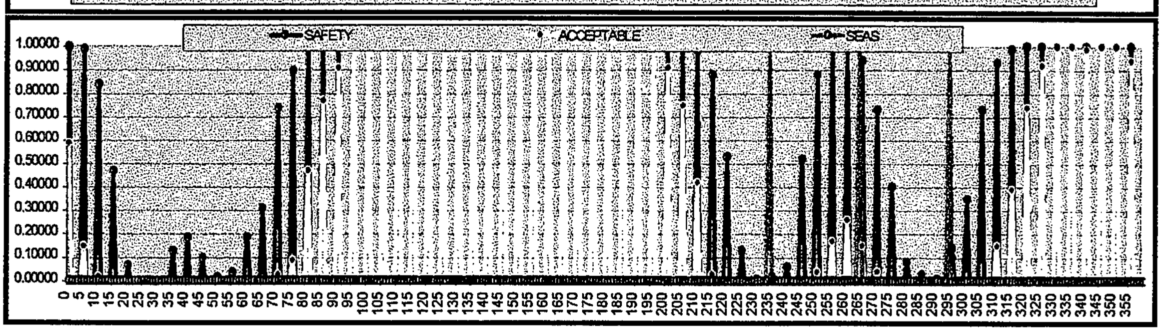
EXAMPLE #1 Scenario Summary			Contact 1	Contact 2	Contact 3	Contact 4	Contact 5
Number Of Contacts	3	Ct	240	0	120	0	0
Acceptable Minimum Range	2500	Dmht	15	15	15	0	0
Safety Range	1500	Rh	10000	10000	10000	0	0
Time On Course	30	By	60	180	300	0	0
Own Ships Speed	10	Ct Error	10	20	30	0	0
Direction of Seas	265	Dmht Error	3	5	7	0	0
Trials per Course	100	Rh Error	1000	1500	2000	0	0
Step Size	1	By Error	2	3	5	0	0



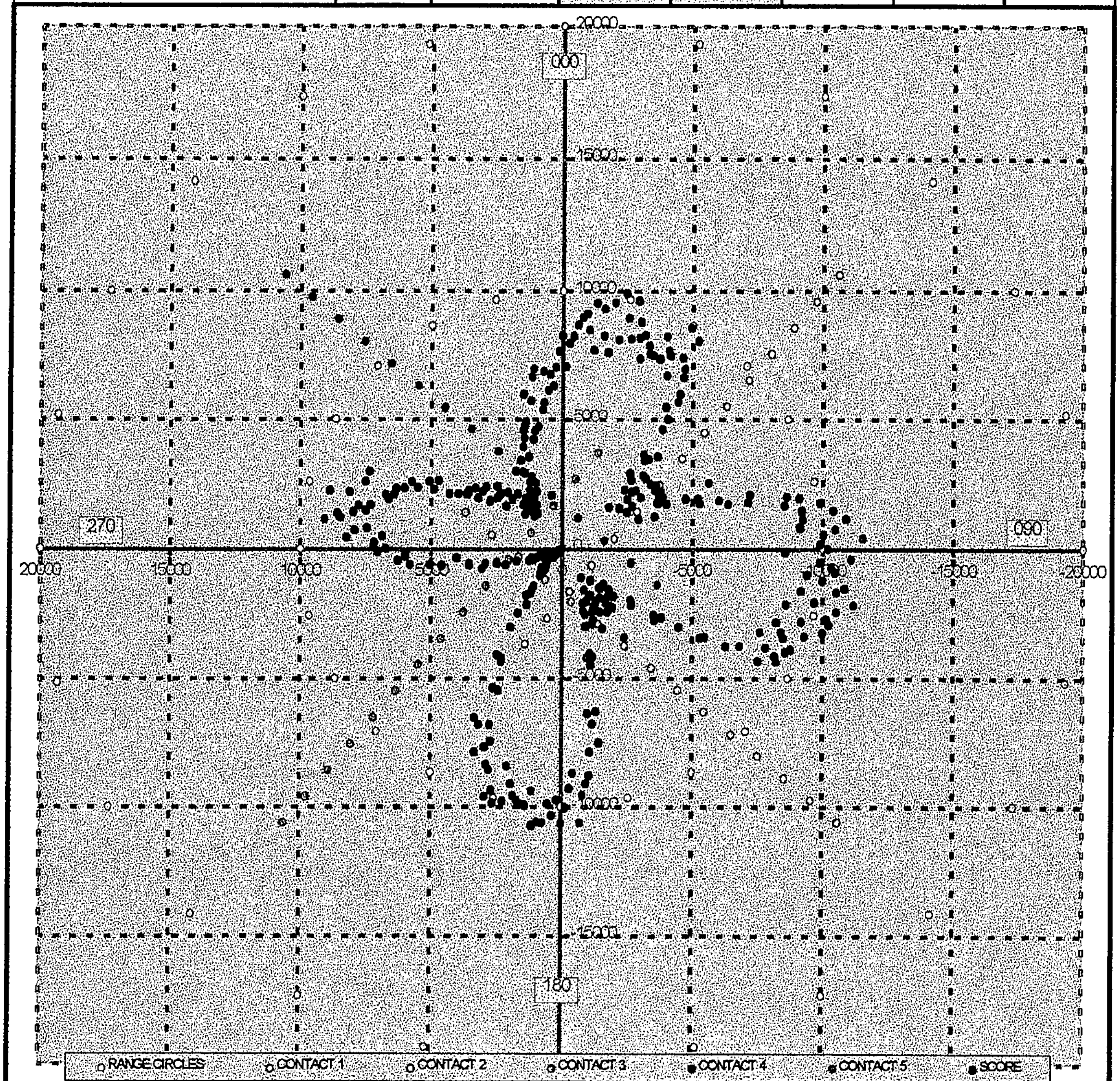


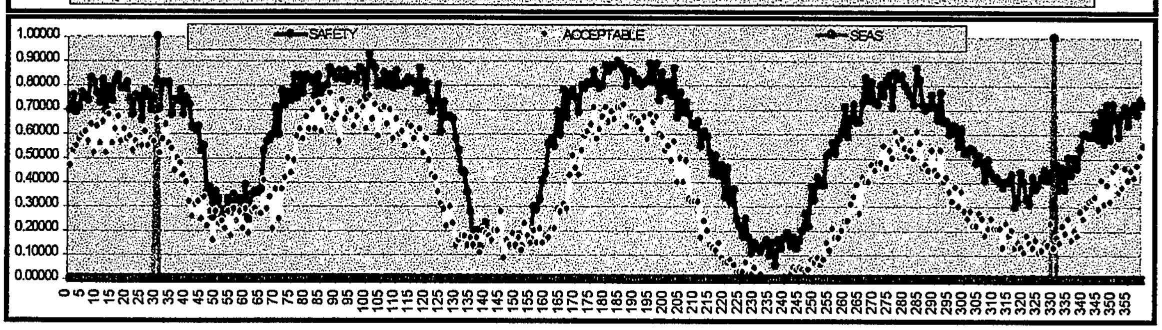
EXAMPLE #2 Scenario Summary			Contact 1	Contact 2	Contact 3	Contact 4	Contact 5
Number Of Contacts	5	Ct	120	65	185	270	185
Acceptable Minimum Range	3000	Dmht	15	11	15	16	11
Safety Range	1500	Rh	15000	15000	15000	15000	15000
Time On Course	30	By	295	239	185	75	15
Own Ships Speed	10	Ct Error	5	5	5	5	5
Direction of Seas	265	Dmht Error	3	3	3	3	3
Trials per Course	100	Rh Error	700	700	700	700	700
Step Size	5	By Error	2	2	2	2	2





EXAMPLE #3 Scenario Summary			Contact 1	Contact 2	Contact 3	Contact 4	Contact 5
Number Of Contacts	4	Ct	220	310	40	130	0
Acceptable Minimum Range	2500	Dmht	20	20	20	20	0
Safety Range	1500	Rh	15000	15000	15000	15000	0
Time On Course	30	By	45	135	225	315	0
Own Ships Speed	10	Ct Error	20	10	10	20	0
Direction of Seas	0	Dmht Error	N/A	N/A	5	10	0
Trials per Course	100	Rh Error	0	0	0	0	0
Step Size	1	By Error	0	0	0	0	0





APPENDIX B. VISUAL BASIC SOURCE CODE FOR DATA ENTRY AND CONTROLS

```
Type Contact
    Ct As Single
    CtSigma As Single
    DMht As Single
    DMhtSigma As Single
    Rh As Single
    RhSigma As Single
    By As Single
    BySigma As Single
End Type
Type Parameter
    NumberOfContacts As Single
    AcceptableMinRange As Single
    SafetyRange As Single
    TimeOnCourse As Single
    DMho As Single
    DirectionOfSeas As Single
    NumberOfTrials As Single
    StepSize As Single
    Bins(0 To 20) As Single
End Type
Type CCR
    Xo As Single
    Yo As Single
End Type

Public Contacts(1 To 5) As Contact
Public Parameters As Parameter
Public CCRider As CCR
```

```
Sub InitializeContacts()
 Dim Count As Single
 For Count = 1 To 5
   With Contacts(Count)
    .Ct = 0
    .DMht = 0
    .Rh = 0
    .By = 0
    .CtSigma = 0
    .DMhtSigma = 0
    .RhSigma = 0
    .BySigma = 0
   End With
 Next Count
End Sub

Sub PrepareDialog()
 Dim Count As Single
 Dim Dialog1 As DialogSheet
 Dim Dialog2 As DialogSheet
 Set Dialog1 = DialogSheets("GetInitialData")
 Set Dialog2 = DialogSheets("GetContactData")

 With Dialog1
  For Count = 1 To 8
   .EditBoxes(Count).Text = "Enter Value"
  Next Count
 End With
 Dialog1.Show
 InitializeContacts
 With Dialog2
  With .DropDowns(1)
   .RemoveAllItems
```

```
    For Count = 1 To Parameters.NumberOfContacts
      .AddItem Text:=Count, Index:=Count
    Next Count
   End With
  End With
  Dialog2.Show
End Sub

Sub SendInputToGraphs()
  Dim Count As Single
  Dim ProgSheet As Worksheet
  Set ProgSheet = Worksheets("Graphs")
  With ProgSheet
   For Count = 1 To 5
    .Cells(1, Count + 5) = "Contact " & Count
    .Cells(2, Count + 5) = Contacts(Count).Ct
    .Cells(3, Count + 5) = Contacts(Count).DMht
    .Cells(4, Count + 5) = Contacts(Count).Rh
    .Cells(5, Count + 5) = Contacts(Count).By
    .Cells(6, Count + 5) = Contacts(Count).CtSigma
    .Cells(7, Count + 5) = Contacts(Count).DMhtSigma
    .Cells(8, Count + 5) = Contacts(Count).RhSigma
    .Cells(9, Count + 5) = Contacts(Count).BySigma
   Next Count
  .Cells(2, 3) = Parameters.NumberOfContacts
  .Cells(3, 3) = Parameters.AcceptableMinRange
  .Cells(4, 3) = Parameters.SafetyRange
  .Cells(5, 3) = Parameters.TimeOnCourse
  .Cells(6, 3) = Parameters.DMho
  .Cells(7, 3) = Parameters.DirectionOfSeas
  .Cells(8, 3) = Parameters.NumberOfTrials
  .Cells(9, 3) = Parameters.StepSize
  End With
End Sub
```

```
Sub ContactList_Change()
 Dim Count As Single
 Dim CurrentDialog As DialogSheet
 Set CurrentDialog = Application.ActiveDialog
 With CurrentDialog
  For Count = 1 To 8
   .EditBoxes(Count).Text = "Enter Value"
  Next Count
 End With
End Sub

Sub NumberOfContacts_Change()
 Dim CurrentDialog As DialogSheet
 Set CurrentDialog = Application.ActiveDialog
 If IsNumeric(CurrentDialog.EditBoxes(1).Text) Then
  Parameters.NumberOfContacts = CurrentDialog.EditBoxes(1).Text
 End If
End Sub

Sub AcceptableMinRange_Change()
 Dim CurrentDialog As DialogSheet
 Set CurrentDialog = Application.ActiveDialog
 If IsNumeric(CurrentDialog.EditBoxes(2).Text) Then
  Parameters.AcceptableMinRange = CurrentDialog.EditBoxes(2).Text
 End If
End Sub

Sub SafetyRange_Change()
 Dim CurrentDialog As DialogSheet
 Set CurrentDialog = Application.ActiveDialog
 If IsNumeric(CurrentDialog.EditBoxes(3).Text) Then
  Parameters.SafetyRange = CurrentDialog.EditBoxes(3).Text
 End If
```

```
End Sub

Sub TimeOnCourse_Change()
  Dim CurrentDialog As DialogSheet
  Set CurrentDialog = Application.ActiveDialog
  If IsNumeric(CurrentDialog.EditBoxes(4).Text) Then
    Parameters.TimeOnCourse = CurrentDialog.EditBoxes(4).Text
  End If
End Sub

Sub DMho_Change()
  Dim CurrentDialog As DialogSheet
  Set CurrentDialog = Application.ActiveDialog
  If IsNumeric(CurrentDialog.EditBoxes(5).Text) Then
    Parameters.DMho = CurrentDialog.EditBoxes(5).Text
  End If
End Sub

Sub DirectionOfSeas_Change()
  Dim CurrentDialog As DialogSheet
  Set CurrentDialog = Application.ActiveDialog
  If IsNumeric(CurrentDialog.EditBoxes(6).Text) Then
    Parameters.DirectionOfSeas = CurrentDialog.EditBoxes(6).Text
  End If
End Sub

Sub NumberOfTrials_Change()
  Dim CurrentDialog As DialogSheet
  Set CurrentDialog = Application.ActiveDialog
  If IsNumeric(CurrentDialog.EditBoxes(7).Text) Then
    Parameters.NumberOfTrials = CurrentDialog.EditBoxes(7).Text
  End If
End Sub
```

```
Sub StepSize_Change()
 Dim CurrentDialog As DialogSheet
 Set CurrentDialog = Application.ActiveDialog
 If IsNumeric(CurrentDialog.EditBoxes(8).Text) Then
  Parameters.StepSize = CurrentDialog.EditBoxes(8).Text
 End If
End Sub


Sub CourseBox_Change()
 Dim CurrentDialog As DialogSheet
 Set CurrentDialog = Application.ActiveDialog
 If IsNumeric(CurrentDialog.EditBoxes(1).Text) Then
  Contacts(CurrentDialog.DropDowns.Value).Ct = CurrentDialog.EditBoxes(1).Text
 End If
End Sub

Sub SpeedBox_Change()
 Dim CurrentDialog As DialogSheet
 Set CurrentDialog = Application.ActiveDialog
 If IsNumeric(CurrentDialog.EditBoxes(2).Text) Then
  Contacts(CurrentDialog.DropDowns.Value).DMht = CurrentDialog.EditBoxes(2).Text
 End If
End Sub

Sub RangeBox_Change()
 Dim CurrentDialog As DialogSheet
 Set CurrentDialog = Application.ActiveDialog
 If IsNumeric(CurrentDialog.EditBoxes(3).Text) Then
  Contacts(CurrentDialog.DropDowns.Value).Rh = CurrentDialog.EditBoxes(3).Text
 End If
End Sub

Sub BearingBox_Change()
```

```
 Dim CurrentDialog As DialogSheet
 Set CurrentDialog = Application.ActiveDialog
 If IsNumeric(CurrentDialog.EditBoxes(4).Text) Then
  Contacts(CurrentDialog.DropDowns.Value).By = CurrentDialog.EditBoxes(4).Text
 End If
End Sub

Sub CourseSigmaBox_Change()
 Dim CurrentDialog As DialogSheet
 Set CurrentDialog = Application.ActiveDialog
 If IsNumeric(CurrentDialog.EditBoxes(5).Text) Then
  Contacts(CurrentDialog.DropDowns.Value).CtSigma = CurrentDialog.EditBoxes(5).Text
 End If
End Sub

Sub SpeedSigmaBox_Change()
 Dim CurrentDialog As DialogSheet
 Set CurrentDialog = Application.ActiveDialog
 If IsNumeric(CurrentDialog.EditBoxes(6).Text) Then
  Contacts(CurrentDialog.DropDowns.Value).DMhtSigma = CurrentDialog.EditBoxes(6).Text
 End If
End Sub

Sub RangeSigmaBox_Change()
 Dim CurrentDialog As DialogSheet
 Set CurrentDialog = Application.ActiveDialog
 If IsNumeric(CurrentDialog.EditBoxes(7).Text) Then
  Contacts(CurrentDialog.DropDowns.Value).RhSigma = CurrentDialog.EditBoxes(7).Text
 End If
End Sub

Sub BearingSigmaBox_Change()
 Dim CurrentDialog As DialogSheet
 Set CurrentDialog = Application.ActiveDialog
```

```
  If IsNumeric(CurrentDialog.EditBoxes(5).Text) Then
    Contacts(CurrentDialog.DropDowns.Value).BySigma = CurrentDialog.EditBoxes(8).Text
  End If
End Sub
```

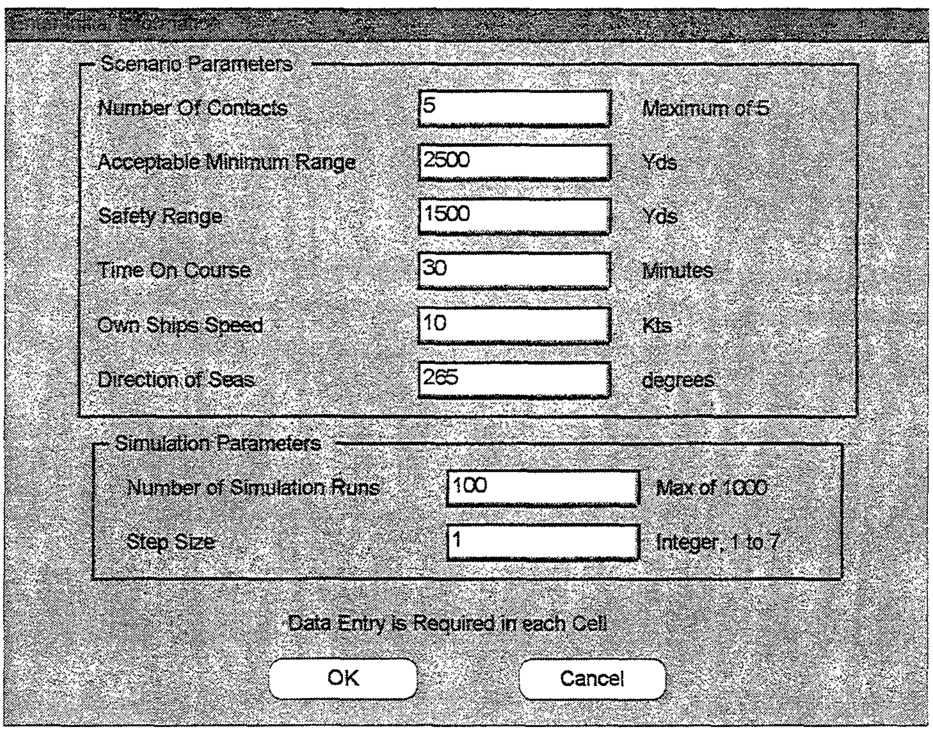

Scenario Parameters
Number Of Contacts
5
Maximum of 5
Acceptable Minimum Range
2500
Yds
Safety Range
1500
Yds
Time On Course
30
Minutes
Own Ships Speed
10
Kts
Direction of Seas
265
degrees
Simulation Parameters
Number of Simulation Runs
100
Max of 1000
Step Size
1
Integer, 1 to 7
Data Entry is Required in each Cell
OK
Cancel

Figure 1.

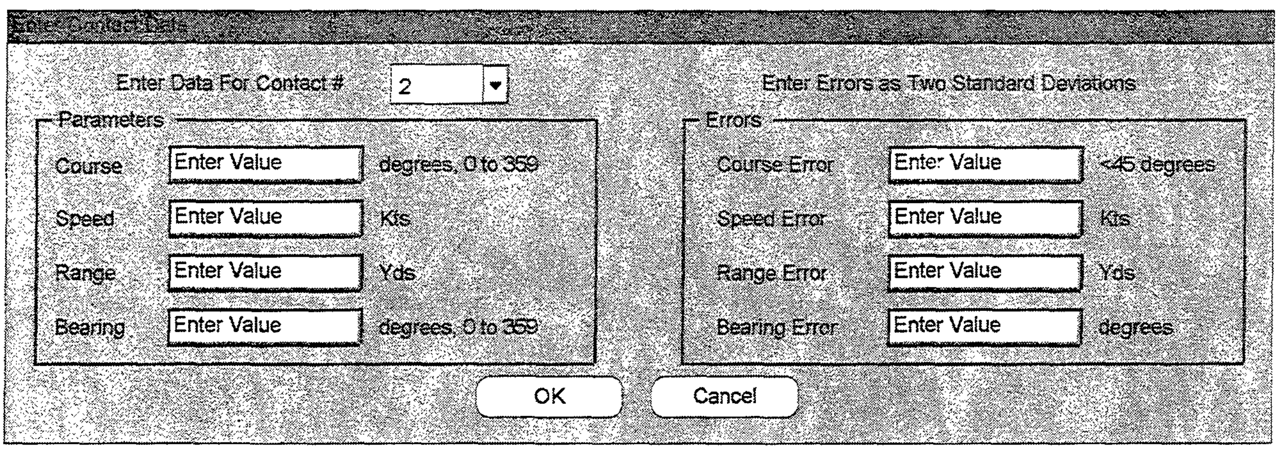

Enter Contact Data
Enter Data For Contact #
2
Enter Errors as Two Standard Deviations
Parameters
Course
Enter Value
degrees, 0 to 359
Speed
Enter Value
Kts
Range
Enter Value
Yds
Bearing
Enter Value
degrees, 0 to 359
Errors
Course Error
Enter Value
<45 degrees
Speed Error
Enter Value
Kts
Range Error
Enter Value
Yds
Bearing Error
Enter Value
degrees
OK
Cancel

Figure 2.

APPENDIX D. VISUAL BASIC SOURCE CODE FOR SIMULATION

```
Option Explicit

Function GetX(Rh As Single, By As Single)
  Application.Volatile
  GetX = Rh * Cos(By)
End Function

Function GetY(Rh As Single, By As Single)
  Application.Volatile
  GetY = Rh * Sin(By)
End Function

Sub GetBins()
  Dim Count As Single
  Dim Bin(0 To 20) As Single
  Parameters.Bins(0) = Application.NormSInv(0.01)
  For Count = 1 To 19
    Parameters.Bins(Count) = Application.NormSInv(0.05 * Count)
  Next Count
  Parameters.Bins(20) = Application.NormSInv(0.99)
End Sub

Sub ConvertCoordinates(Co As Single, Parameters As Parameter)
  Application.Volatile
  CCRider.Xo = GetX(Parameters.DMho, (Co * Application.Pi() / 180))
  CCRider.Yo = GetY(Parameters.DMho, (Co * Application.Pi() / 180))
End Sub

Sub ConvertToRadians()
  Dim Count As Single
  Application.Volatile
  For Count = 1 To Parameters.NumberOfContacts
```

```
    Contacts(Count).By = Contacts(Count).By * Application.Pi() / 180
    Contacts(Count).BySigma = Contacts(Count).BySigma * Application.Pi() / 180
    Contacts(Count).Ct = Contacts(Count).Ct * Application.Pi() / 180
    Contacts(Count).CtSigma = Contacts(Count).CtSigma * Application.Pi() / 180
  Next Count
End Sub

Function GetError(Sigma As Single) As Single
  Dim P1 As Single
  Dim Mean As Single
  Application.Volatile
  Mean = 0  'for all error distributions
  Randomize
  P1 = Rnd
  If Sigma <> 0 Then
    GetError = Parameters.Bins(Int(21 * P1)) * Sigma / 2
  Else
    GetError = 0
  End If
End Function

Function GetTimeToCPA(Xo As Single, Yo As Single, Xc As Single, Yc As Single, Xp As Single, Yp As Single) As Single
  Dim X As Single
  Dim Y As Single
  Dim xDMhr As Single
  Dim yDMhr As Single
  Application.Volatile
  xDMhr = (Xc - Xo) * 33.75
  yDMhr = (Yc - Yo) * 33.75
  If Application.And(xDMhr, yDMhr) <> 0 Then
    GetTimeToCPA = -((xDMhr * Xp + yDMhr * Yp) / (xDMhr * xDMhr + yDMhr * yDMhr))
  Else
    GetTimeToCPA = 0
```

```
    End If
  End Function

  Function GetMinSquaredRange(Xo As Single, Yo As Single, Xc As Single, Yc As Single, Xp As Single, Yp As Single, Time As Single) As Single
    Application.Volatile
    If Time <= 0 Then
      GetMinSquaredRange = GetNextSquaredRange(Xo, Yo, Xc, Yc, Xp, Yp, 0)
    ElseIf Time > Parameters.TimeOnCourse Then
      GetMinSquaredRange = GetNextSquaredRange(Xo, Yo, Xc, Yc, Xp, Yp, Parameters.TimeOnCourse)
    Else
      GetMinSquaredRange = GetNextSquaredRange(Xo, Yo, Xc, Yc, Xp, Yp, Time)
    End If
  End Function

  Function GetNextSquaredRange(Xo As Single, Yo As Single, Xc As Single, Yc As Single, Xp As Single, Yp As Single, Time As Single) As Single
    Dim X As Single
    Dim Y As Single
    Application.Volatile
    X = Xp + ((Xc - Xo) * 33.75) * Time
    Y = Yp + ((Yc - Yo) * 33.75) * Time
    GetNextSquaredRange = X * X + Y * Y
  End Function

  Function Simulate(Parameters As Parameter, Contacts() As Contact) As Variant
    Dim Course As Single
    Dim Trial As Single
    Dim Count As Single
    Dim TrialRh As Single
    Dim TrialBy As Single
    Dim TrialDMht As Single
    Dim TrialCt As Single
```

```
Dim TrialXc As Single
Dim TrialYc As Single
Dim TrialXp As Single
Dim TrialYp As Single
Dim Time As Single
Dim n As Single
Dim k As Single
Dim AbsoluteMinSquaredRange As Single
Dim MOE(1 To 2, 0 To 359) As Single
Dim MinSquaredRange() As Single
ReDim MinSquaredRange(1 To Parameters.NumberOfContacts) As Single
Dim DBy As Single
Application.Volatile
ConvertToRadians
GetBins
For Course = 0 To 359 Step Parameters.StepSize
 n = 0
 k = 0
 ConvertCoordinates Course, Parameters
 For Trial = 1 To Parameters.NumberOfTrials
  For Count = 1 To Parameters.NumberOfContacts
   TrialRh = Contacts(Count).Rh + GetError(Contacts(Count).RhSigma)
   TrialBy = Contacts(Count).By + GetError(Contacts(Count).BySigma)
   TrialDMht = Contacts(Count).DMht + GetError(Contacts(Count).DMhtSigma)
   TrialCt = Contacts(Count).Ct + GetError(Contacts(Count).CtSigma)
   TrialXc = GetX(TrialDMht, TrialCt)
   TrialYc = GetY(TrialDMht, TrialCt)
   TrialXp = GetX(TrialRh, TrialBy)
   TrialYp = GetY(TrialRh, TrialBy)
   Time = GetTimeToCPA(CCRider.Xo, CCRider.Yo, TrialXc, TrialYc, TrialXp, TrialYp)
     MinSquaredRange(Count) = GetMinSquaredRange(CCRider.Xo, CCRider.Yo, TrialXc,
TrialYc, TrialXp, TrialYp, Time)
  Next Count
  AbsoluteMinSquaredRange = Application.Min(MinSquaredRange())
```

```
                If AbsoluteMinSquaredRange >= (Parameters.AcceptableMinRange * Parameters.AcceptableMinRange) Then
     n = n + 1
    End If
    If AbsoluteMinSquaredRange >= (Parameters.SafetyRange * Parameters.SafetyRange) Then
     k = k + 1
    End If
   Next Trial
   MOE(1, Course) = n / Parameters.NumberOfTrials
   MOE(2, Course) = k / Parameters.NumberOfTrials
  Next Course
  Simulate = MOE()
End Function

Sub SendDataToGraphs(MOEDataSet As Variant)
 Dim Count As Single
 Dim GraphSheet As Worksheet
 Set GraphSheet = Worksheets("Graphs")
 Application.Volatile
  For Count = 1 To 360
   With GraphSheet
    .Cells(Count + 1, 17) = MOEDataSet(1, (Count - 1))
    .Cells(Count + 1, 18) = MOEDataSet(2, (Count - 1))
   End With
  Next Count
End Sub
```

APPENDIX E. VISUAL BASIC SOURCE CODE FOR MAIN MODULE

```
Option Explicit
Dim Data As Variant
Sub Main()
   PrepareDialog
   SendInputToGraphs
   Data = Simulate(Parameters, Contacts)
   SendDataToGraphs (Data)
End Sub
```

LIST OF REFERENCES

1. Maneuvering Board Manual, United States Defense Mapping Agency, Hydrographic/Topographic Center, 4th ed.

2. Naval Warfare Publication 3-21.51.1 Target Motion Analysis Techniques, (Urgent Change 5), Unclassified

INITIAL DISTRIBUTION LIST

Copies

1. Defense Technical Information Center..2
   8725 John J. Kingman Road, Suite 0944
   Fort Belvoir, VA 22060-6218
2. Library, Code 52..2
   Naval Postgraduate School
   Monterey, CA 93943-5002
3. COMSUBLANT...1
   7958 Blandy Road
   Norfolk, VA 23551-2492
4. COMSUBPAC...1
   BLDG 619
   100 Morton Street
   Pearl Harbor, HI 96860-6543
5. CAPT S. G. Slaton..1
   SUBGRU 9
   Naval Submarine Base Bangor
   2150 Thresher Avenue
   Silverdale, WA 98315-2150
6. Prof Alan R. Washburn, Code OR/Ws.......................................1
   Naval Postgraduate School
   Monterey, CA 93943-5002
7. LT D. J. Danko..5
   512 Pruner Street
   Osceola Mills, PA 16666