1985

# Simultaneous Conjugate Gradient Search Applied to Target Motion Analysis

John M. Arrigan
*University of Rhode Island*

# SIMULTANEOUS CONJUGATE GRADIENT SEARCH APPLIED TO TARGET MOTION ANALYSIS

BY

JOHN M. ARRIGAN

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF THE

REQUIREMENT FOR THE DEGREE OF

MASTER OF SCIENCE

IN

INDUSTRIAL ENGINEERING

UNIVERSITY OF RHODE ISLAND

1985

MASTER OF SCIENCE THESIS

OF

JOHN M. ARRIGAN

APPROVED:

Thesis Committee

Major Professor ______________________

______________________

______________________

______________________

DEAN OF THE GRADUATE SCHOOL

UNIVERSITY OF RHODE ISLAND

1985

ABSTRACT

This thesis discusses a mathematical approach to solving the bearings only target motion analysis problem. The problem of solving a sonar contact's tract can be reduced to a two variable minimization problem of finding the values of initial and final ranges to contact that minimize the sum of squares of error between the actual sonar bearings and the computed bearings given the estimated initial and final ranges.

Because a submarine can use specific tactics when tracking a contact and because maximum detection ranges can be estimated, the general shape of the sum of squares of error function and its orientation are known. This research develops a search technique in which two conjugate gradient searches converge simultaneously from opposite sides of the optimum. Rules are developed to determine starting points which guarantee that the searches remain on opposite sides of the optimum in both variables. The stopping criterion for the search is the distance between the searches after each iteration. A measure of the progress of the search in terms of maximum distance from the optimum is guaranteed because either search is no further from the optimum at any iteration than the distance between the two searches. The

research shows that a solution to the bearings only target motion analysis problem is obtained with the required accuracy more efficiently by searching simultaneously from opposite sides of the optimum than by searching from one point only.

The advent of multi-processors and co-processors in small computers argues for exploring the concept of searching simultaneously from multiple starting points. Multiple search in itself is not a solution. Criteria are still needed for stopping the searches and choosing a solution from the results of the two searches. This study tests a number of criteria for stopping the search and also evaluation criteria for choosing a solution.

## ACKNOWLEDGEMENTS

The author wishes to express his appreciation to Dr. David M. Shao, Professor of Industrial Engineering, for his assistance in the development of this thesis, and to his wife, Marylin, who produced the art work in this thesis.

TABLE OF CONTENTS

TABLE OF CONTENTS (CON'D)

TABLE OF CONTENTS (CON'D)

TABLE OF CONTENTS (CON'D)

LIST OF TABLES

## LIST OF FIGURES

LIST OF FIGURES (CON'D)

# SECTION 1

# INTRODUCTION

## 1.1 STATEMENT OF THE PROBLEM.

A submarine on patrol is continuously searching for other submarines if for no other reason than to avoid a collision with another submarine, be it friendly or unfriendly. Once a submarine detects another submarine, it must estimate the position, course and speed of the contact. This can be accomplished using active sonar which provides both range and bearing to target. Submarines, however, usually prefer to refrain from using active sonar in order to remain as quiet as possible and thus minimize the probability of being counterdetected. Consequently, submarines usually patrol using passive sonar only. Passive sonar, which listens for sounds from other submarines, provides a bearing to the sound source but does not provide range information. Although passive sonar provides bearings only information (as compared to active sonar which provides bearing and range to target), it is still possible to estimate the sonar contact's track using bearings only information (1). This procedure is called passive target motion analysis.

Developing and improving the efficiency and accuracy of procedures that solve the passive bearings only target motion analysis problem has been a priority of the U.S. Navy for over forty years (2,3). Considerable time and funds have been expended on this effort. A number of techniques have been developed from manual techniques that use hand drawn plots to mathematical techniques that use analog and digital computers. Mathematical techniques using the Kalman filter (4), maximum likelihood estimator (5), and regression techniques (6) have been developed. Some algorithms require hand held calculators (7,8) while others require desk top computers (9). This research, however, is limited to examining the conjugate gradient search as a technique for solving the passive bearings only target motion analysis problem.

The following example describes the problem. A tracker, holding contact on a target submarine, travels east for five time steps and then turns left 135 degrees and travels two additional time steps to the northwest (see figure 1-1). At each time step, the tracker holds a sonar bearing to target ($\hat{B}_i$ solid arrows). $R_{INIT}$ and $R_{FINAL}$, which are the actual initial and final ranges to the target, are not known to the tracker. Since the tracker has initial and final bearings to the target, it can estimate the target's course and speed if it can determine the initial and final ranges to the target. The conjugate gradient search is used to compute a best estimate of $R_{INIT}$ and $R_{FINAL}$.

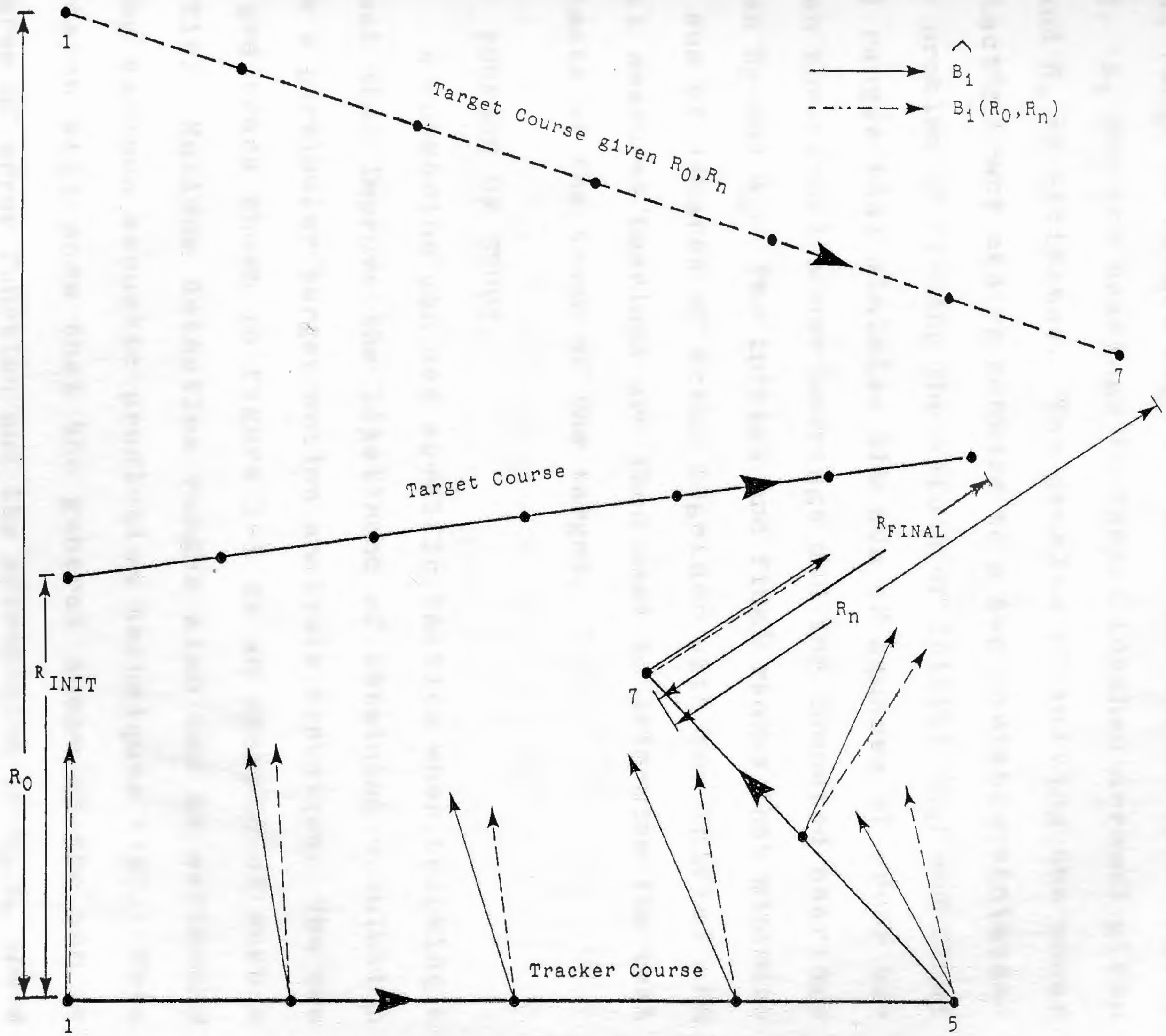



Figure 1-1. Target Motion Analysis Geometry.

Some estimate of the initial range to target ($R_0$) and final range to target ($R_n$) are made and tested (see figure 1-1). $B_i$ are the bearings to target (dashed arrows) given $R_0$ and $R_n$ as estimates. The problem of solving the sonar contact's track can be reduced to a two variable minimization problem of finding the values of initial ($R_0$) and final ($R_n$) ranges that minimize the sum of squares of error between the actual sonar bearings and the computed bearings given $R_0$ and $R_n$. The initial and final ranges that minimize the sum of squares of error together with the initial and final measured bearings are then used to determine the best estimate of the track of the target.

1.2 PURPOSE OF STUDY.

A submarine can use specific tactics when tracking a threat that improve the likelihood of obtaining a solution from a particular target motion analysis technique. The two legged track shown in figure 1-1 is an example of such a tactic. Maximum detection ranges also can be estimated using various acoustic prediction techniques (10). This research will show that the general shape of the sum of squares of error function and its orientation in $R_0$,$R_n$ space are predictable for a specific tracking tactic. The purpose of this research is to develop and test a search technique in which two conjugate gradient searches converge simultaneously from opposite sides of the minimum. This is possible if the function is well behaved and the shape and orientation of the function can be predicted.

A technique is developed to determine starting points which provide a high probability that the searches converge from opposite sides of the minimum in both variables. The stopping criterion is the distance between the searches after each iteration. A measure of the progress of the search in units of maximum distance from the minimum is available because either search is no further from the minimum at any iteration than the distance between the two searches, provided the two searches remain on opposite sides of the minimum. The number of iterations required to obtain an acceptable solution using the simultaneous conjugate gradient search is compared to the number of iterations required to obtain an acceptable solution using a single search and traditional stopping criteria.

The research first develops a form of the conjugate gradient search which consistently converges for all inputs tested. It then tests the hypothesis that a solution to the bearings only target motion analysis problem is obtained with the required accuracy more efficiently by searching simultaneously from opposite sides of the minimum than by searching from one point only.

## 1.3 SCOPE OF STUDY.

This study is primarily concerned with comparing the effectiveness of two simultaneous conjugate gradient searches converging from opposite sides of the minimum to the effectiveness of the single search for solving the bearings only target motion analysis problem. There are many areas of investigation for improving bearings only target motion analysis. Tactical aspects such as tracking geometry, tracking speed and sonar settings (11) are beyond the scope of this analysis. No attempt is made to optimize the conjugate gradient search nor to compare the conjugate gradient technique to other techniques such as the Kalman filter, least squares or maximum likelihood estimators. Because the same conjugate gradient search is used for the single and simultaneous conjugate gradient searches, optimizing the search does not affect the relative performance of the two techniques. Considerable effort is necessary, however, to develop a conjugate gradient search algorithm that consistently converges on the minimum for the number of tracking geometries and the range of starting positions used for the analysis.

## 1.4 COMPUTER CONSIDERATIONS.

The advent of multi-processors and co-processors in small computers (12,13) argues for exploring the concept of searching simultaneously from multiple starting points. Multiple search in itself is not a solution. Criteria are still needed for stopping the searches and choosing a solution from the results of the two searches. The primary advantage of multiple search is that the converging searches provide maximum error measurements in terms of nautical miles if the starting points are properly chosen. Error measurements for the single conjugate gradient search are in terms of sum of squares of error or probability of an acceptable solution after n iterations (14).

All computer runs are made on a Zenith Z-100 desktop computer with the 8087 mathematics chip. The Z-100 does not have multi-processing or co-processing capability. A simultaneous search algorithm is developed which simulates a computer than can calculate two searches simultaneously. All programs are written in Z-BASIC which is Zenith's version of Microsoft's BASIC. Optimization programs are compiled. Some statistical programs are run in the interpretive mode. The Z-100 is chosen primarily because of its ability to perform the necessary calculations in a reasonable amount of time and for its ready availability.

## 1.5 REPORT STRUCTURE.

Section 2 discusses the conjugate gradient search technique and constraints on the solution set that are imposed by the physics of underwater acoustics and the tactics of the searcher. Section 2 also describes the shape of the sum of squares of error function and the technique developed for using the simultaneous conjugate gradient search. Section 3 evaluates the single conjugate gradient search and Section 4 evaluates the simultaneous conjugate gradient search. Conclusions and recommendations are presented in Section 5.

## SECTION 2

## DISCUSSION OF THE PROBLEM

Using the conjugate gradient search to solve the passive bearings only target motion analysis problem is an appropriate application of classical search theory (15,16). A number of constraints are imposed on the theoretical problem of finding the minimum of a contour by operational considerations and the physics of underwater acoustics. These factors constrain both the shape of the function and also the area in $R_0,R_n$ space in which the function is defined. Section 2 discusses some of the constraints imposed by the physical conditions and the implication of these constraints to the solution of the problem. It also discusses the form of the conjugate gradient search used for this analysis and some of the parameters selected for the algorithm.

### 2.1 OPERATIONAL CONSIDERATIONS

The primary rationale for the submarine service is the unique ability of the submarine to operate in a clandestine mode. While various methods of detecting submarines are available, particularly sonar, it is difficult to detect and track a submarine in the open ocean. Because the submarine can compromise its tactical advantage of clandestine operation if it is detected, it is imperative for a submarine to reduce as much as practical all signals eminating from the boat. One of the strongest signals produced by any warship is the pinging of its active sonar. Consequently, it is advantageous for a submarine to avoid using active sonar if it desires to remain clandestine.

A submarine, once it detects another submarine, must track the contact in order to determine its course and speed. This is necessary whether the purpose of the patrol is to avoid contact with other submarines, to follow and collect information on a contact, or to attack a contact. In order to track a contact clandestinely, the submarine relies on passive sonar, that is, listening to noise from the contact. Passive sonar provides the tracking submarine with bearings only information. It knows the bearing to the contact every minute for 10 minutes, for instance, but it does not know the range to the contact.

Clandestine tracking is desired for a number of reasons. If the contact suspects it is being tracked, it may attempt to avoid or evade the tracker. If a submarine is to follow a contact to gain information on enemy tactics, it must be able to track for a long period of time without being counter-detected. In wartime, a tracking submarine could be counter-detected and counter-attacked by the submarine it is tracking.

Various methods have been developed to estimate target range from bearings only information. These methods work best if the bearings change significantly during the tracking sequence. Certain tracking tactics tend to work better than others because they tend to produce bearing changes larger than other tactics. Primarily, some course maneuver is required by the tracker such as using a ziz-zag search tactic or taking an L or V shaped course.

Tracking a submarine is a difficult task. Because the range at which a sonar can detect a submarine is limited (and different for varying environmental and tactical conditions), a tracker must maintain the target within detection range while at the same time avoiding high speed (which tends to reduce sonar performance). It must also keep the contact within detection range while making the necessary tactical maneuvers to provide favorable bearings.

## 2.2 UNDERWATER ACOUSTICS

### 2.2.1 Active and Passive Search

A number of fundamental principles of underwater acoustics are discussed because they are important to an understanding of the target motion analysis problem. The first topic is the difference between active and passive acoustic search. Some of the most effective methods of detecting a submerged submarine are acoustic methods (1). These can be divided into active and passive search. In active search, the searcher projects sound into the water. This sound travels to the target, is deflected off the target and is echoed back to the searcher (provided the target is within detection range). The sonar system on the searcher computes the time delay before the echo is received and also the bearing on which the echo is received. Because the speed of sound in water is known, it is possible to determine both range and bearing to target using active search. Target motion analysis using active search is relatively easy. Knowing own ship course and speed as well as range and bearing to target at two or more time steps, it is possible to estimate the course and speed of the target. For a submarine to detect a contact at 10 nautical miles on active sonar, sound must travel 20 nautical miles, 10 in each direction. The liability of active sonar is that the target can hear the active pinging at twice the distance that the searcher can detect the target.

Passive search, on the other hand, involves listening for noise from the target submarine. When a target is detected on a passive sonar system, a signal is received on a particular bearing. The searcher knows that a target is within maximum detection range on a particular bearing but has no other information as to the range to target. If the searcher can track the target over a period of time, it can collect a series of bearing to target readings. Passive target motion analysis involves determining the most likely target track given the time dependent set of sonar bearings.

### 2.2.2 Maximum Detection Range

2.2.2.1 Propagation Loss. An important element of target motion analysis is estimating maximum detection range. As sound travels through the ocean, it is absorbed and scattered. In general, the intensity of sound decreases with distance from the source (17). For this reason, there is a maximum distance at which a sonar can detect a target in a given environment.

Acoustic range prediction computer models are available which predict propagation loss as a function of distance from source. The primary input to these models is a profile of the temperature of the ocean as a function of depth. A device called an expendable bathythermograph is used to measure ocean temperature as a function of depth as it is dropped through the ocean. The computer prediction model uses the temperature profile and other information such as salinity, bottom type, wave height and frequency to predict the loss in intensity of a sound source versus distance from source. Figure 2-1 shows that the loss in intensity increases from 60 to 110 decibles as range increases from 1 to 50 nautical miles in the sample environment.

The propagation of sound in the ocean is a complex phenomenon. The same sound source which can be heard for hundreds of miles in one ocean environment may be undetectable at 2 or 3 nautical miles in another ocean. The ability

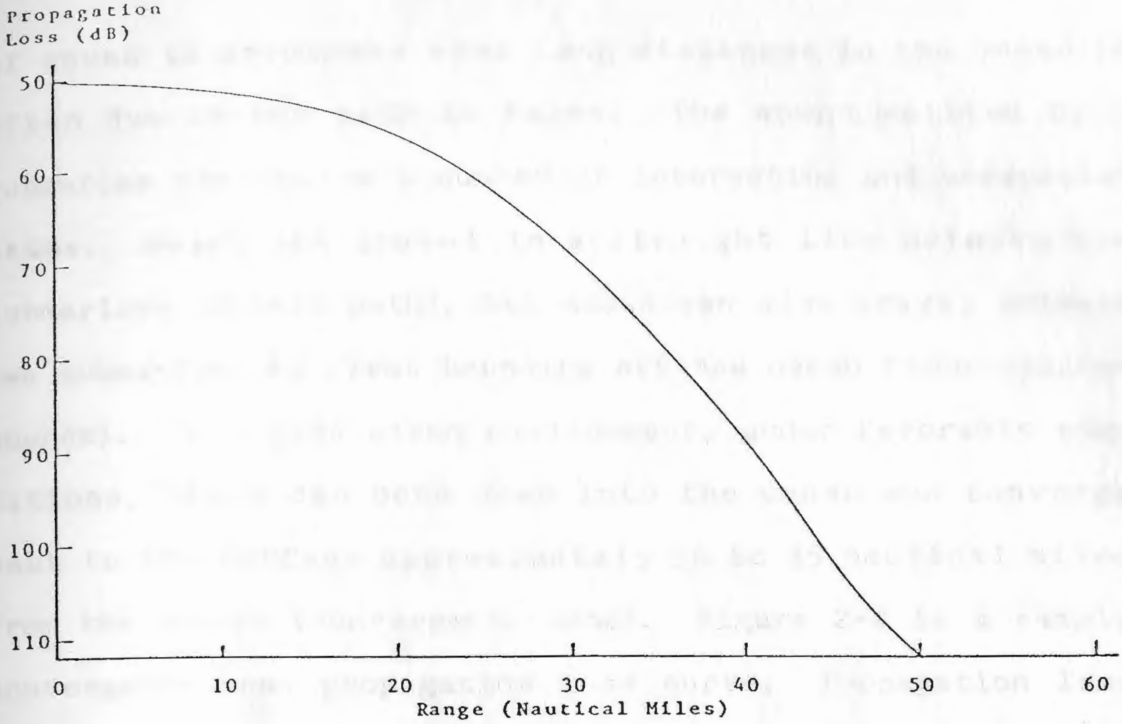



Figure 2-1. Sample Propagation Loss Curve--Direct Path.

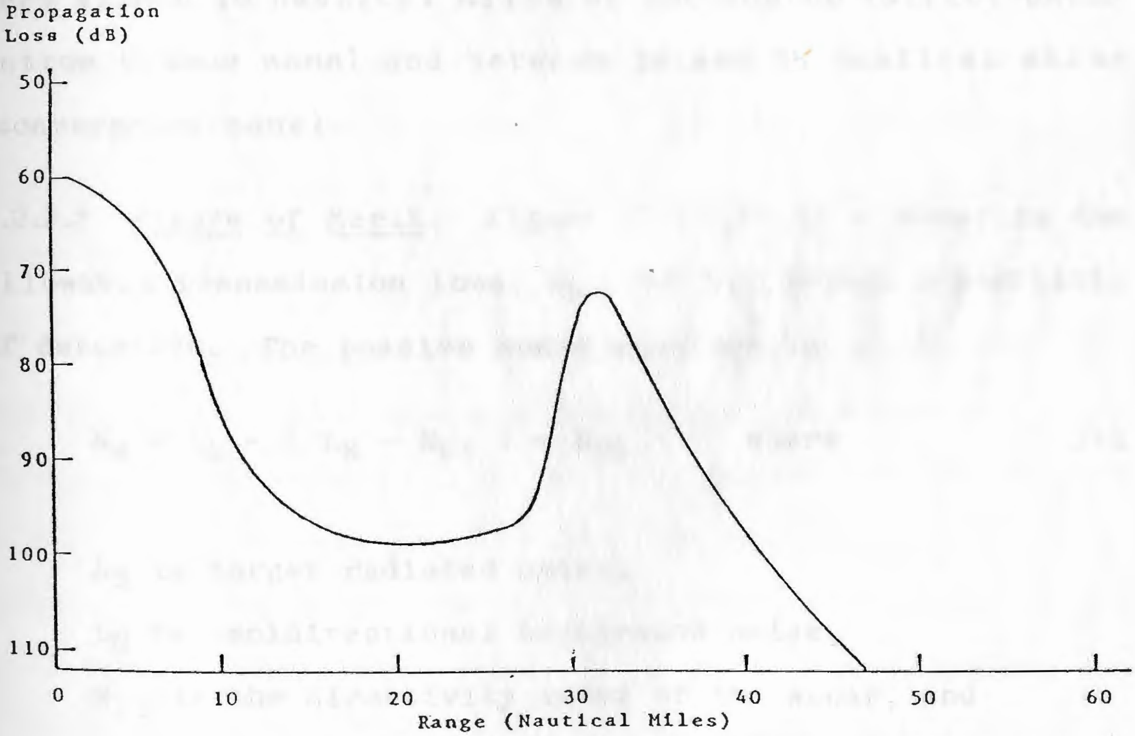



Figure 2-2. Sample Propagation Loss Curve--Convergence Zone.

of sound to propagate over long distances in the ocean is often due to the path it takes. The sound emitted by a submarine can follow a number of interesting and unexpected paths. Sound can travel in a straight line between two submarines (direct path), but sound can also travel between two submarines by first bouncing off the ocean floor (bottom bounce). In a deep ocean environment, under favorable conditions, sound can bend down into the ocean and converge back to the surface approximately 30 to 35 nautical miles from the source (convergence zone). Figure 2-2 is a sample convergence zone propagation loss curve. Propagation loss in this environment is greatest between 10 and 30 nautical miles and beyond 35 nautical miles. Sound propagation is best within 10 nautical miles of the source (direct path/ bottom bounce zone) and between 30 and 35 nautical miles (convergence zone).

2.2.2.2 <u>Figure of Merit</u>. Figure of merit of a sonar is the allowable transmission loss, $N_W$, for 50 percent probability of detection. The passive sonar equation is

$$N_W = L_S - ( L_N - N_{DI} ) - N_{RD} \quad \text{where} \qquad 2\text{-}1$$

$L_S$ is target radiated noise,

$L_N$ is omnidirectional background noise,

$N_{DI}$ is the directivity index of the sonar, and

$N_{RD}$ is the operator's recognition differential.

A line is drawn across the propagation loss curve at the figure of merit (see figure 2-3). The range at which the figure of merit intersects the curve is the range at which the searcher has a 50 percent probability of detection. The probability of detection at any range is determined from the standard normal distribution and the sigma for the sonar. If, for instance, sigma is 10 decibels, figure of merit is 80 decibels and propagation loss is 90 decibels (at 40 nautical miles on figure 2-3), then

$$z = \frac{80 - 90}{10} = -1 \quad \text{and} \qquad 2\text{-}2$$

the probability of detection at the range at which propagation loss is 90 decibels is the probability that z is less than -1 which is approximately 16 percent. Consequently in a direct path/bottom bounce environment, it is possible to determine a maximum detection range for any confidence level desired (10).

In a convergence zone environment, the problem is much more complex. Although maximum detection range may be 35 nautical miles (see figure 2-4), a contact is often much closer (0 to 10 nautical miles). The probability of detecting in a convergence zone is usually less than 1.0. In the environment depicted in figure 2-4, it is approximately 0.6 assuming figure of merit is 80 and sigma is 10 decibels. Consequently there is a 0.4 probability that initial detection occurs within 10 nautical miles. For the sake of the

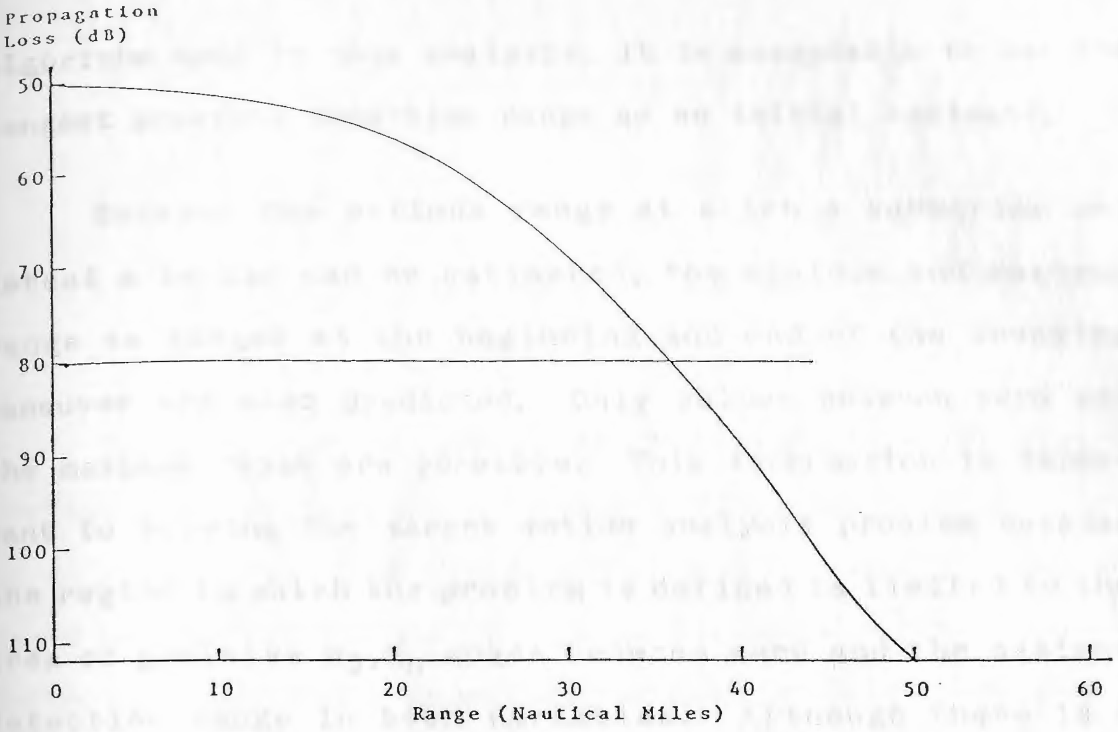



Figure 2-3. Figure of Merit--Direct Path Detection.

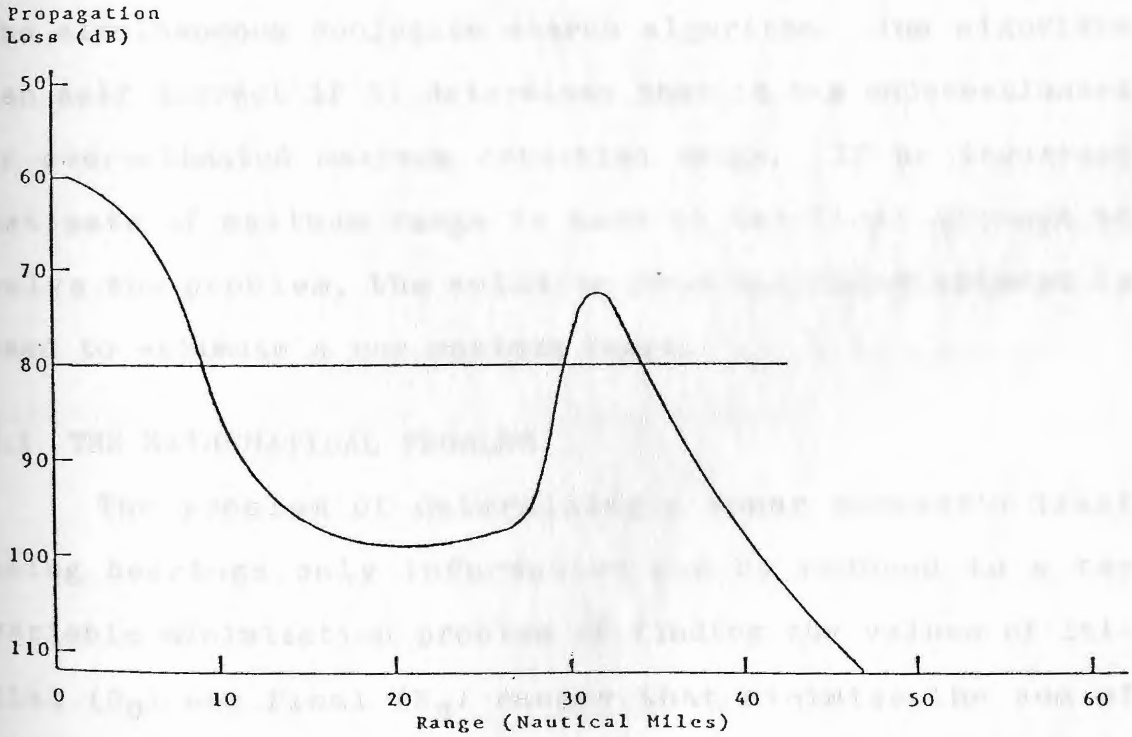



Figure 2-4. Figure of Merit--Convergence Zone Detection.

algorithm used in this analysis, it is acceptable to use the longest possible detection range as an initial estimate.

Because the maximum range at which a submarine can detect a target can be estimated, the minimum and maximum range to target at the beginning and end of the tracking maneuver are also predicted. Only values between zero and the maximum range are possible. This information is important to solving the target motion analysis problem because the region in which the problem is defined is limited to the area of positive $R_0,R_n$ space between zero and the maximum detection range in both variables. Although there is a probability of either underestimating or overestimating maximum detection range, this is not a serious problem for the simultaneous conjugate search algorithm. The algorithm can self correct if it determines that it has underestimated or overestimated maximum detection range. If an incorrect estimate of maximum range is made at the first attempt to solve the problem, the solution from the first attempt is used to estimate a new maximum range.

## 2.3 THE MATHEMATICAL PROBLEM

The problem of determining a sonar contact's tract using bearings only information can be reduced to a two variable minimization problem of finding the values of initial ($R_0$) and final ($R_n$) ranges that minimize the sum of squares of error function

$$g(R_0,R_n) = \sum_{i=1}^{n} \left( \hat{B}_i - B_i(R_0,R_n) \right)^2 \qquad 2\text{-}3$$

where $\hat{B}_i$ is the $i^{th}$ sonar bearing, and

$B_i(R_0,R_n)$ is the $i^{th}$ computed bearing given $R_0$ and $R_n$.

The function is a well behaved canoe or bowl shaped surface that has a single minimum in the area of $R_0,R_n$ space in which the function is defined (see figure 2-5). Consequently the function, which is an arctangent function, is amenable to solution by non-linear optimization techniques.

Several methods are available for solving optimization problems, including direct search methods and descent methods. The descent methods are generally more efficient compared to direct search methods because they use more information about the function, specifically the derivatives of the function. The optimization technique used in this analysis is a form of the conjugate gradient search. This technique uses the gradient and a conjugate gradient to determine the direction of optimum descent.

Although much effort has been expended in developing techniques to solve the passive bearings only target motion analysis problem, almost the entire effort is devoted to techniques other then the conjugate gradient search. There are, for instance, no references in the public literature of using the conjugate gradient search to solve the problem.

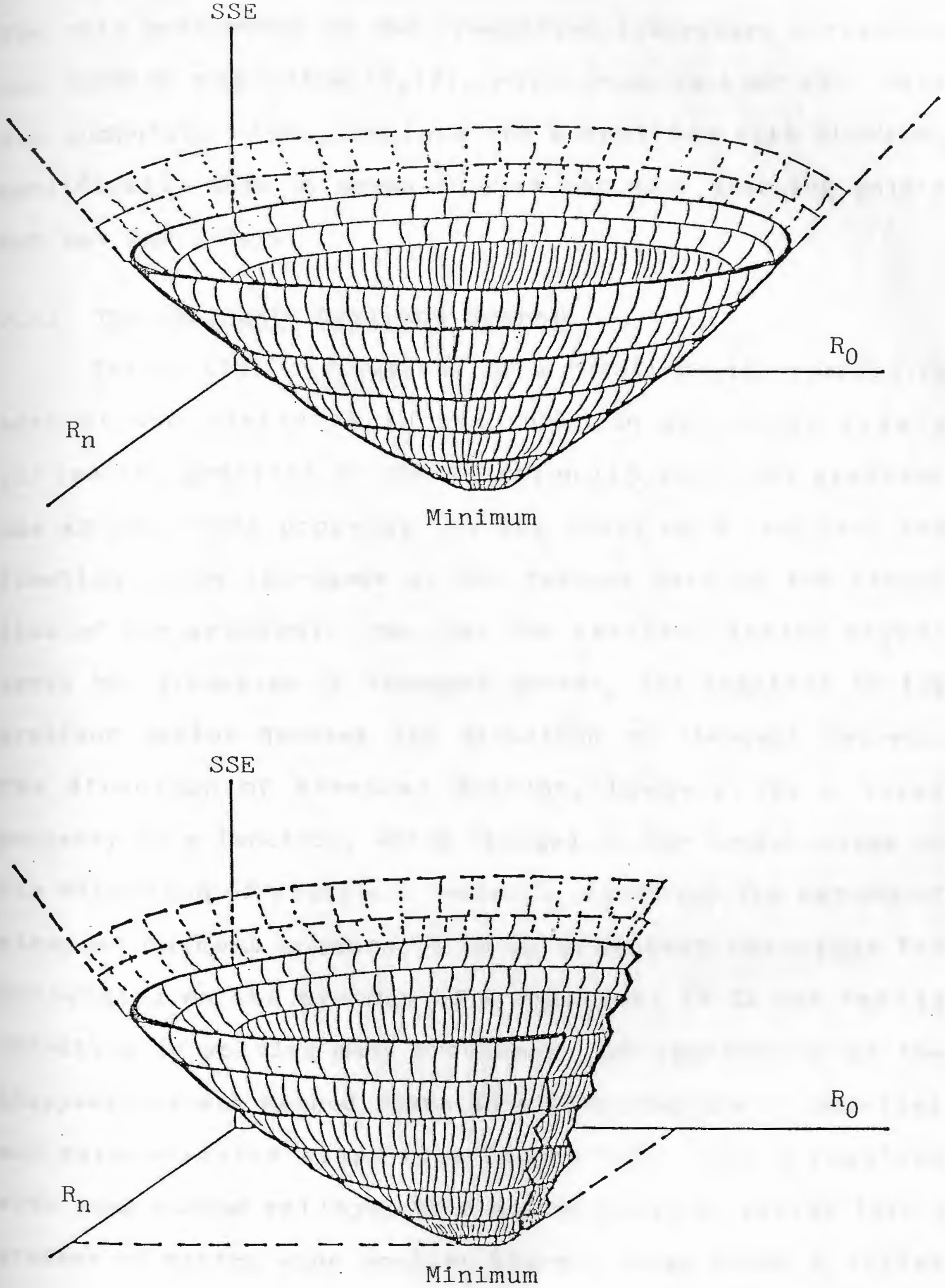



Figure 2-5. Sum of Squares of Error Function.

The only references in the classified literature pertain to the SURFLOC algorithm (9,18), which runs on a HP 9825 desk top computer. Some problems are identified with SURFLOC, specifically that it seems to work for some starting points but not for others.

### 2.3.1 The Conjugate Gradient Search

The partial derivatives of a function with respect to each of the variables in the function are collectively called the gradient of the function (15,16). The gradient has an important property: at any point on a contour, the function value increases at the fastest rate in the direction of the gradient. Because the gradient vector represents the direction of steepest ascent, the negative of the gradient vector denotes the direction of steepest descent. The direction of steepest descent, however, is a local property of a function, which changes as the search steps in the direction of steepest descent. Although the method of steepest descent appears to be an efficient technique for converging on the minimum of a function, it is not really effective in solving many problems. The application of the steepest descent method leads to a path composed of parallel and perpendicular steps (see figure 2-6). For a function with long narrow valleys, this method tends to settle into a process of making ever smaller zig-zag steps along a valley and can become hopelessly slow.

The convergence characteristics of the steepest des-

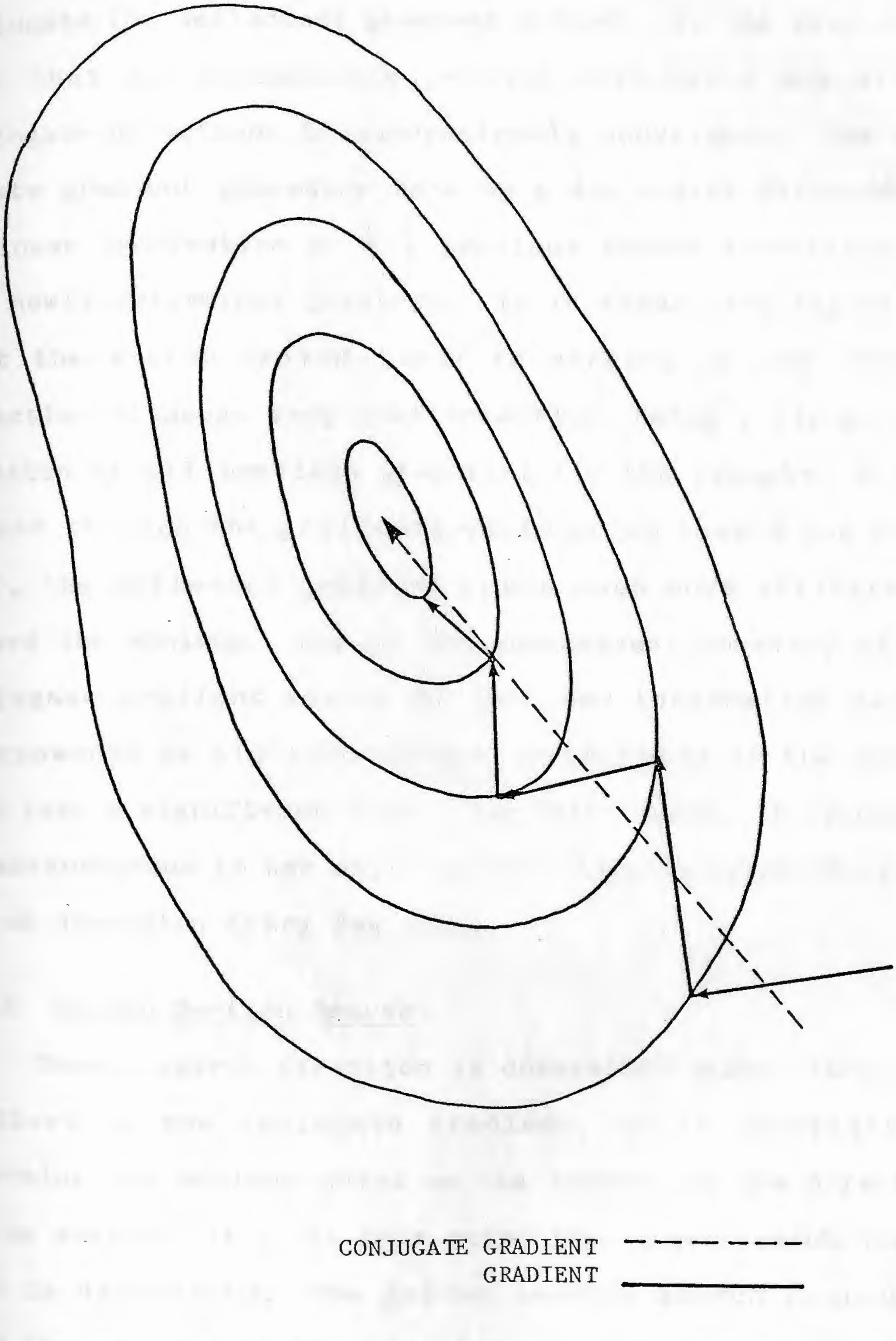
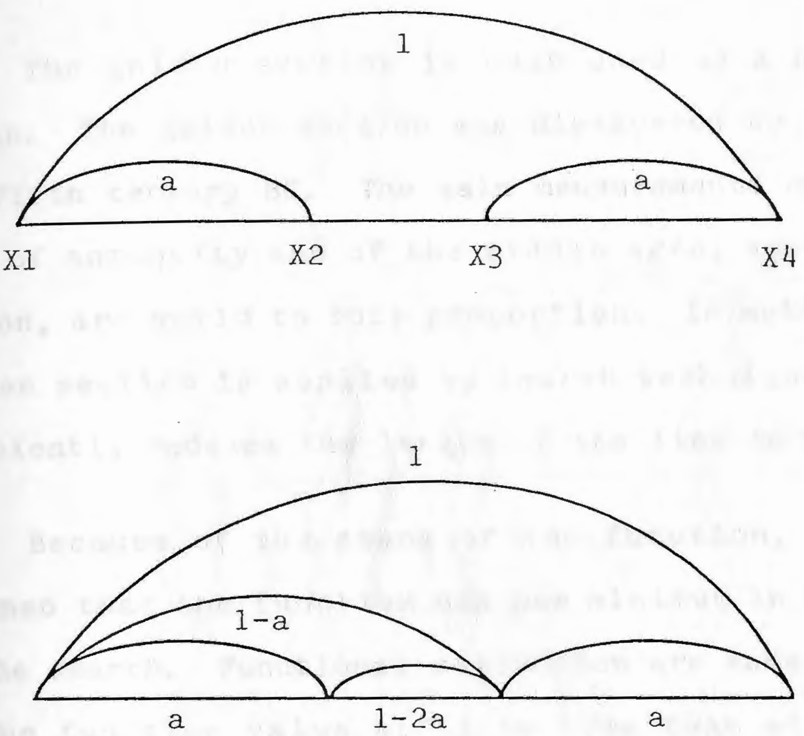



Figure 2-6. Zig-Zag Steps of Steepest Descent Method.

cent method can be greatly improved by modifying it into a conjugate (or deflected) gradient method. It has been shown (15) that any minimization method that makes use of the conjugate directions is quadratically convergent. The conjugate gradient procedure sets up a new search direction as a linear combination of all previous search directions and the newly determined gradient. It is clear from figure 2-6 that the search (solid line) is working in the correct direction although very inefficiently. Using a linear combination of all previous gradients (in the example, a line faired through the gradients would point toward the minimum), the deflected gradient moves much more efficiently toward the minimum. One of the weaknesses, however, of the conjugate gradient search is that new information can be overpowered by old information, especially if the search must take a significant turn. For this reason, it is usually advantageous to use only the new negative gradient as the search direction every few steps.

### 2.3.2 <u>Golden Section Search</u>

Once a search direction is determined using either the gradient or the conjugate gradient, it is necessary to determine the minimum point on the contour in the direction of the search. It is at this point that a new search direction is determined. The golden section search is used to find the minimum in the direction of the search. In this technique, the line to be searched is divided into three parts (see figure 2-7). The two end sections, which are




$$\frac{X4 - X3}{X4 - X1} = \frac{X3 - X2}{X3 - X1}$$

$$\frac{a}{l} = \frac{l - 2a}{l - a}$$

$$a = \frac{3 - \sqrt{5}}{2} = .3892$$

Figure 2-7. Golden Section Search.

equal, are approximately 0.3892 of the total length of the line. The actual length of the line to be searched is arbitrary but important; some considerations for choosing the length of the line are discussed in section 2.5.1.

The golden section is much used as a proportion in design. The golden section was discovered by the Greeks in the fifth century BC. The main measurements of many buildings of antiquity and of the middle ages, such as the Parthenon, are build to this proportion. In mathematics, the golden section is applied to search techniques because it efficiently reduces the length of the line to be searched.

Because of the shape of the function, it is safely assumed that the function has one minimun in the direction of the search. Functional evaluation are made at X2 and X3. If the function value at X3 is less than at X2, then the area from X1 to X2 is eliminated because the function is increasing from X2 to X1 and the minimum, therefore, is to the right of X2. If X2 is less than X3, then X4 is eliminated. The golden section search process is repeated on the remaining 0.6108 of the line. At each iteration the length of the line is reduced by nearly 39 percent and the length of the line after any n iterations is $(0.6108)^n$ of the original length. The search is halted either after n iterations or when the difference between the two functional evaluations at X2 and X3 are less than some value of epsilon.

### 2.3.3 Algorithm Used for the Analysis

The first step in developing the algorithm for the analysis is to determine the gradient of the sum of squares of error function $g(R_0,R_n)$. The gradient is the vector

$$\nabla f = \begin{bmatrix} \dfrac{\partial g}{\partial R_0} = -2\sum_{i=1}^{n}(\hat{B}_i - B_i(R_0,R_n)) * \dfrac{\partial B_i(R_0,R_n)}{\partial R_0} \\ \dfrac{\partial g}{\partial R_n} = -2\sum_{i=1}^{n}(\hat{B}_i - B_i(R_0,R_n)) * \dfrac{\partial B_i(R_0,R_n)}{\partial R_n} \end{bmatrix} \quad 2\text{-}4$$

The lines of the computer algorithm that compute the gradient, the absolute value of the gradient and the negative gradient are shown in figure 2-8.

The multiplier, alpha, used to compute the linear combination of the old and new gradient is

$$\text{alpha} = \left(\frac{\text{absolute value of present gradient}}{\text{absolute value of last gradient}}\right)^2 \quad 2\text{-}5$$

and the conjugate gradient is

present gradient + alpha * last gradient

The conjugate gradient search algorithm used in this analysis (see appendix a) alternately uses the gradient and the conjugate gradient as the search direction. The program repeats the following seven steps for each of 15 iterations:

```
3000 REM COMPUTE GRADIENT
3010  REM
3011  GRO=0:GRN=0
3012  FOR I=0  TO 14
3013  GRO=GRO+(1-I/14)/R(I)*FNSIND(BA(0)-B(I))*(B(I)-BA(I))
3014  GRN=GRN+I/14/R(I)*FNSIND(BA(14)-B(I))*(B(I)-BA(I))
3015  REM PRINT "3015 GRO,GRN ";GRO,GRN
3016  NEXT I
3020 GY=GRO:GZ=GRN
3030 REM PRINT "GRADIENT,GY,GZ ";GY,GZ
3100 REM
3200 REM
4000 REM COMPUTE ABS VALUE OF GRADIENT
4001 REM
4010 ABSG=SQR(GY*GY+GZ*GZ)
4020 REM PRINT "ABS GRADIENT "; ABSG
4030 REM
4040 REM
5000 REM COMPUTE NEGATIVE GRADIENT
5001 REM
5010 NEGGY=-GY:NEGGZ=-GZ
5020 REM PRINT "NEGATIVE GRADIENT ";NEGGY,NEGGZ
5030 RETURN
5050 REM
5060 REM
6000 REM FUNCTIONAL EVALUATION
```

Figure 2-8. Gradient Calculations Algorithm.

1. Compute the gradient.
2. Use golden section search to find the minimum in the direction of the negative gradient.
3. Compute the new gradient.
4. Compute alpha.
5. Compute conjugate gradient.
6. Use golden section search to find the minimum in the direction of the conjugate gradient.
7. Print "SOLUTION" if search is within 300 feet (0.06 nautical miles) of the true minimum in both variables.

Sample computer runs demonstrate that this algorithm finds the minimum of the function for all the tracking geometries tested (see figure 2-9). Note that once the search is within 0.06 nautical miles in both variables, it converges very slowly to the true minimum (9.00, 7.724). For figure 2-9, the search is allowed to continue to 15 iterations after a solution is found. Although it does not always converge in 15 steps from all starting points, it does converge in 15 steps or less in 93 percent of the trials. In the worst case geometry tested (the fifth geometry), the algorithm converges in 15 steps or less from 65 percent of the starting positions. In geometry 5, a maximum of 26 iterations are needed to reach an acceptable solution.

TRIAL: 2
RO=: 2
RN=: 2
10:48:49

| ITERATION | RO | RN | DELTA RO | DELTA RN | SSE |
|---|---|---|---|---|---|
| 1 | 2.838 | 5.700 | 6.162 | 2.026 | 16.10973 |
| 2 | 8.258 | 7.487 | 0.742 | 0.238 | 0.07904 |
| 3 | 8.975 | 7.715 | 0.025 | 0.010 | 0.00033 |
| SOLUTION | | | | | |
| 4 | 8.975 | 7.717 | 0.025 | 0.008 | 0.00009 |
| 5 | 8.975 | 7.717 | 0.025 | 0.008 | 0.00008 |
| 6 | 8.976 | 7.717 | 0.024 | 0.008 | 0.00008 |
| 7 | 8.977 | 7.718 | 0.023 | 0.008 | 0.00007 |
| 8 | 8.978 | 7.718 | 0.022 | 0.007 | 0.00007 |
| 9 | 8.979 | 7.718 | 0.021 | 0.007 | 0.00006 |
| 10 | 8.979 | 7.718 | 0.021 | 0.007 | 0.00006 |
| 11 | 8.983 | 7.719 | 0.017 | 0.006 | 0.00005 |
| 12 | 8.985 | 7.721 | 0.015 | 0.005 | 0.00003 |
| 13 | 8.985 | 7.721 | 0.015 | 0.004 | 0.00003 |
| 14 | 8.986 | 7.721 | 0.014 | 0.004 | 0.00003 |
| 15 | 8.986 | 7.721 | 0.014 | 0.004 | 0.00003 |

FUNCTIONAL EVALUATIONS: 521

TRIAL: 2
RO=: 14
RN=: 14
11:02:46

| ITERATION | RO | RN | DELTA RO | DELTA RN | SSE |
|---|---|---|---|---|---|
| 1 | 15.863 | 9.864 | 6.863 | 2.139 | 3.01447 |
| 2 | 11.207 | 8.577 | 2.207 | 0.852 | 1.22209 |
| 3 | 10.719 | 8.229 | 1.719 | 0.504 | 0.38808 |
| 4 | 10.443 | 8.218 | 1.443 | 0.493 | 0.26254 |
| 5 | 9.331 | 7.861 | 0.331 | 0.136 | 0.05577 |
| 6 | 9.335 | 7.834 | 0.335 | 0.109 | 0.01408 |
| 7 | 9.310 | 7.819 | 0.310 | 0.094 | 0.01288 |
| 8 | 9.305 | 7.824 | 0.305 | 0.099 | 0.01165 |
| 9 | 9.278 | 7.809 | 0.278 | 0.084 | 0.01061 |
| 10 | 9.273 | 7.814 | 0.273 | 0.089 | 0.00940 |
| 11 | 9.267 | 7.809 | 0.267 | 0.083 | 0.00906 |
| 12 | 9.257 | 7.810 | 0.257 | 0.085 | 0.00858 |
| 13 | 9.253 | 7.807 | 0.253 | 0.082 | 0.00810 |
| 14 | 9.186 | 7.790 | 0.186 | 0.065 | 0.00572 |
| 15 | 9.186 | 7.784 | 0.186 | 0.059 | 0.00438 |

FUNCTIONAL EVALUATIONS: 463

11:14:27
TOTAL FUNCTIONAL EVALUATIONS: 463

Figure 2-9. Sample Output for Low and High Search.

## 2.4 NATURE OF THE FUNCTION.

### 2.4.1 Overview

The conjugate gradient search technique converges to a local optimum on a contour. If the contour has no saddle point and only one optimum, then the conjugate gradient search should converge on the optimum in some reasonable number of iterations. Before using the conjugate gradient search for the analysis of the bearings only target motion analysis problem, it is necessary to investigate the shape of the function.

### 2.4.2 Target Motion Analysis Geometries

Fourteen specific geometries are chosen for this analysis. In all cases the tracker uses a two legged tracking maneuver, transiting thirty minutes to the east and then fifteen minutes to the northwest at 7 knots. The target always remains on a constant course and speed from the west to the east (but not parallel to the tracker's course). The search algorithm assumes that the target remains on a constant course and speed.

If it is determined that the target has changed course or speed, it is necessary to restart the tracking sequence. Although an infinite number of tracking geometries is theoretically possible, the geometries are constrained by a number of factors. As is discussed in section 2.2, detection ranges are limited and thus the tracker must keep the target within detection range. The speed at which the tracker can search is limited because of the requirement of minimizing own ship noise that can reduce sonar performance and increase the target's ability to counterdetect the tracker and realize that it is being tracked (1). It is necessary that the tracker either lead or lag the target. If bearing to target remains a constant over the first leg of the track, no information is obtained (21). The target could be on any of a number of tracks (see figure 2-10). The target on tracks 1, 2 and 3 are maintaining the same speed as the tracker while the target on tracks 4, 5 and 6 are faster than the tracker.

The problem of a constant bearing can be avoided either by increasing or decreasing the speed of the tracker or by changing course. Increasing tracker speed causes bearing to target to drift astern. A target on a relative bearing 090 at the beginning of a search may be on 091 after one minute, 092 after two minutes, and so forth. This is a bearing rate of 1 degree per minute. Likewise, decreasing tracker speed below target speed causes bearing to target to drift toward the bow. Bearing rate must be sufficiently high that bearing changes are not overwhelmed by random bearing errors. If bearing rate is low, bearing entries can be made at longer time intervals, thus increasing the bearing delta between entries. This however increases the length of time required to collect a series of bearings and is a potential liability because the probability that the target will change course or speed increases with time.

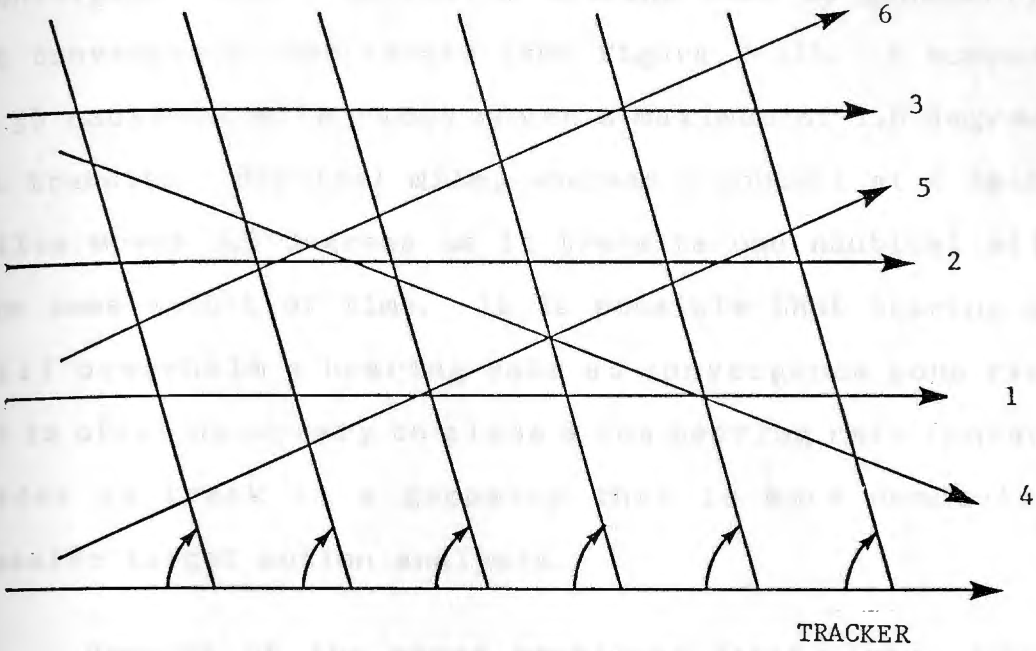



Figure 2-10. Possible Target Course Given Constant Bearing from Tracker.

Convergence zone tracking is especially difficult. First it is difficult to maintain detection in a narrow convergence zone. Secondly, bearing rate is generally low at convergence zone ranges (see figure 2-11). A contact at a 35 nautical mile range moves a maximum of 1.6 degrees as it transits 1 nautical mile, whereas a contact at 6 nautical miles moves 9.5 degrees as it transits one nautical mile in the same amount of time. It is possible that bearing error will overwhelm a bearing rate at convergence zone ranges. It is often necessary to close a low bearing rate contact in order to track in a geometry that is more conducive to passive target motion analysis.

Because of the above mentioned constraints, fourteen specific geometries are chosen for the analysis (see table 2-1). Four of these geometries are depicted with their respective contours later in this chapter. In all cases, tracker course maneuver remains constant. These geometries provide a representative sample of tracking geometries including variations in target speed, target course, and initial bearing to target.

### 2.4.3 Plots of the Contours

The first computer program written for this analysis is called CONTOUR (see appendix b). The first task of this program is to compute the coordinates of the target and the tracker as well as bearing to target for 15 time steps in each geometry. The choice of 15 time steps is arbitrary.

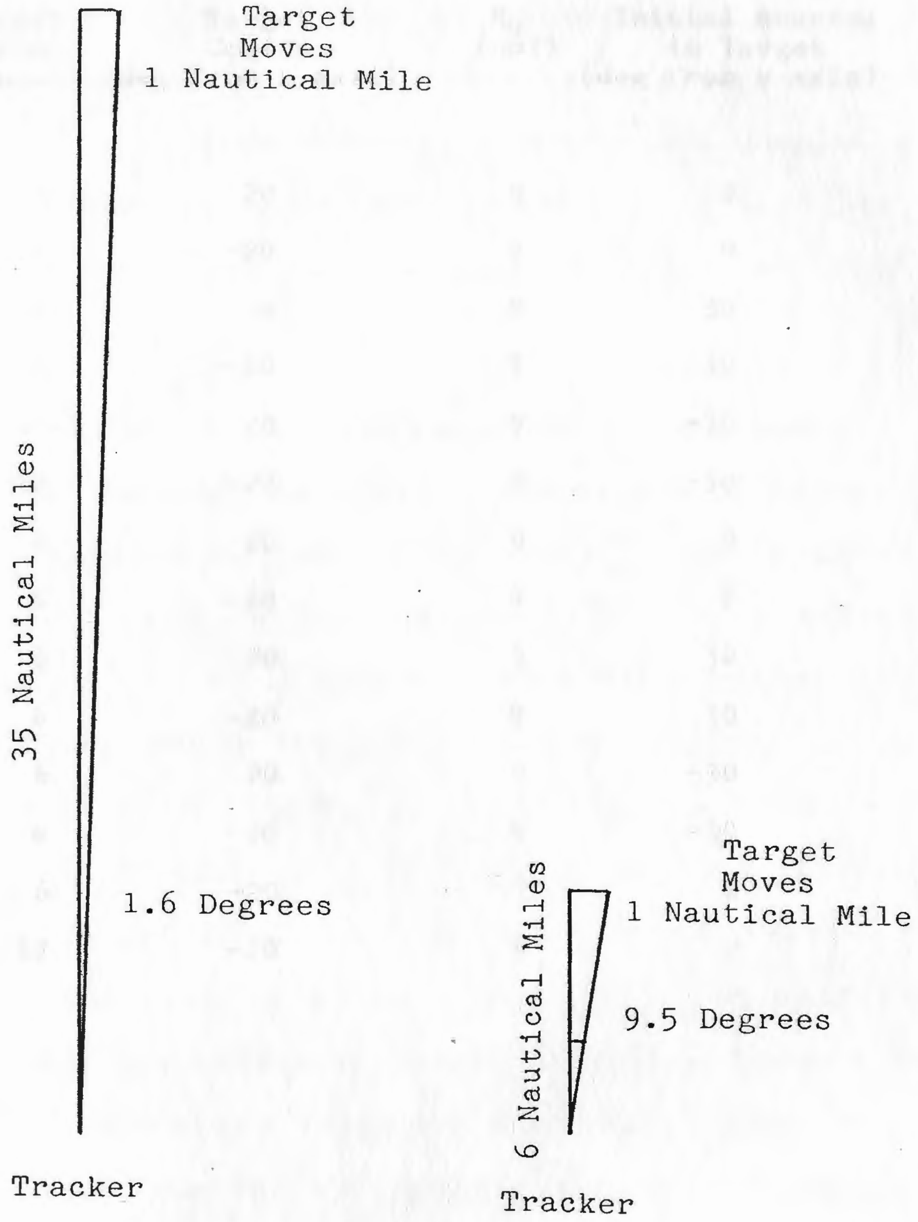



Figure 2-11. Convergence Zone Tracking.

Table 2-1. Geometries Selected for Analysis.

| Geometry Number | Target Speed (knots) | Target Course (deg from x axis) | $R_0$ (nmi) | Initial Bearing to Target (deg from y axis) |
|---|---|---|---|---|
| 1 | 12 | 20 | 9 | 0 |
| 2 | 12 | -20 | 9 | 0 |
| 3 | 12 | 20 | 9 | 30 |
| 4 | 12 | -20 | 9 | 30 |
| 5 | 12 | 20 | 9 | -30 |
| 6 | 12 | -20 | 9 | -30 |
| 7 | 6 | 20 | 9 | 0 |
| 8 | 6 | -20 | 9 | 0 |
| 9 | 6 | 20 | 9 | 30 |
| 10 | 6 | -20 | 9 | 30 |
| 11 | 6 | 20 | 9 | -30 |
| 12 | 6 | -20 | 9 | -30 |
| 13 | 6 | -20 | 4 | 0 |
| 14 | 12 | -20 | 4 | 0 |

This tracking information is printed (see table 2-2) and stored on a disk file for use by subsequent programs. Arithmetic bearings are measured counter-clockwise from the horizontal axis. Relative bearings are measured clockwise from the direction in which the tracker is transiting. "True" bearings are measured clockwise from the vertical axis (North).

The second task of this program is to generate data that can be used to draw the contour map of sum of squares of error for various values of $R_0$ and $R_n$. The program provides data to plot the four contour lines for sum of squares of error = 2, 5, 10 and 20. It first computes the valley line of the surface by using a golden section search to determine the values of $R_n$ at which the function is a minimum for all integer values of $R_0$ from 2 to 25 (see figure 2-12). To determine the sum of squares of error = 2 contour line, the program again uses the golden section search on a penalty function to determine the two points on either side of the valley at which the (sum of squares of error - 2) is a minimum for all integer values of $R_0$ from 2 to 25. If the points on both sides of the valley actually fall in the valley, then the sum of squares of error is always greater than 2 for that value of $R_0$. In this case, the sum of squares of error for all three points at $R_0$ will be identical and greater than 2.0. This process is repeated for SSE - 5, SSE - 10 and SSE - 20 (see figure 2-13).

Table 2-2. Output of CONTOUR -- True Data for a Geometry.

RUN NUMBER: 1

SPEED BLUE (NMI): 7
SPEED TARGET: 6
COURSE TARGET: 110
TIME STEP (MIN): 3
RO (NMI): 9
BEARING TO TARGET: 0

| TRACKER | | TARGET | | MEASURED DATA | | | |
|---|---|---|---|---|---|---|---|
| | | | | ARITH | RELATIVE | TRUE | |
| XB | YB | XT | YT | BEARING | BEARING | BEARING | RANGE |
| 0.00 | 0.00 | 0.00 | 9.00 | 90.00 | 270.00 | 0.00 | 9.00 |
| 0.35 | 0.00 | 0.28 | 8.90 | 90.44 | 269.56 | -0.44 | 8.90 |
| 0.70 | 0.00 | 0.56 | 8.79 | 90.89 | 269.11 | -0.89 | 8.80 |
| 1.05 | 0.00 | 0.85 | 8.69 | 91.35 | 268.65 | -1.35 | 8.69 |
| 1.40 | 0.00 | 1.13 | 8.59 | 91.82 | 268.18 | -1.82 | 8.59 |
| 1.75 | 0.00 | 1.41 | 8.49 | 92.30 | 267.70 | -2.30 | 8.49 |
| 2.10 | 0.00 | 1.69 | 8.38 | 92.79 | 267.21 | -2.79 | 8.39 |
| 2.45 | 0.00 | 1.97 | 8.28 | 93.29 | 266.71 | -3.29 | 8.30 |
| 2.80 | 0.00 | 2.26 | 8.18 | 93.81 | 266.19 | -3.81 | 8.20 |
| 3.15 | 0.00 | 2.54 | 8.08 | 94.34 | 265.66 | -4.34 | 8.10 |
| 2.90 | 0.25 | 2.82 | 7.97 | 90.62 | 44.38 | -0.62 | 7.73 |
| 2.66 | 0.49 | 3.10 | 7.87 | 86.54 | 48.46 | 3.46 | 7.39 |
| 2.41 | 0.74 | 3.38 | 7.77 | 82.10 | 52.90 | 7.90 | 7.09 |
| 2.16 | 0.99 | 3.66 | 7.67 | 77.30 | 57.70 | 12.70 | 6.84 |
| 1.91 | 1.24 | 3.95 | 7.56 | 72.17 | 62.83 | 17.83 | 6.65 |

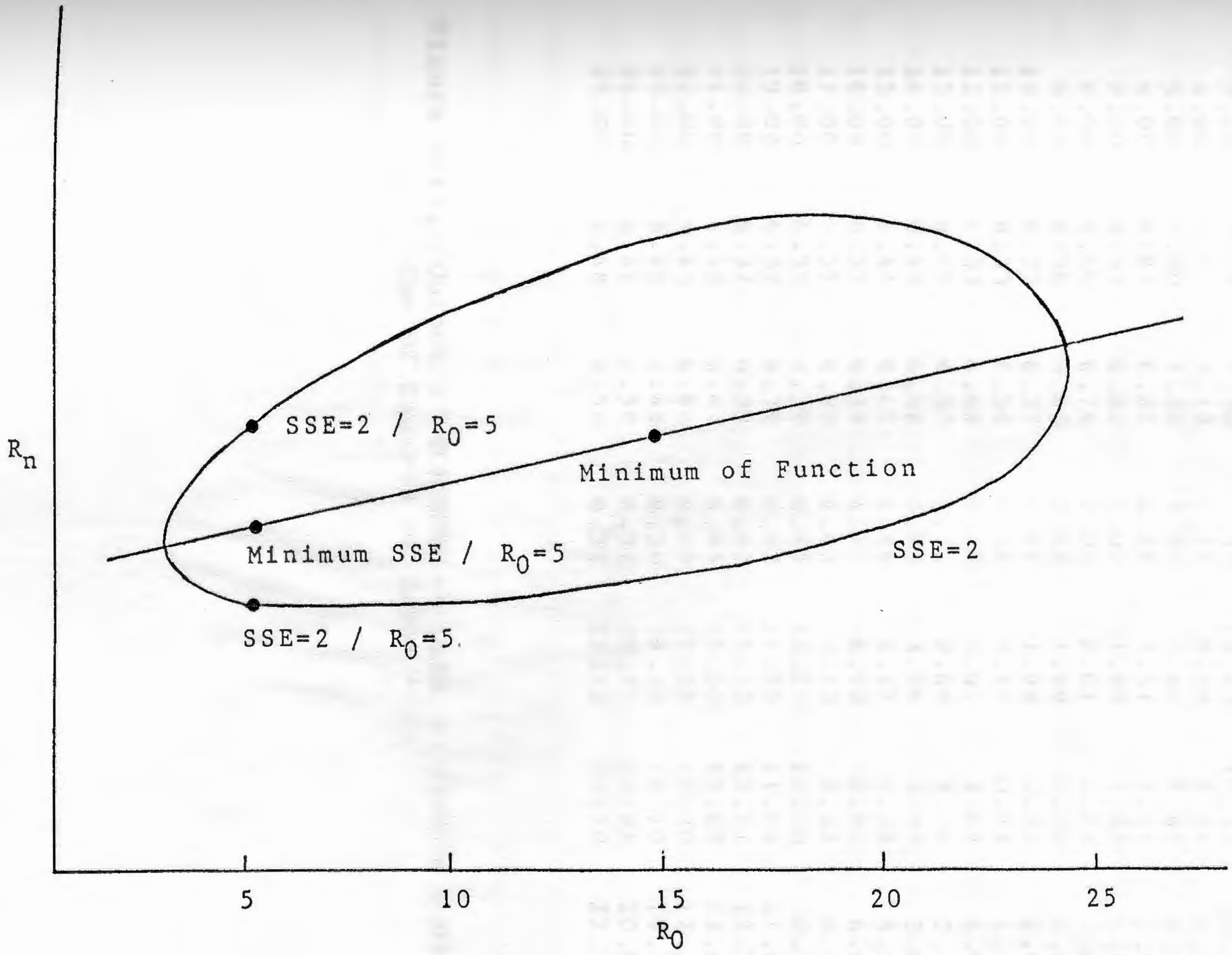



Figure 2-12. Golden Section Search Used to Generate Data to Plot Contours.

08-23-1983
RUN NUMBER: 1

CONTOUR: SSE= 2

| | Rn | | | SSE | | |
|---|---|---|---|---|---|---|
| RO | LOWER | CENTER | UPPER | LOWER | CENTER | UPPER |
| 2.00 | 7.50 | 7.51 | 7.52 | 20.79 | 20.79 | 20.80 |
| 3.00 | 7.29 | 7.30 | 7.32 | 13.43 | 13.43 | 13.44 |
| 4.00 | 7.11 | 7.12 | 7.14 | 8.39 | 8.38 | 8.37 |
| 5.00 | 7.00 | 7.02 | 7.03 | 4.86 | 4.87 | 4.88 |
| 6.00 | 6.87 | 6.87 | 6.89 | 2.51 | 2.51 | 2.50 |
| 7.00 | 6.57 | 6.80 | 7.02 | 1.99 | 1.03 | 1.99 |
| 8.00 | 6.40 | 6.74 | 7.04 | 2.01 | 0.25 | 2.00 |
| 9.00 | 6.30 | 6.63 | 7.01 | 1.99 | 0.00 | 2.01 |
| 10.00 | 6.25 | 6.57 | 6.94 | 1.98 | 0.21 | 2.01 |
| 11.00 | 6.33 | 6.56 | 6.84 | 1.41 | 0.77 | 1.99 |
| 12.00 | 6.33 | 6.48 | 6.67 | 2.01 | 1.62 | 1.99 |
| 13.00 | 6.47 | 6.50 | 6.51 | 2.69 | 2.70 | 2.71 |
| 14.00 | 6.45 | 6.48 | 6.49 | 3.96 | 3.97 | 3.98 |
| 15.00 | 6.41 | 6.42 | 6.44 | 5.37 | 5.37 | 5.37 |
| 16.00 | 6.37 | 6.39 | 6.42 | 6.92 | 6.91 | 6.89 |
| 17.00 | 6.37 | 6.39 | 6.42 | 8.53 | 8.52 | 8.51 |
| 18.00 | 6.37 | 6.39 | 6.42 | 10.21 | 10.20 | 10.19 |
| 19.00 | 6.37 | 6.39 | 6.42 | 11.95 | 11.94 | 11.93 |
| 20.00 | 6.37 | 6.39 | 6.42 | 13.72 | 13.71 | 13.70 |
| 21.00 | 6.40 | 6.42 | 6.44 | 15.50 | 15.49 | 15.49 |
| 22.00 | 6.45 | 6.48 | 6.49 | 17.29 | 17.30 | 17.31 |
| 23.00 | 6.45 | 6.48 | 6.50 | 19.10 | 19.10 | 19.11 |
| 24.00 | 6.47 | 6.50 | 6.52 | 20.91 | 20.91 | 20.92 |
| 25.00 | 6.48 | 6.50 | 6.52 | 22.71 | 22.70 | 22.71 |

Figure 2-13. Output of CONTOUR -- Data to Plot a Contour for Sum of Squares of Error = 2.

Four representative contours are shown in figure 2-14. The surfaces for all fourteen geometries are similar in shape and in orientation. The individual contour line for a particular sum of squares of error is essentially canoe shaped. Recall that these are three dimensional figures; the contour lines lie above the $R_0,R_n$ plane. They are shown in two dimensions because it is easier to portray the shape and orientation of the valley in two dimensions.

The analysis in this report uses bearings without error. Obviously random error is a part of sonar bearings as it is in any measurement. Random bearing error does not, however, substantially change the shape of the function; it lifts the function above the $R_0,R_n$ plane (see figure 2-15). Contours are run for geometry 1 using bearings with random error. These contours follow the same shape as the baseline geometry but the contour is uniformly higher by the sum of squares of random error. Because bearing error does not change the shape of the function, it is possible to do the analysis with true bearings.

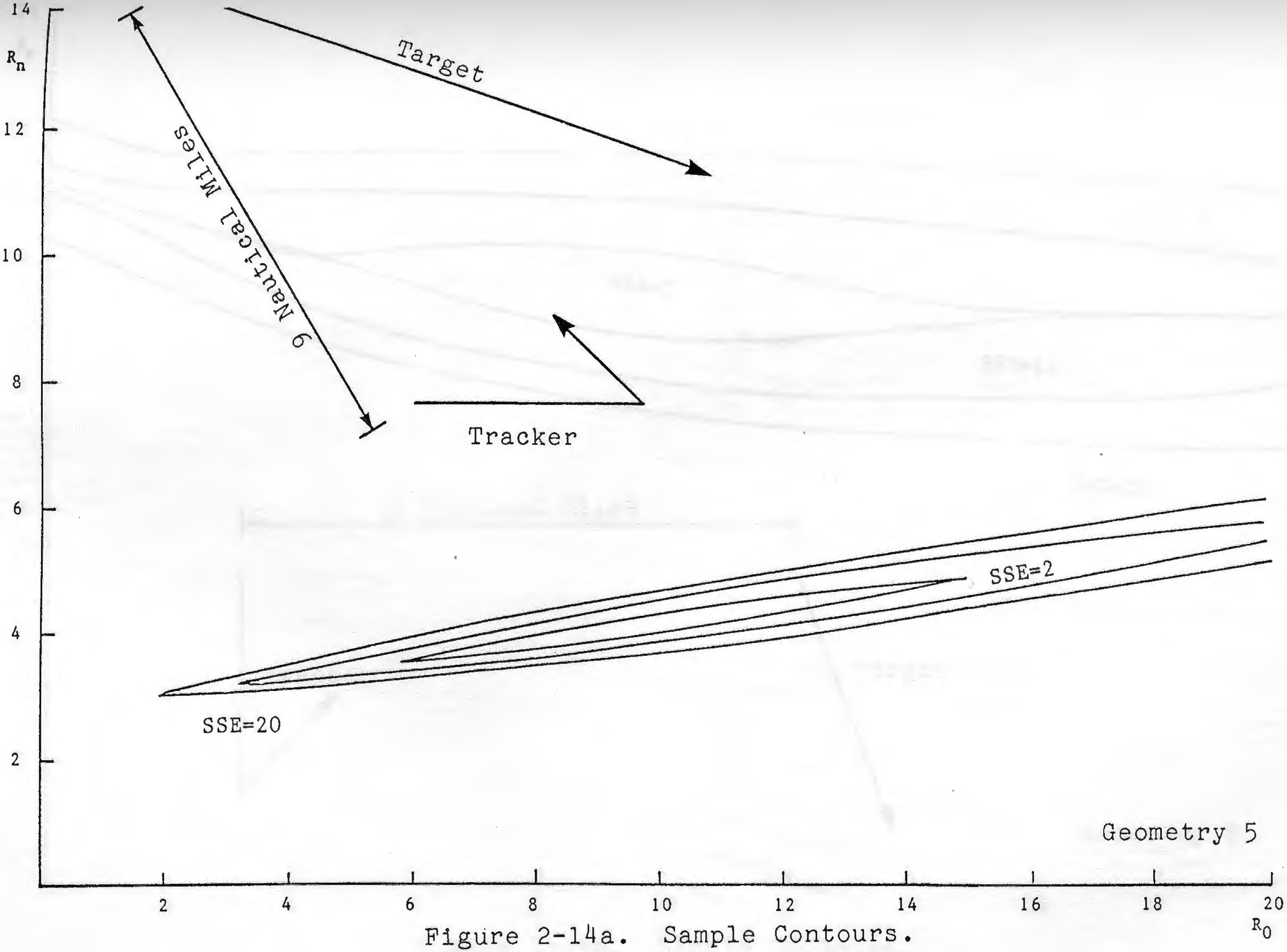



Figure 2-14a. Sample Contours.

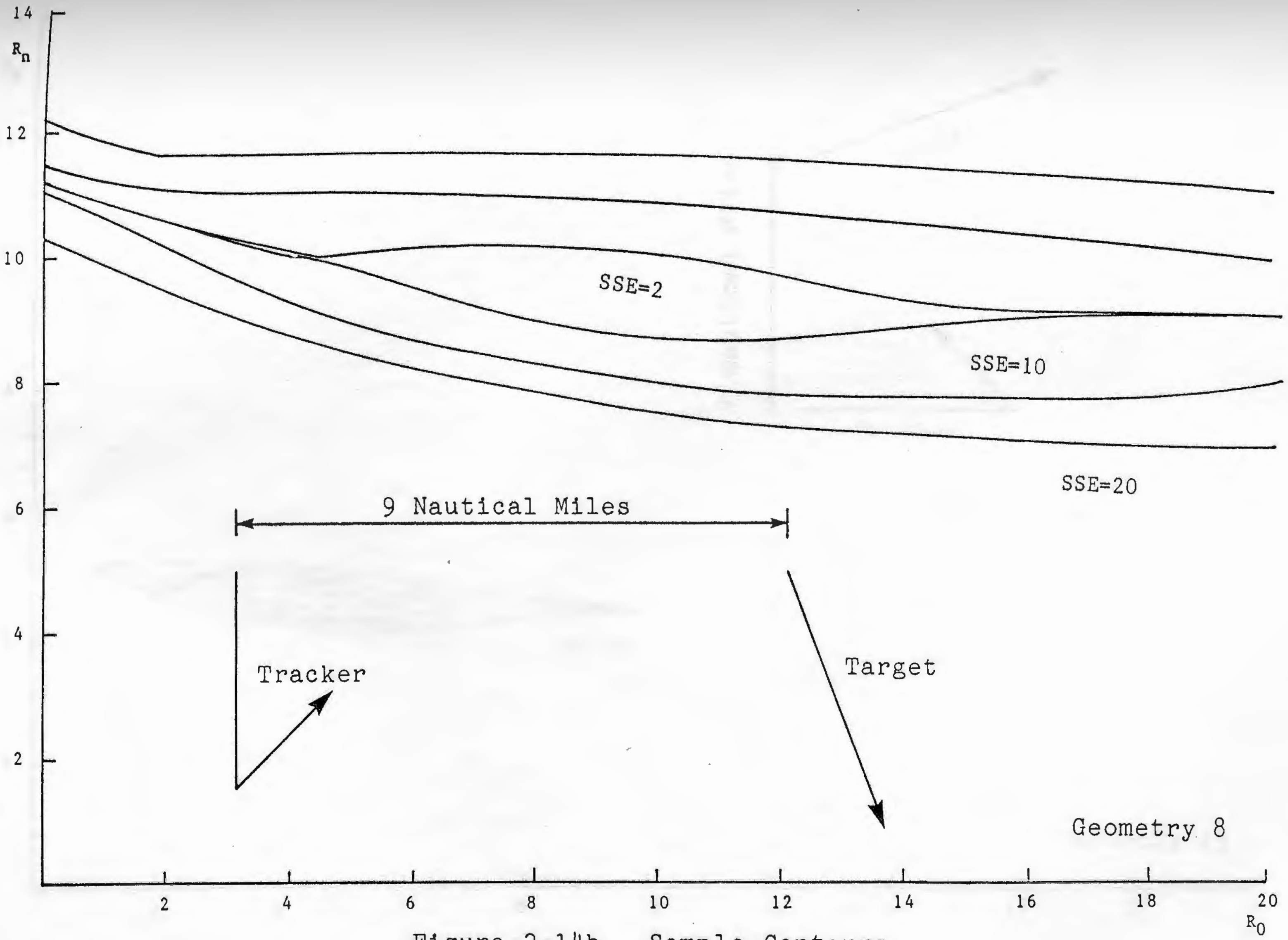



Figure 2-14b. Sample Contours.

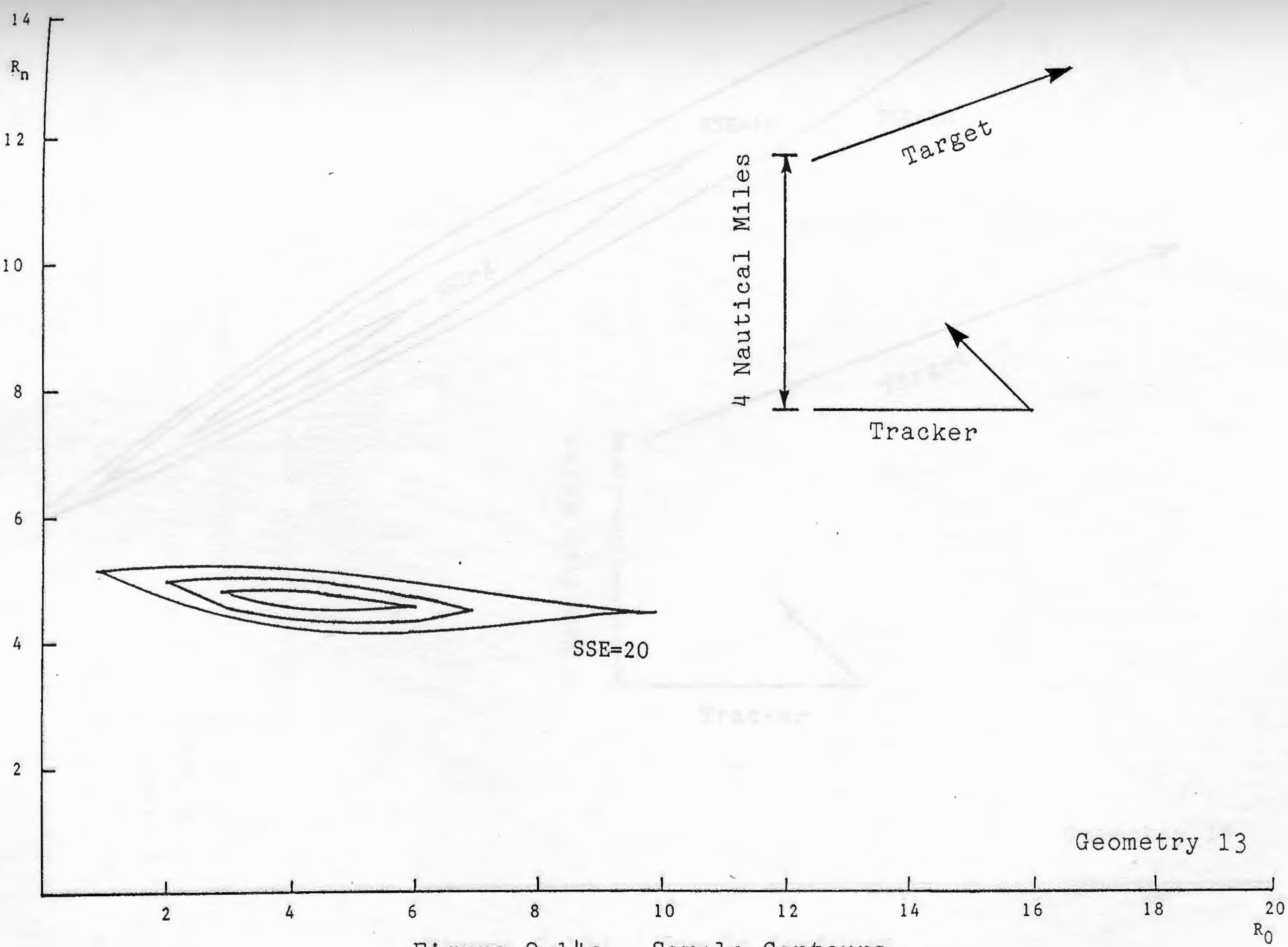



Figure 2-14c. Sample Contours.

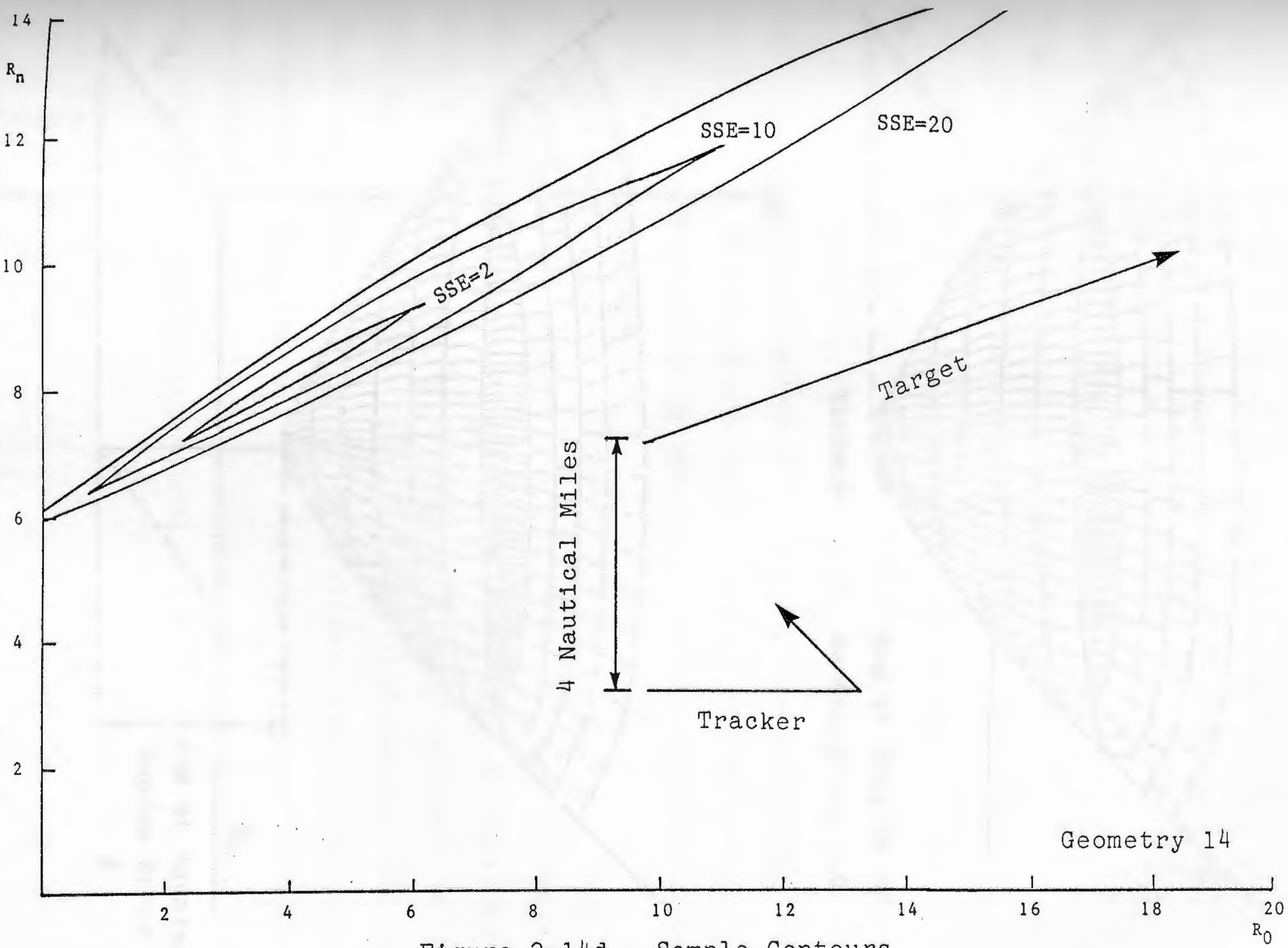



Figure 2-14d. Sample Contours.

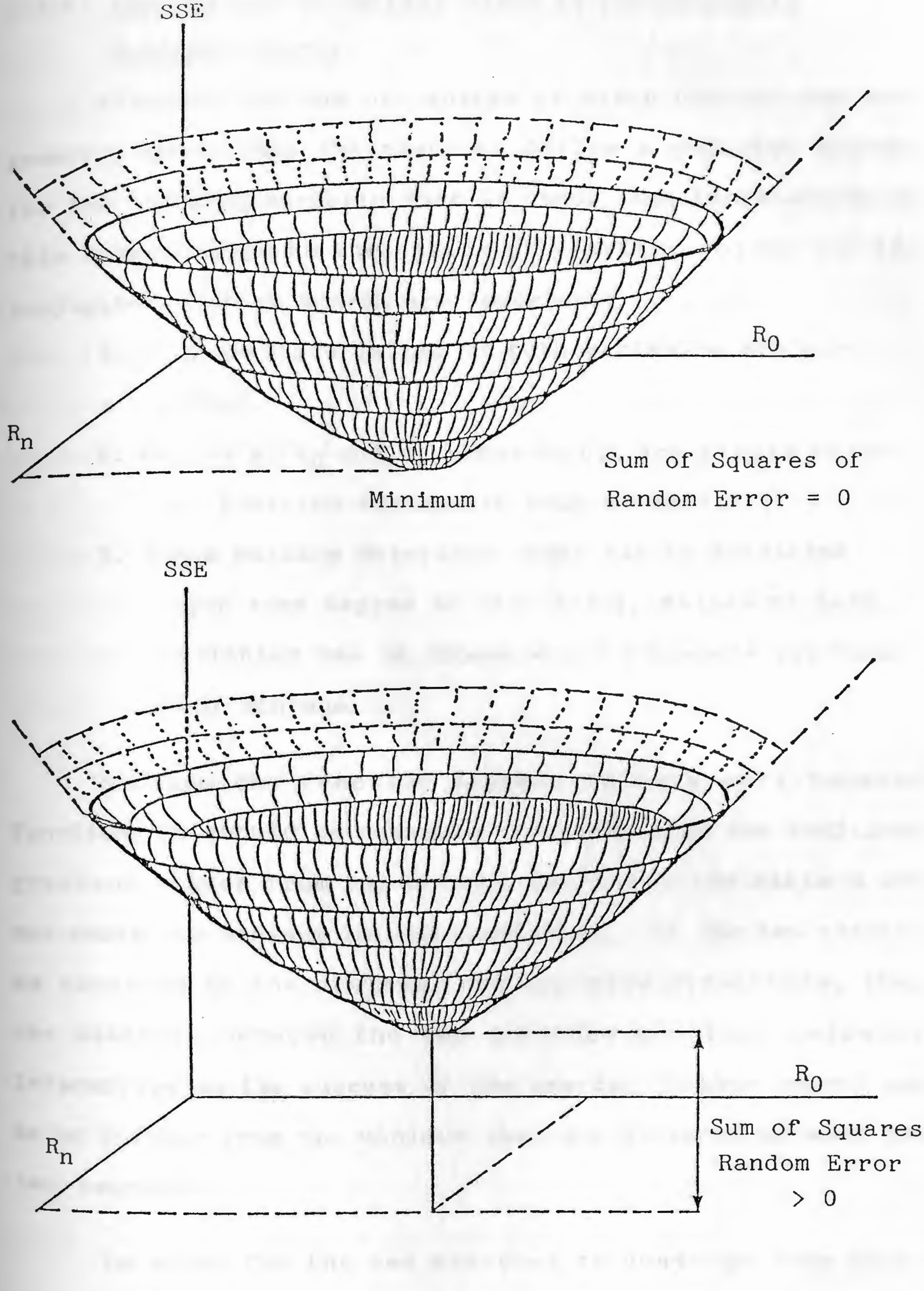



Figure 2-15. Effect of Random Error on Sum of Squares of Error Function.

### 2.4.4 Implication of Surface Shape to the Conjugate Gradient Search

Plotting the sum of squares of error contour for each geometry shows that the contours follow a definite pattern for the tracking scenario that is used. The implications of this common shape to the choice of starting points for the conjugate gradient search are important:

1. Only positive values of both variables are permitted.
2. Values of $R_0$ and $R_n$ close to 0,0 are always below the function minimum in both variables.
3. Since maximum detection range can be predicted (with some degree of certainty), values of both variables can be chosen which are above the function minimum.

Because the function appears to be a well behaved function, it should be possible to search with the conjugate gradient search from two points, one below the minimum and one above the minimum in both variables. If the two searches converge on the minimum from opposite directions, then the distance between the two searches provides valuable information on the success of the search. Either search can be no further from the minimum than the distance between the two searches.

In order for the two searches to converge from opposite directions, it is necessary that the searches begin in opposite quadrants as defined in this paragraph. For purposes of illustration, two lines are drawn through the minimum, one tangent to the valley and the other perpendicular to the valley (see figure 2-16). Although the valley line is not straight, the parallel line is a good approximation of the valley in the area of the search. A conjugate gradient search that begins in the neighborhood of 0,0 should remain in the lower left quadrant and converge on the minimum with steps in which $R_0$ and $R_n$ are always less than the minimum. Likewise a search that begins at high values of $R_0$ and $R_n$ should remain in the upper right quadrant (see figure 2-17). The output of the computer runs indicate that the two searches do consistently converge on the minimum from opposite directions given a proper choice of starting points for the two searches.

Note that it is not sufficient for a starting point for the high search to be above the minimum in both variables. A starting point can be above the minimum in both variables and yet be in the lower right quadrant if the valley has a positive slope. A search that begins in the two lower quadrants could converge on each other and yet be quite distant from the minimum particularly in the vertical direction (see figure 2-18). The high search must commence high enough above the minimum in both variables that it be above the valley line in the vertical variable.

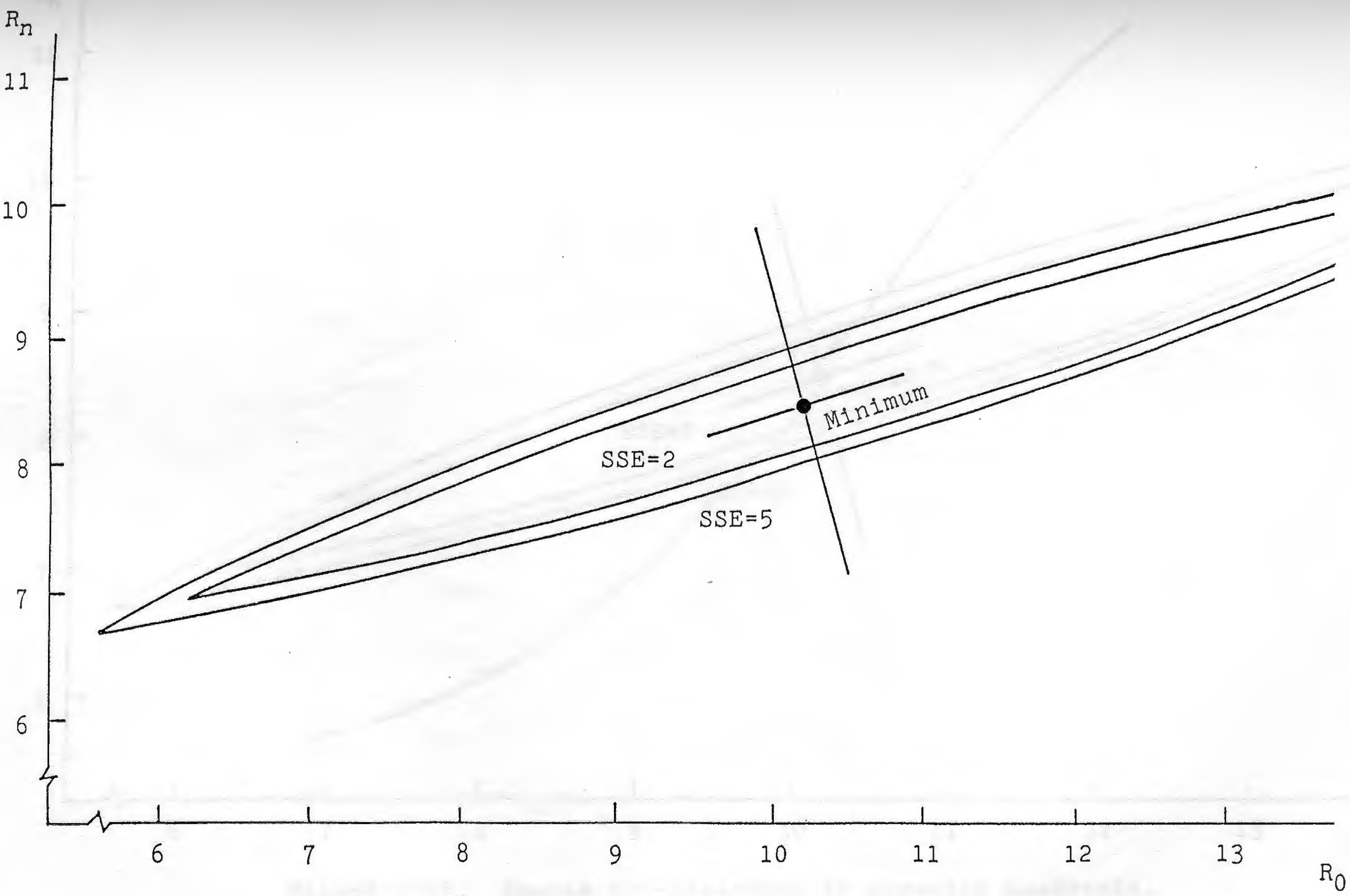



Figure 2-16. Quadrants on a Contour.

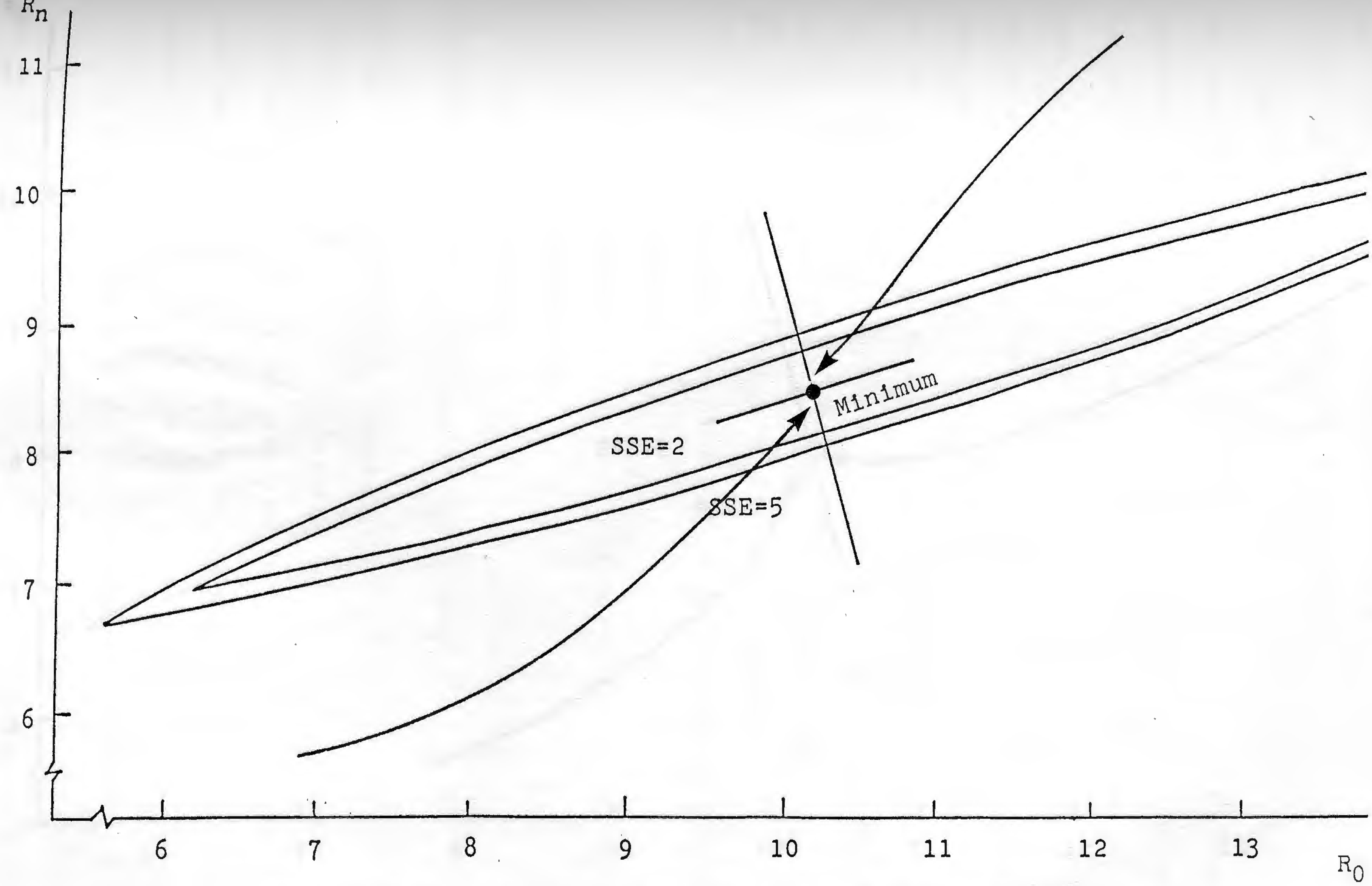



Figure 2-17. Sample Run--Searches in Opposite Quadrants.

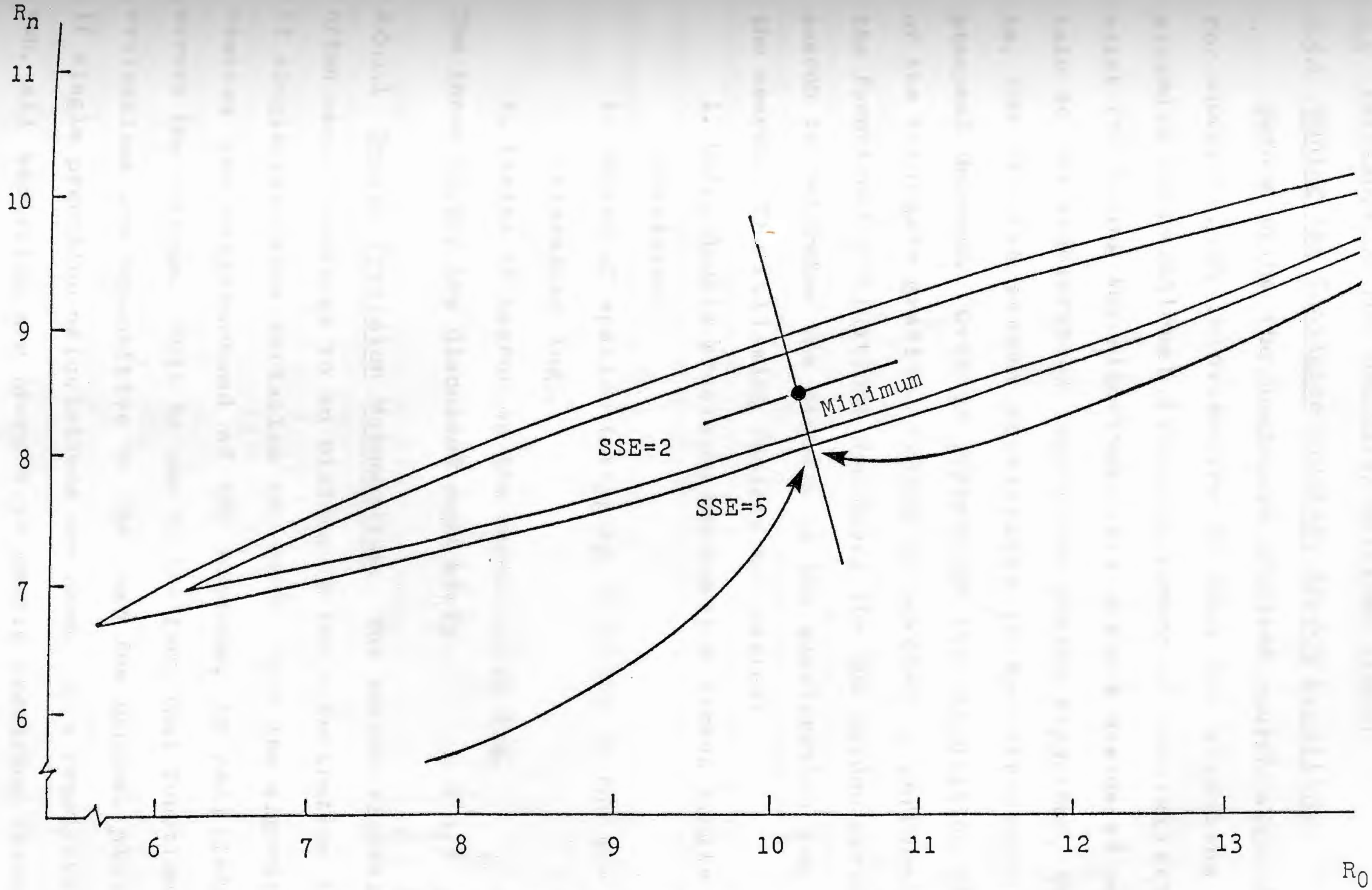



Figure 2-18. Sample Run--Searches in Adjacent Quadrants.

## 2.5 INITIALIZING THE CONJUGATE GRADIENT SEARCH

### 2.5.1 Tuning the Conjugate Gradient Search Algorithm

Before using the conjugate gradient search algorithm for analysis, it is necessary to tune the algorithm to minimize computational time. A number of possibilities exist for tuning the algorithm. All options discussed pertain to the acceleration leg of the search algorithm, that is, how far the search accelerates in the direction of steepest descent. Over 90 percent of the calculation time of the conjugate gradient search is devoted to performing the functional evaluations necessary for the golden section search to determine the minimum on the acceleration leg of the search. The following factors are tested:

1. Using double precision mathematics versus single precision.
2. Choice of epsilon (stopping criterion on the acceleration leg).
3. Limits of search on the acceleration leg.

The three topics are discussed separately.

2.5.1.1 Double Precision Mathematics. The search algorithm often cannot converge to an minimum on the acceleration leg if single precision variables are used. Once the algorithm reaches the neighborhood of the minimum, it oscillates across the optimum. This is due to the fact that functional evaluations are insensitive to the last few decimal places if single precision calculations are used. In a sensitivity run, all variables are changed to double precision (except for loop counters), and a run repeated. Double precision calculations require significantly more computer time than do single precision calculations and do not enable the algorithm to converge with fewer functional evaluations. Double precision is actually counterproductive in the cases tested. In half the cases tried, the algorithm uses all fifteen iterations and converges more slowly toward the minimum. Even with the conjugate gradient search, double precision may keep the search too tight to the valley.

Double precision calculations are required only in one line of the program, specifically the line that computes the sum of squares of error. Note that an error of 0.003 is not significant at the sixth digit if the sum of squares of error is 2.000000. Consequently the error and the sum of squares of error are both double precision variables; all other variables, except for loop counters are single precision. Loop counters are integers.

2.5.1.2 <u>Change of Epsilon</u>. The search on the acceleration leg terminates when the difference between two functional evaluations that are on opposite sides of the minimum is less than epsilon. An initial value of 0.00001 is used. A value smaller than 0.00001 is beyond the sensitivity of single precision. Epsilon set equal to 0.0001 is unacceptable. Sixty percent of the trials tested at epsilon = 0.0001 do not provide solutions to the required tolerance. Twenty percent of the solutions were significantly away from the minimum.

2.5.1.3 Limits of Search on the Acceleration Leg. The choice of lambda, the maximum length of the acceleration leg, is important to computational efficiency. If lambda is small, the algorithm works too slowly toward the minimum (and may not reach the optimum in the maximum fifteen steps allowed by the algorithm.) If lambda is large, the algorithm is slow in searching for the minimum along the acceleration leg (and may not reach the minimum in the twenty iterations allowed for the golden section search). If lambda is sufficiently long that the search evaluates a location on the contour in negative space, the algorithm stalls at that point. The geometry of the problem does not hold for these locations; computed bearings, for instance, are no longer within plus or minus ninety degrees.

A set of ten runs is made comparing various lengths of the accelertion leg (see table 2-3).

Table 2-3. Sensitivity of Search Time to the Length of the Acceleration Leg.

| Lambda (Nautical Miles) | Number of Iterations (Conjugate Gradient Calculations) |
|---|---|
| 100 | 3553 |
| 50 | 2467 |
| 20 | 2190 |
| 10 | 1881 |
| 4 | 2999 |

A large lambda at the beginning of the search is desired because it enables the search to move significantly toward the valley. A shorter lambda is desired in later iterations as the search moves more slowly along the valley. A shorter lambda reduces the number of iterations of the golden section search. If however, lambda becomes too small, the search can move unacceptably slowly.

The following decision is made for the analysis runs. Lambda is initially set to 10. During each of the fifteen iterations, lambda is divided by the iteration number. Lambda is, however, never allowed to be less than 2. This set of values for lambda allows the search to move quickly toward the valley and to search more efficiently along the valley. The results of runs show that if lambda is greater than 2, then the algorithm will, at least occasionally, stall in negative space. If this occurs, the algorithm divides the initial lambda by 2 and repeats the search.

2.5.2 <u>Choice of Starting Position</u>

An important area of investigation is the attempt to identify regions in $R_0,R_n$ space that contains starting points that yield success in a low number of iterations. The number of possible starting points in $R_0,R_n$ space is infinite as are the number of possible contours. Consequently an exhaustive search of $R_0,R_n$ space is impossible. The number of possible starting points and contours is constrained only by the decimal accuracy of the computer being used.

A number of observations are made concerning the starting points for the searches. Some starting points are better in that the search iterates to a solution in two or three steps from these points whereas the search sometimes does not reach an acceptable solution in fifteen steps from other points. As is noted in section 2.5.1.1, a starting point that is good when the single precision variables are used, is not necessarily as good when double precision variables are used. This would suggest that starting coordinates that are favorable on one computer may be unfavorable on another computer if different decimal accuracy is used. Test runs also show that a starting position removed from a good position by a very small distance is not necessarily a good starting position.

Figure 2-19 shows the number of iterations required to reach a solution using $R_n$ = 3.0 and 3.5 nautical miles for $R_0$ varying from 1 to 40. Although lower values of $R_0$ are on the average somewhat better than high values, considerable randomness occurs in the plot. The regression line on figure 2-19 shows that the average number of iterations required to attain a solution does increases from 5.5 to 11.6 as $R_0$ increases from 1 to 40, but the correlation coefficient of 0.43 is explained by the high degree of randomness found in the scatter plot. It is noted that the regression line and correlation coefficients are virtually

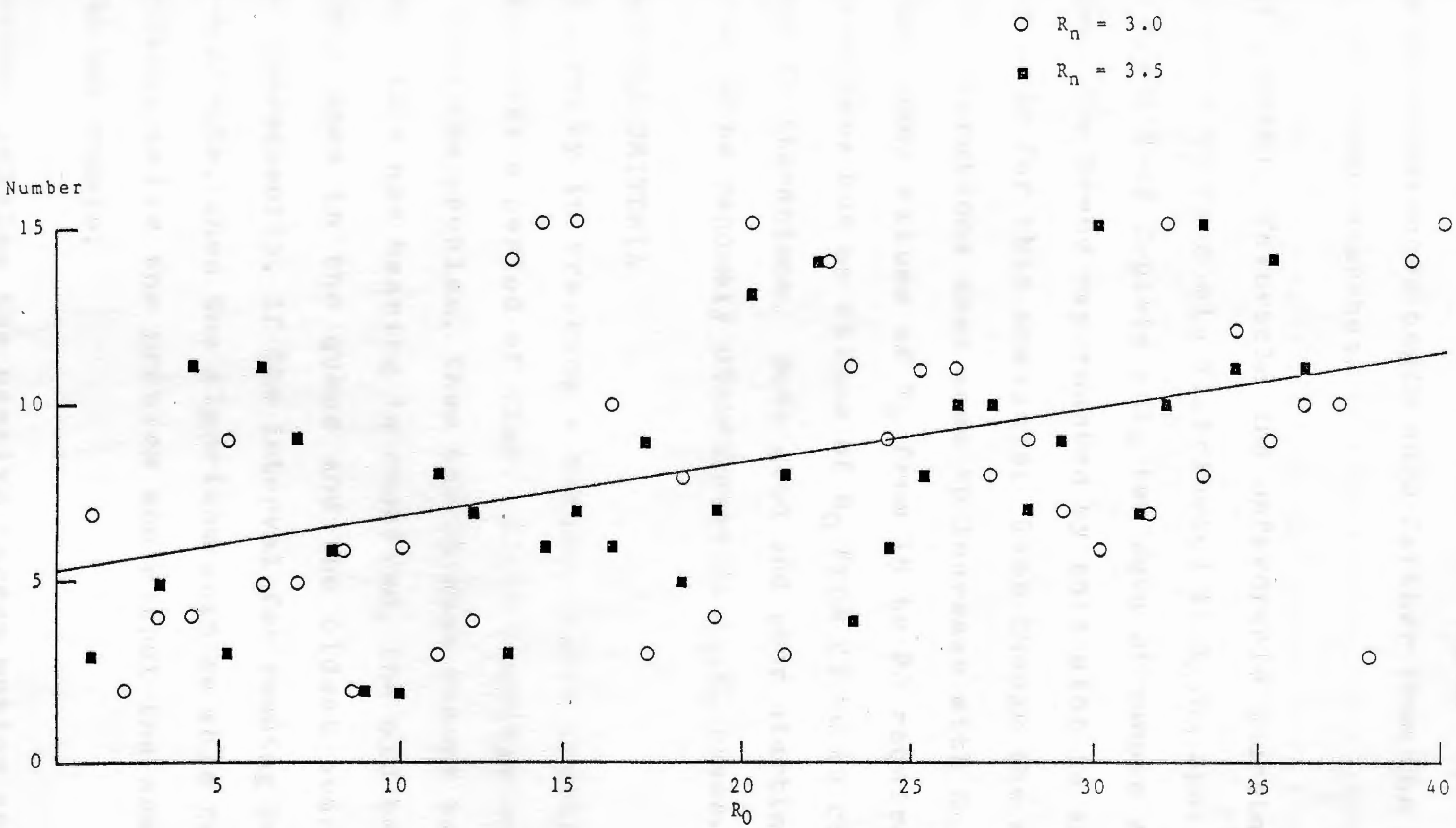



Figure 2-19. Number of Iterations to Reach Solution.

identical for both values of $R_n$. One possible explanation for the average longer search times for higher values of $R_0$ is that these searches begin much farther from the minimum than do the lower searches.

In general, favorable and unfavorable starting positions seem to be randomly distributed in $R_0,R_n$ space. Although figure 2-19 depicts only two sets of sample starting positions, the trend represented by this plot is shown in all runs made for this analysis. Even though the average number of iterations does seem to increase with $R_0$, it is noted that many values of $R_0$ from 15 to 22 require 13 or more iterations but no values of $R_0$ from 23 to 29 requires more than 11 iterations. Both good and poor starting positions seem to be randomly distributed in $R_0,R_n$ space.

## 2.6 STOPPING CRITERIA

Typically in tracking a target, sonar bearings are collected over a period of time. If 15 bearings are used for solving the problem, then the 15 most recent bearings are used. As a new bearing is received, the old bearings are pushed down in the queue and the oldest bearing is dropped. Consequently, if the interval for reading bearings is once a minute, then the algorithm must be able to input the new data, solve the problem and output the answer in less than one minute.

Because solving the passive target motion analysis problem is a real time operation, some stopping criteria must be invoked. Using sum of squares of error less than epsilon is not very helpful since the actual random bearing error is a variable and the minimum of the function is some positive value which includes the sum of the random error and is not zero. A commonly used criterion is n iterations (18). The limitations of using n iterations as a stopping criterion are discussed in section 3.

SECTION 3

ANALYSIS OF SINGLE CONJUGATE GRADIENT SEARCH

3.1 OVERVIEW

The traditional method of solving the bearings only target motion analysis problem using the conjugate gradient search on a computer is to have either the operator or the computer choose starting coordinates for the search (9). The computer program performs n iterations and presents a solution. As is shown in section 2.5, the choice of starting position is one factor in determining the number of iterations required to obtain an acceptable solution. This section establishes a baseline for the effectiveness of the conjugate gradient algorithm in obtaining a solution for the 14 basic geometries discussed in section 2.4.

3.2 TESTING THE CONJUGATE GRADIENT SEARCH

For each geometry, the conjugate gradient search is used to search from ten low starting points and ten high starting points. The 20 starting coordinates are listed in table 3-1. The program used for this part of the analysis is called CGHL (see appendix c. It is similar to the program in appendix a with three exceptions:

Table 3-1. Starting Coordinates for Conjugate Gradient Search.

| Position Number | Low Search $R_0$ | Low Search $R_n$ | High Search $R_0$ | High Search $R_n$ |
|---|---|---|---|---|
| 1 | 1 | 1 | 15 | 15 |
| 2 | 2 | 2 | 14 | 14 |
| 3 | 3 | 3 | 13 | 13 |
| 4 | 1 | 2 | 15 | 14 |
| 5 | 2 | 1 | 15 | 13 |
| 6 | 1 | 3 | 14 | 15 |
| 7 | 3 | 1 | 14 | 13 |
| 8 | 2 | 3 | 13 | 14 |
| 9 | 3 | 2 | 13 | 15 |
| 10 | 1 | 4 | 16 | 16 |

Note: True $R_0$ in all geometries is either 9 or 4 nmi. True $R_n$ is between 4 and 14 nmi.

1. It iterates from a close group of starting points instead of from starting points along a line.
2. It always performs 15 iterations, but prints "SOLUTION" after the iteration that first reaches an acceptable solution (within 0.06 nautical miles in both variables - see figure 3-1).
3. It saves the results in a file.

The computer program CGHL is run 28 times, twice for each geometry. Because each run consists of ten trials, a total of 280 results are stored on disk files. The purpose of these runs is twofold:

1. Determine the number of iterations required to reach an acceptable solution (error less than 300 feet in both variables for a given starting position). The algorithm is given the correct answer which is used only to evaluate the progress of the search.
2. Determine the number of iterations required to reach an acceptable solution in 90, 85 and 80 percent of the trials given that n iterations is the stopping criterion.

Table 3-2 summarizes the number of iterations required to reach an acceptable solution for each of ten starting positions in 14 geometries for both high and low starting positions. The number 16 is a code that an acceptable solution is not reached in fifteen iterations. The average

RUN NUMBER: CONTOURl

TRIAL: 2
RO=: 2
RN=: 2
10:48:49

| ITERATION | RO | RN | DELTA RO | DELTA RN | SSE |
|---|---|---|---|---|---|
| 1 | 2.838 | 5.700 | 6.162 | 2.026 | 16.10973 |
| 2 | 8.258 | 7.487 | 0.742 | 0.238 | 0.07904 |
| 3 | 8.975 | 7.715 | 0.025 | 0.010 | 0.00033 |
| SOLUTION | | | | | |
| 4 | 8.975 | 7.717 | 0.025 | 0.008 | 0.00009 |
| 5 | 8.975 | 7.717 | 0.025 | 0.008 | 0.00008 |
| 6 | 8.976 | 7.717 | 0.024 | 0.008 | 0.00008 |
| 7 | 8.977 | 7.718 | 0.023 | 0.008 | 0.00007 |
| 8 | 8.978 | 7.718 | 0.022 | 0.007 | 0.00007 |
| 9 | 8.979 | 7.718 | 0.021 | 0.007 | 0.00006 |
| 10 | 8.979 | 7.718 | 0.021 | 0.007 | 0.00006 |
| 11 | 8.983 | 7.719 | 0.017 | 0.006 | 0.00005 |
| 12 | 8.985 | 7.721 | 0.015 | 0.005 | 0.00003 |
| 13 | 8.985 | 7.721 | 0.015 | 0.004 | 0.00003 |
| 14 | 8.986 | 7.721 | 0.014 | 0.004 | 0.00003 |
| 15 | 8.986 | 7.721 | 0.014 | 0.004 | 0.00003 |

FUNCTIONAL EVALUATIONS: 521

11:01:50
TOTAL FUNCTIONAL EVALUATIONS: 521

Figure 3-1. Results of Conjugate Gradient Search for Starting Conditions $R_0=2, R_n=2$ on the First Geometry.

Table 3-2. Number of Iterations Required to Reach a Solution.

| Run Number | Starting Position 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 Low | 2 | 7 | 8 | 5 | 4 | 10 | 2 | 13 | 2 | 6 |
| 1 High | 16 | 15 | 6 | 13 | 11 | 16 | 8 | 16 | 16 | 13 |
| 2 Low | 5 | 2 | 2 | 2 | 9 | 2 | 6 | 3 | 2 | 5 |
| 2 High | 10 | 3 | 16 | 15 | 4 | 14 | 9 | 11 | 9 | 16 |
| 3 Low | 4 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 7 |
| 3 High | 8 | 3 | 4 | 10 | 16 | 16 | 12 | 13 | 6 | 16 |
| 4 Low | 5 | 2 | 2 | 4 | 2 | 4 | 4 | 4 | 7 | 2 |
| 4 High | 11 | 9 | 6 | 16 | 4 | 5 | 11 | 9 | 6 | 2 |
| 5 Low | 11 | 11 | 6 | 7 | 16 | 9 | 16 | 3 | 13 | 11 |
| 5 High | 8 | 15 | 11 | 16 | 16 | 13 | 16 | 16 | 16 | 13 |
| 6 Low | 4 | 9 | 16 | 12 | 11 | 5 | 7 | 12 | 6 | 3 |
| 6 High | 9 | 4 | 14 | 14 | 2 | 13 | 5 | 16 | 13 | 13 |
| 7 Low | 7 | 4 | 4 | 7 | 4 | 6 | 5 | 6 | 5 | 5 |
| 7 High | 3 | 4 | 4 | 3 | 3 | 7 | 3 | 3 | 5 | 5 |
| 8 Low | 6 | 2 | 2 | 7 | 5 | 6 | 2 | 3 | 2 | 6 |
| 8 High | 4 | 5 | 2 | 4 | 4 | 5 | 2 | 5 | 15 | 8 |
| 9 Low | 5 | 5 | 5 | 7 | 6 | 5 | 12 | 12 | 6 | 10 |
| 9 High | 3 | 5 | 5 | 3 | 10 | 3 | 2 | 2 | 8 | 3 |
| 10 Low | 2 | 2 | 3 | 2 | 5 | 2 | 3 | 2 | 2 | 2 |
| 10 High | 2 | 2 | 2 | 4 | 2 | 3 | 3 | 2 | 2 | 2 |
| 11 Low | 3 | 3 | 3 | 3 | 2 | 3 | 1 | 3 | 2 | 3 |
| 11 High | 3 | 2 | 2 | 2 | 3 | 2 | 2 | 2 | 2 | 3 |
| 12 Low | 7 | 5 | 4 | 6 | 4 | 9 | 8 | 5 | 8 | 10 |
| 12 High | 8 | 8 | 3 | 6 | 12 | 3 | 16 | 9 | 3 | 6 |
| 13 Low | 2 | 2 | 2 | 3 | 2 | 3 | 1 | 2 | 2 | 3 |
| 13 High | 5 | 4 | 4 | 4 | 4 | 5 | 4 | 4 | 4 | 3 |
| 14 Low | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 1 | 6 |
| 14 High | 5 | 3 | 3 | 2 | 2 | 3 | 6 | 2 | 3 | 7 |

number of iterations required is six with a standard deviation of 4.4.

A modification is made to the algorithm in an attempt to reduce the time required to reach an acceptable solution in the fifth geometry. The algorithm is modified to use two conjugate gradient searches instead of only one before reverting to the gradient. Because this modification substantially changes the search (using three golden section searches per iteration versus two for the baseline), comparisons must be made in terms of functional evaluations instead of iterations. With this modification, the algorithm reaches an acceptable solution using 3371 functional evaluations versus 5890 for the baseline case. In an additional modification, three conjugate gradient searches are used before reverting to the gradient. 3253 functional evaluations are used in this case, a minor improvement over the first modification.

The slowness of the search on the fifth geometry in the baseline case is due to very small steps on the acceleration leg. Using multiple conjugate gradients substantially reduces the time required to reach a solution in this geometry. Because a technique that works well on one geometry does not necessarily work on another, it is not possible to generalize about the effectiveness of these modified algorithms on other geometries. The entire analysis would have be to repeated for both new algorithms in order to determine their general applicability.

### 3.3 THE EFFECT OF STARTING POSITION ON SEARCH EFFECTIVENESS

While the choice of starting position is a deciding factor in determining the success of the search algorithm's ability to converge quickly on a solution in an individual trial, it is difficult to identify any starting points that are universally better than others. The results of these 280 conjugate gradient search runs are grouped in a number of ways (see table 3-3). Standard Deviation pertains to the total number of trials while Standard Deviation of the Means pertains to the average of each group.

Table 3-3. Summary Results of Single Conjugate Gradient Search.

| Grouping | Number of Groups | Total Number Trials | Average Number of Iterations | Standard Deviation | Standard Deviation of Means |
|---|---|---|---|---|---|
| All Runs | 28 | 280 | 5.98 | 4.38 | 3.46 |
| Low Search | 14 | 140 | 4.82 | 3.36 | 2.53 |
| High Search | 14 | 140 | 7.14 | 1.77 | 3.96 |
| All Columns (Low Only) | 10 | 140 | 4.82 | 3.36 | 0.47 |
| All Columns (High Only) | 10 | 140 | 7.14 | 1.77 | 0.82 |

Table 3-3 shows a significant variation in the number of iterations required to reach a solution. The summary statistics correlate with the results of the individual runs: solutions are obtained in as few as one iteration or in as many as 15 or more iterations from many of the starting points. Table 3-3 does show that low starting points provide a solution on the average in two less iterations than the high search. This difference is significant at the three sigma level. The very low standard deviation (0.47) on the average of the ten low searches suggests that no one low point is significantly better than any other. All averages are within two standard deviations of the average of the means. The same observation is made about the high searches. On the other hand, the standard deviation on the average number of iterations for a particular starting point is 4.39. Considerable variability is evident in the ability of a search to converge from a particular starting point. In some instances, (for example, geometry 9), the high search is often much better than the low search. Even though low starting points are more efficient statistically, they are not necessarily more efficient for a particular geometry.

## 3.4 NUMBER OF ITERATIONS REQUIRED TO REACH A SOLUTION

If the criterion for stopping the conjugate gradient search is n iterations, it is necessary to determine the probability that an acceptable solution is reached in n iterations. Table 3-4 presents the percent of trials in

Table 3-4. Percent of Trials in which Solution is Reached in n Iterations.

| Number Iterations | Count | Cumulative Count | Cumulative Percentage |
|---|---|---|---|
| 1 | 3 | 3 | .011 |
| 2 | 70 | 73 | .261 |
| 3 | 40 | 113 | .404 |
| 4 | 29 | 142 | .507 |
| 5 | 28 | 170 | .607 |
| 6 | 20 | 190 | .679 |
| 7 | 12 | 202 | .721 |
| 8 | 10 | 212 | .757 |
| 9 | 10 | 222 | .793 |
| 10 | 8 | 230 | .821 |
| 11 | 7 | 237 | .846 |
| 12 | 6 | 243 | .868 |
| 13 | 10 | 253 | .904 |
| 14 | 3 | 256 | .914 |
| 15 | 4 | 260 | .929 |
| 16+ | 20 | 280 | 1.000 |

which an acceptable solution is reached in n iterations. Thirteen iterations are required if the algorithm is to reach an acceptable solution 90 percent of the time, 12 iterations 85 percent of the time and 10 iterations 80 percent of the time.

## 3.5 OBSERVATIONS ON SINGLE CONJUGATE GRADIENT SEARCH

Although the conjugate gradient search successfully solves the optimization problem at issue, it often requires a large number of iterations, particularly if n iterations is the stopping criterion and a high probability of success is required. If for example, a 90 percent probability of success is required, then 13 iterations are needed. This is more than twice the average number of iterations required to reach the optimum and it represents a significant loss of time in obtaining a solution.

Four decided weaknesses of the single conjugate gradient search are identified:

1. It does not differentiate the acceptable from the unacceptable solutions.
2. It provides no direct information on how far from the optimum a solution may be after n iterations.
3. It requires a large number of iterations on runs in which an adequate solution is actually obtained with a few iterations because of the need to choose a sufficiently large n to guarantee success

in an acceptable percent of the cases.

4. It provides error measurements in terms of sum of squares of error. This level of information may not be meaningful to the user.

Section 4 discusses a solution to these problems.

# SECTION 4

# SIMULTANEOUS CONJUGATE GRADIENT SEARCH

## 4.1 OVERVIEW

The primary purpose of this analysis is to show that the bearings only target motion analysis problem can be solved by using two conjugate gradient searches, one beginning in the neighborhood of 0,0; the other at a point decidedly beyond the optimum in both variables. As is mentioned in sections 2 and 3, these searches, on the contours defined by the geometries, consistently converge on the optimum from opposite directions (see figure 4-1). Consequently this technique provides a measure in units of feet of the maximum error at every iteration. Either search is never any further from the optimum at any iteration than the difference between the two searches.

## 4.2 EXPERIMENTAL DESIGN

### 4.2.1 Simultaneous Search Algorithm

The simultaneous search algorithm can be used on a conventional computer, evaluating an iteration of the first search, an iteration of the second search and then computing the distance between the two searches. The concept of simultaneous search, however, is really intended for a computer with co-processing capability. With co-processing,

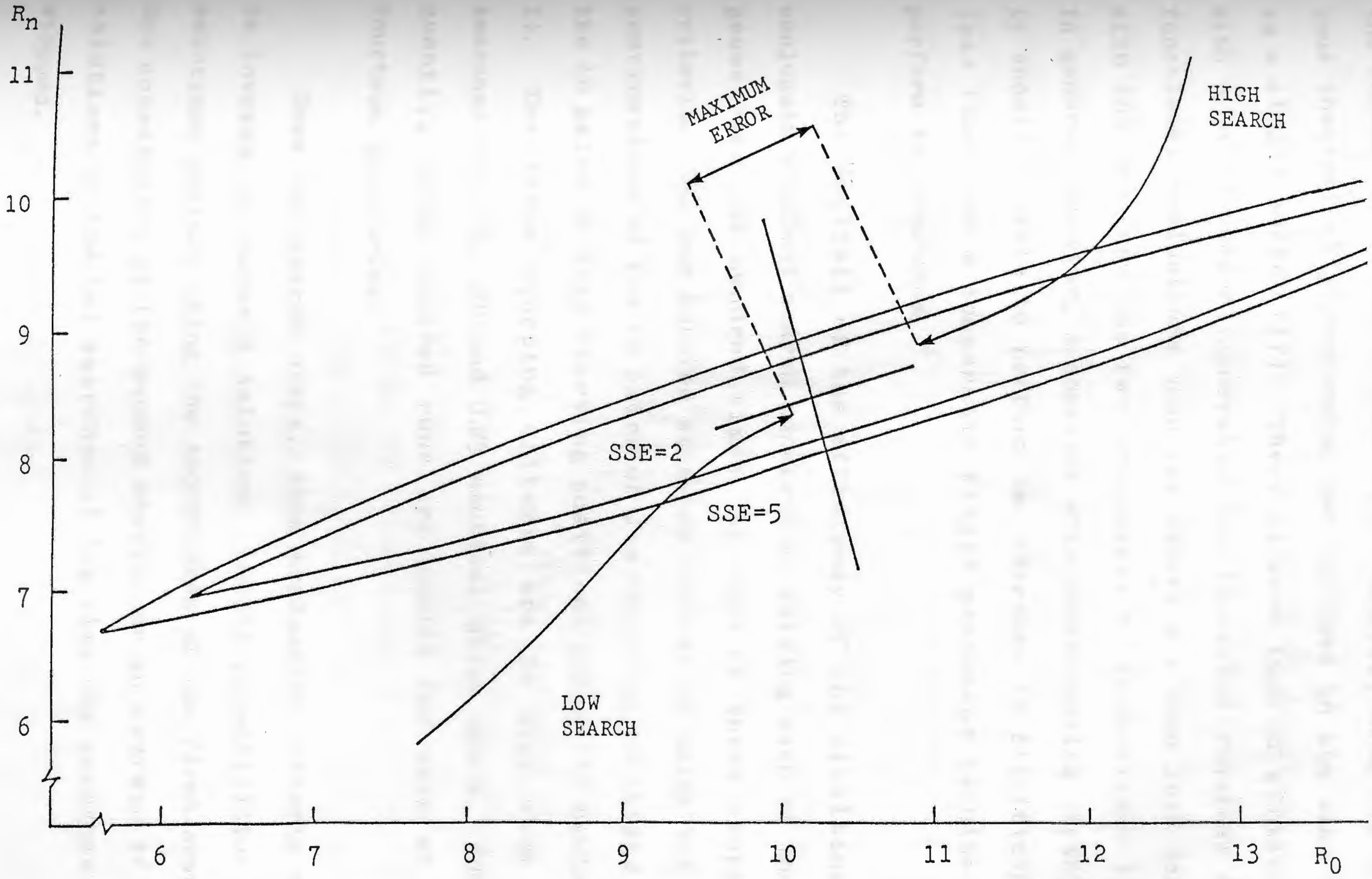



Figure 4-1. Simultaneous Conjugate Gradient Search.

the second search can be done simultaneously with the first, thus theoretically performing two searches in the same time as a single search (12). There is some loss of efficiency with a co-processor (generally one iteration requires more functional evaluations than the other) and some loss occurs with the need for the two processors to communicate (13). In general, however, a computer with co-processing capability should be able to perform two searches in significantly less time than a comparable single processor machine can perform two searches.

The analysis of the efficiency of the simultaneous conjugate gradient search consists of solving each of the 14 geometries one hundred times for each of three stopping criteria. The one hundred attempts consist of using the 100 combinations of the 10 pairs of low starting positions and the 10 pairs of high starting positions (refer to table 3-1). The three stopping criteria are to stop when the searches are .15, .20 and 0.25 nautical miles apart. Consequently, three hundred runs are needed for each of the fourteen geometries.

Once the search stops, some evaluation criteria must be invoked to choose a solution. Likely possibilities for solutions include using the coordinates of the first search, the coordinates of the second search, or an average of the solutions of the two searches at the time the searches are stopped.

4.2.2 Sets of Runs

Seven sets of runs are made for each geometry to test seven evaluation criteria that can be used to choose a solution. A set consists of three groups of 100 runs, one group for each stopping criterion. Figure 4-2 summarizes the computer runs used to test the simultaneous conjugate gradient search. The seven sets are differentiated by the evaluation criteria used to choose a solution when the stopping criteria are invoked. The seven sets are:

1. Choose the solution that has the lowest sum of squares of error when the stopping criteria are invoked.
2. Choose the average of the two solutions when the stopping criteria are invoked.
3. Choose the solution that has the lowest sum of squares of error (set 1) after a minimum of 5 iterations. A minimum of 5 iterations are required for all runs in sets 3 to 7.
4. Choose the average of the two solutions when the stopping criteria are invoked (set 2) after a minimum of 5 iterations.
5. Choose the solution that has the least change in sum of squares of error when the stopping criteria are invoked. Use the difference between the last three iterations to determine the most stable sum of squares of error.

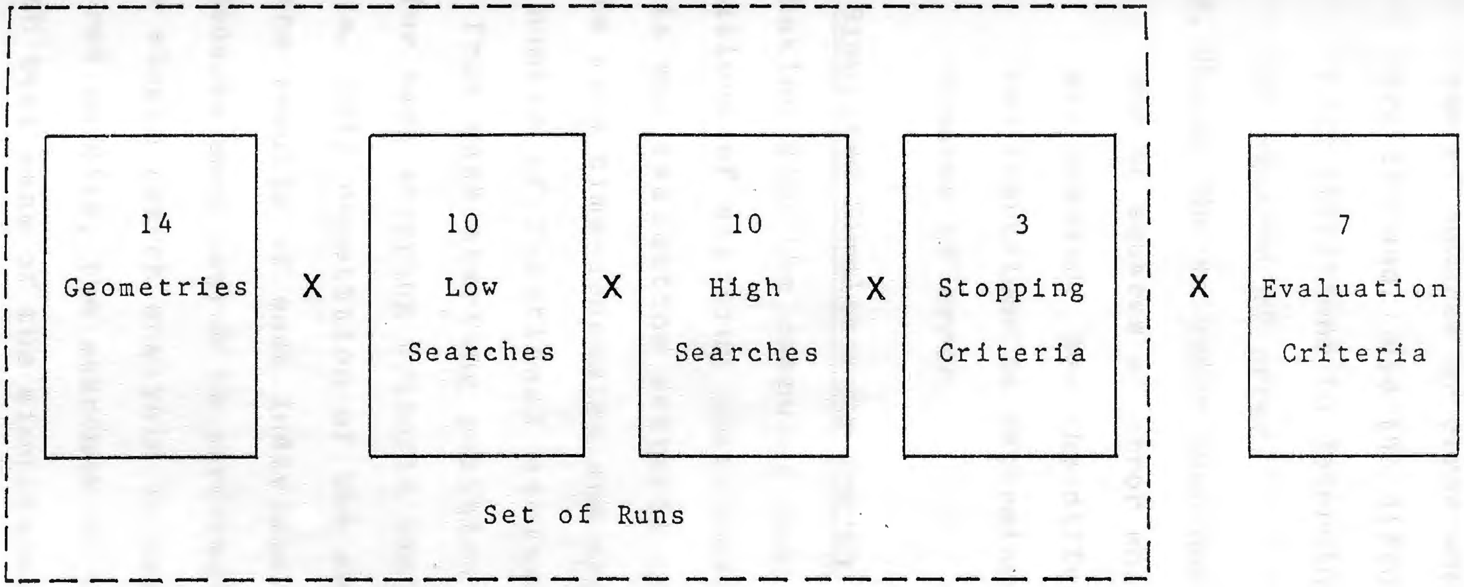



Figure 4-2. Summary of Simultaneous Search Runs.

6. Choose the solution that has the least change in sum of squares of error when the stopping criteria are invoked. Use the difference between the last four iterations to determine the most stable sum of squares of error.
7. Choose the solution that has the least change in sum of squares of error when the stopping criteria are invoked. Use the difference between the last two iterations to determine the most stable sum of squares of error.

### 4.2.3 Simulated Simultaneous Search

Making all the computer runs required for all the combinations of starting positions, geometries, stopping criteria and evaluation criteria needed for the analysis would be very time-consuming and expensive because of the large number of functional evaluations required. Each search from each starting position would be repeated 10 times for each stopping criteria and each set of evaluation criteria. This repetition of the searches is unnecessary since the results of each individual search is deterministic. Because each search is performed to fifteen iterations for the single search analysis in section 3 and the results are stored on disk, the searches do not have to be repeated for each test case of the simultaneous conjugate gradient search. Only a simple statistical program is required for this part of the analysis. The program simulates a simultaneous conjugate gradient search by reading the results of each iteration from a disk file instead of actually performing the calculations. The simulated simultaneous search performs six functions for each test case:

1. Read the results of a low search from a disk file.
2. Read the results of a high search from a disk file.
3. Determine the number of iterations required before the searches are .15, .20 or .25 nautical miles apart.
4. Choose an estimate of the optimum (low search, high search, average of two searches) using one of seven sets of evaluation criteria.
5. Determine if the solution is acceptable (within 300 feet of the true optimum in both variables).
6. Collect statistics.
   a. Good solution.
   b. Stopping criteria not invoked, but solution available on one of the two searches.
   c. Stopped iterating too soon.
   d. No solution possible by either search in 15 iterations.

4.3 STOPPING AND EVALUATION CRITERIA

As is discussed in section 3, the conjugate gradient search is an effective tool to solve the bearings only target motion analysis problem. Solving the problem in real time applications, however, requires invoking some stopping criteria. Because of the canoe shape of the curve, two solutions with the same sum of squares of error are not necessarily equally close to the optimum. The simultaneous conjugate gradient search provides some absolute information on the success of the search in that it provides at every iteration a maximum error, namely the distance between both searches. This type of absolute information on the success of the search is not available from a single search.

One of the major difficulties in evaluating the success of the simultaneous conjugate gradient search is the lack of information on which search is closer to the optimum. In general, one search is likely to be closer to the optimum than the other as is discussed in section 2. Although the distance between the two searches provides a limit on the maximum error, it does not identify which search is farther from the optimum.

4.3.1 <u>Stopping Criteria</u>

Clearly it is possible to continue the two conjugate gradient searches until they are within 300 yards in both variables. Because searches that do not converge quickly tend to converge after quite different numbers of iterations, the approach would be very inefficient. In many cases, one search would be at the optimum many iterations before the other search is within the stopping criterion. Because the purpose of the simultaneous conjugate gradient search is to increase the probability of stopping the search at an acceptable solution while at the same time reducing the number of iterations, this stopping criterion would be too strict. Three stopping criteria are selected for this analysis:

1. Stop when searches are within .15 nautical miles (900 feet).
2. Stop when searches are within .20 nautical miles (1200 feet).
3. Stop when searches are within .25 nautical miles (1500 feet).

Despite the fact that these distances are significantly greater than .06 nautical miles, the criterion used in single search, in both variables (a maximum of .085 nautical miles), they incorporate the statistical probability that one search tends to converge before the other and thus one search may reach an acceptable solution while the other is 1200 feet away. There is some probability of choosing the search with the poorer solution. This analysis is testing the statistical probability of obtaining an acceptable solution given that the search terminates once the two searches converge to a particular range. The three values are chosen as baseline parameters for the anlaysis. Other values can be investigated if necessary.

### 4.3.2 Evaluation Criteria

4.3.2.1 Lowest Sum of Squares of Error. Although it is established that the solution with the lowest sum of squares of error is not necessarily the best solution, the first test of the simultaneous search is to choose the solution that has the lower sum of squares of error when the stopping criteria are invoked. Using the minimum sum of squares of error as an evaluation criterion is important because it is an obvious criterion to test. Even though it does not always provide the best solution, it is important to establish how well minimum sum of squares of error functions as an evalution criterion.

4.3.2.2 Average of Two Solutions. Because the two searches are converging from opposite directions, some point between the two searches is actually closer to the optimum than either search. The average values of the two variables from both searches is used instead of either solution in the second set of runs. One search, however, usually converges more quickly than the other. It is likely in many cases that one of the solutions is actually closer to the optimum than the average of the two solutions. Because the probability that both searches are equidistant from the optimum is low, especially for searches that converge after a large number of iterations, the average distance may not be the best choice to use as an estimate of the optimum. It is worthwhile, however, to test a value between the two searches and the average search is an obvious choice.

4.3.2.3 Minimum of Five Iterations. 61 percent of the test runs reaches an acceptable solution in five iterations or less. If all runs were independent, then there would be an 85 percent probability that an acceptable solution would be reached by at least one of the two searches in five iterations or less. Although there is some definite dependence between the high and low search, there is also a fair amount of independence both between high and low search and within high and low search. For these reasons, sets three and four explore the advantage of calculating at least five iterations in all cases. Set 3 repeats set 1 and set 4 repeats set 2 under the additional stopping criterion that a minimum of five iterations be completed.

4.3.2.4 Most Stable Sum of Squares of Error. As the conjugate gradient search iterates toward the optimum, the change in the sum of squares of error between two iterations becomes consistently smaller. Sets 5, 6 and 7 test the change in the sum of squares of error as an evaluation tool in choosing a solution. The coordinates of the search that have the smaller change in sum of squares of error when the stopping criteria are invoked are chosen as the estimate of the optimum. Set five tests the difference between the last three iterations (n and n - 2). Set six tests the difference between the last four iterations and set seven the last two iterations.

## 4.4 RESULTS OF SIMULTANEOUS CONJUGATE GRADIENT SEARCH

### 4.4.1 Explanation of Output

A statistical computer program called STAT (see appendix d), which simulates the simultaneous conjugate gradient search by reading the result of each iteration from a disk file, is used to quantify the performance of the simultaneous conjugate gradient search. Sample output of a set of runs is presented in figure 4-3. The output is explained by discussing three specific trials. In the first trial, which uses the first set of low coordinates (0,0) and the first set of high coordinates (15,15) as starting points, the two searches converge to 0.15 nautical miles after 13 iterations. The evaluation criterion is lowest sum of squares of error. Since the low search (RN1,R01) has the lowest sum of squares of error and both variables are within 0.06 nautical miles of the optimum (see DELTA RN1,DELTA R01), the values of R01 and RN1 after 13 iterations are an acceptable solution to the problem. In the second test which uses the first set of low coordinates (2,2) and the second set of high coordinates (14,14), the two searches do not converge to 0.15 nautical miles in 15 iterations. Although the low search does have an acceptable solution, the solution is missed because the searches do not converge to the stopping criterion. The program prints "SOLUTION ON LOW" to indicate that a good solution is missed. Trial 55 is a case in which the search stops too early (see figure 4-3c). The two searches are within 0.15 nautical miles but the high search has the lower sum of squares of error. Because $R_0$ on the second search is more than 0.06 nautical miles from the function optimum (DELTA R02 = 0.065), the program prints "NO SOLUTION" and counts the trial as a BADSTOP.

| | |
|---|---|
| TEST DISTANCE 1 | .15 |
| CONTOUR | GEOMETRY #1 |
| FIRST FILE | B:OUTPUT1L.ASC |
| SECOND FILE | B:OUTPUT1H.ASC |

| TRIAL | NUMBER ITERATIONS | DELTA RN1 | DELTA R01 | DELTA RN2 | DELTA R02 | |
|---|---|---|---|---|---|---|
| 1 | 13 | 0.003 | 0.011 | 0.032 | 0.119 | |
| 2 | 15 | 0.003 | 0.010 | 0.051 | 0.148 | NO SOLUTION |
| | SOLUTION ON LOW | | | | | |
| 3 | 6 | 0.004 | 0.015 | 0.011 | 0.014 | |
| 4 | 11 | 0.004 | 0.012 | 0.039 | 0.115 | |
| 5 | 7 | 0.005 | 0.013 | 0.023 | 0.119 | |
| 6 | 15 | 0.003 | 0.010 | 0.102 | 0.328 | NO SOLUTION |
| | SOLUTION ON LOW | | | | | |
| 7 | 8 | 0.004 | 0.013 | 0.002 | 0.007 | |
| 8 | 15 | 0.003 | 0.010 | 0.044 | 0.160 | NO SOLUTION |
| | SOLUTION ON LOW | | | | | |
| 9 | 15 | 0.003 | 0.010 | 0.026 | 0.094 | |
| 10 | 13 | 0.003 | 0.011 | 0.000 | 0.004 | |
| 11 | 13 | 0.001 | 0.003 | 0.032 | 0.119 | |
| 12 | 15 | 0.001 | 0.003 | 0.051 | 0.148 | NO SOLUTION |
| | SOLUTION ON LOW | | | | | |
| 13 | 7 | 0.020 | 0.038 | 0.005 | 0.016 | |
| 14 | 11 | 0.001 | 0.003 | 0.039 | 0.115 | |
| 15 | 9 | 0.012 | 0.038 | 0.019 | 0.066 | |
| 16 | 15 | 0.001 | 0.003 | 0.102 | 0.328 | NO SOLUTION |
| | SOLUTION ON LOW | | | | | |
| 17 | 8 | 0.013 | 0.040 | 0.002 | 0.007 | |
| 18 | 15 | 0.001 | 0.003 | 0.044 | 0.160 | NO SOLUTION |
| | SOLUTION ON LOW | | | | | |
| 19 | 15 | 0.001 | 0.003 | 0.026 | 0.094 | |
| 20 | 13 | 0.001 | 0.003 | 0.000 | 0.004 | |
| 21 | 13 | 0.000 | 0.002 | 0.032 | 0.119 | |
| 22 | 15 | 0.001 | 0.002 | 0.051 | 0.148 | NO SOLUTION |
| | SOLUTION ON LOW | | | | | |
| 23 | 8 | 0.001 | 0.002 | 0.005 | 0.014 | |
| 24 | 11 | 0.000 | 0.002 | 0.039 | 0.115 | |
| 25 | 8 | 0.001 | 0.002 | 0.036 | 0.114 | |

Figure 4-3a. Sample Output -- Simultaneous Conjugate Gradient Search.

| | |
|---|---|
| TEST DISTANCE 1 | .15 |
| CONTOUR | GEOMETRY #1 |
| FIRST FILE | B:OUTPUT1L.ASC |
| SECOND FILE | B:OUTPUT1H.ASC |

| TRIAL | NUMBER ITERATIONS | DELTA RN1 | DELTA R01 | DELTA RN2 | DELTA R02 | |
|---|---|---|---|---|---|---|
| 26 | 15 | 0.001 | 0.002 | 0.102 | 0.328 | NO SOLUTION |
| | SOLUTION ON LOW | | | | | |
| 27 | 8 | 0.001 | 0.002 | 0.002 | 0.007 | |
| 28 | 15 | 0.001 | 0.002 | 0.044 | 0.160 | NO SOLUTION |
| | SOLUTION ON LOW | | | | | |
| 29 | 15 | 0.001 | 0.002 | 0.026 | 0.094 | |
| 30 | 13 | 0.000 | 0.002 | 0.000 | 0.004 | |
| 31 | 14 | 0.009 | 0.026 | 0.037 | 0.116 | |
| 32 | 15 | 0.008 | 0.025 | 0.051 | 0.148 | NO SOLUTION |
| | SOLUTION ON LOW | | | | | |
| 33 | 6 | 0.019 | 0.058 | 0.011 | 0.014 | |
| 34 | 12 | 0.009 | 0.027 | 0.037 | 0.115 | |
| 35 | 9 | 0.009 | 0.030 | 0.019 | 0.066 | |
| 36 | 15 | 0.008 | 0.025 | 0.102 | 0.328 | NO SOLUTION |
| | SOLUTION ON LOW | | | | | |
| 37 | 8 | 0.010 | 0.032 | 0.002 | 0.007 | |
| 38 | 15 | 0.008 | 0.025 | 0.044 | 0.160 | NO SOLUTION |
| | SOLUTION ON LOW | | | | | |
| 39 | 15 | 0.008 | 0.025 | 0.026 | 0.094 | |
| 40 | 13 | 0.009 | 0.027 | 0.000 | 0.004 | |
| 41 | 13 | 0.002 | 0.006 | 0.032 | 0.119 | |
| 42 | 15 | 0.002 | 0.006 | 0.051 | 0.148 | NO SOLUTION |
| | SOLUTION ON LOW | | | | | |
| 43 | 6 | 0.002 | 0.007 | 0.011 | 0.014 | |
| 44 | 11 | 0.002 | 0.006 | 0.039 | 0.115 | |
| 45 | 7 | 0.002 | 0.007 | 0.023 | 0.119 | |
| 46 | 15 | 0.002 | 0.006 | 0.102 | 0.328 | NO SOLUTION |
| | SOLUTION ON LOW | | | | | |
| 47 | 7 | 0.002 | 0.007 | 0.048 | 0.149 | |
| 48 | 15 | 0.002 | 0.006 | 0.044 | 0.160 | NO SOLUTION |
| | SOLUTION ON LOW | | | | | |
| 49 | 15 | 0.002 | 0.006 | 0.026 | 0.094 | |
| 50 | 13 | 0.002 | 0.006 | 0.000 | 0.004 | |

Figure 4-3b. Sample Output -- Simultaneous Conjugate Gradient Search.

| | |
|---|---|
| TEST DISTANCE 1 | .15 |
| CONTOUR | GEOMETRY #1 |
| FIRST FILE | B:OUTPUT1L.ASC |
| SECOND FILE | B:OUTPUT1H.ASC |

| TRIAL | NUMBER ITERATIONS | DELTA RN1 | DELTA R01 | DELTA RN2 | DELTA R02 | |
|---|---|---|---|---|---|---|
| 51 | 15 | 0.009 | 0.026 | 0.020 | 0.055 | |
| 52 | 15 | 0.009 | 0.026 | 0.051 | 0.148 | NO SOLUTION |
| | SOLUTION ON LOW | | | | | |
| 53 | 10 | 0.014 | 0.035 | 0.004 | 0.013 | |
| 54 | 13 | 0.010 | 0.029 | 0.001 | 0.000 | |
| 55 | 10 | 0.014 | 0.035 | 0.021 | 0.065 | |
| | NO SOLUTION | | | | | |
| 56 | 15 | 0.009 | 0.026 | 0.102 | 0.328 | NO SOLUTION |
| | SOLUTION ON LOW | | | | | |
| 57 | 10 | 0.014 | 0.035 | 0.002 | 0.006 | |
| 58 | 15 | 0.009 | 0.026 | 0.044 | 0.160 | NO SOLUTION |
| | SOLUTION ON LOW | | | | | |
| 59 | 15 | 0.009 | 0.026 | 0.026 | 0.094 | |
| 60 | 13 | 0.010 | 0.029 | 0.000 | 0.004 | |
| 61 | 13 | 0.004 | 0.013 | 0.032 | 0.119 | |
| 62 | 15 | 0.004 | 0.012 | 0.051 | 0.148 | |
| 63 | 6 | 0.004 | 0.015 | 0.011 | 0.014 | |
| 64 | 9 | 0.005 | 0.014 | 0.054 | 0.154 | |
| 65 | 7 | 0.004 | 0.015 | 0.023 | 0.119 | |
| 66 | 15 | 0.004 | 0.012 | 0.102 | 0.328 | NO SOLUTION |
| | SOLUTION ON LOW | | | | | |
| 67 | 6 | 0.004 | 0.015 | 0.063 | 0.145 | |
| 68 | 15 | 0.004 | 0.012 | 0.044 | 0.160 | NO SOLUTION |
| | SOLUTION ON LOW | | | | | |
| 69 | 15 | 0.004 | 0.012 | 0.026 | 0.094 | |
| 70 | 13 | 0.004 | 0.013 | 0.000 | 0.004 | |
| 71 | 13 | 0.006 | 0.025 | 0.032 | 0.119 | |
| 72 | 15 | 0.008 | 0.023 | 0.051 | 0.148 | NO SOLUTION |
| | SOLUTION ON LOW | | | | | |
| 73 | 13 | 0.006 | 0.025 | 0.004 | 0.012 | |
| 74 | 13 | 0.006 | 0.025 | 0.001 | 0.000 | |
| 75 | 13 | 0.006 | 0.025 | 0.014 | 0.041 | |

Figure 4-3c. Sample Output -- Simultaneous Conjugate Gradient Search.

| | | |
|---|---|---|
| TEST DISTANCE 1 | .15 | |
| CONTOUR | GEOMETRY #1 | |
| FIRST FILE | B:OUTPUT1L.ASC | |
| SECOND FILE | B:OUTPUT1H.ASC | |

| TRIAL | NUMBER ITERATIONS | DELTA RN1 | DELTA RO1 | DELTA RN2 | DELTA RO2 | |
|---|---|---|---|---|---|---|
| 76 | 15 | 0.008 | 0.023 | 0.102 | 0.328 | NO SOLUTION |
| | SOLUTION ON LOW | | | | | |
| 77 | 12 | 0.044 | 0.137 | 0.002 | 0.006 | |
| 78 | 15 | 0.008 | 0.023 | 0.044 | 0.160 | NO SOLUTION |
| | SOLUTION ON LOW | | | | | |
| 79 | 15 | 0.008 | 0.023 | 0.026 | 0.094 | |
| 80 | 13 | 0.006 | 0.025 | 0.000 | 0.004 | |
| 81 | 13 | 0.001 | 0.005 | 0.032 | 0.119 | |
| 82 | 15 | 0.001 | 0.005 | 0.051 | 0.148 | NO SOLUTION |
| | SOLUTION ON LOW | | | | | |
| 83 | 6 | 0.002 | 0.006 | 0.011 | 0.014 | |
| 84 | 11 | 0.001 | 0.005 | 0.039 | 0.115 | |
| 85 | 7 | 0.002 | 0.006 | 0.023 | 0.119 | |
| 86 | 15 | 0.001 | 0.005 | 0.102 | 0.328 | NO SOLUTION |
| | SOLUTION ON LOW | | | | | |
| 87 | 8 | 0.002 | 0.005 | 0.002 | 0.007 | |
| 88 | 15 | 0.001 | 0.005 | 0.044 | 0.160 | NO SOLUTION |
| | SOLUTION ON LOW | | | | | |
| 89 | 15 | 0.001 | 0.005 | 0.026 | 0.094 | |
| 90 | 13 | 0.001 | 0.005 | 0.000 | 0.004 | |
| 91 | 13 | 0.005 | 0.017 | 0.032 | 0.119 | |
| 92 | 15 | 0.006 | 0.016 | 0.051 | 0.148 | NO SOLUTION |
| | SOLUTION ON LOW | | | | | |
| 93 | 6 | 0.007 | 0.035 | 0.011 | 0.014 | |
| 94 | 11 | 0.006 | 0.019 | 0.039 | 0.115 | |
| 95 | 9 | 0.009 | 0.029 | 0.019 | 0.066 | |
| 96 | 15 | 0.006 | 0.016 | 0.102 | 0.328 | NO SOLUTION |
| | SOLUTION ON LOW | | | | | |
| 97 | 8 | 0.011 | 0.033 | 0.002 | 0.007 | |
| 98 | 15 | 0.006 | 0.016 | 0.044 | 0.160 | NO SOLUTION |
| | SOLUTION ON LOW | | | | | |
| 99 | 15 | 0.006 | 0.016 | 0.026 | 0.094 | |
| 100 | 13 | 0.005 | 0.017 | 0.000 | 0.004 | |

Figure 4-3d. Sample Output -- Simultaneous Conjugate Gradient Search.

Summary output of the 100 trials is found after the last trial on figure 4-4. The distribution of the number of iterations is shown. 16 iterations is a code for trials that do not close to the stopping criterion. The 29 trials listed at 16 iterations are the 29 trials that do not converge to 0.15 nautical miles. In all 29 trials, however, one of the two searches does have an acceptable solution (MISSED = 29).

STAT is run for all 14 geometries and the results stored on disk. A program called SUMMARY (see appendix e) is used to print results of a set of 14 STAT runs on one table (see figure 4-5). Figure 4-5 summarizes 70 pages of output from STAT. Each numbered row of figure 4-5 contains the summary results of 100 runs (the 100 combinations of 10 low and 10 high searches) on a particular geometry. The next to the last line is the mean of the fourteen statistics in each column and the last row is the standard deviation of the fourteen sample means.

The first column of the output is the number of the geometry. The second column is the number of trials (out of 100) in which a successful solution is found under the stopping criteria and evaluation criteria for that run. The third column is the average number of iterations required to reach a solution. The last three columns summarize the three reasons that a correct solution is not identified:

| Number Iterations | Number Trials |
|---|---|
| 1 | 0 |
| 2 | 0 |
| 3 | 0 |
| 4 | 0 |
| 5 | 0 |
| 6 | 7 |
| 7 | 6 |
| 8 | 8 |
| 9 | 4 |
| 10 | 3 |
| 11 | 6 |
| 12 | 2 |
| 13 | 22 |
| 14 | 1 |
| 15 | 12 |
| 16+ | 29 |

| | |
|---|---|
| Number of Successes: | 70 |
| Average # Iterations: | 12.19 |
| Missed Solutions: | 29 |
| Bad Stop: | 1 |
| No Solution: | 0 |

Figure 4-4. Summary Output -- Simultaneous Conjugate Gradient Search.

SET #1: R=.15 09-05-1985

| # | SUCCESS | ITERATIONS | MISSED | BADSTOP | NO SOLUTION |
|---|---|---|---|---|---|
| 1 | 70.00 | 12.19 | 29 | 1 | 0 |
| 2 | 74.00 | 10.10 | 20 | 6 | 0 |
| 3 | 78.00 | 9.89 | 22 | 0 | 0 |
| 4 | 100.00 | 7.45 | 0 | 0 | 0 |
| 5 | 36.00 | 14.15 | 50 | 0 | 14 |
| 6 | 79.00 | 10.89 | 17 | 3 | 1 |
| 7 | 95.00 | 4.35 | 0 | 5 | 0 |
| 8 | 93.00 | 3.64 | 0 | 7 | 0 |
| 9 | 96.00 | 5.85 | 0 | 4 | 0 |
| 10 | 87.00 | 2.04 | 0 | 13 | 0 |
| 11 | 97.00 | 2.52 | 0 | 3 | 0 |
| 12 | 93.00 | 8.28 | 6 | 1 | 0 |
| 13 | 100.00 | 4.00 | 0 | 0 | 0 |
| 14 | 94.00 | 2.84 | 0 | 6 | 0 |
| AVG | 85.14 | 7.01 | 10.29 | 3.50 | 1.07 |
| SD | 17.23 | 3.96 | 15.36 | 3.72 | 3.73 |

Figure 4-5. Sample Output -- Summary of Fourteen Geometries.

1. MISSED - fifteen iterations are completed for both searches but the two searches never converge to the stopping criterion. One of the searches does, however, have an acceptable solution. Note that this number is a constant over all seven sets of runs for a given stopping criterion, for example, .15 nautical miles.
2. BADSTOP - the searches converge to the stopping distance before either search reaches an acceptable solution.
3. NONE - Neither search reaches an acceptable solution in 15 iterations and the searches do not converge to the stopping criterion.

Columns 2, 4, 5 and 6 sum to 100 trails.

4.4.2 <u>Results of Set 1 - Minimum Sum of Squares of Error</u>

In the first set of runs, the choice of stopping criteria has no effect on the number of successful trials (85, 86 and 85) but does effect the number of iterations (see table 4-1). This is expected since the number of iterations required for the two searches to converge to .25 nautical miles should be less than the number required to reach .15. Because the process is a step function, however, the number of iterations required to bring the searches to .25 may actually bring them to .15 in some trials. Increasing the stopping criterion from .15 to .25 nautical miles affects the number of solutions missed and the number of bad stops. When the stopping criterion is .15 nautical miles,

Table 4-1. Results of Set 1 - Simultaneous Conjugate Gradient Search.

SET #1: R=.15 09-05-1985

| # | SUCCESS | ITERATIONS | MISSED | BADSTOP | NO SOLUTION |
|---|---|---|---|---|---|
| 1 | 70.00 | 12.19 | 29 | 1 | 0 |
| 2 | 74.00 | 10.10 | 20 | 6 | 0 |
| 3 | 78.00 | 9.89 | 22 | 0 | 0 |
| 4 | 100.00 | 7.45 | 0 | 0 | 0 |
| 5 | 36.00 | 14.15 | 50 | 0 | 14 |
| 6 | 79.00 | 10.89 | 17 | 3 | 1 |
| 7 | 95.00 | 4.35 | 0 | 5 | 0 |
| 8 | 93.00 | 3.64 | 0 | 7 | 0 |
| 9 | 96.00 | 5.85 | 0 | 4 | 0 |
| 10 | 87.00 | 2.04 | 0 | 13 | 0 |
| 11 | 97.00 | 2.52 | 0 | 3 | 0 |
| 12 | 93.00 | 8.28 | 6 | 1 | 0 |
| 13 | 100.00 | 4.00 | 0 | 0 | 0 |
| 14 | 94.00 | 2.84 | 0 | 6 | 0 |
| AVG | 85.14 | 7.01 | 10.29 | 3.50 | 1.07 |
| SD | 17.23 | 3.96 | 15.36 | 3.72 | 3.73 |

SET #1: R=.20 09-05-1985

| # | SUCCESS | ITERATIONS | MISSED | BADSTOP | NO SOLUTION |
|---|---|---|---|---|---|
| 1 | 88.00 | 11.54 | 10 | 2 | 0 |
| 2 | 74.00 | 9.51 | 20 | 6 | 0 |
| 3 | 89.00 | 9.40 | 11 | 0 | 0 |
| 4 | 98.00 | 7.15 | 0 | 2 | 0 |
| 5 | 40.00 | 14.00 | 46 | 0 | 14 |
| 6 | 93.00 | 10.11 | 0 | 6 | 1 |
| 7 | 86.00 | 3.84 | 0 | 14 | 0 |
| 8 | 84.00 | 3.27 | 0 | 16 | 0 |
| 9 | 90.00 | 5.15 | 0 | 10 | 0 |
| 10 | 84.00 | 2.00 | 0 | 16 | 0 |
| 11 | 93.00 | 2.43 | 0 | 7 | 0 |
| 12 | 95.00 | 7.62 | 3 | 2 | 0 |
| 13 | 100.00 | 3.90 | 0 | 0 | 0 |
| 14 | 90.00 | 2.68 | 0 | 10 | 0 |
| AVG | 86.00 | 6.61 | 6.43 | 6.50 | 1.07 |
| SD | 14.76 | 3.83 | 12.91 | 5.88 | 3.73 |

SET #1: R=.25 09-05-1985

| # | SUCCESS | ITERATIONS | MISSED | BADSTOP | NO SOLUTION |
|---|---|---|---|---|---|
| 1 | 88.00 | 10.85 | 10 | 2 | 0 |
| 2 | 80.00 | 9.13 | 13 | 7 | 0 |
| 3 | 90.00 | 8.54 | 10 | 0 | 0 |
| 4 | 98.00 | 6.94 | 0 | 2 | 0 |
| 5 | 41.00 | 13.90 | 44 | 1 | 14 |
| 6 | 90.00 | 9.41 | 0 | 9 | 1 |
| 7 | 78.00 | 3.55 | 0 | 22 | 0 |
| 8 | 75.00 | 2.92 | 0 | 25 | 0 |
| 9 | 86.00 | 4.39 | 0 | 14 | 0 |
| 10 | 84.00 | 2.00 | 0 | 16 | 0 |
| 11 | 88.00 | 2.34 | 0 | 12 | 0 |
| 12 | 96.00 | 7.09 | 2 | 2 | 0 |
| 13 | 100.00 | 3.90 | 0 | 0 | 0 |
| 14 | 90.00 | 2.58 | 0 | 10 | 0 |
| AVG | 84.57 | 6.25 | 5.64 | 8.71 | 1.07 |
| SD | 14.46 | 3.71 | 11.97 | 8.25 | 3.73 |

the solution is missed in over 10 percent of the trails. Even though one search converges to an acceptable solution, it is not used since the distance between the two searches is still greater than 0.15 after 15 iterations. On the other hand, as the stopping criterion is increased to .25, the percent of trials in which the search is stopped before either search reaches an acceptable solution increases from 3.5 to 8.7 percent. In this particular set, the average number missed and the average number of bad stops is essentially a constant. This is not the case, however, in individual runs. The number of bad stops remains fairly constant over the first five runs under all three stopping criteria, while the number of missed solutions is essentially unchanged over the last eight geometries. Geometry 5 is difficult to solve under all stopping criteria in 15 iterations or less with the particular form of the conjugate gradient search used for the analysis.

### 4.4.3 Results of Set 2 - Average Distance

Using the average distance between two searches as the estimate of the optimum when the two searches converge to the stopping criterion provides a reduction in performance in virtually all cases (see table 4-2). The only exception is geometry 10 in which performance increases from 84 to 94 percent in all 3 cases (using only 2 iterations in almost all trials). The unique characteristic of geometry 10 is that virtually all searches reach a solution in three iterations or less and are very close after two iterations.

Table 4-2. Results of Set 2 - Simultaneous Conjugate Gradient Search.

SET #2: R=.15 09-05-1985

| # | SUCCESS | ITERATIONS | MISSED | BADSTOP | NO SOLUTION |
|---|---|---|---|---|---|
| 1 | 60.00 | 12.19 | 29 | 11 | -0 |
| 2 | 75.00 | 10.10 | 20 | 5 | 0 |
| 3 | 74.00 | 9.89 | 22 | 4 | 0 |
| 4 | 98.00 | 7.45 | 0 | 2 | 0 |
| 5 | 24.00 | 14.15 | 50 | 12 | 14 |
| 6 | 80.00 | 10.89 | 17 | 2 | 1 |
| 7 | 89.00 | 4.35 | 0 | 11 | 0 |
| 8 | 82.00 | 3.64 | 0 | 18 | 0 |
| 9 | 79.00 | 5.85 | 0 | 21 | 0 |
| 10 | 94.00 | 2.04 | 0 | 6 | 0 |
| 11 | 97.00 | 2.52 | 0 | 3 | 0 |
| 12 | 85.00 | 8.28 | 6 | 9 | 0 |
| 13 | 94.00 | 4.00 | 0 | 6 | 0 |
| 14 | 94.00 | 2.84 | 0 | 6 | 0 |
| AVG | 80.36 | 7.01 | 10.29 | 8.29 | 1.07 |
| SD | 19.43 | 3.96 | 15.36 | 5.80 | 3.73 |

SET #2: R=.20 09-05-1985

| # | SUCCESS | ITERATIONS | MISSED | BADSTOP | NO SOLUTION |
|---|---|---|---|---|---|
| 1 | 44.00 | 11.54 | 10 | 46 | 0 |
| 2 | 52.00 | 9.51 | 20 | 28 | 0 |
| 3 | 67.00 | 9.40 | 11 | 22 | 0 |
| 4 | 84.00 | 7.15 | 0 | 16 | 0 |
| 5 | 20.00 | 14.00 | 46 | 20 | 14 |
| 6 | 68.00 | 10.11 | 0 | 31 | 1 |
| 7 | 77.00 | 3.84 | 0 | 23 | 0 |
| 8 | 72.00 | 3.27 | 0 | 28 | 0 |
| 9 | 57.00 | 5.15 | 0 | 43 | 0 |
| 10 | 94.00 | 2.00 | 0 | 6 | 0 |
| 11 | 89.00 | 2.43 | 0 | 11 | 0 |
| 12 | 77.00 | 7.62 | 3 | 20 | 0 |
| 13 | 88.00 | 3.90 | 0 | 12 | 0 |
| 14 | 84.00 | 2.68 | 0 | 16 | 0 |
| AVG | 69.50 | 6.61 | 6.43 | 23.00 | 1.07 |
| SD | 20.44 | 3.83 | 12.91 | 11.48 | 3.73 |

SET #2: R=.25 09-05-1985

| # | SUCCESS | ITERATIONS | MISSED | BADSTOP | NO SOLUTION |
|---|---|---|---|---|---|
| 1 | 39.00 | 10.85 | 10 | 51 | 0 |
| 2 | 46.00 | 9.13 | 13 | 41 | 0 |
| 3 | 46.00 | 8.54 | 10 | 44 | 0 |
| 4 | 75.00 | 6.94 | 0 | 25 | 0 |
| 5 | 17.00 | 13.90 | 44 | 25 | 14 |
| 6 | 61.00 | 9.41 | 0 | 38 | 1 |
| 7 | 65.00 | 3.55 | 0 | 35 | 0 |
| 8 | 68.00 | 2.92 | 0 | 32 | 0 |
| 9 | 40.00 | 4.39 | 0 | 60 | 0 |
| 10 | 94.00 | 2.00 | 0 | 6 | 0 |
| 11 | 81.00 | 2.34 | 0 | 19 | 0 |
| 12 | 61.00 | 7.09 | 2 | 37 | 0 |
| 13 | 88.00 | 3.90 | 0 | 12 | 0 |
| 14 | 77.00 | 2.58 | 0 | 23 | 0 |
| AVG | 61.29 | 6.25 | 5.64 | 32.00 | 1.07 |
| SD | 21.56 | 3.71 | 11.97 | 14.84 | 3.73 |

Since both iterations are converging simultaneously, averaging the two results is very effective in this geometry.

It is noted that the average number of iterations does not change from set 1 to set 2 for a given geometry and stopping conditions. Likewise the number of missed solutions for a geometry and stopping conditions does not change since this is the number of combinations of starting conditions that do not converge to the stopping distance with 15 iterations even though one of the searches reaches an acceptable solution.

#### 4.4.4 Results of Set 3 - Minimum Sum of Squares of Error and Five Iterations Minimum

The upper limit of performance is constrained by the column labeled "MISSED" as this column is a constant for a given stopping criterion. A number of solutions are not found in set 1 because the search stops before either search has iterated to an acceptable solution (see table 4-1). Adding the additional constraint of a minimum of five iterations to the first set of runs provides a significant increase in performance, virtually eliminating bad stops (see table 4-3). Using a lower stopping criterion would prevent bad stops. The use of a stopping criterion greater than 0.10 nautical miles, however, is based on the assumption that two opposing searches do not regularly converge in the same number of iterations. In general, either both searches converge quickly (five or less iterations each) or the two

Table 4-3. Results of Set 3 - Simultaneous Conjugate Gradient Search.

SET #3: R=.15 09-05-1985

| # | SUCCESS | ITERATIONS | MISSED | BADSTOP | NO SOLUTION |
|---|---|---|---|---|---|
| 1 | 70.00 | 12.19 | 29 | 1 | 0 |
| 2 | 80.00 | 10.38 | 20 | 0 | 0 |
| 3 | 78.00 | 10.15 | 22 | 0 | 0 |
| 4 | 100.00 | 7.82 | 0 | 0 | 0 |
| 5 | 36.00 | 14.15 | 50 | 0 | 14 |
| 6 | 81.00 | 11.00 | 17 | 1 | 1 |
| 7 | 97.00 | 5.05 | 0 | 3 | 0 |
| 8 | 99.00 | 5.04 | 0 | 1 | 0 |
| 9 | 98.00 | 6.33 | 0 | 2 | 0 |
| 10 | 100.00 | 5.00 | 0 | 0 | 0 |
| 11 | 100.00 | 5.00 | 0 | 0 | 0 |
| 12 | 94.00 | 8.39 | 6 | 0 | 0 |
| 13 | 100.00 | 5.00 | 0 | 0 | 0 |
| 14 | 100.00 | 5.02 | 0 | 0 | 0 |
| AVG | 88.07 | 7.89 | 10.29 | 0.57 | 1.07 |
| SD | 18.18 | 3.17 | 15.36 | 0.94 | 3.73 |

SET #3: R=.20 09-05-1985

| # | SUCCESS | ITERATIONS | MISSED | BADSTOP | NO SOLUTION |
|---|---|---|---|---|---|
| 1 | 88.00 | 11.54 | 10 | 2 | 0 |
| 2 | 80.00 | 9.80 | 20 | 0 | 0 |
| 3 | 89.00 | 9.75 | 11 | 0 | 0 |
| 4 | 99.00 | 7.56 | 0 | 1 | 0 |
| 5 | 40.00 | 14.00 | 46 | 0 | 14 |
| 6 | 96.00 | 10.28 | 0 | 3 | 1 |
| 7 | 96.00 | 5.00 | 0 | 4 | 0 |
| 8 | 97.00 | 5.00 | 0 | 3 | 0 |
| 9 | 95.00 | 5.89 | 0 | 5 | 0 |
| 10 | 100.00 | 5.00 | 0 | 0 | 0 |
| 11 | 100.00 | 5.00 | 0 | 0 | 0 |
| 12 | 97.00 | 7.77 | 3 | 0 | 0 |
| 13 | 100.00 | 5.00 | 0 | 0 | 0 |
| 14 | 98.00 | 5.00 | 0 | 2 | 0 |
| AVG | 91.07 | 7.61 | 6.43 | 1.43 | 1.07 |
| SD | 15.76 | 3.00 | 12.91 | 1.74 | 3.73 |

SET #3: R=.25 09-05-1985

| # | SUCCESS | ITERATIONS | MISSED | BADSTOP | NO SOLUTION |
|---|---|---|---|---|---|
| 1 | 88.00 | 10.85 | 10 | 2 | 0 |
| 2 | 86.00 | 9.44 | 13 | 1 | 0 |
| 3 | 90.00 | 8.93 | 10 | 0 | 0 |
| 4 | 99.00 | 7.40 | 0 | 1 | 0 |
| 5 | 41.00 | 13.90 | 44 | 1 | 14 |
| 6 | 93.00 | 9.59 | 0 | 6 | 1 |
| 7 | 96.00 | 5.00 | 0 | 4 | 0 |
| 8 | 97.00 | 5.00 | 0 | 3 | 0 |
| 9 | 93.00 | 5.44 | 0 | 7 | 0 |
| 10 | 100.00 | 5.00 | 0 | 0 | 0 |
| 11 | 100.00 | 5.00 | 0 | 0 | 0 |
| 12 | 98.00 | 7.30 | 2 | 0 | 0 |
| 13 | 100.00 | 5.00 | 0 | 0 | 0 |
| 14 | 95.00 | 5.00 | 0 | 2 | 3 |
| AVG | 91.14 | 7.35 | 5.64 | 1.93 | 1.29 |
| SD | 15.13 | 2.83 | 11.97 | 2.30 | 3.75 |

searches converge after a significantly different number of iterations (2 versus 16+, 5 verses 13, and so forth). Requiring a minimum of five iterations allows for the geometries in which the two searches converge simultaneously in a few iterations.

### 4.4.5 Results of Set 4 - Average Distance and Five Iteration Minimum

Adding the five iteration minimum as an additional stopping criterion to set 2 provides a significant improvement in performance (see table 4-4). The runs with the 0.15 nautical mile stopping criterion are virtually as good as the runs in set 1 in terms of the number of successes but not in terms of the number of iterations required.

The effectiveness of the stopping criterion is degraded significantly by any increase in the stopping distance beyond 0.15 nautical miles. Using a larger average distance as a stopping criterion is counterproductive if the two searches do not converge simultaneously. It is noted, however, that this technique produced 100 percent success in five geometries. No other technique tested provided 100 percent success for as many geometries. The common trend in these five geometries is that most searches in these geometries converge in 5 or less iterations, that is, the two searches tended to converge quickly and simultaneously, the conditions under which using the average between two searches tends to provide a good solution.

Table 4-4. Results of Set 4 - Simultaneous Conjugate Gradient Search.

SET #4: R=.15 09-05-1985

| # | SUCCESS | ITERATIONS | MISSED | BADSTOP | NO SOLUTION |
|---|---|---|---|---|---|
| 1 | 60.00 | 12.19 | 29 | 11 | -0 |
| 2 | 75.00 | 10.38 | 20 | 5 | 0 |
| 3 | 74.00 | 10.15 | 22 | 4 | 0 |
| 4 | 100.00 | 7.82 | 0 | 0 | 0 |
| 5 | 24.00 | 14.15 | 50 | 12 | 14 |
| 6 | 80.00 | 11.00 | 17 | 2 | 1 |
| 7 | 95.00 | 5.05 | 0 | 5 | 0 |
| 8 | 100.00 | 5.04 | 0 | 0 | 0 |
| 9 | 85.00 | 6.33 | 0 | 15 | 0 |
| 10 | 100.00 | 5.00 | 0 | 0 | 0 |
| 11 | 100.00 | 5.00 | 0 | 0 | 0 |
| 12 | 85.00 | 8.39 | 6 | 9 | 0 |
| 13 | 100.00 | 5.00 | 0 | 0 | 0 |
| 14 | 100.00 | 5.02 | 0 | 0 | 0 |
| AVG | 84.14 | 7.89 | 10.29 | 4.50 | 1.07 |
| SD | 21.54 | 3.17 | 15.36 | 5.24 | 3.73 |

SET #4: R=.20 09-05-1985

| # | SUCCESS | ITERATIONS | MISSED | BADSTOP | NO SOLUTION |
|---|---|---|---|---|---|
| 1 | 44.00 | 11.54 | 10 | 46 | 0 |
| 2 | 52.00 | 9.80 | 20 | 28 | 0 |
| 3 | 68.00 | 9.75 | 11 | 21 | 0 |
| 4 | 88.00 | 7.56 | 0 | 12 | 0 |
| 5 | 20.00 | 14.00 | 46 | 20 | 14 |
| 6 | 69.00 | 10.28 | 0 | 30 | 1 |
| 7 | 95.00 | 5.00 | 0 | 5 | 0 |
| 8 | 100.00 | 5.00 | 0 | 0 | 0 |
| 9 | 70.00 | 5.89 | 0 | 30 | 0 |
| 10 | 100.00 | 5.00 | 0 | 0 | 0 |
| 11 | 100.00 | 5.00 | 0 | 0 | 0 |
| 12 | 79.00 | 7.77 | 3 | 18 | 0 |
| 13 | 100.00 | 5.00 | 0 | 0 | 0 |
| 14 | 100.00 | 5.00 | 0 | 0 | 0 |
| AVG | 77.50 | 7.61 | 6.43 | 15.00 | 1.07 |
| SD | 25.19 | 3.00 | 12.91 | 14.89 | 3.73 |

SET #4: R=.25 09-05-1985

| # | SUCCESS | ITERATIONS | MISSED | BADSTOP | NO SOLUTION |
|---|---|---|---|---|---|
| 1 | 39.00 | 10.85 | 10 | 51 | 0 |
| 2 | 45.00 | 9.44 | 13 | 42 | 0 |
| 3 | 50.00 | 8.93 | 10 | 40 | 0 |
| 4 | 81.00 | 7.40 | 0 | 19 | 0 |
| 5 | 17.00 | 13.90 | 44 | 25 | 14 |
| 6 | 63.00 | 9.59 | 0 | 36 | 1 |
| 7 | 95.00 | 5.00 | 0 | 5 | 0 |
| 8 | 100.00 | 5.00 | 0 | 0 | 0 |
| 9 | 62.00 | 5.44 | 0 | 38 | 0 |
| 10 | 100.00 | 5.00 | 0 | 0 | 0 |
| 11 | 100.00 | 5.00 | 0 | 0 | 0 |
| 12 | 66.00 | 7.30 | 2 | 32 | 0 |
| 13 | 100.00 | 5.00 | 0 | 0 | 0 |
| 14 | 100.00 | 5.00 | 0 | 0 | 0 |
| AVG | 72.71 | 7.35 | 5.64 | 20.57 | 1.07 |
| SD | 27.83 | 2.83 | 11.97 | 19.25 | 3.73 |

### 4.4.6 Results of Set 5 - Most Stable Sum of Squares of Error: Last Three Iterations

As the conjugate gradient search converges toward the optimum, the differences between the sum of squares of error tend to be very small (refer to figure 2-5). The evaluation criterion for choosing a solution in this set of runs is to choose the solution that has the most stable sum of squares of error when the stopping criteria are invoked. "Most stable" is defined as the search that has the least change over the last three iterations. The results of set 5 show capability comparable to set 1 in obtaining an acceptable solution but at the expense of an increase of one in the average number of iterations required (see table 4-5).

### 4.4.7 Results of Set 6 - Most Stable Sum of Squares of Error: Last Four Iterations

The results of set 6 are virtually identical to the results of set 5 (see table 4-6). Some minor differences are noted in some of the geometries, but the averages are identical. With few exceptions, the search that changes least between the last three iterations also changes least between the last four iterations.

Table 4-5. Results of Set 5 - Simultaneous Conjugate Gradient Search.

SET #5: R=.15 09-05-1985

| # | SUCCESS | ITERATIONS | MISSED | BADSTOP | NO SOLUTION |
|---|---|---|---|---|---|
| 1 | 70.00 | 12.19 | 29 | 1 | 0 |
| 2 | 80.00 | 10.38 | 20 | 0 | 0 |
| 3 | 78.00 | 10.15 | 22 | 0 | 0 |
| 4 | 96.00 | 7.82 | 0 | 4 | 0 |
| 5 | 35.00 | 14.15 | 50 | 1 | 14 |
| 6 | 74.00 | 11.00 | 17 | 8 | 1 |
| 7 | 94.00 | 5.05 | 0 | 6 | 0 |
| 8 | 93.00 | 5.04 | 0 | 7 | 0 |
| 9 | 89.00 | 6.33 | 0 | 11 | 0 |
| 10 | 100.00 | 5.00 | 0 | 0 | 0 |
| 11 | 100.00 | 5.00 | 0 | 0 | 0 |
| 12 | 89.00 | 8.39 | 6 | 5 | 0 |
| 13 | 100.00 | 5.00 | 0 | 0 | 0 |
| 14 | 96.00 | 5.02 | 0 | 4 | 0 |
| AVG | 85.29 | 7.89 | 10.29 | 3.36 | 1.07 |
| SD | 17.57 | 3.17 | 15.36 | 3.63 | 3.73 |

SET #5: R=.20 09-05-1985

| # | SUCCESS | ITERATIONS | MISSED | BADSTOP | NO SOLUTION |
|---|---|---|---|---|---|
| 1 | 87.00 | 11.54 | 10 | 3 | 0 |
| 2 | 79.00 | 9.80 | 20 | 1 | 0 |
| 3 | 89.00 | 9.75 | 11 | 0 | 0 |
| 4 | 96.00 | 7.56 | 0 | 4 | 0 |
| 5 | 39.00 | 14.00 | 46 | 1 | 14 |
| 6 | 87.00 | 10.28 | 0 | 12 | 1 |
| 7 | 91.00 | 5.00 | 0 | 9 | 0 |
| 8 | 92.00 | 5.00 | 0 | 8 | 0 |
| 9 | 84.00 | 5.89 | 0 | 16 | 0 |
| 10 | 100.00 | 5.00 | 0 | 0 | 0 |
| 11 | 100.00 | 5.00 | 0 | 0 | 0 |
| 12 | 90.00 | 7.77 | 3 | 7 | 0 |
| 13 | 100.00 | 5.00 | 0 | 0 | 0 |
| 14 | 94.00 | 5.00 | 0 | 6 | 0 |
| AVG | 87.71 | 7.61 | 6.43 | 4.79 | 1.07 |
| SD | 15.34 | 3.00 | 12.91 | 5.09 | 3.73 |

SET #5: R=.25 09-05-1985

| # | SUCCESS | ITERATIONS | MISSED | BADSTOP | NO SOLUTION |
|---|---|---|---|---|---|
| 1 | 85.00 | 10.85 | 10 | 5 | 0 |
| 2 | 85.00 | 9.44 | 13 | 2 | 0 |
| 3 | 88.00 | 8.93 | 10 | 2 | 0 |
| 4 | 96.00 | 7.40 | 0 | 4 | 0 |
| 5 | 40.00 | 13.90 | 44 | 2 | 14 |
| 6 | 80.00 | 9.59 | 0 | 19 | 1 |
| 7 | 91.00 | 5.00 | 0 | 9 | 0 |
| 8 | 92.00 | 5.00 | 0 | 8 | 0 |
| 9 | 80.00 | 5.44 | 0 | 20 | 0 |
| 10 | 100.00 | 5.00 | 0 | 0 | 0 |
| 11 | 100.00 | 5.00 | 0 | 0 | 0 |
| 12 | 85.00 | 7.30 | 2 | 13 | 0 |
| 13 | 100.00 | 5.00 | 0 | 0 | 0 |
| 14 | 94.00 | 5.00 | 0 | 6 | 0 |
| AVG | 86.86 | 7.35 | 5.64 | 6.43 | 1.07 |
| SD | 15.19 | 2.83 | 11.97 | 6.71 | 3.73 |

Table 4-6. Results of Set 6 - Simultaneous Conjugate Gradient Search.

SET #6: R=.15 09-05-1985

| # | SUCCESS | ITERATIONS | MISSED | BADSTOP | NO SOLUTION |
|---|---|---|---|---|---|
| 1 | 69.00 | 12.19 | 29 | 2 | 0 |
| 2 | 80.00 | 10.38 | 20 | 0 | 0 |
| 3 | 78.00 | 10.15 | 22 | 0 | 0 |
| 4 | 97.00 | 7.82 | 0 | 3 | 0 |
| 5 | 36.00 | 14.15 | 50 | 0 | 14 |
| 6 | 74.00 | 11.00 | 17 | 8 | 1 |
| 7 | 92.00 | 5.05 | 0 | 8 | 0 |
| 8 | 90.00 | 5.04 | 0 | 10 | 0 |
| 9 | 91.00 | 6.33 | 0 | 9 | 0 |
| 10 | 100.00 | 5.00 | 0 | 0 | 0 |
| 11 | 100.00 | 5.00 | 0 | 0 | 0 |
| 12 | 88.00 | 8.39 | 6 | 6 | 0 |
| 13 | 100.00 | 5.00 | 0 | 0 | 0 |
| 14 | 98.00 | 5.02 | 0 | 2 | 0 |
| AVG | 85.21 | 7.89 | 10.29 | 3.43 | 1.07 |
| SD | 17.45 | 3.17 | 15.36 | 3.90 | 3.73 |

SET #6: R=.20 09-05-1985

| # | SUCCESS | ITERATIONS | MISSED | BADSTOP | NO SOLUTION |
|---|---|---|---|---|---|
| 1 | 87.00 | 11.54 | 10 | 3 | 0 |
| 2 | 80.00 | 9.80 | 20 | 0 | 0 |
| 3 | 89.00 | 9.75 | 11 | 0 | 0 |
| 4 | 97.00 | 7.56 | 0 | 3 | 0 |
| 5 | 39.00 | 14.00 | 46 | 1 | 14 |
| 6 | 87.00 | 10.28 | 0 | 12 | 1 |
| 7 | 91.00 | 5.00 | 0 | 9 | 0 |
| 8 | 89.00 | 5.00 | 0 | 11 | 0 |
| 9 | 84.00 | 5.89 | 0 | 16 | 0 |
| 10 | 100.00 | 5.00 | 0 | 0 | 0 |
| 11 | 100.00 | 5.00 | 0 | 0 | 0 |
| 12 | 87.00 | 7.77 | 3 | 10 | 0 |
| 13 | 100.00 | 5.00 | 0 | 0 | 0 |
| 14 | 97.00 | 5.00 | 0 | 3 | 0 |
| AVG | 87.64 | 7.61 | 6.43 | 4.86 | 1.07 |
| SD | 15.41 | 3.00 | 12.91 | 5.55 | 3.73 |

SET #6: R=.25 09-05-1985

| # | SUCCESS | ITERATIONS | MISSED | BADSTOP | NO SOLUTION |
|---|---|---|---|---|---|
| 1 | 84.00 | 10.85 | 10 | 6 | 0 |
| 2 | 84.00 | 9.44 | 13 | 3 | 0 |
| 3 | 88.00 | 8.93 | 10 | 2 | 0 |
| 4 | 97.00 | 7.40 | 0 | 3 | 0 |
| 5 | 41.00 | 13.90 | 44 | 1 | 14 |
| 6 | 81.00 | 9.59 | 0 | 18 | 1 |
| 7 | 91.00 | 5.00 | 0 | 9 | 0 |
| 8 | 89.00 | 5.00 | 0 | 11 | 0 |
| 9 | 80.00 | 5.44 | 0 | 20 | 0 |
| 10 | 100.00 | 5.00 | 0 | 0 | 0 |
| 11 | 100.00 | 5.00 | 0 | 0 | 0 |
| 12 | 84.00 | 7.30 | 2 | 14 | 0 |
| 13 | 100.00 | 5.00 | 0 | 0 | 0 |
| 14 | 97.00 | 5.00 | 0 | 3 | 0 |
| AVG | 86.86 | 7.35 | 5.64 | 6.43 | 1.07 |
| SD | 15.08 | 2.83 | 11.97 | 6.86 | 3.73 |

#### 4.4.8 Results of Set 7 - Most Stable Sum of Squares of Error: Last Two Iterations

The evaluation criteria for set 7 is to choose the solution that has the least reduction in sum of squares of error between the last and next to last iteration. The results of set 7 show that the probability of success is two percentage points less than in sets 5 and 6 (see table 4-7).

#### 4.4.9 Summary of Results

The results of the seven sets of runs are presented in table 4-8. Performance varies from a low of 60 to a high of 91 percent probability of obtaining an acceptable solution in 15 iterations or less for the 21 combinations of stopping and evaluation criteria tested. The average number of iterations varies from 6.3 to 7.9. 16 of the 21 combinations tested provided acceptable solutions for 85 percent of the trials in an average of 7.9 iterations or less while four of the combinations tested provided acceptable solutions for 88 percent of the trials in an average of 7.6 iterations or less. The only tests that fail to provide 84 percent success use the average distance between the two searches. Even in this case, however, the technique provides 84 percent success if the stopping distance is low and a minimum of five iterations is required. Two evaluation criteria consistently provide good results: lowest sum of squares of error and most stable sum of squares of error. Lowest sum of squares of error provides an 85 percent success after an average of only 6.3 iterations for a stopping

Table 4-7. Results of Set 7 - Simultaneous Conjugate Gradient Search.

SET #7: R=.15 09-05-1985

| # | SUCCESS | ITERATIONS | MISSED | BADSTOP | NO SOLUTION |
|---|---|---|---|---|---|
| 1 | 70.00 | 12.19 | 29 | 1 | 0 |
| 2 | 80.00 | 10.38 | 20 | 0 | 0 |
| 3 | 78.00 | 10.15 | 22 | 0 | 0 |
| 4 | 96.00 | 7.82 | 0 | 4 | 0 |
| 5 | 35.00 | 14.15 | 50 | 1 | 14 |
| 6 | 79.00 | 11.00 | 17 | 3 | 1 |
| 7 | 86.00 | 5.05 | 0 | 14 | 0 |
| 8 | 93.00 | 5.04 | 0 | 7 | 0 |
| 9 | 77.00 | 6.33 | 0 | 23 | 0 |
| 10 | 100.00 | 5.00 | 0 | 0 | 0 |
| 11 | 100.00 | 5.00 | 0 | 0 | 0 |
| 12 | 88.00 | 8.39 | 6 | 6 | 0 |
| 13 | 100.00 | 5.00 | 0 | 0 | 0 |
| 14 | 96.00 | 5.02 | 0 | 4 | 0 |
| AVG | 84.14 | 7.89 | 10.29 | 4.50 | 1.07 |
| SD | 17.28 | 3.17 | 15.36 | 6.62 | 3.73 |

SET #7: R=.20 09-05-1985

| # | SUCCESS | ITERATIONS | MISSED | BADSTOP | NO SOLUTION |
|---|---|---|---|---|---|
| 1 | 83.00 | 11.54 | 10 | 7 | 0 |
| 2 | 78.00 | 9.80 | 20 | 2 | 0 |
| 3 | 89.00 | 9.75 | 11 | 0 | 0 |
| 4 | 96.00 | 7.56 | 0 | 4 | 0 |
| 5 | 37.00 | 14.00 | 46 | 3 | 14 |
| 6 | 89.00 | 10.28 | 0 | 10 | 1 |
| 7 | 83.00 | 5.00 | 0 | 17 | 0 |
| 8 | 94.00 | 5.00 | 0 | 6 | 0 |
| 9 | 71.00 | 5.89 | 0 | 29 | 0 |
| 10 | 100.00 | 5.00 | 0 | 0 | 0 |
| 11 | 100.00 | 5.00 | 0 | 0 | 0 |
| 12 | 87.00 | 7.77 | 3 | 10 | 0 |
| 13 | 100.00 | 5.00 | 0 | 0 | 0 |
| 14 | 94.00 | 5.00 | 0 | 6 | 0 |
| AVG | 85.79 | 7.61 | 6.43 | 6.71 | 1.07 |
| SD | 16.51 | 3.00 | 12.91 | 8.08 | 3.73 |

SET #7: R=.25 09-05-1985

| # | SUCCESS | ITERATIONS | MISSED | BADSTOP | NO SOLUTION |
|---|---|---|---|---|---|
| 1 | 81.00 | 10.85 | 10 | 9 | 0 |
| 2 | 84.00 | 9.44 | 13 | 3 | 0 |
| 3 | 89.00 | 8.93 | 10 | 1 | 0 |
| 4 | 96.00 | 7.40 | 0 | 4 | 0 |
| 5 | 35.00 | 13.90 | 44 | 7 | 14 |
| 6 | 82.00 | 9.59 | 0 | 17 | 1 |
| 7 | 83.00 | 5.00 | 0 | 17 | 0 |
| 8 | 94.00 | 5.00 | 0 | 6 | 0 |
| 9 | 64.00 | 5.44 | 0 | 36 | 0 |
| 10 | 100.00 | 5.00 | 0 | 0 | 0 |
| 11 | 100.00 | 5.00 | 0 | 0 | 0 |
| 12 | 79.00 | 7.30 | 2 | 19 | 0 |
| 13 | 100.00 | 5.00 | 0 | 0 | 0 |
| 14 | 94.00 | 5.00 | 0 | 6 | 0 |
| AVG | 84.36 | 7.35 | 5.64 | 8.93 | 1.07 |
| SD | 17.49 | 2.83 | 11.97 | 10.18 | 3.73 |

Table 4-8. Summary Results of Simultaneous Conjugate Gradient Search.

| Stopping Criterion | Statistic | Set | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| Searches Within .15 Nautical Miles | Percent Success | 85 | 80 | 88 | 84 | 85 | 85 | 84 |
| | Number of Iterations | 7.0 | 7.0 | 7.9 | 7.9 | 7.9 | 7.9 | 7.9 |
| Searches Within .20 Nautical Miles | Percent Success | 86 | 70 | 91 | 78 | 88 | 88 | 86 |
| | Number of Iterations | 6.6 | 6.6 | 7.6 | 7.6 | 7.6 | 7.6 | 7.6 |
| Searches Within .25 Nautical Miles | Percent Success | 85 | 61 | 91 | 73 | 87 | 87 | 84 |
| | Number of Iterations | 6.3 | 6.3 | 7.4 | 7.4 | 7.4 | 7.4 | 7.4 |

distance of 0.25 nautical miles. The evaluation criteria used in set 3 consistently provide a high probability of success (except in geometry 5). It is noted, however, that geometry 5 is unique in that 7 of the 20 individual searches in geometry 5 fail to reach an acceptable solution in 15 iterations. Consequently any simultaneous search that uses either one or both of any of those seven searches will not converge to the stopping criterion.

The optimal technique identified by this analysis is the following:

1. Perform a minimum of 5 iterations.
2. Perform a maximum of 15 iterations.
3. Stop iterating when searches are within .25 nautical miles.
4. Choose the search with the lower sum of squares of error at the time the stopping criteria are invoked.

This technique provides a 91 percent success rate in an average of 7.4 iterations.

SECTION 5

CONCLUSIONS AND RECOMMENDATIONS

5.1 SINGLE CONJUGATE GRADIENT SEARCH

The single conjugate gradient search, with the fine tuning discussed in section 3, does consistently find the minimum of thesum of squares of error function for all the geometries tested in a finite number of iterations (26 or less). Table 3-4 shows that there is better than a ninety percent probability of obtaining an adequate solution using 13 iterations. A much higher probability of success is possible if the algorithm performs 20 or 25 iterations but the time required to perform a large number of iterations is prohibitive in the tactical environment in which computational power and time to reach a solution are limited.

It is not necessary that every individual attempt at solving the problem be successful. As is discussed in section 2, target motion analysis is a continuous process. As a new bearing is added to a full set of bearings, the earliest bearing is dropped and a new solution estimated. Because a new set of bearings makes at least minor changes to the contour, it is possible that a solution is obtained in one time step even if it is not obtained in the previous time step.

The use of different starting points on subsequent searches may increase the probability of obtaining a solution at one of the time steps. Unfortunately, it is often difficult to determine which of two successive solutions is better.

The major problem with the single conjugate gradient search is that it provides, at best, vague indicators as to the success of an individual search. It is possible to use regression analysis or exponential smoothing on the last n solutions. There is the long term probability that a particular solution is correct, but this says nothing about the accuracy of an individual solution, and there is also information available from the sum of squares of error. It is likely, however, that the end user of these tactical decision aids is unfamiliar with analysis of variance. Moreover, such information as a stable sum of squares of error over the last n iterations could indicate only that the search is stalled at a non-optimal point.

## 5.2 SIMULTANEOUS CONJUGATE GRADIENT SEARCH

The simultaneous conjugate gradient search provides a solution to the two main problems with the single conjugate gradient search: it reduces the average number of iterations and it provides an indication if a solution is acceptable. It provides an acceptable answer in 91 percent of the trials in an average of 7.4 iterations versus a 92 percent success rate in 13 iterations for the single search. More important, however, the simultaneous conjugate gradient search provides clear indications as to which solutions are acceptable, which ones are close to acceptable, which ones are inconclusive, and which ones are stalled. It also provides a meaningful measure of the maximum error at every iteration (in units of feet).

A simple CRT graphic or digital printout showing the progress of each search in $R_0, R_n$ space would provide an operator with a clear indication if the algorithm is converging or if one or both of the searches is stalled. Because this technique differentiates acceptable from non-acceptable solutions, the probability of obtaining an acceptable solution on one of three successive calculations is very high given a 91 percent probability of success at an individual time step.

The simultaneous conjugate gradient search provides a solution in half the iterations required by the single conjugate gradient search. Even if true simultaneous co-processing capability is not available, this technique would on the average require no more total computer time than the single search. The simultaneous search, moreover, provides evaluation criteria that are not possible with the single search. If computers with simultaneous co-processing are made available for this application, then a significant speed advantage is also possible.

## 5.3 OBSERVATIONS PERTAINING TO THE CONJUGATE GRADIENT SEARCH

The results of this research identify a number of potential problems with using the conjugate gradient search to solve the passive bearings only target motion analysis problem. The most serious problem is that the search fails to work in a large number of trials if the length of the acceleration leg is too long. In these cases the search is taken into negative space in one or both variables. The basic geometry of the problem does not hold for negative values of $R_0$ and $R_n$, and the algorithm stalls in negative space. On the other hand, if the length of the acceleration leg is too short, the search will not move quickly enough to the optimum in some other geometries. Since it is determined that no one length for the acceleration leg works for all geometries, a technique is developed to successively reduce the length of the acceleration leg and restart the search if the search is taken into negative space.

The choice of the stopping criteria for the golden section search is also critical. The conjugate gradient search does not work if the golden section search on the acceleration leg is stopped too soon. Because nearly all the computational time of the conjugate gradient search is spent doing functional evaluations on the acceleration leg, it is important that the golden section search be terminated as early as possible. Although the lower limits of the stopping criteria for the conjugate gradient search are not tested, it is found that a maximum of 20 iterations or a maximum distance of 0.00001 between two functional evaluations does work for all geometries and starting positions tested.

For the algorithm tested in this analysis, the use of double precision calculations is counterproductive on the Z-100 computer. Computers that typically use a high level of precision for computations such as the HP 9845 and the VAX may keep the conjugate gradient search too close to the ridges of the function and thus keep it from moving off the ridges to the optimum.

Although the conjugate gradient search developed for this analysis finds the optimum of all the geometries from all the tested starting points, it is inefficient in solving the fifth geometry, requiring as many as 26 iterations, over four times the average. This problem can be resolved by modifying the conjugate gradient search.

This research has identified a number of areas in which the conjugate gradient search must be tuned to the particular problem. Otherwise there is a probability that the search will either stall or move unacceptably slowly toward the optimum.

## 5.4 RECOMMENDATIONS

Although this research demonstrates the effectiveness of the simultaneous conjugate gradient search in solving the bearings only target motion analysis problem in the representative geometries, the results of this analysis are not necessarily conclusive. One of the more important findings of this research is that minor changes to the starting positions for the search or to the contour can cause significant differences in the ability of the search to converge quickly on the minimum of the function.

The next step in this analysis is to test and fine tune the algorithm to solve the problem using typical bearings that are generated on sea tests. A number of variables are included in sea test data that affect the contour. Among these are random bearing error and random error on the time steps. Another important concern is the issue of lost data points. A bearing can be lost at a given time step for a number of reasons including temporary loss of detection by the sonar. Bearing readings also must be rejected at times if a bearing seams to be clearly erroneous. If the target is on a constant course and speed (as the algorithm assumes), then the bearing rate is fairly constant between time steps. A student's t or other suitable test is used to determine if a new bearing is acceptable and to be added to the algorithm or if it is to be rejected. Because all these factors affect the shape of a contour, it is important to tune and test the algorithm with data taken from sea tests or at least with simulated sea test data that include random bearing error.

The geometries used in this study assume an environment with a 15 nautical mile direct path/bottom-bounce detection range. It is important to test the technique both in other environments and with different tracking tactics.

Lastly the conjugate gradient search itself must be tested and optimized. Various conjugate gradients need to be tested as well as the optimal ratio of conjugate gradient to gradient directions.

The results of this analysis on theoretical tracking geometries and theoretical bearings suggest that a properly tuned conjugate gradient search is an effective tool for solving the passive target motion analysis problem. The technique of using two simultaneous conjugate gradient searches converging from opposite directions provides both an effective search and also criteria for evaluating the success of a search at any iteration.

REFERENCES


1. Operations Analysis Study Group, Naval Operations Analysis, Naval Institute Press, Annapolis, MD., 1977.
2. Ghen, D., Passive Localization TMA Reference Manual, Vol I, Inventory of Techniques Analysis & Technology, North Stonington, CT., 23 June 1981.
3. Ghen, D., Passive Localization TMA Reference Manual, Vol II, Techniques and Performance, Analysis & Technology, North Stonington, CT., 23 June 1981.
4. Lindren, A.G., and Gong, K.F., Position and Velocity Estimation via Bearing Observations, Naval Underwater Systems Center, Newport, RI, 15 June 1977.
5. Weinstein, Ehud, Decentralization of the Gaussian Maximum Likelihood Estimator and Its Application to Passive Array Localization, Woods Hole Oceanographic Institution, MA, 17 August 1979.
6. McQuitty, J. B., Target Motion Analysis Algorithms for Rapid Localization Methods, Mandex, Inc., Rockville, MD., 27 February 1985.
7. Shudde, R.H., Naval Applications: Ten Algorithms for the HP-67 and HP-97 Calculators, Naval Postgraduate School, Monterey, CA, February 1979.

8. Shudde, R.H., Multiple Leg TMA Procedures with Programs for the HP-41CV, HP-75C, the Sharp PC-1500 and the TRS-80 Model 100 Portable Computers, Naval Postgraduate School, Monterey, CA, September 1983.

9. Robinson, G., A Computerized Technique for Passive Localization and Target Motion Analysis, Analysis & Technology, North Stonington, CT., 4 May 1977.

10. Callahan, J.E., ASW Directions Environmental Data Book, Analysis & Technology, North Stonington, CT., 1 May 1980.

11. Cohen, Marvin S. and Brown, Rex V., Decision Support for Attack Sumbarine Commanders, Decision Science Consortium, Inc., Falls Church, VA, October 1980.

12. Krajewski, R., "Multiprocessing: An Overview", BYTE, Vol. 10, No. 5, May 1985.

13. Kumar, Vipin, "Thoughts on Parallel Processing", BYTE, Vol. 10, No. 5, May 1985.

14. Walpole, Ronald E., and Myers, Raymond H., Probability and Statistics for Engineers and Scientists, Macmillan Publishing Company, New York, 1972.

15. Rao, S.S., Optimization, Theory and Applications, John Wiley Eastern Limited, New Delhi, 1979.

16. Bazaraa, Mokhtar S. and Shetty, C.M., Nonlinear Programming, Theory and Algorithms, John Wiley & Sons, New York, 1979.

17. Urick, Robert J., Sound Propagation in the Sea, Peninsula Publishing, Los Altos, CA., 1982.

18. Thompson, E., Design Document for the SURFLOC Computer Program, Analysis & Technology, North Stonington, Ct., 19 September 1980.

19. Pelletier, D., Design Document for the Data Management System HP9845 Program, Analysis & Technology, North Stonington, CT., March 1984.

APPENDIX A

Conjugate Gradient Search Algorithm

```
1 REM *****************************************************************
2 REM *                                                               *
10 REM *                   PROGRAM NAME: CONTOUR                       *
12 REM *  GENERATE GEOMETRIES FOR TARGET MOTION ANALYSIS PROBLEM       *
13 REM *                      07 JULY 1985                             *
18 REM *                                                               *
19 REM *****************************************************************
20 REM
50 PI=3.14159:EPSILON=.00001
60 REM
70 REM DEFINE TRIG FUNCTIONS IN DEGREES
80 REM
90 DEF FNSIND(X)=SIN(X*PI/180)
100 DEF FNCOSD(X)=COS(X*PI/180)
110 DEF FNTAND(X)=TAN(X*PI/180)
120 DEF FNATND(X)=180/PI*ATN(X)
130 ALPHA=(3-SQR(5))/2:PRINT "ALPHA ",ALPHA
140 DIM XB(15),YB(15),XT(15),YT(15),B(15),R(15)
150 DIM XTARG(15),YTARG(15),BA(15),BM(15),BR(15),ERRORSQ(15)
160 REM
170 REM READ FIX DATA FOR GEOMETRY FROM DATA STATEMENTS
180 REM
190 DATA 7,225,12,20,3,9
200 READ SPEEDBLUE,MANEUVER,SPEEDTARGET,COURSETARGET,TIMESTEP,RO
210 REM
220 REM READ VARIABLE DATA FOR GEOMETRY FROM INPUT STATEMENTS
230 REM
240 INPUT "ENTER RUN NUMBER ",RUNNO$
250 INPUT "ENTER SPEED OF TARGET ",SPEEDTARGET
260 INPUT "ENTER COURSE OF TARGET ",COURSETARGET
270 INPUT "ENTER STARTING RANGE   ",RO
280 INPUT "ENTER BEARING TO TARGET       ",BEARTOTARG
290 REM
300 REM PRINT VARIABLE DATA FOR GEOMETRY
310 REM
320 LPRINT DATE$:LPRINT "RUN NUMBER: ";RUNNO$:LPRINT
330 LPRINT "SPEED BLUE (NMI):    ";SPEEDBLUE
340 LPRINT "SPEED TARGET:        ";SPEEDTARGET
350 IF COURSETARGET=20 THEN LPRINT "COURSE TARGET:        110"
360 IF COURSETARGET=-20 THEN LPRINT "COURSE TARGET:        070"
370 LPRINT "TIME STEP (MIN):     ";TIMESTEP
380 LPRINT "RO (NMI):            ";RO
390 LPRINT "BEARING TO TARGET:  ";BEARTOTARG:LPRINT:LPRINT
400 FILE$=RUNNO$+".ASC":OPEN "O",#1,FILE$
410 REM
420 REM COMPUTE POSITION OF TRACKER
430 REM
440 XB(0)=0:YB(0)=0
450 FOR I=1 TO 9
460 YB(I)=0
470 INCREMENT=SPEEDBLUE/60*TIMESTEP
480 XB(I)=XB(I-1)+INCREMENT
490 NEXT I
```

```
500 FOR I=10 TO 14
510 XB(I)=XB(I-1)+INCREMENT*FNCOSD(360-MANEUVER)
520 YB(I)=YB(I-1)+INCREMENT*FNSIND(360-MANEUVER)
530 NEXT I
540 REM
550 REM COMPUTE POSITION OF TARGET
560 REM
570 XTARG(0)=RO*FNSIND(BEARTOTARG)
580 YTARG(0)=RO*FNCOSD(BEARTOTARG)
590 INCREMENT=SPEEDTARGET/60*TIMESTEP
600 FOR I=1 TO 14
610 XTARG(I)=XTARG(I-1)+INCREMENT*FNCOSD(360-COURSETARGET)
620 YTARG(I)=YTARG(I-1)+INCREMENT*FNSIND(360-COURSETARGET)
630 NEXT I
640 REM
650 REM COMPUTE RANGE AND BEARINGS AT EACH TIME STEP
660 REM
670 FOR I=0 TO 14
680 R(I)=((XTARG(I)-XB(I))^2+(YTARG(I)-YB(I))^2)^.5
690 IF XTARG(I)=XB(I) THEN BEARING=90
     ELSE BEARING=FNATND((YTARG(I)-YB(I))/(XTARG(I)-XB(I)))
700 IF BEARING<0 THEN BEARING=180+BEARING
710 IF I<10 THEN BR(I)= 360-BEARING
     ELSE BR(I)=135-BEARING
720 BM(I)=BEARING
730 BA(I)=FNATND((XTARG(I)-XB(I))/(YTARG(I)-YB(I)))
740 NEXT I
750 REM
760 REM PRINT GEOMETRY DATA TO PRINTER
770 REM
780 LPRINT "                                               MEASURED DATA
790 LPRINT "     TRACKER            TARGET          ARITH  RELATIVE TRUE
800 LPRINT "  XB        YB        XT        YT   BEARING BEARING BEARING R
ANGE
810 LPRINT
820 FOR I=0 TO 14
830 LPRINT USING "###.##  ";XB(I),YB(I),XTARG(I),YTARG(I),BM(I),BR(I
),BA(I),R(I)
840 NEXT I
850 REM
860 REM PRINT GEOMETRY DATA TO DISK
870 REM
880 PRINT #1,RUNNO$;",";SPEEDBLUE;MANEUVER;SPEEDTARGET;COURSETARGET;
TIMESTEP;RO;BEARTOTARG
890 FOR I=0 TO 14
900 PRINT #1,XB(I);YB(I);XTARG(I);YTARG(I);BM(I);BR(I);BA(I);R(I)
910 NEXT I
920 CLOSE
930 LPRINT CHR$(12)
940 REM
950 REM LOOP FOR THE 4 CONTOURS
960 REM
970 N=14 :REM 15 BEARINGS COUNTED FROM 0 TO 14
```

```
980 FOR CC=1  TO 4
990 IF CC=1  THEN LPRINT DATE$:LPRINT "RUN NUMBER: ";RUNNO$:LPRINT
1000 IF CC=3 THEN LPRINT DATE$:LPRINT "RUN NUMBER: ";RUNNO$:LPRINT
1010 IF CC=1 THEN CONTOUR=2 ELSE IF CC=2 THEN CONTOUR=5
1020 IF CC=3 THEN CONTOUR=10 ELSE IF CC=4 THEN CONTOUR=20
1030 LPRINT "CONTOUR: SSE= ";CONTOUR
1040 LPRINT "                 Ro                        SSE"
1050 LPRINT "     RO     LOWER    CENTER  UPPER    LOWER  CENTER   UPPE
R
1060 REM
1070 REM LOOP FOR THE 24 VALUES OF RO
1080 REM
1090 FOR RO=2 TO 25
1100 X1=0:X4=30
1110 XSTART=XB(0)+RO*FNSIND(BA(0))
1120 YSTART=YB(0)+RO*FNCOSD(BA(0))
1130 REM
1140 REM COMPUTE THE MINIMUM OF THE FUNCTION
1150 REM
1160 TEST=0: REM FIND MINIMUM
1170 GOSUB 1500 : REM  GOLDEN SECTION SEARCH
1180 CENTER=X:CSSE=SSE
1190 PRINT "CENTER ",CENTER
1200 REM
1210 REM  COMPUTE THE LOWER POINT ON THE CONTOUR
1220 REM
1230 X1=0
1240 X4=CENTER
1250 TEST=CONTOUR : REM FIND THE POINT ON THE CONTOUR
1260 GOSUB 1500 : REM GOLDEN SECTION SEARCH
1270 LOWER=X:LSSE=SSE
1280 PRINT "LOWER ",LOWER
1290 REM
1300 REM FIND THE HIGH POINT ON THE CONTOUR
1310 REM
1320 X1=CENTER
1330 X4=50
1340 GOSUB 1500 : REM GOLDEN SECTION SEARCH
1350 UPPER=X:USSE=SSE
1360 PRINT "UPPER ",UPPER
1370 PRINT USING "#####.##";RO,LOWER,CENTER,UPPER,LSSE,CSSE,USSE
1380 LPRINT USING "#####.##";RO,LOWER,CENTER,UPPER,LSSE,CSSE,USSE
1390 NEXT RO
1400 IF CC=1 THEN LPRINT:LPRINT
1410 IF CC=3 THEN LPRINT:LPRINT
1420 IF CC=4 THEN LPRINT CHR$(12)
1430 IF CC=2 THEN LPRINT CHR$(12)
1440 NEXT CC
1450 END
1460 REM
1470 REM
```

```
1480 REM                           ****** SUBROUTINES ******
1490 REM
1500 REM GOLDEN SECTION SEARCH
1510 REM
1520 X2=X1+ALPHA*(X4-X1)
1530 X3=X4-ALPHA*(X4-X1)
1540 X=X3:GOSUB 1810:FX3=FX
1550 X=X2:GOSUB 1810:FX2=FX
1560 FOR I=1 TO 25
1570 IF FX3<=FX2 THEN GOSUB 1650
              ELSE GOSUB 1730
1580 IF ABS(FX2-FX3)<EPSILON THEN RETURN
1590 NEXT I
1600 REM LPRINT I,X2,FX2,FX3,X3-X2
1610 RETURN
1620 REM
1630 REM
1640 REM
1650 REM ELIMINATE X1   LOW VARIABLE
1660 REM
1670 X1=X2:X2=X3:FX2=FX3
1680 X3=X4-ALPHA*(X4-X1)
1690 X=X3:GOSUB 1810:FX3=FX:RETURN
1700 REM
1710 REM
1720 REM
1730 REM ELIMINATE X4   HIGH VARIABLE
1740 REM
1750 X4=X3:X3=X2:FX3=FX2
1760 X2=X1+ALPHA*(X4-X1)
1770 X=X2:GOSUB 1810:FX2=FX:RETURN
1780 REM
1790 REM
1800 REM
1810 REM FUNCTIONAL EVALUATION
1820 REM
1830 SSE=0
1840 XEND=XB(N)+X*FNSIND(BA(N))
1850 YEND=YB(N)+X*FNCOSD(BA(N))
1860 FOR I=0 TO N
1870 XT(I)=(1-I/N)*XSTART+I/N*XEND
1880 YT(I)=(1-I/N)*YSTART+I/N*YEND
1890 REM
1900 REM DETERMINE COMPUTED BEARING GIVEN ESTIMATE OF R0 AND RN
1910 REM
1920 B(I)=FNATND((XT(I)-XB(I))/(YT(I)-YB(I)))
1930 REM
1940 REM COMPUTE SUM OF SQUARES OF ERROR
1950 REM
1960 ESQ=(BA(I)-B(I))^2:SSE=SSE+ESQ
1970 NEXT I
1980 REM PRINT "R0,X,SSE ",R0,X,SSE
1990 FX=ABS(SSE-TEST)
2000 RETURN
```

APPENDIX B

Points on a Contour Algorithm

```
1 REM ************************************************************
10 REM * SECOND PROGRAM FOR THESIS - CC     18 JUL 85    DISK #67 *
11 REM *                                                          *
12 REM *     PROGRAM PERFORMS CONJUGATE GRADIENT SEARCH           *
13 REM *                                                          *
14 REM ************************************************************
15 REM
20 DEFDBL E:DEFINT I-K
30  DIM XB(15),YB(15),XT(15),YT(15),B(15),R(15),RNG(15)
35  DIM XTARG(15),YTARG(15),BA(15),BM(15),BR(15)
40 PI=3.14159:EPSILON=.00001:COUNTALL=0:L4IN=10
45 INPUT "RN ",RN
46 INPUT "RO START ",ROSTART
47 INPUT "RO END   ",ROEND
50 DEF FNSIND(X)=SIN(X*PI/180)
60 DEF FNCOSD(X)=COS(X*PI/180)
80 DEF FNATND(X)=180/PI*ATN(X)
90 ALPHA=(3-SQR(5))/2:PRINT "ALPHA ",ALPHA
95  FORMAT1$="###          ###.###     ###.###     ###.###     ###.###
###.#####"
105 REM
109 GOSUB 30000 : REM READ INPUTS
110 REM
200 IN=14
210 FOR IR=ROSTART TO ROEND:RO=IR:LPRINT:LPRINT:LPRINT "RO=: ";RO:LP
RINT "RN=:";RN
215 LPRINT TIME$
220 LPRINT "ITERATION       RO          RN       DELTA RO     DELTA RN
SSE"
230 L1=0:L4HOLD=L4IN:COUNT=0  :REM    L4 NEEDED TO BE VARIED
235 YY=RO:ZZ=RN
236 REM
237 REM  EVALUATE FUNCTION
238 REM
240 GOSUB 6000:FX=ESS
245 REM
246 REM COMPUTE GRADIENT
247 REM
250 GOSUB 3000
280 Y=RO:Z=RN
284 REM
285 REM  PERFORM 15 ITERATIONS
286 REM
290 FOR K=1 TO 15:PRINT "K:    ";K
292 REM
293 REM DECREMENT LENGTH OF ACCELERATION LEG
294 REM
295 L4=L4HOLD/K:IF L4<2 THEN L4=2
296 REM
297 REM PERFORM TWO LEGS FOR EACH ITERATION
298 REM
300 FOR J=1 TO  2:PRINT "     J  L4 :";J,L4
310 REM         SET LAMBDA ON ACCELERATION LEG
320 X1=L1:X4=L4
```

```
330 GOSUB 1000 :REM GOLDEN SECTION SEARCH
340 REM PRINT "X,Y,FXY  ",YY,ZZ,FX2
350 REM
352 REM COMPUTE CONJUGATE GRADIENT ON FIRST LEG
354 REM COMPUTE GRADIENT ON SECOND LEG
356 REM
360 IF J=1 THEN GOSUB 10000 ELSE GOSUB 11000
370 NEXT J
374 REM
375 REM COMPUTE DISTANCE FROM MINIMUM
376 REM
380 DELTARO=ABS(RNG(0)-Y):DELTARN=ABS(RNG(14)-Z)
400 LPRINT USING FORMAT1$;K,Y,Z,DELTARO,DELTARN,ESS
410  PRINT USING FORMAT1$;K,Y,Z,DELTARO,DELTARN,ESS
414 REM
415 REM EXIT IF SEARCH WITHIN LIMITS OF MINIMUM
416 REM
420 IF DELTARO<.06 AND DELTARN<.06 THEN 435
421 REM
422 REM  START SEARCH OVER WITH SHORTER ACCELERATION LEG IF SSE >20
423 REM  THIS USUALLY MEANS SEARCH IS IN NEGATIVE SPACE
424 REM
425 IF K>=2 AND ESS>20 THEN L4HOLD=L4HOLD/2:GOTO 235
430 NEXT K
435 LPRINT "FUNCTIONAL EVALUATIONS: ";COUNT
440 NEXT IR
444 LPRINT:LPRINT:LPRINT TIME$
445 LPRINT "TOTAL FUNCTIONAL EVALUATIONS: ";COUNTALL
450 END
1000 REM
1001 REM
1002 REM                    ****** SUBROUTINES ******
1003 REM
1019 REM
1020 REM GOLDEN SECTION SEARCH
1021 REM
1040 X2=X1+ALPHA*(X4-X1)
1050 X3=X4-ALPHA*(X4-X1)
1060 X=X3:GOSUB 1430:FX3=FX
1090 X=X2:GOSUB 1430:FX2=FX
1120 FOR IJ=1 TO 20
1125 PRINT IJ;
1130 IF FX3<=FX2 THEN GOSUB 1340
              ELSE GOSUB 1390
1140 IF ABS(FX2-FX3)<EPSILON THEN PRINT:RETURN
1150 NEXT IJ
1330 REM LPRINT I,X2,FX2,FX3,X3-X2
1333 PRINT
1335 RETURN
1336 REM
1337 REM
1340 REM ELIMINATE X1
1341 REM
1345 X1=X2:X2=X3:FX2=FX3
```

```
1357 X3=X4-ALPHA*(X4-X1)
1360 X=X3:GOSUB 1430:FX3=FX:RETURN
1370 REM
1372 REM
1390 REM ELIMINATE X4
1391 REM
1392 X4=X3:X3=X2:FX3=FX2
1398 X2=X1+ALPHA*(X4-X1)
1400 X=X2:GOSUB 1430:FX2=FX:RETURN
1420 REM
1421 REM
1430 REM FUNCTIONAL EVALUATION
1435 REM
1440 ZZ=Z+NEGGZ/ABSG*X
1450 YY=Y+NEGGY/ABSG*X
1460 REM PRINT "YY,ZZ,ABSG ";YY,ZZ,ABSG
1470 GOSUB 6000 :FX=ESS:RETURN
1480 REM
1490 REM
3000 REM COMPUTE GRADIENT
3010  REM
3011  GRO=0:GRN=0
3012  FOR I=0 TO 14
3013  GRO=GRO+(1-I/14)/R(I)*FNSIND(BA(0)-B(I))*(B(I)-BA(I))
3014  GRN=GRN+I/14/R(I)*FNSIND(BA(14)-B(I))*(B(I)-BA(I))
3015  REM PRINT "3015 GRO,GRN ";GRO,GRN
3016  NEXT I
3020 GY=GRO:GZ=GRN
3030 REM PRINT "GRADIENT,GY,GZ ";GY,GZ
3100 REM
3200 REM
4000 REM COMPUTE ABSOLUTE VALUE OF GRADIENT
4001 REM
4010 ABSG=SQR(GY*GY+GZ*GZ)
4020 REM PRINT "ABS GRADIENT "; ABSG
4030 REM
4040 REM
5000 REM COMPUTE NEGATIVE GRADIENT
5001 REM
5010 NEGGY=-GY:NEGGZ=-GZ
5020 REM PRINT "NEGATIVE GRADIENT ";NEGGY,NEGGZ
5030 RETURN
5050 REM
5060 REM
6000 REM FUNCTIONAL EVALUATION
6005 REM
6010 ESS=0:COUNT=COUNT+1:COUNTALL=COUNTALL+1
6012 XSTART=XB(0)+YY*FNSIND(BA(0))
6014 YSTART=YB(0)+YY*FNCOSD(BA(0))
6020 XEND=XB(IN)+ZZ*FNSIND(BA(IN))
6030 YEND=YB(IN)+ZZ*FNCOSD(BA(IN))
6040 FOR I=0 TO IN
6050 XT(I)=(1-I/IN)*XSTART+I/IN*XEND
6060 YT(I)=(1-I/IN)*YSTART+I/IN*YEND
```

```
6065 DX=XT(I)-XB(I):DY=YT(I)-YB(I)
6070 B(I)=FNATND(DX/DY):R(I)=SQR(DX*DX+DY*DY)
6074 REM
6075 REM  COMPUTE SUM OF SQUARES OF ERROR
6076 REM
6080 ER=BA(I)-B(I):ESS=ESS+ER*ER
6090 NEXT I
6100 REM PRINT "RO,X,SSE ",RO,X,ESS
6120 RETURN
6200 REM
6300 REM
10000 REM J LESS THAN 2
10001 REM COMPUTE CONJUGATE GRADIENT
10002 REM
10010 OLDNEGGY=NEGGY:OLDNEGGZ=NEGGZ:OLDABSG=ABSG:Y=YY:Z=ZZ
10020 GOSUB 3000 : REM COMPUTE GRADIENT
10030 ALPHAJ=ABSG*ABSG/OLDABSG/OLDABSG
10040 REM PRINT "ALPHA J ";ALPHAJ
10050 NEGGY=NEGGY+ALPHAJ*OLDNEGGY:NEGGZ=NEGGZ+ALPHAJ*OLDNEGGZ
10060 REM PRINT "NEW GRADIENT ";NEGGY,NEGGZ
10070 RETURN
10080 REM
10090 REM
11000 REM J EQUAL TO  2
11001 REM COMPUTE GRADIENT
11010 OLDNEGGY=NEGGY:OLDNEGGZ=NEGGZ:OLDABSG=ABSG:Y=YY:Z=ZZ
11020 GOSUB 3000 : REM COMPUTE GRADIENT
11070 RETURN
11100 REM
11110 REM
30000 REM INPUT DATA FROM FILE
30005 REM
30010 INPUT "ENTER DATA FILE ",FILE$
30015 FILE$=FILE$+".ASC"
30020 OPEN "I",#1,FILE$
30030 INPUT #1,RUNNO$,SPEEDBLUE,MANEUVER,SPEEDTARGET,COURSETARGET,TI
MESTEP,RO,BEARTOTARG
30032 FOR I=0 TO 14
30034 INPUT #1,XB(I),YB(I),XTARG(I),YTARG(I),BM(I),BR(I),BA(I),RNG(I
)
30036 NEXT I
30040 LPRINT DATE$:LPRINT "RUN NUMBER: ";RUNNO$:LPRINT
30045      RETURN:REM &&&&&&&&&&&&&&&&&&&&&&
30046 REM
30047 REM ALL INPUTS CAN BE PRINTED AS AN OPTION
30048 REM
30050 LPRINT "SPEED BLUE (NMI):   ";SPEEDBLUE
30060 REM LPRINT "MANEUVER:            ";MANEUVER
30070 LPRINT "SPEED TARGET:       ";SPEEDTARGET
30080 IF COURSETARGET=20 THEN LPRINT "COURSE TARGET:       110"
30090 IF COURSETARGET=-20 THEN LPRINT "COURSE TARGET:       070"
30100 LPRINT "TIME STEP (MIN):    ";TIMESTEP
30110 LPRINT "RO (NMI):           ";RO
30120 LPRINT "BEARING TO TARGET:  ";BEARTOTARG:LPRINT:LPRINT
```

```
30130           PRINT "                                            MEASURED DAT
A"
30140 LPRINT "                                       MEASURED DATA"
30150 PRINT "   TRACKER           TARGET      ARITH  RELATIVE TRUE"
30160 LPRINT "   TRACKER           TARGET      ARITH  RELATIVE TRUE"
30170 PRINT "  XB        YB       XT       YT   BEARING BEARING BEARING
RANGE"
30180 LPRINT "  XB        YB       XT       YT   BEARING BEARING BEARING
 RANGE"
30190 PRINT
30200 LPRINT
30210 FOR I=0 TO 14
30230 PRINT USING "###.##  ";XB(I),YB(I),XTARG(I),YTARG(I),BM(I),BR(
I),BA(I),RNG(I)
30240 LPRINT USING "###.##  ";XB(I),YB(I),XTARG(I),YTARG(I),BM(I),BR
(I),BA(I),RNG(I)
30250 NEXT I
30260 CLOSE
30270 LPRINT CHR$(12)
30300 RETURN
```

APPENDIX C

Algorithm to Search from Ten Points

```
1 REM **************************************************************
2 REM *                                                            *
10 REM *  THIRD  PROGRAM FOR THESIS - CGLH15  20 JUL 85 DISK #68   *
11 REM *          PERFORMS 15 ITERATIONS FOR HIGH AND LOW SEARCH    *
14 REM *                                                            *
15 REM **************************************************************
20 DEFDBL E:DEFINT I-K
30  DIM XB(15),YB(15),XT(15),YT(15),B(15),R(15),RNG(15)
40  DIM XTARG(15),YTARG(15),BA(15),BM(15),BR(15)
50 PI=3.14159:EPSILON=.00001:COUNTALL=0:L4IN=10
60 INPUT "OUTPUT FILE NAME ",OUTFILE$:OUTFILE$=OUTFILE$+".ASC"
70 OPEN "O",#2,OUTFILE$
80 REM
90 REM PAIRS OF STARTING POINTS FOR LOW SEARCH
100 REM
110 DATA 1,2,3,1,2,1,3,2,3,1
120 DATA 1,2,3,2,1,3,1,3,2,4
130 REM
140 REM PAIRS OF STARTING POINTS FOR HIGH SEARCH
150 REM
160 DATA 15,14,13,15,15,14,14,13,13,16
170 DATA 15,14,13,14,13,15,13,14,15,16
180 INPUT "ENTER HIGH OR LOW SEARCH (H OR L )",HL$
190 IF HL$="H" THEN RESTORE 160 ELSE RESTORE 110
200 FOR I=1 TO 10: READ RO(I):PRINT RO(I):NEXT
210 FOR I=1 TO 10: READ RN(I):PRINT RN(I):NEXT
220 DEF FNCOSD(X)=COS(X*PI/180)
230 DEF FNSIND(X)=SIN(X*PI/180)
240 DEF FNATND(X)=180/PI*ATN(X)
250 ALPHA=(3-SQR(5))/2:PRINT "ALPHA ",ALPHA
260  FORMAT1$="###       ###.###      ###.###      ###.###      ###.###
 ###.#####"
270 GOSUB 30000 :REM INPUT DATA FROM DISK FILE
280 IN=14:REM RN=3
290 FOR IR=1 TO 10
300 RO=RO(IR):RN=RN(IR):PRINT #2,RO,RN:FLAG=0
310 LPRINT:LPRINT:LPRINT "TRIAL: ";IR:LPRINT "RO=: ";RO:LPRINT "RN=:
 ";RN
320 LPRINT TIME$
330 LPRINT "ITERATION      RO             RN      DELTA RO     DELTA RN
 SSE"
340 L1=0:L4HOLD=L4IN:COUNT=0  :REM    L4 NEEDED TO BE VARIED
350 YY=RO:ZZ=RN
360 GOSUB 6000:FX=ESS
370 GOSUB 3000
380 Y=RO:Z=RN
390 FOR K=1 TO 15:PRINT "K:    ";K
400 L4=L4HOLD/K:IF L4<2 THEN L4=2
410 FOR J=1 TO  2:PRINT "     J   L4 :";J,L4
420 REM       SET LAMBDA ON ACCELERATION LEG
430 X1=L1:X4=L4
440 GOSUB 1000
450 REM PRINT "X,Y,FXY ",YY,ZZ,FX2
460 IF J=1 THEN GOSUB 10000 ELSE GOSUB 11000
```

```
470 NEXT J
480 DELTARO=ABS(RNG(0)-Y):DELTARN=ABS(RNG(14)-Z)
490 LPRINT USING FORMATI$;K,Y,Z,DELTARO,DELTARN,ESS
500  PRINT USING FORMATI$;K,Y,Z,DELTARO,DELTARN,ESS
510 REM
520 REM PRINT RESULTS OF ITERATION TO DISK FILE
530 REM
540  PRINT #2,USING FORMATI$;K,Y,Z,DELTARO,DELTARN,ESS
550 REM
560 REM PRINT FLAG IS SOLUTION IS REACHED AT THIS ITERATION
570 REM
580  IF DELTARO<.06 AND DELTARN<.06 AND FLAG=0 THEN LPRINT "SOLUTION
":FLAG=1
590 IF K>=2 AND ESS>20 THEN L4HOLD=L4HOLD/2:GOTO 350
600 NEXT K
610 LPRINT "FUNCTIONAL EVALUATIONS: ";COUNT
620 NEXT IR
630 LPRINT:LPRINT:LPRINT TIME$
640 LPRINT "TOTAL FUNCTIONAL EVALUATIONS: ";COUNTALL
650 END
1000 REM
1001 REM
1002 REM             ****** SUBROUTINES ******
1003 REM
1019 REM
1020 REM GOLDEN SECTION SEARCH
1021 REM
1040 X2=X1+ALPHA*(X4-X1)
1050 X3=X4-ALPHA*(X4-X1)
1060 X=X3:GOSUB 1430:FX3=FX
1090 X=X2:GOSUB 1430:FX2=FX
1120 FOR IJ=1 TO 20
1125 PRINT IJ;
1130 IF FX3<=FX2 THEN GOSUB 1340
                ELSE GOSUB 1390
1140 IF ABS(FX2-FX3)<EPSILON THEN PRINT:RETURN
1150 NEXT IJ
1330 REM LPRINT I,X2,FX2,FX3,X3-X2
1333 PRINT
1335 RETURN
1336 REM
1337 REM
1340 REM ELIMINATE X1
1341 REM
1345 X1=X2:X2=X3:FX2=FX3
1357 X3=X4-ALPHA*(X4-X1)
1360 X=X3:GOSUB 1430:FX3=FX:RETURN
1370 REM
1372 REM
1390 REM ELIMINATE X4
1391 REM
1392 X4=X3:X3=X2:FX3=FX2
1398 X2=X1+ALPHA*(X4-X1)
1400 X=X2:GOSUB 1430:FX2=FX:RETURN
```

```
1420 REM
1421 REM
1430 REM FUNCTIONAL EVALUATION
1435 REM
1440 ZZ=Z+NEGGZ/ABSG*X
1450 YY=Y+NEGGY/ABSG*X
1460 REM PRINT "YY,ZZ,ABSG ";YY,ZZ,ABSG
1470 GOSUB 6000 :FX=ESS:RETURN
1480 REM
1490 REM
3000 REM COMPUTE GRADIENT
3010  REM
3011  GRO=0:GRN=0
3012  FOR I=0 TO 14
3013  GRO=GRO+(1-I/14)/R(I)*FNSIND(BA(0)-B(I))*(B(I)-BA(I))
3014  GRN=GRN+I/14/R(I)*FNSIND(BA(14)-B(I))*(B(I)-BA(I))
3015  REM PRINT "3015 GRO,GRN ";GRO,GRN
3016  NEXT I
3020 GY=GRO:GZ=GRN
3030 REM PRINT "GRADIENT,GY,GZ ";GY,GZ
3100 REM
3200 REM
4000 REM COMPUTE ABS VALUE OF GRADIENT
4001 REM
4010 ABSG=SQR(GY*GY+GZ*GZ)
4020 REM PRINT "ABS GRADIENT "; ABSG
4030 REM
4040 REM
5000 REM COMPUTE NEGATIVE GRADIENT
5001 REM
5010 NEGGY=-GY:NEGGZ=-GZ
5020 REM PRINT "NEGATIVE GRADIENT ";NEGGY,NEGGZ
5030 RETURN
5050 REM
5060 REM
6000 REM FUNCTIONAL EVALUATION
6005 REM
6010 ESS=0:COUNT=COUNT+1:COUNTALL=COUNTALL+1
6012 XSTART=XB(0)+YY*FNSIND(BA(0))
6014 YSTART=YB(0)+YY*FNCOSD(BA(0))
6020 XEND=XB(IN)+ZZ*FNSIND(BA(IN))
6030 YEND=YB(IN)+ZZ*FNCOSD(BA(IN))
6040 FOR I=0 TO IN
6050 XT(I)=(1-I/IN)*XSTART+I/IN*XEND
6060 YT(I)=(1-I/IN)*YSTART+I/IN*YEND
6065 DX=XT(I)-XB(I):DY=YT(I)-YB(I)
6070 B(I)=FNATND(DX/DY):R(I)=SQR(DX*DX+DY*DY)
6074 REM
6075 REM  COMPUTE SUM OF SQUARES OF ERROR
6076 REM
6080 ER=BA(I)-B(I):ESS=ESS+ER*ER
6090 NEXT I
6100 REM PRINT "RO,X,SSE ",RO,X,ESS
6120 RETURN
6200 REM
```

```
6300 REM
10000 REM J LESS THAN 2
10001 REM COMPUTE CONJUGATE GRADIENT
10002 REM
10010 OLDNEGGY=NEGGY:OLDNEGGZ=NEGGZ:OLDABSG=ABSG:Y=YY:Z=ZZ
10020 GOSUB 3000 : REM COMPUTE GRADIENT
10030 ALPHAJ=ABSG*ABSG/OLDABSG/OLDABSG
10040 REM PRINT "ALPHA J ";ALPHAJ
10050 NEGGY=NEGGY+ALPHAJ*OLDNEGGY:NEGGZ=NEGGZ+ALPHAJ*OLDNEGGZ
10060 REM PRINT "NEW GRADIENT ";NEGGY,NEGGZ
10070 RETURN
10080 REM
10090 REM
11000 REM J EQUAL TO  2
11001 REM COMPUTE GRADIENT
11010 OLDNEGGY=NEGGY:OLDNEGGZ=NEGGZ:OLDABSG=ABSG:Y=YY:Z=ZZ
11020 GOSUB 3000 : REM COMPUTE GRADIENT
11070 RETURN
11100 REM
11110 REM
30000 REM INPUT DATA FROM FILE
30005 REM
30010 INPUT "ENTER DATA FILE ",FILE$
30015 FILE$=FILE$+".ASC"
30020 OPEN "I",#1,FILE$
30030 INPUT #1,RUNNO$,SPEEDBLUE,MANEUVER,SPEEDTARGET,COURSETARGET,TI
MESTEP,RO,BEARTOTARG
30032 FOR I=0 TO 14
30034 INPUT #1,XB(I),YB(I),XTARG(I),YTARG(I),BM(I),BR(I),BA(I),RNG(I
)
30036 NEXT I
30040 LPRINT DATE$:LPRINT "RUN NUMBER: ";RUNNO$:LPRINT
30045      RETURN:REM &&&&&&&&&&&&&&&&&&&&&
30046 REM
30047 REM ALL INPUTS CAN BE PRINTED AS AN OPTION
30048 REM
30050 LPRINT "SPEED BLUE (NMI):   ";SPEEDBLUE
30060 REM LPRINT "MANEUVER:              ";MANEUVER
30070 LPRINT "SPEED TARGET:       ";SPEEDTARGET
30080 IF COURSETARGET=20 THEN LPRINT "COURSE TARGET:       110"
30090 IF COURSETARGET=-20 THEN LPRINT "COURSE TARGET:       070"
30100 LPRINT "TIME STEP (MIN):    ";TIMESTEP
30110 LPRINT "RO (NMI):           ";RO
30120 LPRINT "BEARING TO TARGET:  ";BEARTOTARG:LPRINT:LPRINT
30130          PRINT "                                    MEASURED DAT
A"
30140 LPRINT "                                    MEASURED DATA"
30150 PRINT "   TRACKER           TARGET      ARITH  RELATIVE TRUE"
```

```
6300 REM
10000 REM J LESS THAN 2
10001 REM COMPUTE CONJUGATE GRADIENT
10002 REM
10010 OLDNEGGY=NEGGY:OLDNEGGZ=NEGGZ:OLDABSG=ABSG:Y=YY:Z=ZZ
10020 GOSUB 3000 : REM COMPUTE GRADIENT
10030 ALPHAJ=ABSG*ABSG/OLDABSG/OLDABSG
10040 REM PRINT "ALPHA J ";ALPHAJ
10050 NEGGY=NEGGY+ALPHAJ*OLDNEGGY:NEGGZ=NEGGZ+ALPHAJ*OLDNEGGZ
10060 REM PRINT "NEW GRADIENT ";NEGGY,NEGGZ
10070 RETURN
10080 REM
10090 REM
11000 REM J EQUAL TO  2
11001 REM COMPUTE GRADIENT
11010 OLDNEGGY=NEGGY:OLDNEGGZ=NEGGZ:OLDABSG=ABSG:Y=YY:Z=ZZ
11020 GOSUB 3000 : REM COMPUTE GRADIENT
11070 RETURN
11100 REM
11110 REM
30000 REM INPUT DATA FROM FILE
30005 REM
30010 INPUT "ENTER DATA FILE ",FILE$
30015 FILE$=FILE$+".ASC"
30020 OPEN "I",#1,FILE$
30030 INPUT #1,RUNNO$,SPEEDBLUE,MANEUVER,SPEEDTARGET,COURSETARGET,TI
MESTEP,RO,BEARTOTARG
30032 FOR I=0 TO 14
30034 INPUT #1,XB(I),YB(I),XTARG(I),YTARG(I),BM(I),BR(I),BA(I),RNG(I
)
30036 NEXT I
30040 LPRINT DATE$:LPRINT "RUN NUMBER: ";RUNNO$:LPRINT
30045      RETURN:REM &&&&&&&&&&&&&&&&&&&&&
30046 REM
30047 REM ALL INPUTS CAN BE PRINTED AS AN OPTION
30048 REM
30050 LPRINT "SPEED BLUE (NMI):    ";SPEEDBLUE
30060 REM LPRINT "MANEUVER:             ";MANEUVER
30070 LPRINT "SPEED TARGET:        ";SPEEDTARGET
30080 IF COURSETARGET=20 THEN LPRINT "COURSE TARGET:       110"
30090 IF COURSETARGET=-20 THEN LPRINT "COURSE TARGET:       070"
30100 LPRINT "TIME STEP (MIN):     ";TIMESTEP
30110 LPRINT "RO (NMI):            ";RO
30120 LPRINT "BEARING TO TARGET:   ";BEARTOTARG:LPRINT:LPRINT
30130          PRINT "                                  MEASURED DAT
A"
30140 LPRINT "                                 MEASURED DATA"
30150 PRINT "   TRACKER          TARGET      ARITH  RELATIVE TRUE"
30160 LPRINT "   TRACKER          TARGET      ARITH  RELATIVE TRUE"
30170 PRINT "  XB       YB       XT       YT   BEARING BEARING BEARING
RANGE"
30180 LPRINT "  XB       YB       XT       YT   BEARING BEARING BEARING
 RANGE"
30190 PRINT
```

```
30200 LPRINT
30210 FOR I=0 TO 14
30230 PRINT USING "###.##  ";XB(I),YB(I),XTARG(I),YTARG(I),BM(I),BR(
I),BA(I),RNG(I)
30240 LPRINT USING "###.##  ";XB(I),YB(I),XTARG(I),YTARG(I),BM(I),BR
(I),BA(I),RNG(I)
30250 NEXT I
30260 CLOSE
30270 LPRINT CHR$(12)
30300 RETURN
```

# APPENDIX D

## Simulated Simultaneous Search Algorithm

```
1  REM ************************************************************
10 REM * FOURTH PROGRAM FOR THESIS - STAT  23 AUG 85 DISK #68      *
20 REM *                                                           *
30 REM *              SIMULATED CONJUGATE GRADIENT SEARCH          *
40 REM *                                                           *
50 REM *        PROGRAM COLLECTS STATISTICS ON EFFECTIVENESS OF    *
60 REM *            SIMULATED CONJUGATE GRADIENT SEARCH            *
70 REM *                                                           *
75 REM ************************************************************
80  DIM K2(15),Y2(15),Z2(15),DRO2(15),DRN2(15),SSE2(15)
90  DIM COUNTER1(16)
100  DIM K1(15),Y1(15),Z1(15),DRO1(15),DRN1(15),SSE1(15)
110 REM
120 INPUT "ENTER TEST DISTANCE 1 ",TESTDIST1 : REM STOPPING DISTANCE
 FOR SEARCH
130 INPUT "ENTER CONTOUR           ",CONTOUR   : REM GEOMETRY NUMBER 1
 - 14
140 REM
150 LPRINT "        TEST DISTANCE 1 ",TESTDIST1
160 LPRINT "        CONTOUR         ",CONTOUR
170 FIRSTFILE$=FIRSTFILE$+".ASC":SECONDFILE$=SECONDFILE$+".ASC"
180 COUNT=0:BADSTOP=0:MISSED=0
190 REM
200 REM INPUT FILES NAMES FOR EACH CONTOUR AND TRUE MINIMUM OF FUNCT
ION
210 REM
220 IF COUNTOUR<13 THEN ROTRUE=9
230 IF CONTOUR=1 THEN RNTRUE=7.73:FF$="B:OUTPUT1L":SF$="B:OUTPUT1H"
240 IF CONTOUR=2 THEN RNTRUE=12.2:FF$="B:OUTPUT2L":SF$="B:OUTPUT2H"
250 IF CONTOUR=3 THEN RNTRUE=11.11:FF$="B:OUTPUT3L":SF$="B:OUTPUT3H"
260 IF CONTOUR=4 THEN RNTRUE=14.1:FF$="B:OUTPUT4L":SF$="B:OUTPUT4H"
270 IF CONTOUR=5 THEN RNTRUE=3.97:FF$="B:OUTPUT5L":SF$="B:OUTPUT5H"
280 IF CONTOUR=6 THEN RNTRUE=9.55:FF$="B:OUTPUT6L":SF$="B:OUTPUT6H"
290 IF CONTOUR=7 THEN RNTRUE=6.65:FF$="B:OUTPUT7L":SF$="B:OUTPUT7H"
300 IF CONTOUR=8 THEN RNTRUE=9.42:FF$="B:OUTPUT8L":SF$="B:OUTPUT8H"
310 IF CONTOUR=9 THEN RNTRUE=8.3:FF$="B:OUTPUT9L":SF$="B:OUTPUT9H"
320 IF CONTOUR=10 THEN RNTRUE=10.32:FF$="B:OUTPT10L":SF$="B:OUTPT10H
"
330 IF CONTOUR=11 THEN RNTRUE=5.68:FF$="B:OUTPT11L":SF$="B:OUTPT11H"
340 IF CONTOUR=12 THEN RNTRUE=8.36:FF$="B:OUTPT12L":SF$="B:OUTPT12H"
350 IF CONTOUR=13 THEN ROTRUE= 4:RNTRUE=4.67:FF$="B:OUTPT13L":SF$="B
:OUTPT13H"
360 IF CONTOUR=14 THEN ROTRUE= 4:RNTRUE=8.22:FF$="B:OUTPT14L":SF$="B
:OUTPT14H"
370 FIRSTFILE$=FF$+".ASC":SECONDFILE$=SF$+".ASC"
380 LPRINT "        FIRST FILE ",FIRSTFILE$
390 LPRINT "        SECOND FILE ",SECONDFILE$
400 FIRSTFILE$=FF$+".ASC":SECONDFILE$=SF$+".ASC"
410 LPRINT
420 LPRINT "                    NUMBER ,        DELTA          DELTA
DELTA        DELTA"
430 LPRINT " TRIAL      ITERATIONS         RN1            RO1
 RN2         RO2"
440 OPEN "I",#1,FIRSTFILE$
```

```
450 INPUT #1,DDATE$,NNAME$
460 PRINT DDATE$,NNAME$
470 REM
480 REM   LOOP FOR LOW SEARCH FILE
490 REM
500 FOR IONE=1 TO 10
510 INPUT #1,RO1,RN1
520 PRINT RO1,RN1
530 FOR J=1 TO 15
540 REM
550 REM   READ RESULTS OF 15 ITERATIONS
560 REM
570 INPUT #1,K1(J),Y1(J),Z1(J),DRO1(J),DRN1(J),SSE1(J)
580 REM PRINT  K1(J);Y1(J),Z1(J),DRO1(J),DRN1(J),SSE1(J)
590 NEXT J
600 OPEN "I",#2,SECONDFILE$
610 INPUT #2,DDDATE$,NNAME$
620 PRINT DDDATE$,NNAME$
630 REM
640 REM   LOOP FOR HIGH SEARCH FILE
650 REM
660 FOR ITWO=1 TO 10
670 COUNT=COUNT+1
680 INPUT #2,RO2,RN2
690 PRINT RO2,RN2
700 REM
710 REM   READ RESULTS OF 15 ITERATIONS
720 REM
730 FOR J=1 TO 15
740 INPUT #2,K2(J),Y2(J),Z2(J),DRO2(J),DRN2(J),SSE2(J)
750 REM PRINT  K2(J);Y2(J),Z2(J),DRO2(J),DRN2(J),SSE2(J)
760 NEXT J
770 REM FIRST TEST
780 PRINT "COUNT: ";COUNT
790 FOR J=1 TO 15
800 REM
810 REM  COMPUTE DISTANCE BETWEEN SEARCHES AT EACH ITERATION
820 REM
830 DISTANCE=SQR((Y2(J)-Y1(J))^2+(Z2(J)-Z1(J))^2)
840 REM PRINT J;DISTANCE,DRN2(J),DRN1(J),DRO2(J),DRO1(J)
850 REM PRINT J;DISTANCE
860 IF DISTANCE<=TESTDIST1 THEN PRINT "   OK",J,DISTANCE:COUNTER1(J)
=COUNTER1(J)+1:GOSUB 1040:GOTO 890: REM  PRINT STATUS AT TIME SEARCH
ES STOP
870 NEXT J
880 GOSUB 1290 : REM SEARCHES DO NOT CONVERGE IN 15 ITERATIONS
890 REM
900 NEXT ITWO
910 CLOSE #2
920 NEXT IONE
930 CLOSE
940 REM
950 REM PRINT SUMMARY STATISTICS
960 REM
```

```
970 GOSUB 1410
980 LPRINT CHR$(12)
990 END
1000 REM
1010 REM
1020 REM                 ****** SUBROUTINES ******
1030 REM
1040 REM DISTANCE FROM OPTIMUM AT SOLUTION
1050 REM SEARCHES CONVERGED TO STOPPING CRITERION
1060 REN  1. COMPUTE ERROR
1070 REM  2. DETERMINE IF SOLUTION ACCEPTABLE
1080 REM  3. SET FLAGS IF STOPPED TOO SOON
1090 REM  4. PRINT RESULTS
1100 REM
1110 OKL=0:OKH=0:SOLUTION=0
1120 XDISTL=DRN1(J)-RNTRUE
1130 YDISTL=DRO1(J)-ROTRUE
1140 XDISTH=DRN2(J)-RNTRUE
1150 YDISTH=DRO2(J)-ROTRUE
1160 LPRINT COUNT,J,DRN1(J),DRO1(J),DRN2(J),DRO2(J)
1170  PRINT COUNT,J,XDISTL,YDISTL,XDISTH,YDISTH
1180  PRINT COUNT,J,DRN1(J),DRN2(J),DRO1(J),DRO2(J)
1190 IF SSE1(J)<SSE2(J) THEN D1=DRO1(J):D2=DRN1(J)
                        ELSE D1=DRO2(J):D2=DRN2(J)
1200 IF DRO1(J)<=.06 AND DRN1(J)<=.06 THEN  PRINT "SOLUTION ON LOW":
OKL=1
1210 IF DRO2(J)<=.06 AND DRN2(J)<=.06 THEN  PRINT "SOLUTION ON HIGH"
:OKH=1
1220 IF D1<=.06 AND D2<=.06 THEN  PRINT "SOLUTION ":SOLUTION=1
1230 IF SOLUTION=0 THEN LPRINT "        NO SOLUTION":BADSTOP=BADSTO
P+1
1240 PRINT "D1,D2,SSE1,SSE2 ";D1,D2,SSE1(J),SSE2(J)
1250 RETURN
1260 REM
1270 REM
1280 REM
1290 REM    SEARCHES DO NOT CONVERGE TO STOPPING CRITERION IN 15 ITE
RATIONS
1300 REM
1310 J=15:SOLUTION=0
1320 COUNTER1(16)=COUNTER1(16)+1
1330 LPRINT COUNT,J,DRN1(J),DRO1(J),DRN2(J),DRO2(J),"NO SOLUTION"
1340  PRINT COUNT,J,DRN1(J),DRO1(J),DRN2(J),DRO2(J),"NO SOLUTION"
1350 IF DRO1(J)<=.06 AND DRN1(J)<=.06 THEN LPRINT "             SOLU
TION ON LOW":SOLUTION=1
1360 IF DRO2(J)<=.06 AND DRN2(J)<=.06 THEN LPRINT "             SOLU
TION ON HIGH":SOLUTION=1
1370 IF SOLUTION=1 THEN MISSED=MISSED+1
1380 RETURN
1390 REM
1400 REM
1410 REM PRINT SUMMARY RESULTS OF 100 TRIALS
1420 REM
1430 LPRINT:PRINT
```

```
1440 LPRINT "NUMBER          NUMBER"
1450 LPRINT "ITERATIONS      TRAILS"
1460 FOR J=1 TO 16
1470 PRINT J,COUNTER1(J)
1480 LPRINT J,COUNTER1(J)
1490 NEXT J
1500 LPRINT:PRINT
1510 PERCENTSUCCESS=1-(BADSTOP+COUNTER1(16))/100
1520 LPRINT "PERCENT SUCCESS ";PERCENTSUCCESS
1530 REM :COMPUTE AVERAGE NUMBER OF ITERATIONS
1540 TOTAL=0:FOR J=1 TO 15:TOTAL=TOTAL+COUNTER1(J)*J:NEXT J
1550 TOTAL=TOTAL+15*COUNTER1(16):AVERAGE=TOTAL/100
1560 LPRINT USING "\                             \##.##";"AVERAGE #
ITERATIONS",AVERAGE
1570 LPRINT:LPRINT "DID NOT RECOGNIZE SOLUTION ";MISSED
1580 LPRINT:LPRINT "STOPPED TOO EARLY          ";BADSTOP
1590 RETURN
```

APPENDIX E

Summary Statistics Algorithm

```
1   REM ***************************************************************
2   REM *                                                             *
10  REM *           BUILD FILE OF STATISTICAL INFO FOR THESIS         *
15  REM *                 MEAN AND STANDARD DEVIATION                 *
16  REM *                27 AUG 85  DISK  #49  PRINT-ST               *
18  REM *                                                             *
19  REM ***************************************************************
20  REM
30  INPUT "ENTER FILE NAME ",OUTFILE$
40  OUTFILE$=OUTFILE$+".ASC"
42  OPEN "I",#5,OUTFILE$
43  INPUT #5,OUTFILE$
46  INPUT "ENTER THE DATA TITLE ",TITLE$
47  LPRINT "                   "TITLE$,,DATE$
48  LPRINT
49  LPRINT      "                     #    SUCCESS  ITERATIONS   MISSED
BADSTOP   NO SOLUTION"
50  FOR J=1 TO 14
80  INPUT #5,J,A,B,C,D
81  A=A*100
82  E=100-A-C-D
83  REM COMPUTE SUM AND SUM OF SQUARES
84  T1=T1+A:SS1=SS1+A*A
85  T2=T2+B:SS2=SS2+B*B
86  T3=T3+C:SS3=SS3+C*C
87  T4=T4+D:SS4=SS4+D*D
88  T5=T5+E:SS5=SS5+E*E
89  LPRINT USING "                   ##   ###.##    ###.##         ##
 ##          ##";J,A,B,C,D,E
90  NEXT J
95  REM COMPUTE AVERAGE AND STANDARD DEVIATION
100  A1=T1/14:SD1=SQR((14*SS1-T1*T1)/14/13)
110  A2=T2/14:SD2=SQR((14*SS2-T2*T2)/14/13)
130  A3=T3/14:SD3=SQR((14*SS3-T3*T3)/14/13)
140  A4=T4/14:SD4=SQR((14*SS4-T4*T4)/14/13)
145  A5=T5/14:SD5=SQR((14*SS5-T5*T5)/14/13)
150  LPRINT
160  LPRINT USING "\              \\ \   ##.##    ###.##        ##.##
 ##.##       ##.##";DUMMY$,"AVG",A1,A2,A3,A4,A5
170  LPRINT USING "\              \\ \   ##.##    ###.##        ##.##
 ##.##       ##.##";DUMMY$,"SD ",SD1,SD2,SD3,SD4,SD5
180  LPRINT
185  LPRINT
190  LPRINT
1100  CLOSE
1110  END
```

# BIBLIOGRAPHY

# BIBLIOGRAPHY


Bazaraa, Mokhtar S. and Shetty, C.M., Nonlinear Programming, Theory and Algorithms, John Wiley & Sons, New York, 1979.

Callahan, J.E., ASW Directions Environmental Data Book, Analysis & Technology, North Stonington, CT., 1 May 1980.

Cohen, Marvin S. and Brown, Rex V., Decision Support for Attack Sumbarine Commanders, Decision Science Consortium, Inc., Falls Church, VA, October 1980.

Ghen, D., Passive Localization TMA Reference Manual, Vol I, Inventory of Techniques Analysis & Technology, North Stonington, CT., 23 June 1981.

Ghen, D., Passive Localization TMA Reference Manual, Vol II, Techniques and Performance, Analysis & Technology, North Stonington, CT., 23 June 1981.

Krajewski, R., "Multiprocessing: An Overview", BYTE, Vol. 10, No. 5, May 1985.

Kumar, Vipin, "Thoughts on Parallel Processing", BYTE, Vol. 10, No. 5, May 1985.

Lindren, A.G., and Gong, K.F., Position and Velocity Estimation via Bearing Observations, Naval Underwater Systems Center, Newport, RI, 15 June 1977.

McQuitty, J. B., Target Motion Analysis Algorithms for Rapid Localization Methods, Mandex, Inc., Rockville, MD., 27 February 1985.

Operations Analysis Study Group, Naval Operations Analysis, Naval Institute Press, Annapolis, MD., 1977.

Pelletier, D., Design Document for the Data Management System HP9845 Program, Analysis & Technology, North Stonington, CT., March 1984.

Rao, S.S., Optimization, Theory and Applications, John Wiley Eastern Limited, New Delhi, 1979.

Robinson, G., A Computerized Technique for Passive Localization and Target Motion Analysis, Analysis & Technology, North Stonington, CT., 4 May 1977.

Shudde, R.H., Multiple Leg TMA Procedures with Programs for the HP-41CV, HP-75C, the Sharp PC-1500 and the TRS-80 Model 100 Portable Computers, Naval Postgraduate School, Monterey, CA, September 1983.

Shudde, R.H., Naval Applications: Ten Algorithms for the HP-67 and HP-97 Calculators, Naval Postgraduate School, Monterey, CA, February 1979.

Thompson, E., Design Document for the SURFLOC Computer Program, Analysis & Technology, North Stonington, Ct., 19 September 1980.

Urick, Robert J., Sound Propagation in the Sea, Peninsula Publishing, Los Altos, CA., 1982.

Walpole, Ronald E., and Myers, Raymond H., _Probability and Statistics for Engineers and Scientists_, Macmillan Publishing Company, New York, 1972.

Weinstein, Ehud, _Decentralization of the Gaussian Maximum Likelihood Estimator and Its Application to Passive Array Localization_, Woods Hole Oceanographic Institution, MA, 17 August 1979.