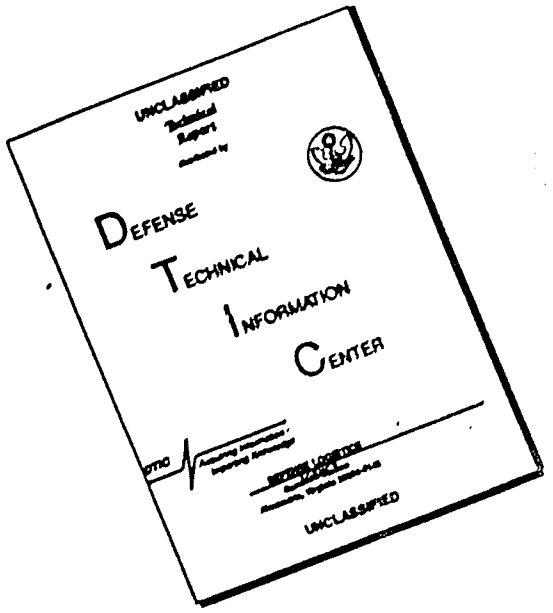
TIMING SCHEDULE OPTIMIZATION FOR EARTH ORBIT

by

John H. Fagan, Jr.
and
David W. Whitlow

S.M. Course XVI June 1965

DISCLAIMER NOTICE



THIS DOCUMENT IS BEST QUALITY AVAILABLE. THE COPY FURNISHED TO DTIC CONTAINED A SIGNIFICANT NUMBER OF PAGES WHICH DO NOT REPRODUCE LEGIBLY.

AD 618571

CLEARINGHOUSE FOR FEDERAL SCIENTIFIC AND TECHNICAL INFORMATION, CFSTI
INPUT SECTION 410.11

LIMITATIONS IN REPRODUCTION QUALITY OF TECHNICAL ABSTRACT BULLETIN
DOCUMENTS, DEFENSE DOCUMENTATION CENTER (DDC)

☒ 1. AVAILABLE ONLY FOR REFERENCE USE AT DDC FIELD SERVICES.
COPY IS NOT AVAILABLE FOR PUBLIC SALE.

☐ 2. AVAILABLE COPY WILL NOT PERMIT FULLY LEGIBLE REPRODUCTION.
REPRODUCTION WILL BE MADE IF REQUESTED BY USERS OF DDC.

☐ A. COPY IS AVAILABLE FOR PUBLIC SALE.

☐ B. COPY IS NOT AVAILABLE FOR PUBLIC SALE.

☐ 3. LIMITED NUMBER OF COPIES CONTAINING COLOR OTHER THAN BLACK
AND WHITE ARE AVAILABLE UNTIL STOCK IS EXHAUSTED. REPRODUCTIONS
WILL BE MADE IN BLACK AND WHITE ONLY.

TSL-121-2/65

DATE PROCESSED: 11 Aug 65

PROCESSOR: R. F. Zimmer

TIMING SCHEDULE OPTIMIZATION

FOR EARTH ORBIT

by

JOHN H. FAGAN, JR.

B. S., United States Military Academy
(1962)

DAVID W. WHITLOW, 1st Lt. USAF

B.S.E.E., North Carolina State College
(1963)

SUBMITTED IN PARTIAL FULFILLMENT

OF THE REQUIREMENTS FOR THE

DEGREE OF MASTER OF SCIENCE

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June, 1965

Signature of Authors ..... John H. Fagan Jr.
David W. W. Whitlow
Department of Aeronautics and
Astronautics, May 1965

Certified by ..... James E. Potter
Thesis Supervisor

Accepted by ..... Leon Trilling
Chairman, Departmental
Graduate Committee

TIMING SCHEDULE OPTIMIZATION

FOR EARTH ORBIT

by

John H. Fagan, Jr.

David W. Whitlow

Submitted to the Department of Aeronautics and Astronautics, Massachusetts Institute of Technology on 21 May, 1965, in partial fulfillment of the requirements for the degree of Master of Science.

ABSTRACT

The location times of thirty measurement points for a spacecraft in a circular, planar earth orbit are varied to minimize a cost function, the sum of the squared components of position uncertainty, at a pre-determined target. In addition, the optimum schedule of horizon references for the star-elevation measurement to be used at each point is determined with respect to the same cost function. A steepest-descent computer program was written to perform the optimization in each case. It is shown that the measurement times collect into four clusters from a nominal schedule in which they are equally spaced. A cost reduction greater than 80% is realized. The horizon-selection procedure defines certain areas along the trajectory where one or the other horizon is preferred. When carried out simultaneously with a time optimization, this procedure results in only a slight improvement over the case where a single horizon is used for each measurement.

Thesis Supervisor: James E. Potter

Title: Assistant Professor of
Aeronautics and Astronautics

ACKNOWLEDGMENTS

The authors wish to express their appreciation and thanks to: Professor James E. Potter for his direction as thesis supervisor, Dr. Richard H. Battin for his aid in selecting the problem, Mr. Peter Kachmar for his assistance in writing the computer program, and Mrs. Sandra Peterson for the conscientious effort which she devoted to typing this thesis.

Lieutenant Whitlow also wishes to express his appreciation and thanks to the United States Air Force for providing the opportunity for graduate study at the Massachusetts Institute of Technology.

Acknowledgment is due the M.I.T. Computation Center for use of the IBM 7094 Computer. The numerical studies were performed as problem M4052.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

LIST OF ILLUSTRATIONS (continued)

LIST OF TABLES

LIST OF SYMBOLS

General Notation Rules

A capital letter indicates a matrix unless otherwise noted. Standard matrix notation is used.

An underlined lower-case letter indicates a vector, which is equivalent to a one-column matrix. A vector symbol without the underlining indicates the magnitude of the vector.

A bar over a symbol signifies the average value of the quantity represented by the symbol. A bar over a group of symbols signifies the average value of the product of the quantities represented by the symbols.

CHAPTER 1

INTRODUCTION

The optimization of a celestial measurement schedule for a manned space mission has become a subject of great importance in the last few years. Walter F. Denham and Jason L. Speyer of Raytheon Company considered this problem in a recent report (Reference 5). They sought to minimize the position estimation uncertainty at the terminal point of a free-fall mission by comparing various sequences of star and star-horizon measurements. A steepest-descent numerical procedure was used to obtain the optimization. The authors' results showed a 10% improvement over a schedule earlier proposed by Richard H. Battin, of the MIT Instrumentation Lab. In this and similar studies, the locations in time for the various measurements were held fixed and spaced at nearly equal intervals. The purpose of this thesis is to investigate the behavior of such a nominal time schedule as the measurement times are changed to decrease terminal position uncertainty. The model used is a planar, circular earth orbit where the target point appears in the first revolution. Only one type of measurement is considered, the star-elevation angle, and use of both horizons is investigated. In addition, the time optimization problem is coupled with a horizon-selection procedure, to compare with the single-horizon mode.

It is expected that there are certain preferred measurement positions along the circular path. If the times of the measurements are free to change, they should eventually cluster about these points in order to effect a reduction in position uncertainty. A similar hypothesis can be stated for the horizon selection. It is probable that there are certain areas along the planar trajectory where it would be more beneficial to use one horizon instead of the other. The steepest-descent procedure will be used to determine the optimum schedules in both cases.

CHAPTER 2

STATEMENT OF THE PROBLEM

The objective of changing the measurement times is to decrease the position uncertainty at the target. The expression for the covariance matrix of estimation errors is developed in Appendix B.

$$E_a^{-1} = A = (C_{ao} E_o C_{ao}^T)^{-1} + \sum_{k=1}^{N} \frac{(C_{ak}^T)^{-1} \underline{\beta}_k \underline{\beta}_k^T C_{ak}^{-1}}{\sigma^2} \qquad (2\text{-}1)$$

Where:

E_a = estimation error covariance matrix at target

E_o = initial estimation error covariance matrix

C = state transition matrix (Appendix C)

$\underline{\beta}$ = measurement vector (Appendix A)

σ^2 = variance of measurement error

Certain assumptions are made in stating the problem which simplify the elements of the above equation, making it easier to manipulate.

As stated in Chapter 1, a planar, circular earth orbit is assumed for the spacecraft. The in-plane navigational problem can be considered alone since, as is shown by Stern (Ref. 8) and others, the in-and out-of-plane error propagations are uncoupled. The inherent simplicity of the circular orbit is especially obvious in the reduction of Stern's formula for the transition matrix (Appendix C)

to a less complicated form. As explained in Appendix C, the local vertical co-ordinate system was chosen to coincide with Stern's equations.

It was decided to select a target point in the first orbit so that the resultant time changes would be more clearly defined. The entire trajectory is included in a central angle of 290°, and the zero-angle reference is arbitrary.

The star-elevation measurement is a reasonable selection since it has been found to be superior in the vicinity of a planet (Chap. 8, Ref. 1). Also, the characteristic vector of the measurement, developed in Appendix A, has the same simple form at all points in the trajectory, when expressed in local-vertical co-ordinates.

As implied in equation (2-1), the variance of the measurement error is assumed constant for all measurements. This seems reasonable enough, since the type of measurement is the same each time and it is always taken at the same altitude. In order to provide ample spacing for an adequate sample of measurement points, the altitude chosen for the problem is 11,000 miles. From this altitude, an optical instrument can be expected to be about a mile in error in discerning the horizon. Considering an error of about .8 miles, the angular variation, as shown in Figure 2.1, is given in equation (2-2).

$$\sin \delta_B \approx \delta_B = \frac{.8}{10^3\sqrt{(15)^2-(4)^2}}$$

$$\delta_B = .554 \text{ mr} \qquad (2\text{-}2)$$

Expressed in arc-seconds, this value is about 3.5×10^{-3} seconds. Hence, the variance used in the problem, assuming the mean of measure-

ment errors is zero, is 12×10^{-8} sec^2.

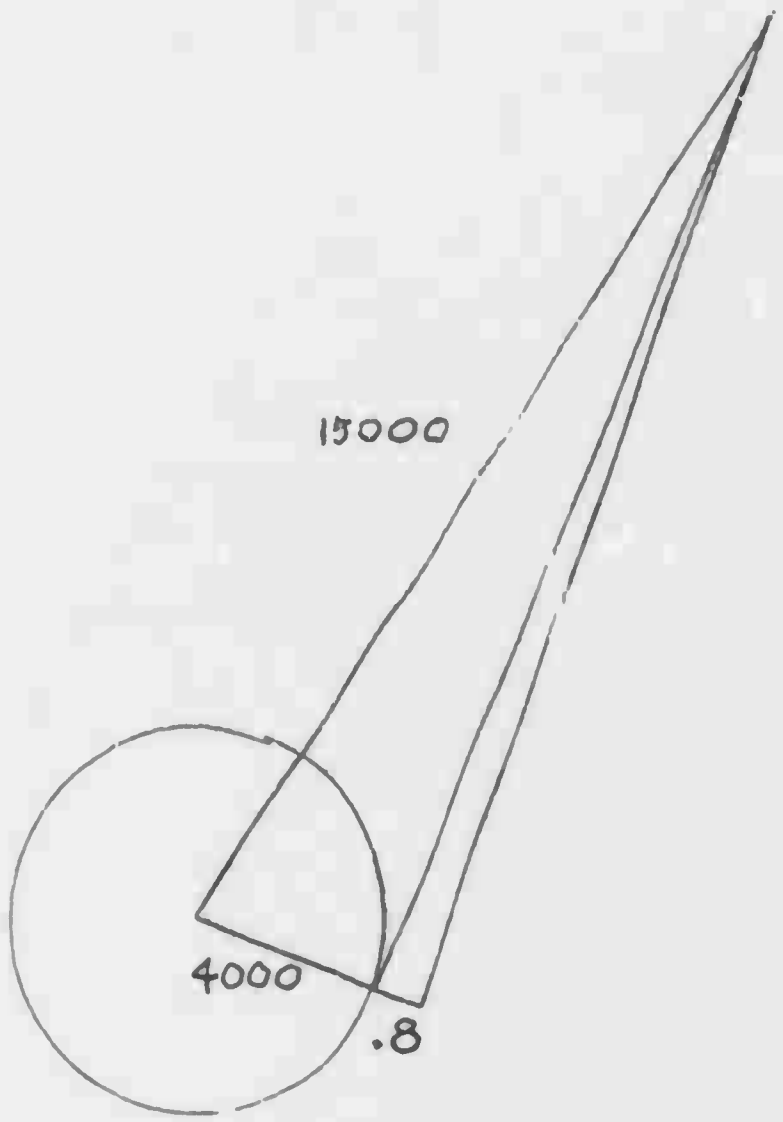


Figure 2.1

The quantities for the initial estimation errors are chosen to be five miles and ten miles per second in position and velocity in each co-ordinate direction. These errors are assumed uncorrelated so that the initial estimation error covariance matrix is diagonal.

Since the quantity to be optimized is the position uncertainty,

only the first two diagonal elements of the 4 X 4 E_a matrix are considered. A convenient way to write this cost function is given in equation (2-3).

$$\text{Cost} = \text{tr}\left[Q\, E_a\right]$$

$$Q = \begin{bmatrix} I_2 & 0_2 \\ 0_2 & 0_2 \end{bmatrix} \qquad (2\text{-}3)$$

A more sophisticated cost function for a manned mission might be a weighted average of the target position and velocity errors, such as that used by Denham and Speyer. This would imply a different Q than that used above. Another possible cost function is the determinant of the E_a matrix, describing the volume of the target error ellipsoid.

Stated briefly, the problem is to find the time schedule, out of all possible schedules of thirty measurements, that minimizes the cost function given in equation (2-3). The nominal schedule has thirty measurement points, spaced at an interval of about 900 seconds in time, between central angles of zero and 290°. Similarly, the horizon-selection problem seeks to find the sequence of horizon references which minimizes cost. There is a choice between two references at each point. The nominal schedule in this case is the use of the "right" horizon, opposite to the direction of motion, at each point. The method of solution in each case is the steepest-descent numerical procedure, which is the subject of the next chapter.

CHAPTER 3

APPLICATION OF STEEPEST-DESCENT

The steepest-descent, or -ascent, method is one of a number of numerical techniques developed over a century ago by Cauchy and others of that era. The advent of the high-speed computer has brought many such procedures back to life. Largely responsible for the revival of steepest-descent are Kelley and Bryson who, working independently, recognized its superiority in certain classes of problems. It eliminates much of the guesswork associated with other methods by assuming a non-optimal, nominal solution, and proceeding to the optimum by a series of linear, incremental changes. The nominal solution need only be a reasonable first guess and may or may not satisfy the boundary conditions.

An analogy, credited to Bryson, illustrates the method quite well. A hiker, climbing a mountain in a dense fog, will climb where the slope rises the sharpest to minimize his time of ascent. Because of the fog, he must relocate the direction of steepest rise at regular intervals. In equation form, the direction in which he climbs from his starting point is:

$$\underline{r} = \frac{\partial z}{\partial x}\bigg|_{\substack{x=x_1\\y=y_1}} \underline{i} + \frac{\partial z}{\partial y}\bigg|_{\substack{x=x_1\\y=y_1}} \underline{j} = z_{x_1}\,\underline{i} + z_{y_1}\,\underline{j} \qquad (3\text{-}1)$$

where z is the function describing the hill. The horizontal distance moved in a certain direction is directly proportional to the slope in that direction.

$$\Delta x = x_2 - x_1 = Kz_{x_1}$$
$$\Delta y = y_2 - y_1 = Kz_{y_1} \qquad (3\text{-}2)$$

The linearizing assumption is that the total vertical distance climbed equals the sum of the computed vertical distances for the x and y directions.

$$\Delta z \cong z_{x_1} \Delta x + z_{y_1} \Delta y \qquad (3\text{-}3)$$

The climber will decide before he starts how far he will climb vertically before re-assessing the direction. Hence, Δz is a known quantity.

$$\Delta z \cong K \left(z_{x_1}\right)^2 + \left(z_{y_1}\right)^2 \qquad (3\text{-}4)$$

The constant K, which governs the horizontal distance, can then be determined.

$$K = \frac{\Delta z}{\left(z_{x_1}\right)^2 + \left(z_{y_1}\right)^2} \qquad (3\text{-}5)$$

The climber predicts that his new altitude, when he has arrived at point 2, will be $z_1 + \Delta z$. The actual altitude will normally be less than this straight line extrapolation of slope. After determining the new direction of steepest ascent, the climber repeats the procedure until finally, the actual change in altitude is much less than he predicted, indicating he is approaching the top of the moun-

tain.

Two disadvantages of the steepest-ascent technique are brought out in the analogy. The proper step size, Δz, is important because the climber may miss a better path if he climbs too far in any direction. Unfortunately, a reasonable step size can only be selected by a trial and error process. Also, the climber may venture onto an isolated peak and, because of the fog, think he has reached the top. A fresh start with new initial conditions is the only way to effectively reduce the probability of converging on a local maximum.

In References 3 and 4, Bryson has outlined the mathematical approach to a series of general problems. His formulation of a problem without constraints will be considered here since it is somewhat similar to the thesis problem.

A nominal spacecraft trajectory is postulated, which is described by the following set of ordinary differential equations.

$$\frac{dy_i}{dt} = f_i(y, b, t) \qquad i = 1, 2, \ldots, n \qquad (3\text{-}6)$$

The known quantities f_i, are functions of the independent variable t, the dependent variables y_i and the driving, or control, function, b(t). The cost, a function of the dependent variables, is increased (or decreased) by varying b(t). Variations about this nominal trajectory are considered and it is assumed that they can be accurately described by first-order differentials in the perturbation equation.

$$\frac{d}{dt}(\delta y_i) = \sum_{j=1}^{n} \frac{\partial f_i}{\partial y_j} \delta y_j + \frac{\partial f_i}{\partial b} \delta b \qquad (3\text{-}7)$$

The partial derivatives in (3-7) are evaluated along the nominal tra-

jactory. The dependent variables are functions of time so that (3-7) implies a set of n linear equations with variable coefficients. The δy_i terms represent a small variation of the dependent variables from their nominal time history. A set of equations adjoint to (3-7) is defined in equation (3-8).

$$\frac{dL_i}{dt} = - \sum_{j=1}^{n} \frac{\partial f_j}{\partial y_i} L_j \qquad (3\text{-}8)$$

The partial derivative in (3-8) is the negative transpose of the similar quantity in equation (3-7). The reason for the adjoint equation is made clear in the following sequence:

$$\sum_{i=1}^{n} \left(L_i \frac{d}{dt}(\delta y_i) + \delta y_i \frac{dL_i}{dt}\right) = \sum_{i=1}^{n} L_i \frac{\partial f_i}{\partial b} \delta b$$

$$+ \sum_{i=1}^{n} \sum_{j=1}^{n} \left(L_i \frac{\partial f_i}{\partial y_j} \delta y_j - L_j \frac{\partial f_j}{\partial y_i} \delta y_i\right) \qquad (3\text{-}9)$$

The double summation term in (3-9) equals zero since only the indices differ. The left hand side of the equation is equivalent to the time derivative of $L_i \delta y_i$.

$$\frac{d}{dt} \sum_{i=1}^{n} L_i \delta y_i = \sum_{i=1}^{n} L_i \frac{\partial f_i}{\partial b} \delta b \qquad (3\text{-}10)$$

The expression which relates incremental changes in the control variables to the resulting changes in the dependent variables is obtained by integrating equation (3-10) over the flight time.

$$\sum_{i=1}^{n} L_i \, \delta y_i \Big|_{t_o}^{T} = \int_{t_o}^{T} \sum_{i=1}^{n} L_i \, \frac{\partial f_i}{\partial b} \, \delta b(t) dt \qquad (3\text{-}11)$$

$$L_b(t) \triangleq \sum_{i=1}^{n} L_i \, \frac{\partial f_i}{\partial b}$$

$$\sum_{i=1}^{n} L_i \, \delta y_i \Big|_{t_o}^{T} = \int_{t_o}^{T} L_b(t) \, \delta b(t) dt$$

The quantity $L_b(t)$, defined in (3-11), is the influence function associated with the control function, b(t). The definition of the adjoint variable L_i, defined in equation (3-8), is justified by this simple expression. L_i is a known function of the nominal trajectory and its boundary condition is a function of the cost, which is usually determined at the terminal point of the flight.

$$L_i(T) = \frac{\partial \, \text{Cost}}{\partial \, y_i}\Big|_{t=T} \qquad \text{Cost} = \text{Cost}\left[y(T)\right] \qquad (3\text{-}12)$$

The objective is to relate changes in cost to changes in the control function by the use of the adjoint variable. By definition, the differential cost change is a sum of partial derivatives. Using equation (3-12):

$$\delta \text{Cost}\Big|_{t=T} = \left[\sum_{i=1}^{n} \frac{\partial \text{Cost}}{\partial y_i}\,\delta y_i\right]_{t=T} = \left[\sum_{i=1}^{n} L_i\,\delta y_i\right]_{t=T} \qquad (3\text{-}13)$$

Substituting (3-13) into (3-11):

$$\delta \text{Cost} = \int_{t_o}^{T} L_b(t)\,\delta b(t)dt + \left[\sum_{i=1}^{n} L_i\,\delta y_i\right]_{t=t_o} \qquad (3\text{-}14)$$

The adjoint variables L_i, can be interpreted as the influence functions for the initial conditions of the dependent variables. The δ Cost term in equation (3-14) is pre-selected. For a given value of δ Cost, it is desirable to require the smallest possible changes in the driving function and initial conditions so that the linear perturbation equation is valid. Stated another way, the problem is to minimize the effect of the second-order $\delta b(t)$ and δy_i terms for a constant cost change. The summation term in equation (3-14) can be rewritten as a dot product to simplify the mathematics:

$$\left[\sum_{i=1}^{n} L_i\,\delta y_i\right]_{t=t_o} = \underline{L}_o \cdot \delta \underline{Y}_o \qquad (3\text{-}15)$$

where

$$\underline{L}_o = \begin{bmatrix} L_1 \\ L_2 \\ \cdot \\ \cdot \\ \cdot \\ L_n \end{bmatrix} \qquad \underline{Y}_o = \begin{bmatrix} y_1 \\ y_2 \\ \cdot \\ \cdot \\ \cdot \\ y_n \end{bmatrix}$$

If the i^{th} initial condition is specified, δy_i is zero and the i^{th} term does not contribute to the dot product. The variational calculus problem can then be stated as follows:

$$\text{minimize:} \quad J = \int_{t_o}^{T} \left|\delta b(t)\right|^2 dt + \alpha \left|\delta \underline{\chi}_o\right|^2 \tag{3-16}$$

$$\text{subject to:} \quad \delta \text{Cost} = A = \text{constant}$$

The positive constant α in (3-16) is chosen to make the dimensions compatible in the J expression. The problem can be rewritten using Lagrange multipliers;

$$J' = J + \Lambda (\delta \text{Cost} - A) \tag{3-17}$$

where J' is the quantity to be minimized. Substituting from equations (3-14) and (3-15):

$$J' = \int_{t_o}^{T} \left[\Lambda L_b(t)\, \delta b(t) + \left|\delta b(t)\right|^2\right] dt \tag{3-18}$$

$$+ \Lambda \underline{L}_o \cdot \delta \underline{\chi}_o + \alpha \left|\delta \underline{\chi}_o\right|^2 - \Lambda A$$

J' can be divided into three parts, a function of $\delta b(t)$, a function of $\delta \underline{\chi}_o$ and a constant.

$$J' = \int_{t_o}^{T} F_1 \left[\delta b(t)\right] dt + F_2(\delta \underline{\chi}_o) + \text{constant} \tag{3-19}$$

To minimize J':

$$\frac{\partial F_1}{\partial \delta b(t)} = \lambda L_b(t) + 2 \delta b(t) = 0 \qquad (3\text{-}20)$$

$$\frac{\partial F_2}{\partial \delta \underline{Y}_o} = \lambda \underline{L}_o + 2\alpha \delta \underline{Y}_o = 0 \qquad (3\text{-}21)$$

The second derivatives in (3-20) and (3-21) are both positive and a minimum for J' is assured. These equations show that the smallest changes in the driving function and the initial conditions to result in a given cost increment are changes proportional to their respective influence coefficients.

$$\delta b(t) = -\frac{\lambda}{2} L_b(t) = K L_b(t) \qquad (3\text{-}22)$$

$$\delta \underline{Y}_o = -\frac{\lambda}{2\alpha} \underline{L}_o = \frac{K}{\alpha} \underline{L}_o \qquad (3\text{-}23)$$

The sign of the constant K is chosen positive or negative for a desired cost increase or decrease. Substituting equations (3-22) and (3-23) into (3-14) results in the cost expression as a function of K.

$$\delta \text{Cost} = K \int_{t_o}^{T} \left[L_b(t)\right]^2 dt + \frac{K}{\alpha} \left|\underline{L}_o\right|^2 \qquad (3\text{-}24)$$

Since δ Cost is pre-selected and $L_b(t)$ and $\underline{L}_o$ are known functions defined by equations (3-11) and (3-15) the unknown K is determined by equation (3-25).

$$K = \frac{\delta \text{ Cost}}{\int_{t_o}^{T} \left[L_b(t)\right]^2 dt + \frac{\left|\underline{L}_o\right|^2}{\alpha}} \qquad (3\text{-}25)$$

The influence coefficients, $L_b(t)$ and $\underline{L}_o$, determine the nature of the changes, and K determines the direction and magnitude.

$$b(t)_{NEW} = b(t)_{OLD} + K\, L_b(t) \tag{3-26}$$

$$\underline{Y}_{o_{NEW}} = \underline{Y}_{o_{OLD}} + \frac{K}{\alpha}\,\underline{L}_o \tag{3-27}$$

As mentioned previously, only the unspecified initial conditions are available for change.

The general procedure can be summarized as follows:

(i) A reasonable first estimate of b(t) is chosen, according to the particular problem.

(ii) The partial derivatives of the known functions f_i with respect to the dependent variables and the control function are evaluated along the nominal trajectory.

(iii) The adjoint variables L_i are determined from equation (3-8), integrating backward over the nominal trajectory with equation (3-12) as initial conditions. The influence function $L_b(t)$ can then be computed from equation (3-11).

(iv) An arbitrary cost change is chosen, depending on the nature of the problem. A value of 5 to 10% might be a reasonable initial value for δ Cost if a substantial overall cost change is anticipated. K is then determined from equation (3-25).

(v) The new control function is found from equation (3-26) and the new initial conditions from (3-27). Equations

(3-6) are then integrated to obtain the new trajectory and the process is repeated.

(vi) The ratio of the predicted cost change to the actual cost change will increase as the optimum is approached. When this ratio becomes greater than about 5, the values of δ Cost and/or K should be decreased to reduce step size. In this way, the optimum can be approached as closely as desired.

The essential part of this general formulation is, of course, relating control function changes to the resultant cost change. The adjoint variables were necessary to obtain such an expression because a direct relation between δ Cost and δ b(t) did not exist. In the thesis problem, the estimation error covariance matrices, E_k, are analogous to the f_i in the general formulation. For the time selection procedure, the times of the measurement points, t_k, correspond to the driving function b(t). The schedule of measurement vectors, g_k, is the driving function for horizon selection. The cost function, explained in Chapter 2, is the same for both cases. If it can be expressed as an explicit function of t_k and, for the other case, g_k, the adjoint equations, defined in (3-8), will not be needed. Instead, the influence functions for both cases would be defined by the following equations:

$$\delta \text{Cost} = \sum_{k=1}^{N} \frac{\partial \text{Cost}}{\partial t_k} \delta t_k = \sum_{k=1}^{N} L_k \delta t_k \qquad (3\text{-}28)$$

$$\delta \text{Cost} \quad \sum_{k=1}^{N} \left(\frac{\partial \text{Cost}}{\partial \xi_k}\right)^T \delta \xi_k = \sum_{k=1}^{N} \underline{L}_k^T \delta \xi_k \qquad (3\text{-}29)$$

Equations (3-28) and (3-29) are both analogous to equations (3-14). The initial conditions in the thesis problem are specified so that the term corresponding to the rightmost term in (3-14) is zero. Also, the summations are used since the driving functions, unlike b(t), are not continuous functions. Using the cost function explained in Chapter 2, and the E_a expression developed in Appendix B, the influence coefficients for the two cases are derived in Appendicen D and E.

Time selection:

$$L_k = -\frac{2}{\sigma^2} \xi_k^T C_{ak}^{-1} E_a Q E_a \frac{\partial (C_{ak}^T)^{-1}}{\partial t_k} \xi_k \qquad (D\text{-}13)$$

Horizon selection:

$$\underline{L}_k = -\frac{2}{\sigma^2} \xi_k^T C_{ak}^{-1} E_a Q E_a (C_{ak}^T)^{-1} \qquad (E\text{-}17)$$

For the time optimization problem, the N state-transition matrices are evaluated from the nominal schedule using equation (C-4). Due to the symplectic properties of C, the inverse can be found using the elements of C. Since C is a function of time, the determination of its derivative is straightforward. These results are given in equations (C-7) and (C-5). The estimation error covariance matrix is then computed from equation (B-26). Using these quantities, and the measurement vector determined in Appendix A, the time selection influence coefficient is obtained from equation (D-13). Since

a decrease in cost is desired, the time change at each point is opposite in sign from the corresponding influence coefficient. Instead of specifying a specific cost change as in Bryson's formulation, it is more convenient to first specify a maximum time increment. If the resultant cost change is too small, the time increment can be increased until a change greater than 5% is obtained. Of course, as the optimum is approached, a smaller percentage change is required. The time changes are scaled according to the size of their respective influence coefficients as in equation (3-22), so that the measurement point having the greatest effect on cost is changed the most. The change follows the direction of steepest descent in an N-dimensional space. The intricacies of this procedure are clarified by the flow charts in the next chapter.

The horizon-selection procedure is similar up to a point. Since it is carried out simultaneously with the time-optimization procedure, E_a and C_{ak} must be re-evaluated after each iteration. There is not as much control in this problem however, since there are only two possible values for δg_k at each point. The N individual elements of equation (3-29) must be examined to determine the incremental cost changes. If a proposed horizon change results in a decrease in cost, the change is made. If not, the original horizon is retained. With so little control, it is possible that the proposed change violates the assumption of linearity utilized in the perturbation equation. This problem is discussed in Chapter 5. The procedure for horizon selection is also illustrated in Chapter 4.

CHAPTER 4

COMPUTER SOLUTION

The specifics of the computer program used to implement the theory developed in Chapters 2 and 3 and Appendices A thru E are covered in Appendix F. However, it will be useful to understand how the problem solution is carried out. The flow charts in Figures 4.1-5 will help in understanding the methods used.

Figure 4.1 gives the flow chart for Block One (no horizon change). Here the new measurement time schedule is computed using the same measurement vector, either left or right horizon.

The input data needed is covered in Appendix F. From equation (A-14), the measurement vectors can be computed.

$$\underline{g}(\text{RIGHT}) = \begin{bmatrix} \dfrac{r_E}{z(z^2 - r_E^2)^{\frac{1}{2}}} \\ -\dfrac{1}{z} \\ 0 \\ 0 \end{bmatrix} \tag{4.1}$$

$$\underline{g}(\text{LEFT}) = \begin{bmatrix} \dfrac{r_E}{z(z^2 - r_E^2)^{\frac{1}{2}}} \\ \dfrac{1}{z} \\ 0 \\ 0 \end{bmatrix} \tag{4.2}$$

Using these vectors and the state transition matrix equation (C-4) the target estimation error covariance matrix, equation (B-26) can be computed.

As shown in Appendix B, the symplectic property of the state transition matrix allows one to simply compute the matrix inverse and matrix transpose inverse by rearranging the elements. Equation (C-7) is computed in a simple subroutine.

The matrix Q is covered in Appendix D. Now, the cost computed will be the following:

$$\text{Cost} = \text{tr}\left[QE_a\right] \qquad \text{(D-2)}$$

The cost will be designated the old cost, oc, when the computation uses a measurement time schedule which is either the initial one or a result of a previous iteration.

Using the present target estimation error covariance matrix and equation (D-13) the influence coefficients are computed. The logic used will change the measurement time schedule by an amount depending on the influence coefficient having the largest magnitude. Now, if we define a scale factor, sf, as:

$$sf = \frac{|\text{maximum influence coefficient}|}{\text{maximum time increment}}$$

or

$$sf = \frac{|EC_m|}{\Delta t_m} \qquad \text{(4-3)}$$

then the new measurement time depends on the old measurement time and the value of the influence coefficient at the old measurement

time. Now:

$$\text{new time} = \text{old time} - \frac{\text{influence coefficient}}{\text{scale factor}}$$

or

$$t_{i_n} = t_{i_o} - \frac{EC_i}{sf} \qquad (4\text{-}4)$$

and this procedure is applied to all the measurement times. Obviously the time with the influence coefficient having the largest magnitude will have the greatest change. Also the sign of the influence coefficient will determine which way the measurement time will move.

After all the measurement times have been changed, a new target estimation error covariance matrix and a new cost, nc, can be computed. The actual cost change, acc, is clearly:

$$acc = oc - nc \qquad (4\text{-}5)$$

and this number should be positive. A predicted cost change, pcc, can be defined as:

$$pcc = \sum_{i=1}^{30} EC_i\ (\Delta t_i) = \sum_{i=1}^{30} EC_i (t_{i_o} - t_{i_n}) \qquad (4\text{-}6)$$

and then it is compared to acc. The percentage cost change is then:

$$pc = \frac{acc}{oc} \qquad (4\text{-}7)$$

When pc is positive, it is then compared to some minimum desired percent change, mpc. If pc is less than mpc, then the maximum time increment is multiplied by the ratio of mpc to pc.

When pc is zero, the maximum time increment is cut in half. When pc is negative, the new maximum time increment is changed by an amount depending on how much negative it is. The logic is then

if pc + mpc is zero, then:

$$\Delta t_m = \frac{\Delta t_m}{2} \tag{4-8}$$

if pc + mpc is greater than zero then:

$$\Delta t_m = \Delta t_m(1 - \frac{|pc|}{mpc}) \tag{4-9}$$

and if pc + mpc is less than zero then:

$$\Delta t_m = \Delta t_m \frac{mpc}{pc} \tag{4-10}$$

Having changed the maximum time increment, it may be used to repeat the measurement time change procedure shown previously. The same influence coefficients are used since they were calculated before any measurement time change was made. The same procedure of selecting a scale factor, and of then changing each measurement time is carried out. This procedure may continue until pc is equal to or greater than mpc. Any additional changes made will be added to those previously made. Since the logic does not return to the measurement time schedule used to compute the influence coefficients, the optimum may be missed, much as the hill climber in Chapter 3 missed a better path by climbing too far in one direction. The initial time increment may have forced the optimum over the top of the hill and any further change in this time increment will merely cause changes that will put the optimum further over the top. The program may then run into trouble. When this happens, the best procedure to follow is to return to the measurement time schedule computed before the trouble was encountered and reduce either the maximum time

increment or the minimum desired percent change.

Therefore, the program control is either on the percent change or the maximum time increment. Another possible program control could be the ratio of pcc to acc.

The above procedures are shown better in Figures 4.2 and 4.3. Subroutine CHECK, Figure 4.2, changes only one time. This routine is used right after the influence coefficients are computed. The result shows that the cost does decrease by changing the measurement time, and that the influence coefficients are correct.

Subroutine LOGIC, Figure 4.3, computes the entire new measurement time schedule. Also this routine limits the new times to the end conditions:

$$0 \leq t_i \leq \frac{F_a}{n} \tag{4-11}$$

where F_a is the final angle of the orbit.

This whole procedure can be repeated any number of times. But as mentioned before, after several iterations, it may be impossible to achieve a given mpc. Increasing the maximum time increment may place the optimum over the hill top. Then a new mpc must be chosen and this can only be done in a heuristic manner. However, the results after a set of iterations help determine what size steps must be made to bring the measurement time schedule to an optimum.

Figure 4.4 shows the flow chart for the second block. Here a new measurement time schedule and a new measurement horizon schedule are computed. The main difference lies in the fact that at each measurement time the measurement vector is different, and the results of the program are optimum measurement time and horizon schedules.

Except for using a different measurement vector at each time, the measurement time schedule optimization is the same as in Block One. Also since linearity is assumed superposition holds, and the measurement time and horizon schedule optimization can be carried out independently.

The measurement time schedule optimization is completed first and then using equation (E-17) the horizon influence vectors are computed.

Using equations (E-20) the change in cost is computed. Only if this change in cost is negative, will the total cost be reduced by changing the measurement vector.

To avoid going outside the linear range by making the change in cost too large, only one measurement vector will be changed at a time. This will keep the change in cost small. Therefore, only the measurement time having the negative change in cost with the largest magnitude will have its measurement vector changed.

After this particular measurement vector is changed, the new target estimation error covariance matrix and a new cost are computed. As before:

$$acc = oc - nc \tag{4-5}$$

and the predicted change in cost is defined by equation (4-12):

$$pcc = -\delta \text{ Cost} \tag{4-12}$$

where:

$$\delta \text{ Cost} = -\frac{2}{\sigma^2}\left(\ell_k^T C_{ak}^{-1} E_a Q E_a C_{ak}^{T-1}\right)\delta \ell_k \tag{E-16}$$

The ratio of pcc to acc shows how the change effected the cost.

The new cost can be compared to the old cost to see if change did decrease the cost.

Subroutine JUMP, Figure 4.5 shows the logic used in changing the measurement vector.

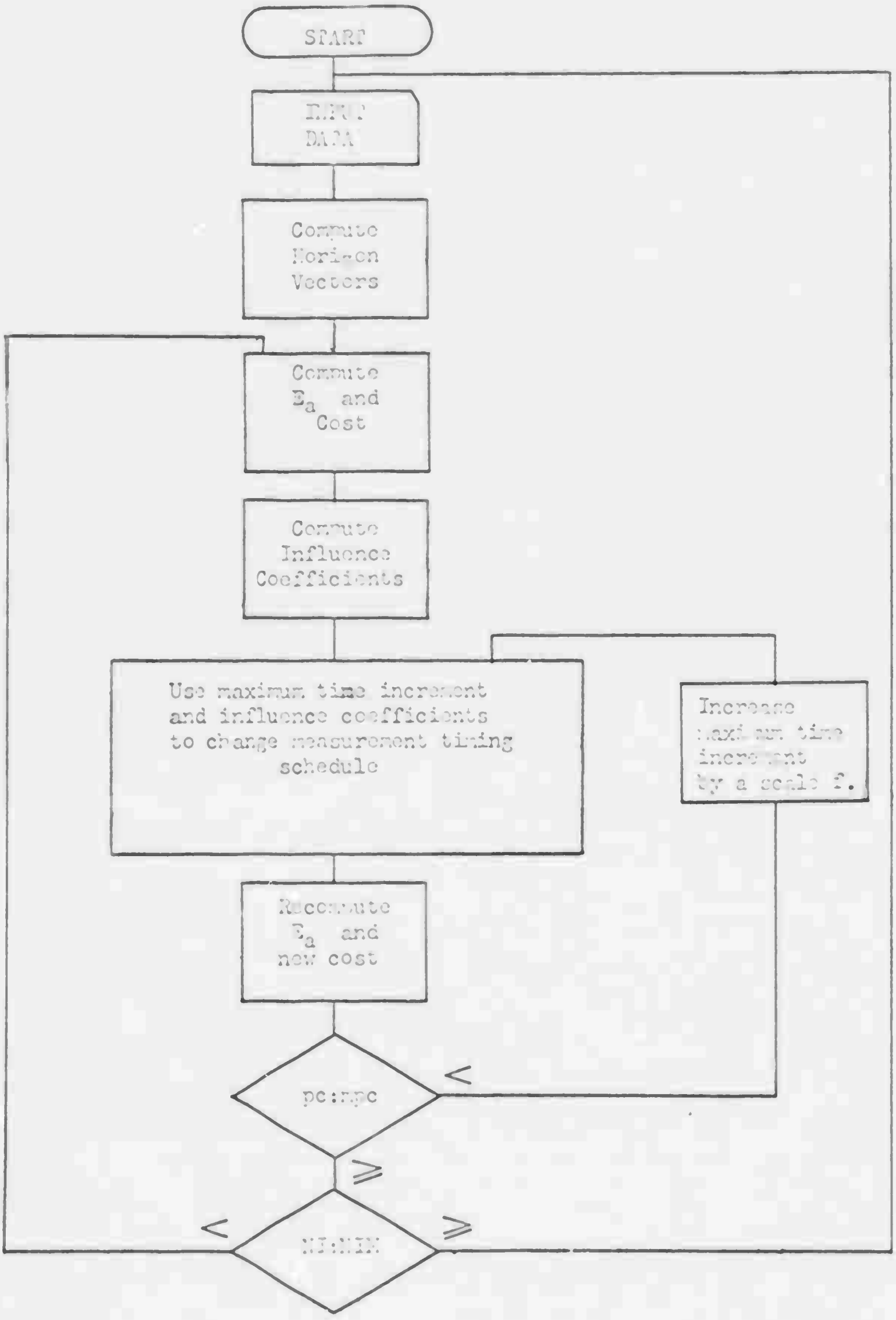


FIGURE 4.1

FLOW CHART BLOCK ONE

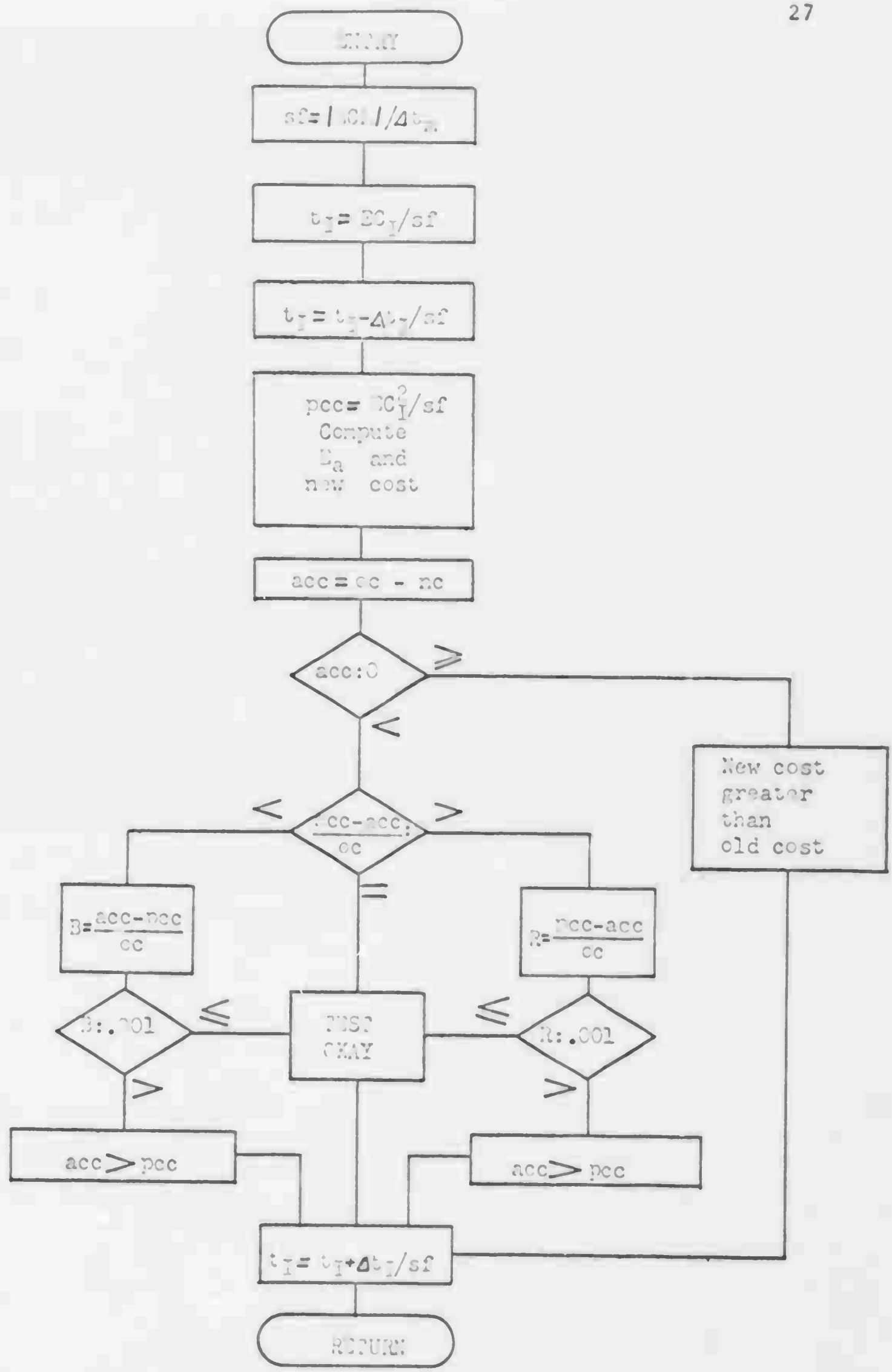


Figure 4.2

SUBROUTINE CHECK

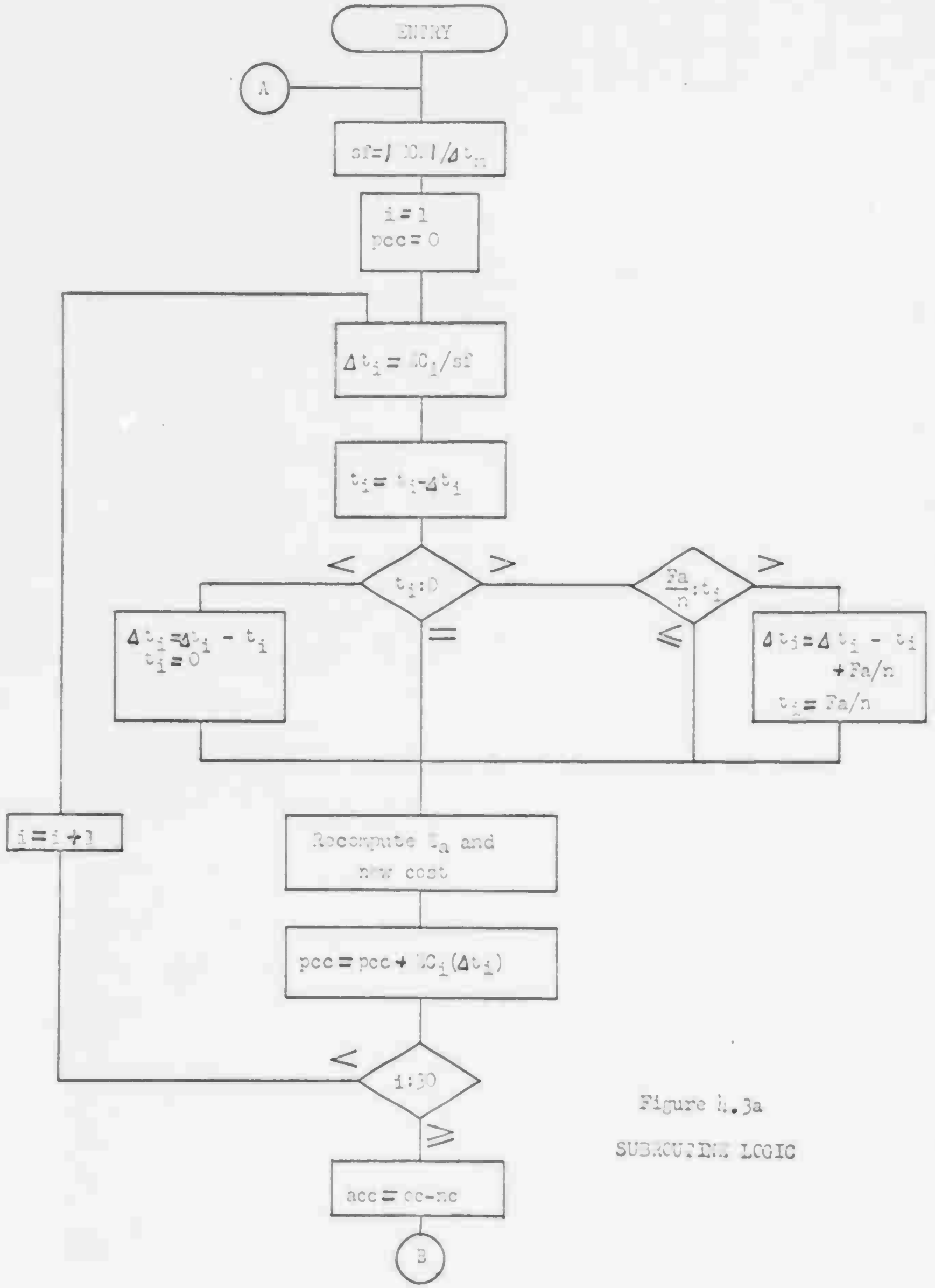


Figure 4.3a

SUBROUTINE LOGIC

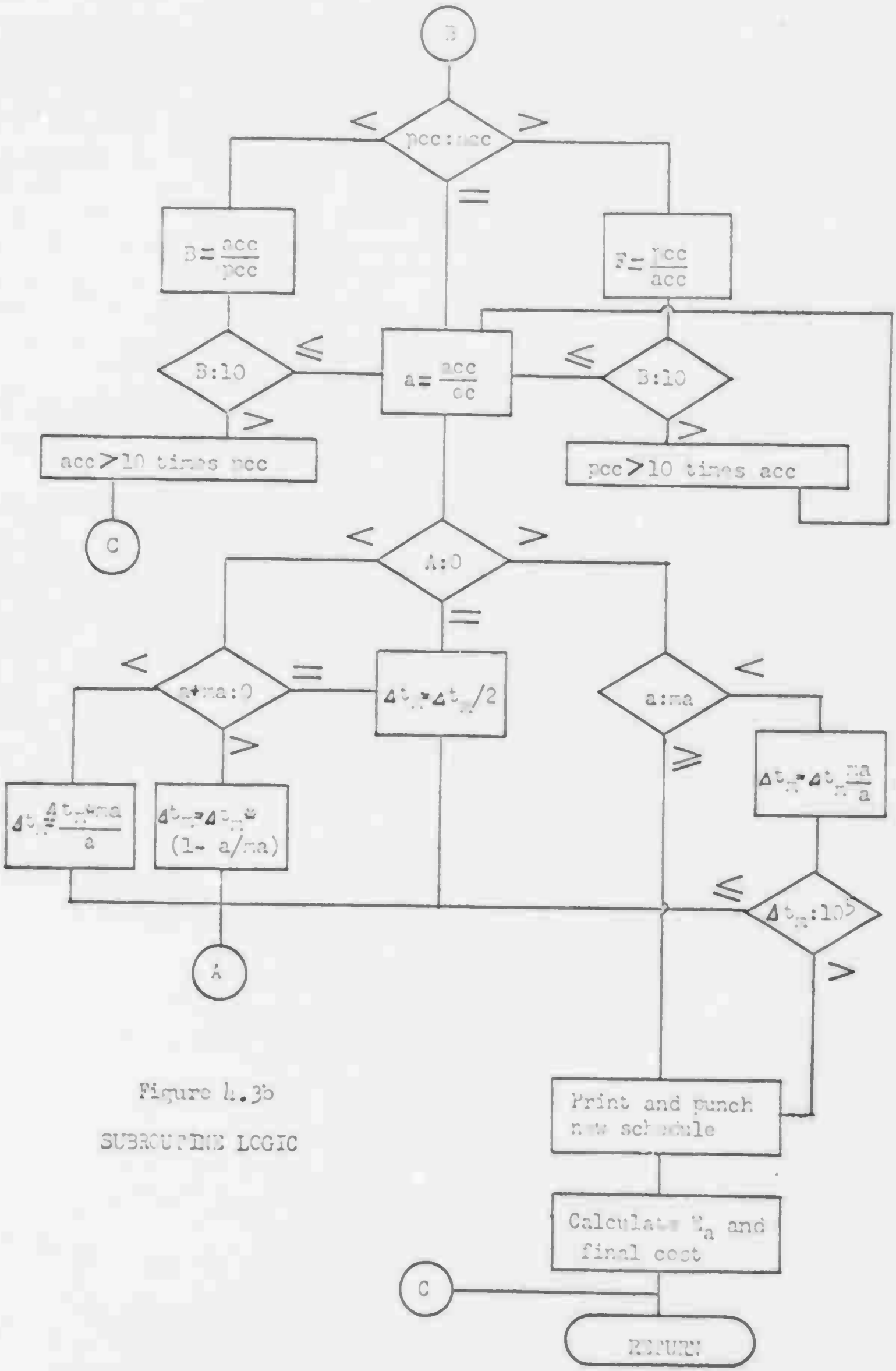


Figure 4.3b

SUBROUTINE LOGIC

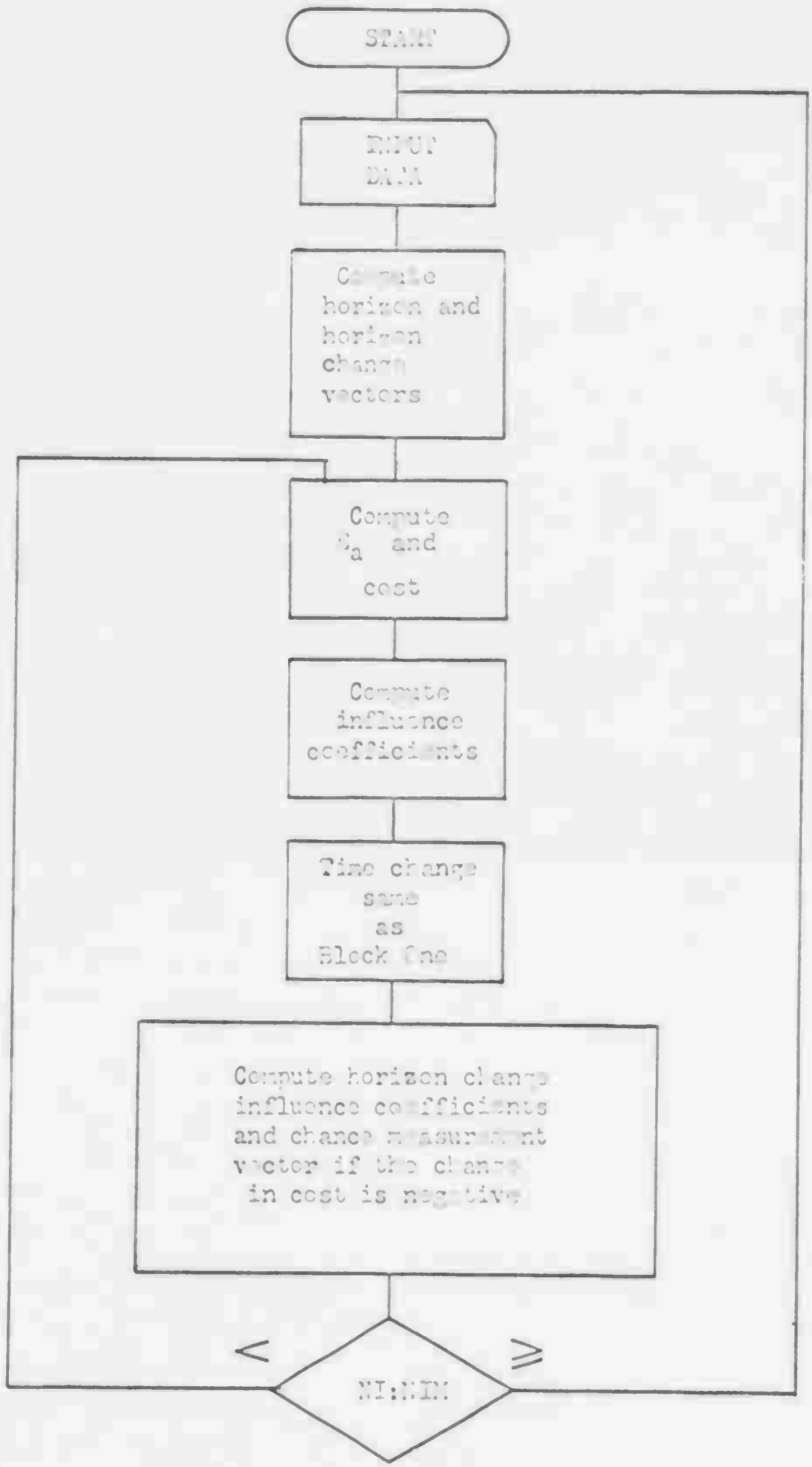


Figure 4.4

BLOCK TWO

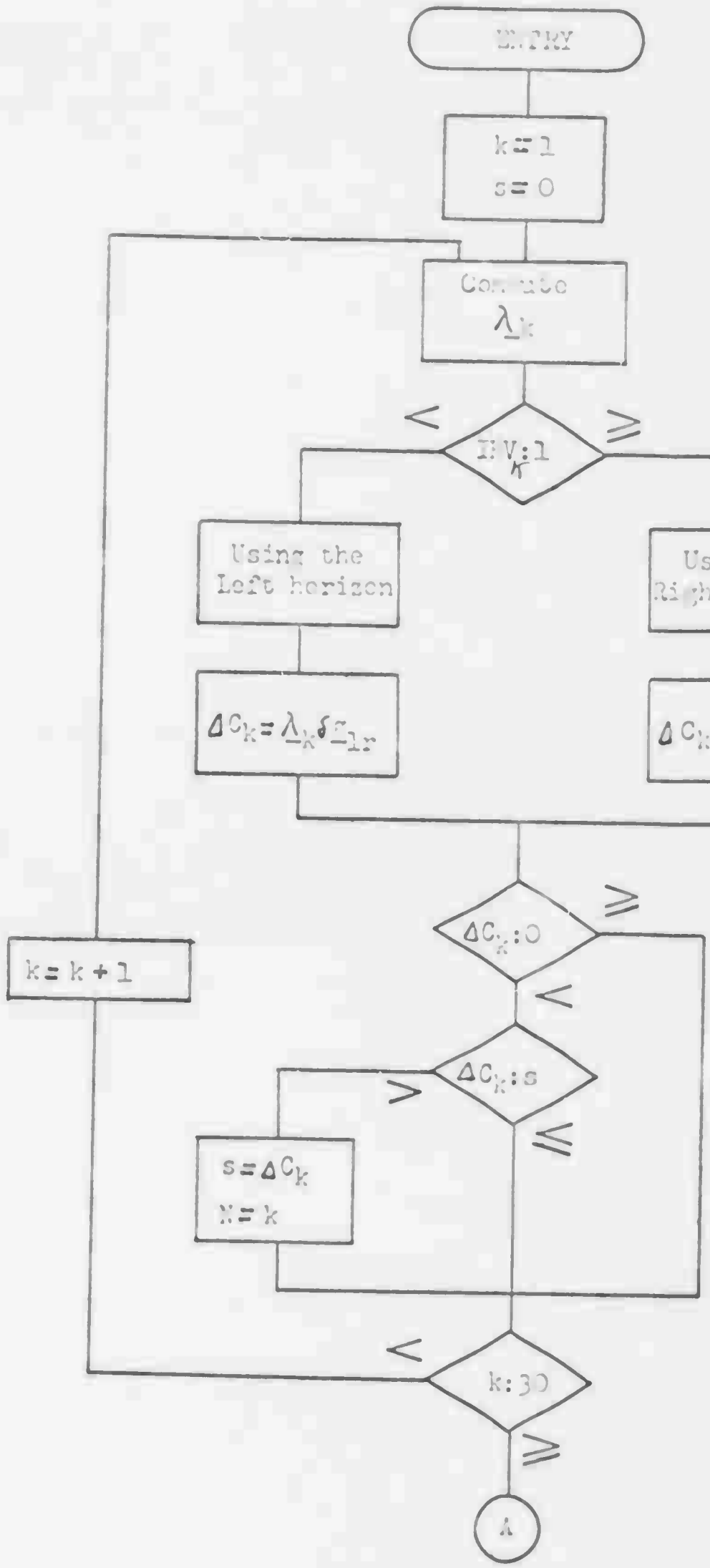


Figure 4.5a
SUBROUTINE JUMP

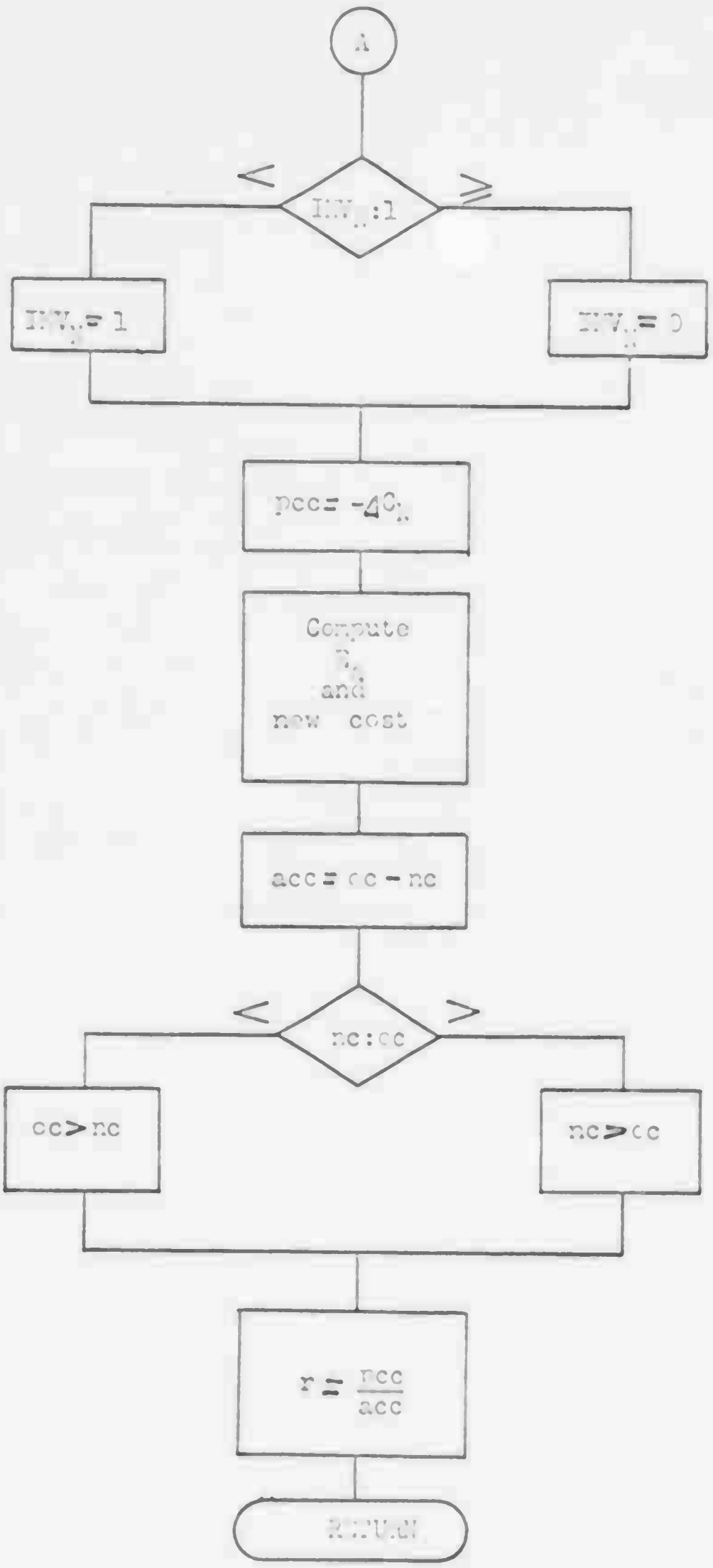


Figure 4.5b

SUBROUTINE JUMP

TABLE 4-1

SYMBOLS FOR FLOW CHARTS

a	ratio of actual cost change to old cost
acc	actual cost change
ΔC_k	cost change due to change in measurement vector
E_a	estimation error covariance matrix at target
EC_i	influence coefficient for measurement time change
ECM	largest value for EC_i
Fa	final angle of the orbit
$\delta \mathcal{L}_{rl}$	measurement vector change from right to left horizon
$\delta \mathcal{L}_{lr}$	measurement vector change from left to right horizon
IHV_{lf}	horizon flag
	= 0 using left horizon
	= 1 using right horizon
mpc	maximum desired percent cost change
n	mean angular motion
nc	new cost
NI	number of iterations
oc	old cost
pc	percentage cost change
pcc	predicted cost change
r	ratio of predicted cost change to actual cost change
sf	scale factor
t_{i_n}	new measurement time
t_{i_o}	old measurement time

Δt_m	maximum time increment
Δt_i	actual time increment
$\underline{\lambda}_k$	influence vector = $\underline{L}_k$

Subscripts:

i, k	measurement times
I	one particular measurement time
N	measurement time having the a - C_k with the largest magnitude

CHAPTER 5

RESULTS AND CONCLUSIONS

As mentioned in Chapter 2, the hypothesis associated with the time optimization problem is that, in a schedule of thirty equally spaced measurement points, there are a certain number of preferred positions. Measurements made at or near these positions should result in a lower cost, the sum of the squares of terminal position uncertainty, than measurements made at other points along the trajectory. If the times of the various points are allowed to change to effect a cost decrease, they should cluster about the preferred, or optimum, points. The method of steepest descent is particularly applicable to this type of problem since the relative size of the thirty influence coefficients indicates the sensitivity of their corresponding points. The time changes are proportional and of opposite sign from their respective coefficients, so that a relatively large value for L_k indicates that a substantial time change should be made in a specific direction. If the optimum points are well-defined, their position on the trajectory should not change appreciably as the times of the measurements are changed. Therefore, a plot of the influence coefficients as a function of the corresponding central angles for each case should serve to locate them. These

influence functions are plotted in Figures 5-1 and 5-2 for the cost values listed in Table 5-1.

	NOMINAL	CHANGE = 25%	CHANGE = 50%
RIGHT	$.6405 \times 10^9$	$.4686 \times 10^9$	$.3221 \times 10^9$
LEFT	1.1628×10^9	$.8432 \times 10^9$	$.5732 \times 10^9$

COST VALUES - FT^2

TABLE 5-1

The "RIGHT" horizon is that opposite to the direction of motion, as defined in Appendix A. It is evident from Table 5-1 that, for a single horizon reference, the right horizon is preferable. The angular dispersion of the measurement points corresponding to the above cost values are more clearly shown in the polar plots, Figures 5-3 through 5-7. The general configurations of the influence functions in Figures 5-1 and 5-2 remain the same, even after substantial changes in cost. The increase in amplitude indicates that the times are driven harder toward the optimum as the optimum is approached. The arrows in both figures indicate the direction of time change, and serve to define the circled stable points. It appears that four clusters should result, two at the end points and two in the middle. The cluster locations predicted from Figures 5-1 and 5-2 are given in Table 5-2.

	1st	2nd	3rd	4th
RIGHT	0°	58.5°	170°	290°
LEFT	0°	70°	176°	290°

CLUSTER PREDICTIONS

TABLE 5-2

For a total cost change greater than 50%, the dispersion of the points is not sufficient to accurately identify the zero-crossings in an influence function plot. Subsequent iterations were carried out, periodically decreasing the maximum time increment and required percent change, until the clusters were clearly defined. The angular dispersion for a cost change of about 75% is shown in Figures 5-8 and 5-9. At this point, it is obvious that there will be only four clusters. The influence coefficients at this stage tend to drive a number of the times beyond the end points. As noted in Chapter 4, the measurement positions are constrained once they reach 0° and 290°, and the large end-point influence coefficients are ignored in computing the predicted cost change for each iteration.

Accurate identification of the position of the clusters was not possible until after several iterations requiring a .1% cost decrease or less. The size of the clusters cannot be predicted since, in the early iterations, the program drove the time locations quite hard until a substantial cost decrease was realized. There were several instances of points "jumping" from one cluster to another. It is reasonable to assume that a tighter tolerance on the maximum time increment would result in different cluster sizes. The loose

tolerance was used to shorten the convergence time.

When the program had changed the times as much as possible, the most likely cluster positions were chosen, and all points were assigned one of these four time values. The cost function values resulting from the selected solution show only a slight decrease from the computer solution. The results are listed in Table 5-3.

	COST COMPUTER	COST SELECTED	% CHANGE
RIGHT	$.1108 \times 10^9$	$.1108 \times 10^9$	82.7%
LEFT	$.2162 \times 10^9$	$.2161 \times 10^9$	81.4%

FINAL COST VALUES

TABLE 5-3

The final angular positions and numbers of included points for the clusters are given in Table 5-4, along with the positions predicted from Figures 5-1 and 5-2.

	RIGHT			LEFT		
	Included Points	Angle	Pred.	Included Points	Angle	Pred.
1st	2	0°	0°	3	0°	0°
2nd	11	69.3°	58.5°	11	75.4°	70°
3rd	7	208.1°	170°	10	197.2°	176°
4th	10	290°	290°	6	290°	290°

FINAL CLUSTER POSITIONS

TABLE 5-4

The angular position of the clusters is more clearly shown in Figures

5-10 and 5-11. The influence function plots provide a fair prediction of the number and position of the clusters even before any cost reduction is obtained.

The selected cluster positions can be justified only if the influence coefficients for these points approach zero, indicating that there are no further changes to be made. The influence coefficients for the end points are still quite large but, as noted, they tend to drive the times beyond the constraints. The sign of the coefficients is positive at 0° and negative at 290°, so that the times are being driven in the proper direction. The values of the coefficients for the middle two clusters are compared with the values for points close to the cluster position in Table 5-5.

		ON CLUSTER	OFF CLUSTER	ANGULAR DIFFERENCE
RIGHT	2nd	+ 4.40	- 68.0	2.5°
	3rd	- 6.36	+ 69.3	3.4°
LEFT	2nd	-21.32	-173.5	3.9°
	3rd	+ 5.59	- 22.0	4.6°

INFLUENCE COEFFICIENTS - FT^2/SEC

TABLE 5-5

It is evident that the optimum positions have been closely approximated.

The cost reductions in each case will be more meaningful if compared in terms of position uncertainty in the radial and tangential directions. As noted in Chapter 2, the initial estimation error

was chosen to be about five miles in each direction. The corresponding values after making thirty measurements for the nominal and optimum solutions are compared in Table 5-6.

		NOMINAL	OPTIMUM	% IMPROVEMENT
RIGHT	RADIAL	2.53	1.33	47.5%
	TANGENTIAL	4.08	1.49	63.5%
LEFT	RADIAL	2.98	1.50	49.6%
	TANGENTIAL	5.74	2.34	59.2%

POSITION UNCERTAINTY - MILES

TABLE 5-6

Changing the times of the measurement points results in a significant cost reduction in both cases and the right horizon reference gives the best results.

The purpose of the horizon-selection procedure was to investigate the possibility of a cost reduction by providing a choice of two references at each measurement point. Using the measurement vectors defined in Appendix A for the right and left horizons, the $\delta \underline{z}$ vectors were defined in equations (E-18) and (E-19). The time optimization problem described in the first part of this chapter was carried out first. Since it was evident after the first run that the right horizon reference would result in a lower cost value, a reasonable nominal schedule for the horizon selection was to use this horizon at each point. The idea was to switch to the left horizon where the steepest descent procedure predicted a decrease. Since the switching problem was paired with a timing schedule optimization, it was anticipated that the horizon schedule would not stabilize until

the optimum time schedule was approached. Stated another way, the portions of the trajectory which preferred one horizon over another were expected to be a function of the central angle only.

As explained in Appendix E, the horizon selection differed from the time optimization in that there was no control over the step size. The measurement vector could not be driven in a direction to effect a cost decrease since the two values of $\delta\xi$ were pre-determined. If the cost change predicted from a proposed horizon change was negative, the switch was made. If not, the original horizon was retained. The problem does not have the continuous nature of the time optimization and the lack of step size control caused trouble. Early results using the scheme described above did not provide accurate predictions of cost change. When the program changed the horizon at all points where a cost decrease was predicted, the resultant cost value was greater than before. The influence vectors were correct, so it seemed best to change only one horizon at a time before re-evaluating the vectors. The problem persisted, however, and at that point the step size was investigated. It was found that the right and left horizon vectors were separated by an angle of 149° at 11,000 miles as shown in Figure 5-12.

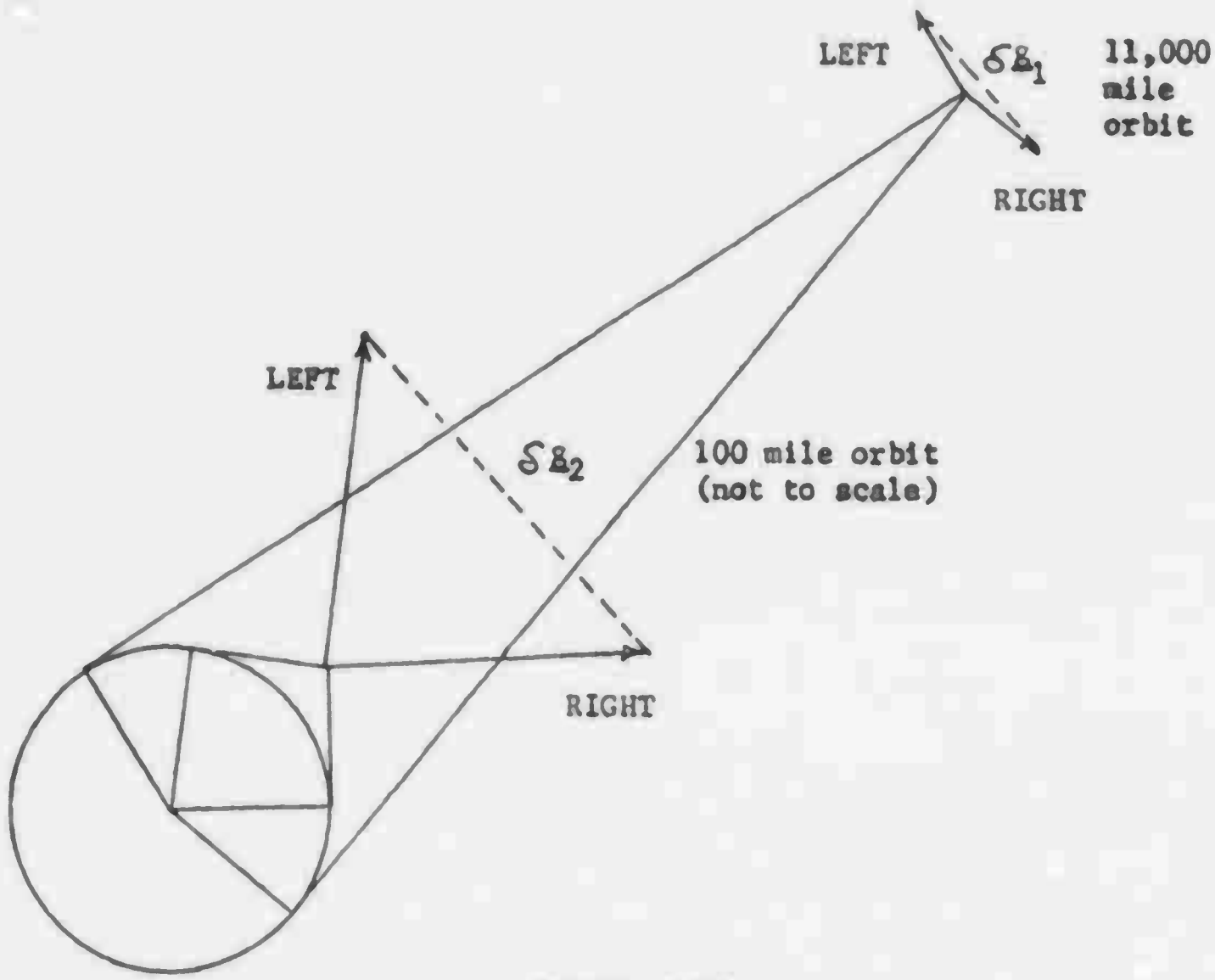
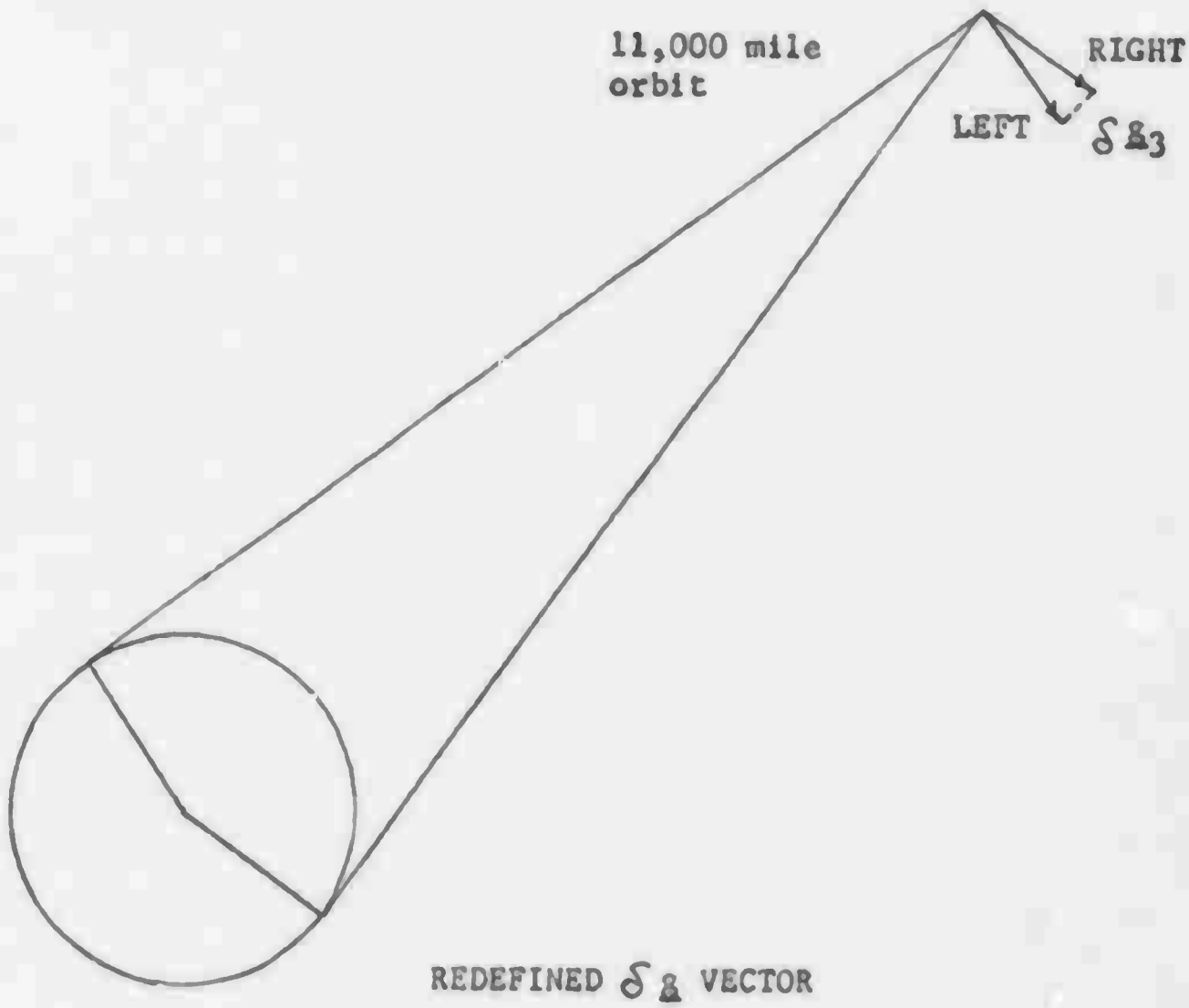


FIGURE 5-12
COMPARISON OF $\delta \underline{g}$ VECTOR AT DIFFERENT ALTITUDES

If the altitude was reduced to its lowest practical limit, 100 miles, the angle is reduced to 12^{o}; but, since the $\delta \underline{g}$ vectors are inversely proportional to the altitude, it was felt that the step size would still be too large. The alternative solution was to redefine the "left" horizon vector, using the negative of the $\underline{g}$ (LEFT) defined in Appendix E. The physical meaning of this change is that the star-elevation angle would be measured in a counter-clockwise, rather than clockwise, direction. An examination of Figure 1 in Appendix A shows that this is true. The time optimization procedure is not affected by this change since, in the expressions for E_a, L_k and δ Cost (Equations (D-10), (D-13), (D-1)), the measurement vectors appear in the form $\underline{g}\underline{g}^T$. Therefore only the square of the ele-

ments is critical and changing the sign of the measurement vector does not weaken the comparison with the straight time-optimization problem. As shown in Figure 5-13, the step size was considerably decreased. The step size could be reduced more, if necessary, by increasing the altitude.



REDEFINED δg VECTOR

FIGURE 5-13

As a further precaution, only one horizon change was made before re-evaluating the influence vectors. The δg vectors replacing those in Appendix E are given in equations (5-1) and (5-2).

$$\delta_{\mathcal{E}_k}\ (\text{LEFT TO RIGHT}) = \begin{bmatrix} \dfrac{2r_E}{z(z^2 - r_E^2)^{\frac{1}{2}}} \\ 0 \\ 0 \\ 0 \end{bmatrix} \qquad (5\text{-}1)$$

$$\delta_{\mathcal{E}_k}\ (\text{RIGHT TO LEFT}) = \begin{bmatrix} \dfrac{-2r_E}{z(z^2 - r_E^2)^{\frac{1}{2}}} \\ 0 \\ 0 \\ 0 \end{bmatrix} \qquad (5\text{-}2)$$

The procedure described in Chapters 3 and 4 was carried out, with more encouraging results. The predicted incremental cost changes are plotted in Figure 5-14 as a function of the central angle. Since the nominal schedule uses the right horizon at all points, the areas of negative cost predict a favorable change to the left horizon. It is evident that the points preferring one horizon or the other are not scattered randomly over the trajectory but lie together in certain well defined areas.

Since the time and horizon selection procedures are independent, the shape of Figure 5-14 should not be affected by changing measurement times. The angular limits corresponding to either horizon reference can be predicted from the figure. These predictions are listed in Table 5-7.

RIGHT HORIZON	LEFT HORIZON
16° to 109°	0° to 16°
212° to 290°	109° to 212°

AREA PREDICTIONS

TABLE 5-7

A similar line of reasoning applies to the time-optimization clusters. Changing the horizon from right to left should not affect the number of clusters although their positions may be slightly altered. The predictions for the cluster positions are given in Table 5-8 along with the predicted horizon obtained from Table 5-7.

	ANGLE	HORIZON
1st	0°	LEFT
2nd	72°	RIGHT
3rd	202°	LEFT
4th	290°	RIGHT

CLUSTER PREDICTIONS

TABLE 5-8

As expected, the effect of two optimization procedures is to provide for more rapid convergence. The horizon changes made along with the respective central angles are listed in order of their occurrence in Table 5-9. For each iteration, the proposed horizon change which results in the greatest cost decrease is the only change made. The numbers associated with the points are the identification numbers in the program. After several iterations, these numbers lose their meaning since the points may pass each other on the way to the optimum.

POINT NO.	CHANGE	ANGLE AT CHANGE
19	RIGHT TO LEFT	165.5°
18	"	157.0°
17	"	148.5°
20	"	174.0°
16	"	140.0°
21	"	182.4°
15	"	135.6°
1	"	8.2°
14	"	122.2°
22	"	190.8°
13	"	112.1°
13	LEFT TO RIGHT	112.0°
2	RIGHT TO LEFT	17.2°
22	LEFT TO RIGHT	240.9°
21	LEFT TO RIGHT	252.9°

HORIZON CHANGES

TABLE 5-9

The predictions in Table 5-7 were quite accurate for several iterations. At one point however, while seeking an overall cost reduction of 5% or greater, the program made changes which were obviously outside the linear range. A number of the angles were changed by 30° or more. A cost reduction was realized from these new values but the large changes, in effect, altered the nature of the problem. If such violations of linearity were not allowed the new values would be arrived at from a different nominal schedule. This "new" nominal schedule would probably result in different zero-crossings in Figure 5-14 and hence different predictions in Table 5-7. This line of reasoning seeks to explain the apparent discrepancies in the last two entries of Table 5-9.

As the clusters become more clearly defined, the horizon selection stabilizes since there is no further movement across the

boundaries. The result of the time optimization is given in Table 5-10.

	INCLUDED POINTS	ANGLE	HORIZON
1st	2	0°	LEFT
2nd	11	67°	RIGHT
3rd	6	209°	LEFT
4th	11	290°	RIGHT

FINAL CLUSTERS
TABLE 5-10

The table shows that the horizon selection procedure does not change appreciably the strength and position of the clusters. Comparing Tables 5-8 and 5-10 shows the accuracy of the predictions. The dispersion of the points, at different stages in the optimization, is shown in the polar plots, Figures 5-15 through 5-18. The nominal positions are the same as shown in Figure 5-3. Note that the improvement in overall percent change is % over the "RIGHT" case in Table 5-3, which has the same initial conditions. The sizes of the time-selection influence coefficients for points at and near the cluster positions are compared in Table 5-11, in order to justify the final position of the clusters.

	ON CLUSTER	OFF CLUSTER	ANGULAR DIFFERENCE
1st	15.8	33.0	.03°
2nd	92.8	872.9	.02°

INFLUENCE COEFFICIENTS - FT^2/SEC

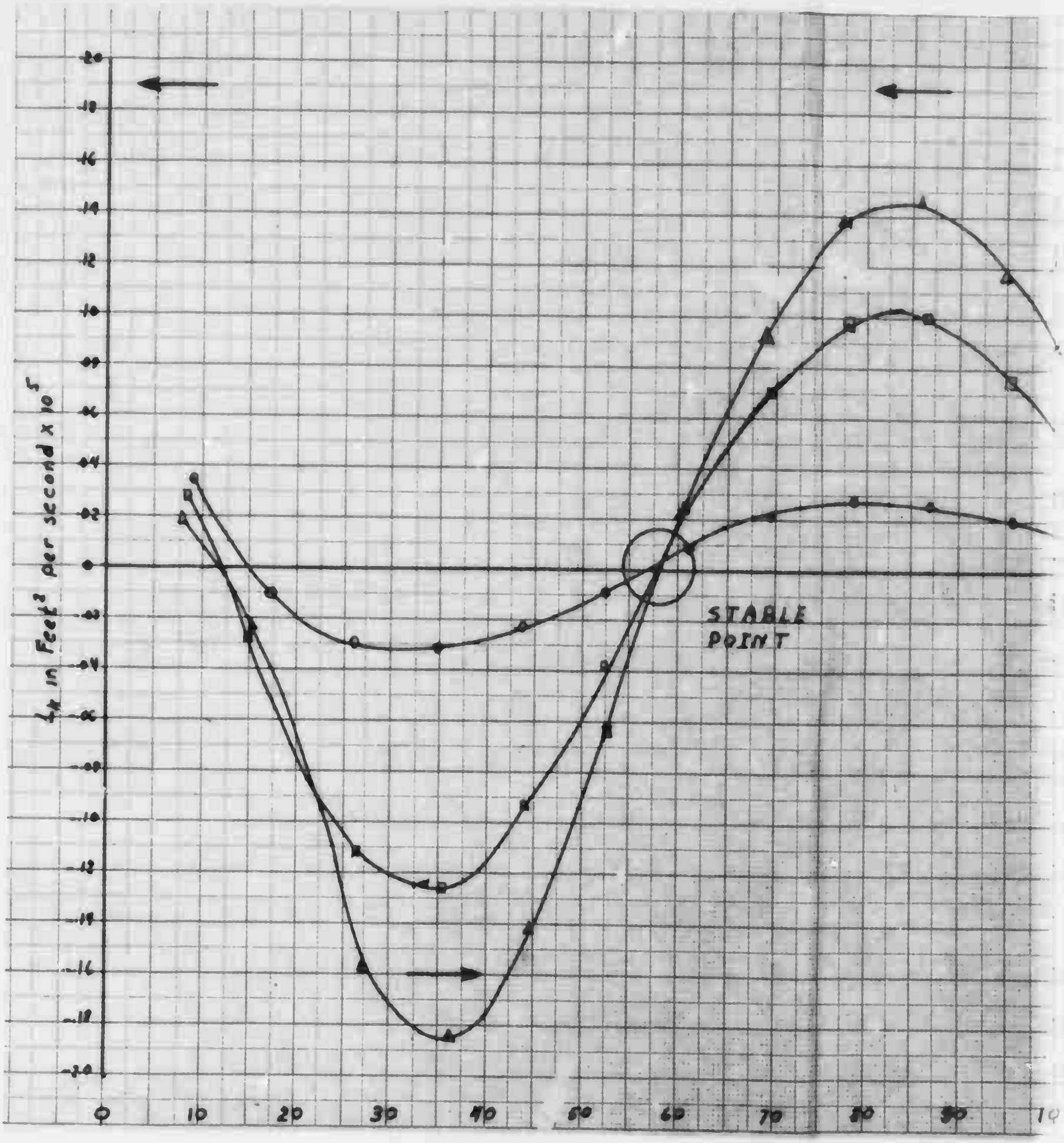
TABLE 5-11

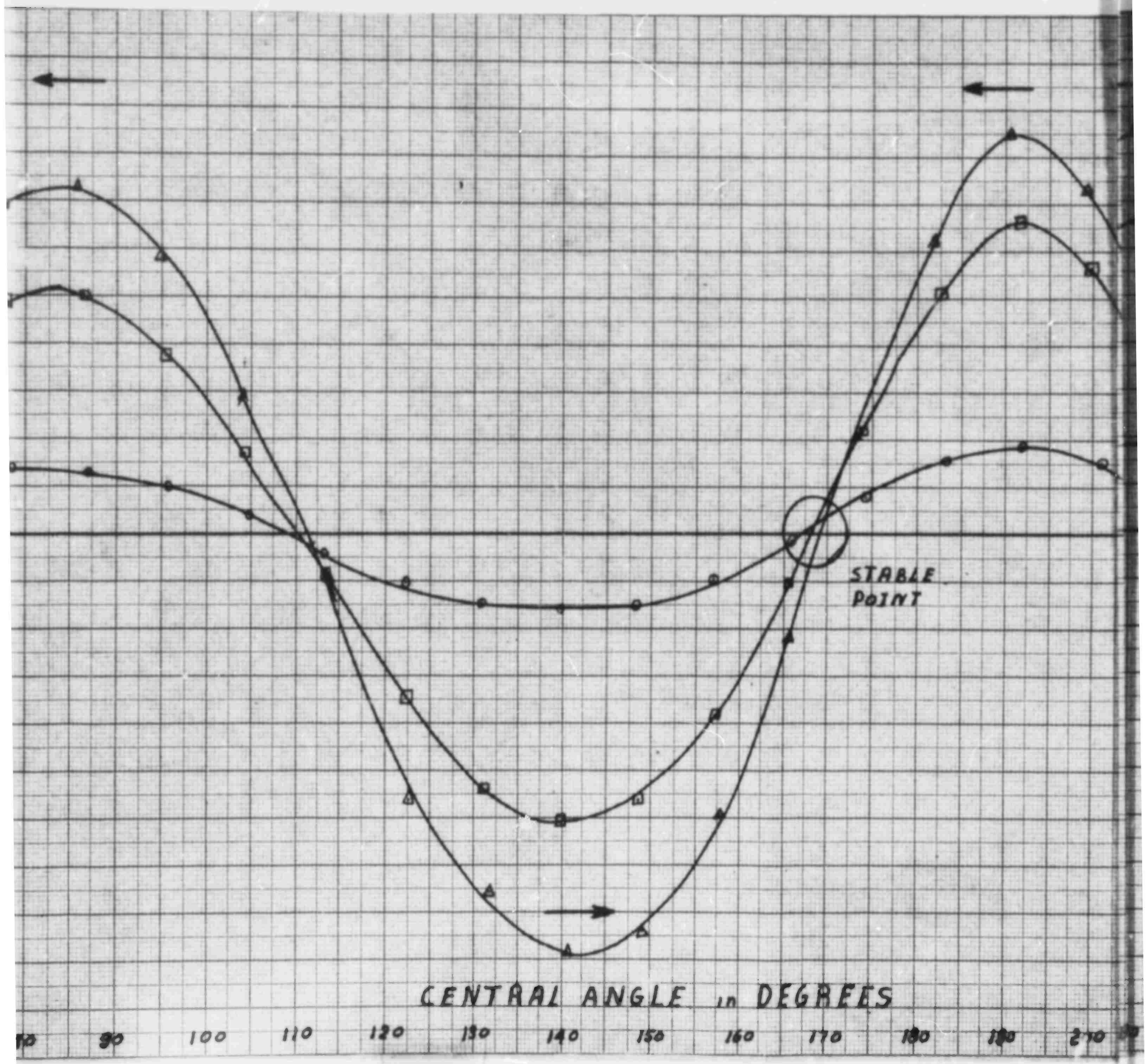
The radial and tangential components of the final position uncertainty are listed in Table 5-12. In these more familiar units, the improvement over the previous time optimization is more obvious.

	NOMINAL	OPTIMUM	% IMPROVEMENT
RADIAL	2.53	1.06	58.2%
TANGENTIAL	4.08	1.46	64.3%

POSITION UNCERTAINTY - MILES

TABLE 5-12


$\dot{h}_x$ in Feet² per second x
20
18
16
14
12
10
8
6
4
2
0
-2
-4
-6
-8
-10
-12
-14
-16
-18
-20
0
10
20
30
40
50
60
70
80
90
100
STABLE POINT


STABLE
POINT
CENTRAL ANGLE in DEGREES
90
100
110
120
130
140
150
160
170
180
190
210

B

FIGURE 5.1

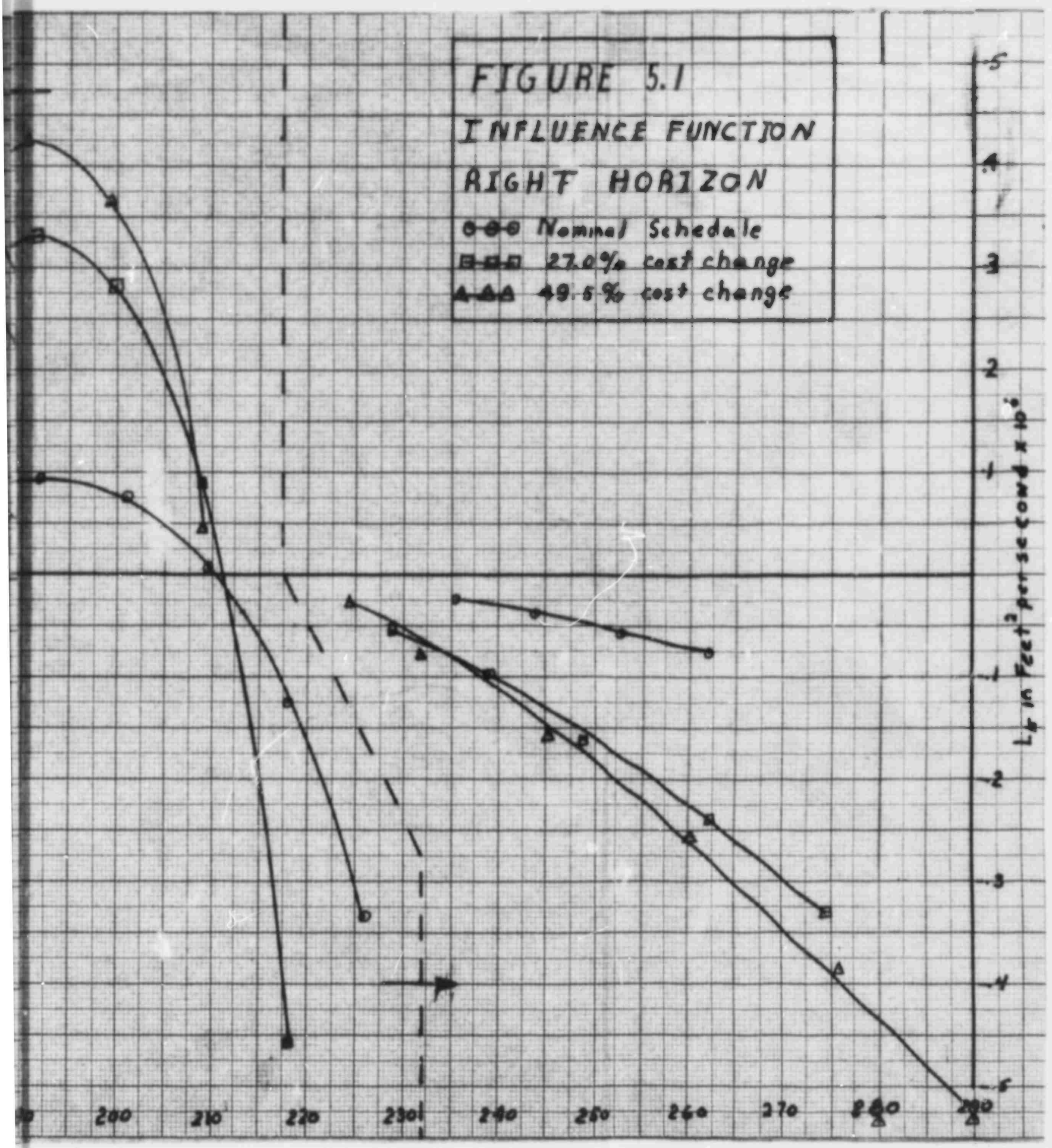
INFLUENCE FUNCTION

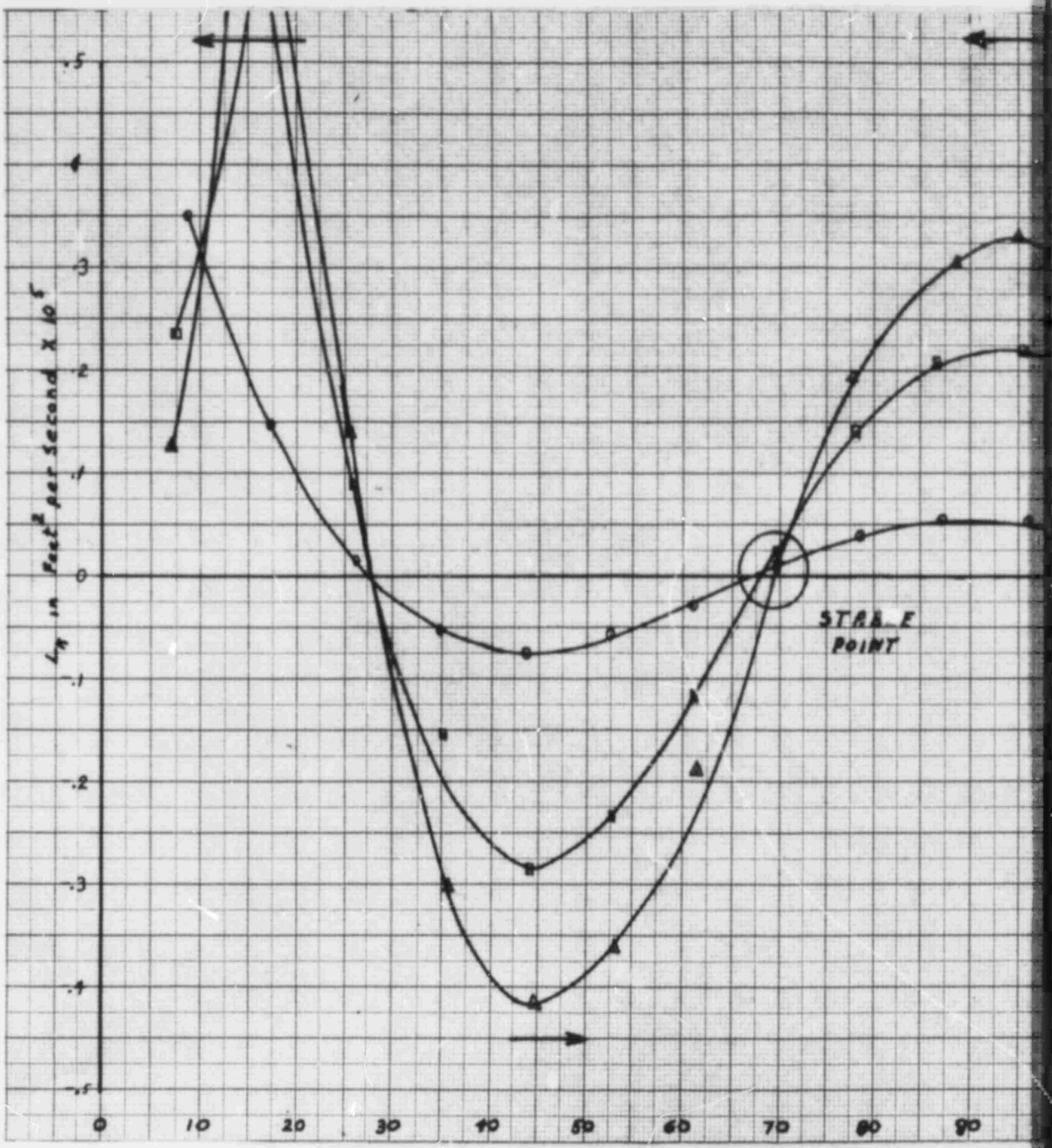
RIGHT HORIZON

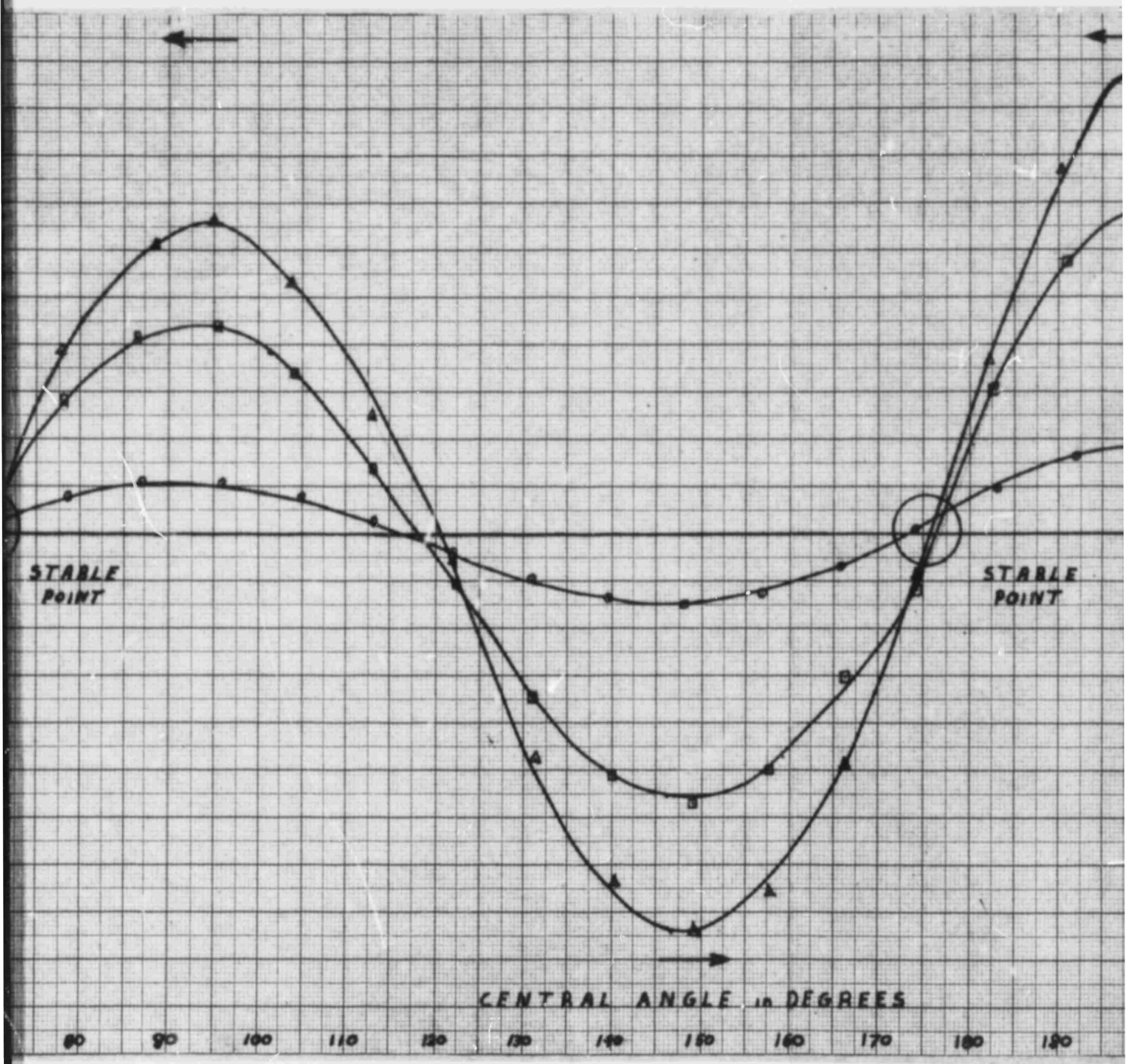
o-o-o Nominal Schedule

□-□-□ 27.0% cost change

Δ-Δ-Δ 49.5% cost change




.5
.4
.3
.2
.1
0
-.1
-.2
-.3
-.4
-.5
$\Delta\eta$ in Feet^2 per Second X 10^5
0
10
20
30
40
50
60
70
80
90
STABLE POINT


STABLE POINT
STABLE POINT
A
B
CENTRAL ANGLE in DEGREES
80
90
100
110
120
130
140
150
160
170
180
190

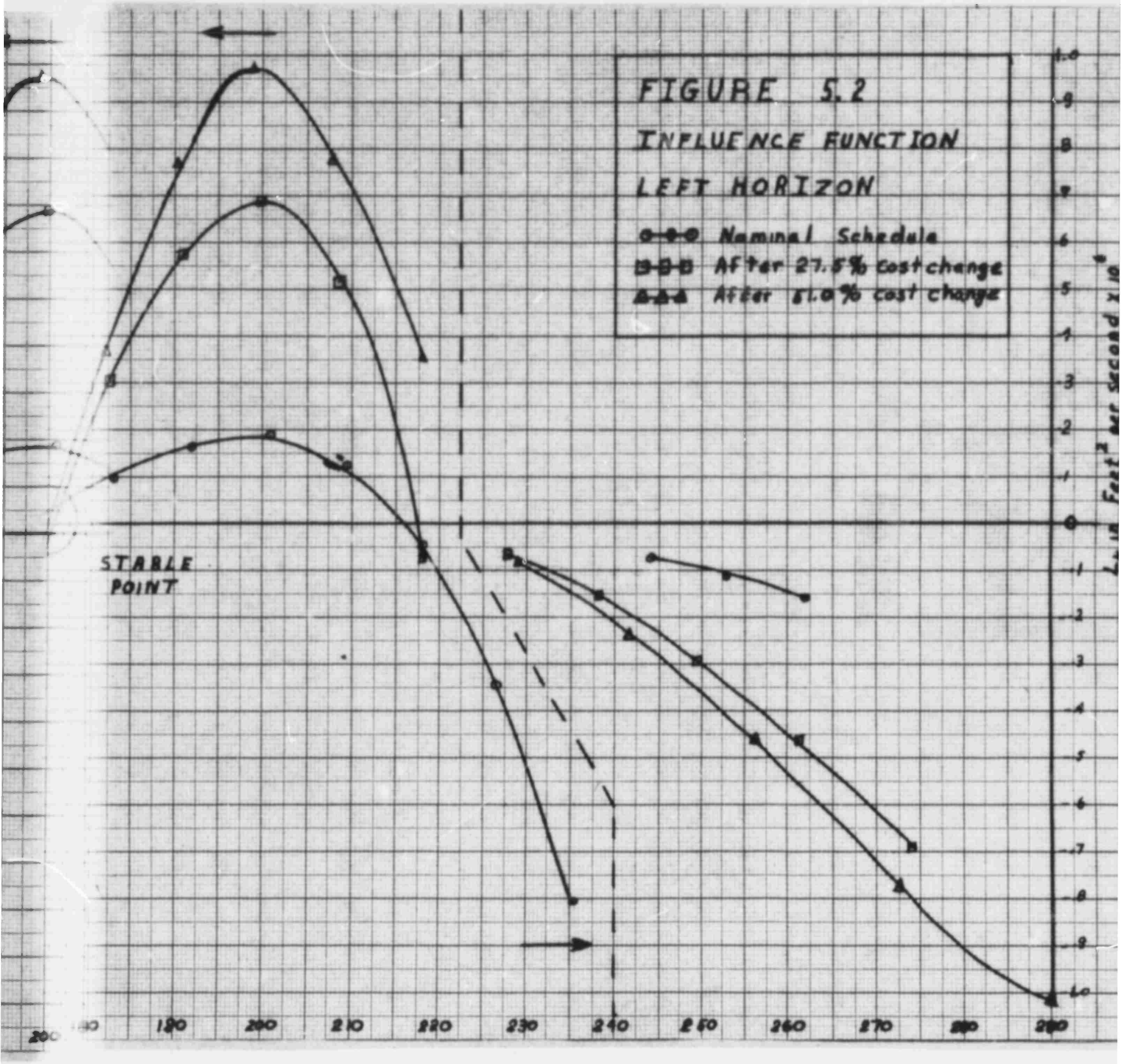


FIGURE 5.2
INFLUENCE FUNCTION
LEFT HORIZON
Nominal Schedule
After 27.5% cost change
After 51.0% cost change
STABLE POINT
180
190
200
210
220
230
240
250
260
270
280
290
1.0
9
8
7
6
5
4
3
2
1
0
-1
-2
-3
-4
-5
-6
-7
-8
-9
-10

C

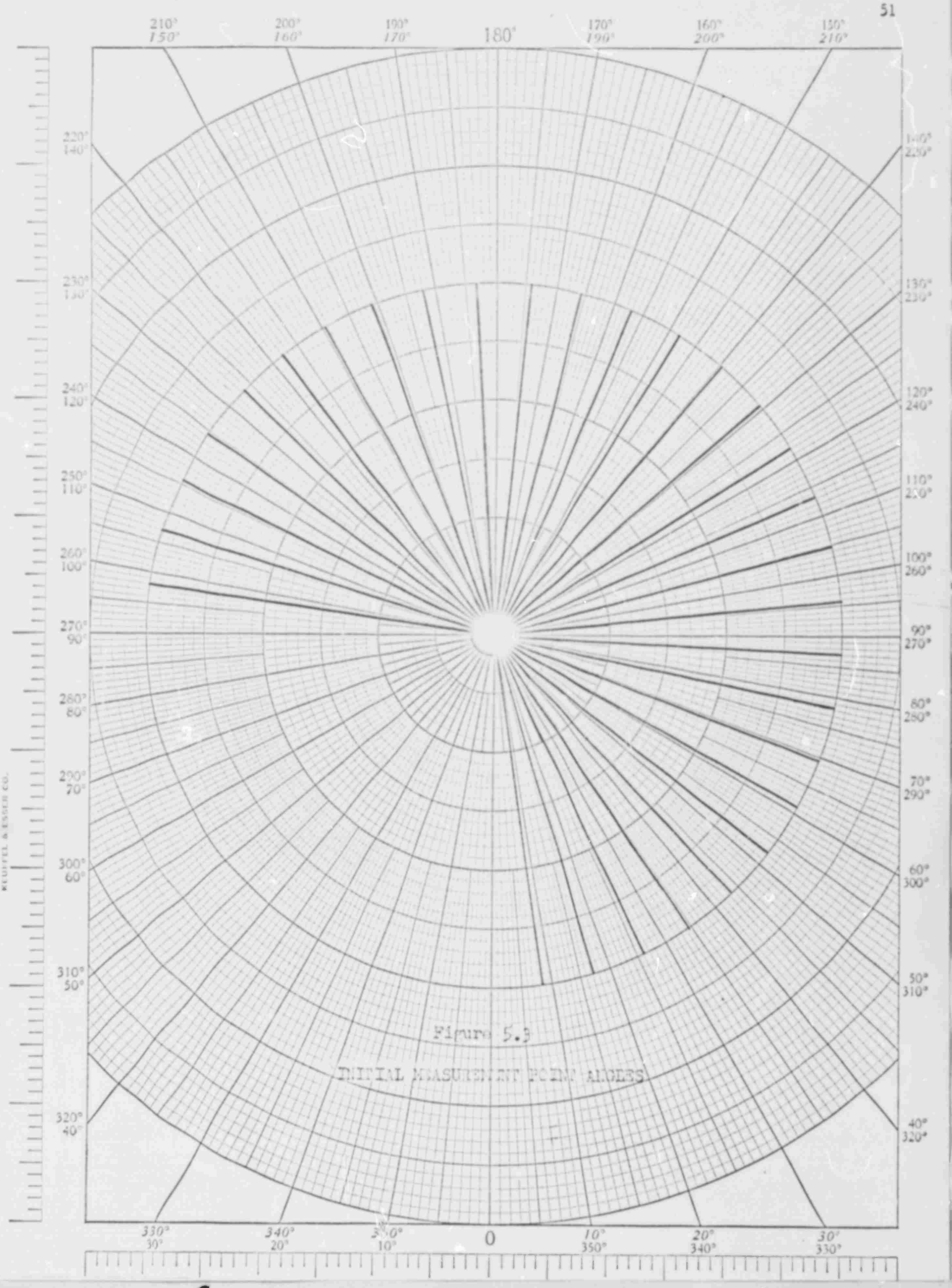


Figure 5.3

INITIAL MEASUREMENT POINT ANGLES

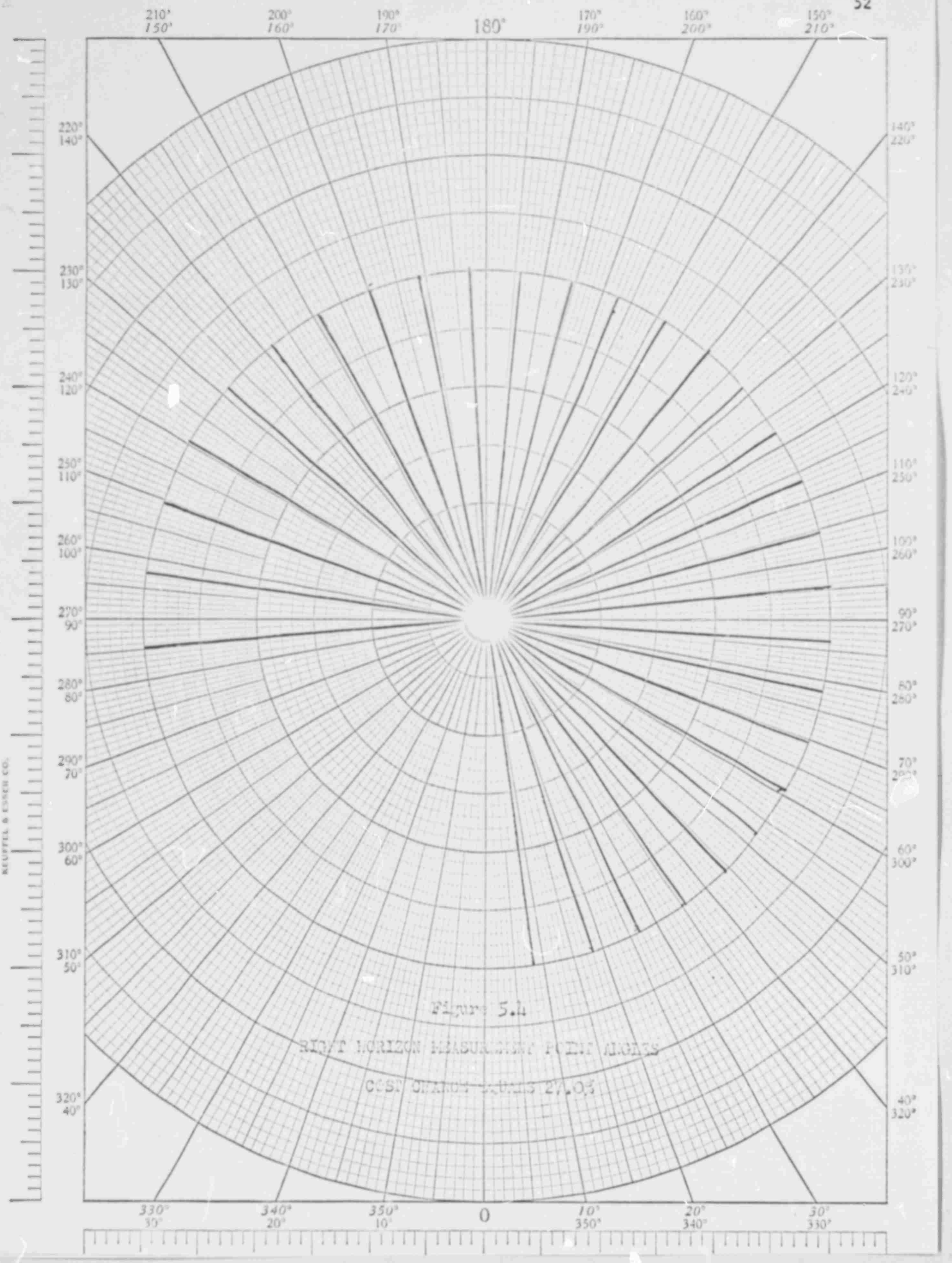
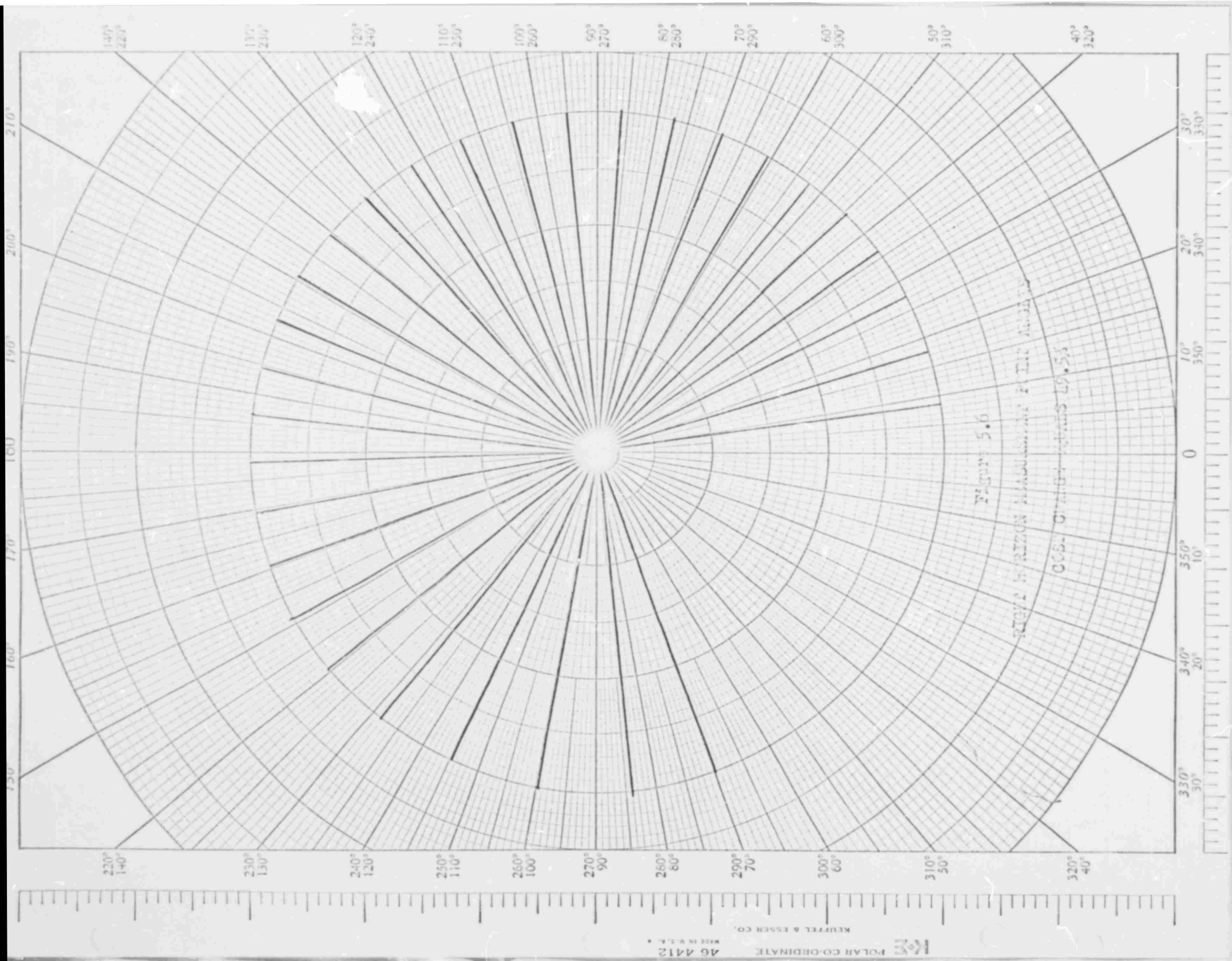


Figure 5.4

RIGHT HORIZON MEASUREMENT POINT ANGLES

CCSP ORBIT 2 DAYS 27.05


140° 220°
130° 230°
120° 240°
110° 250°
100° 260°
90° 270°
80° 280°
70° 290°
60° 300°
50° 310°
40° 320°
30° 330°
20° 340°
10° 350°
0
170°
160°
150°
220° 140°
230° 130°
240° 120°
250° 110°
260° 100°
270° 90°
280° 80°
290° 70°
300° 60°
310° 50°
320° 40°
330° 30°
340° 20°
350° 10°
KEUFFEL & ESSER CO.
POLAR CO-ORDINATE 46 4412

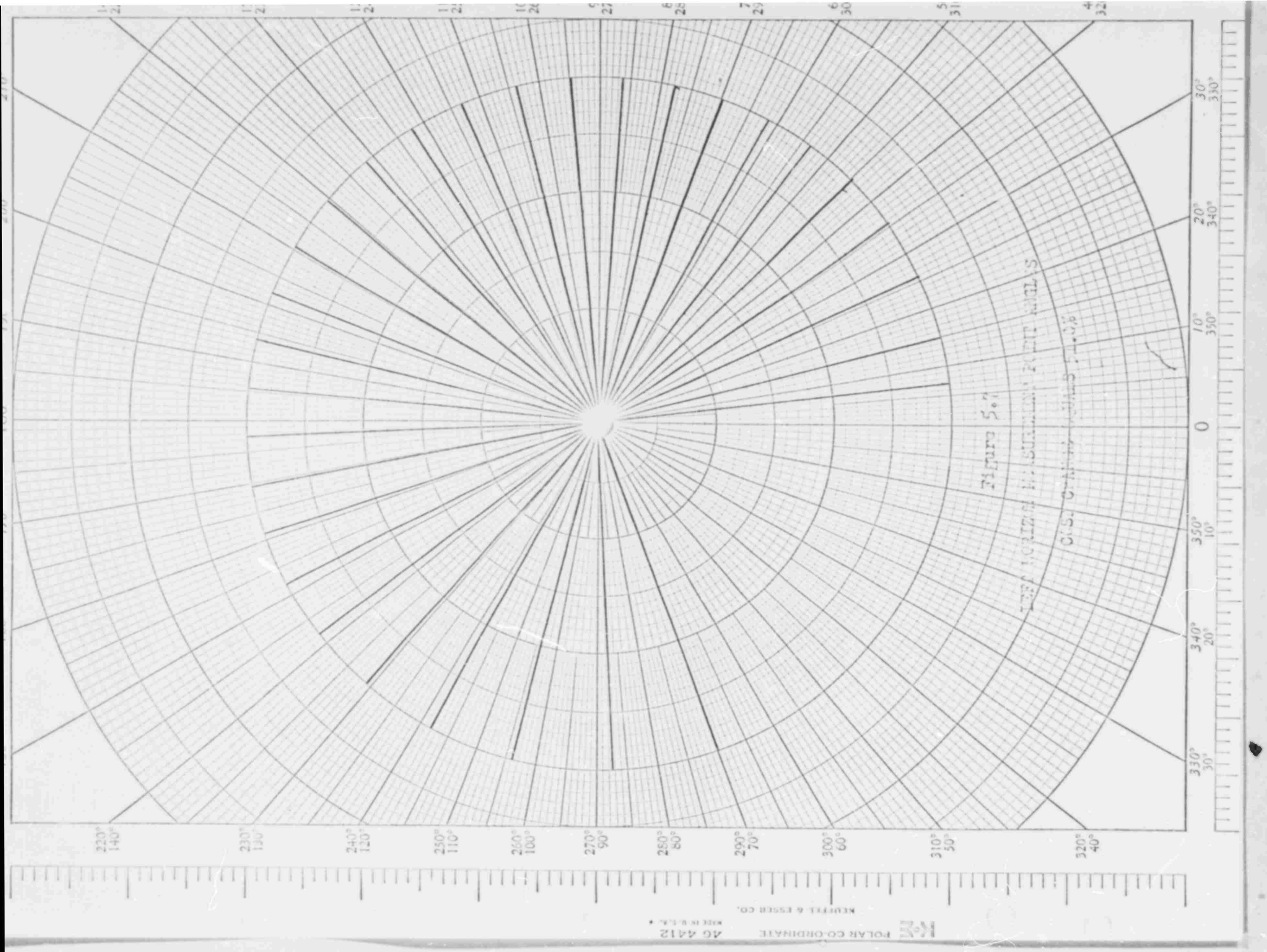


FIGURE 5.21
POLAR CO-ORDINATE
KEUFFEL & ESSER CO.

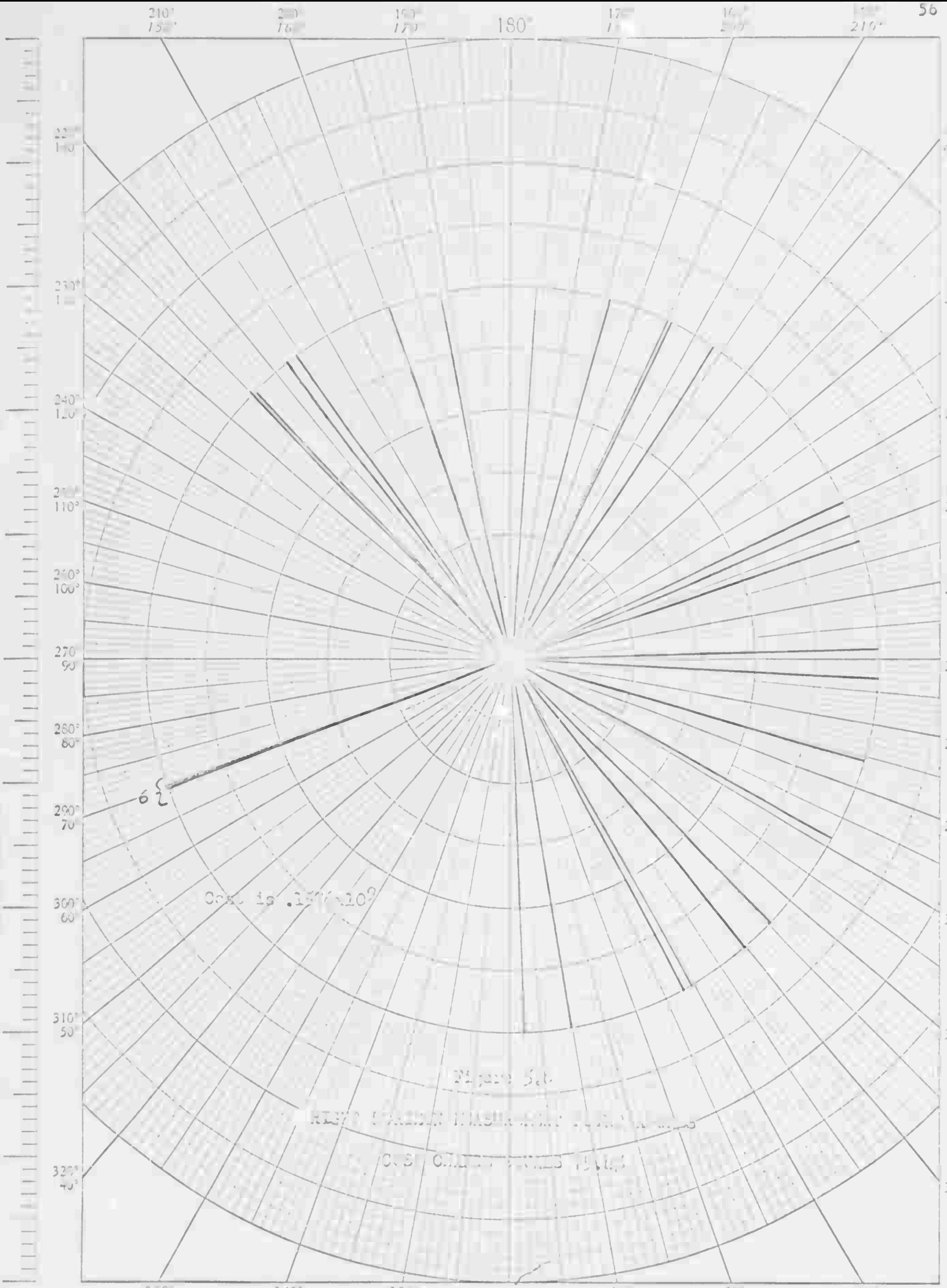
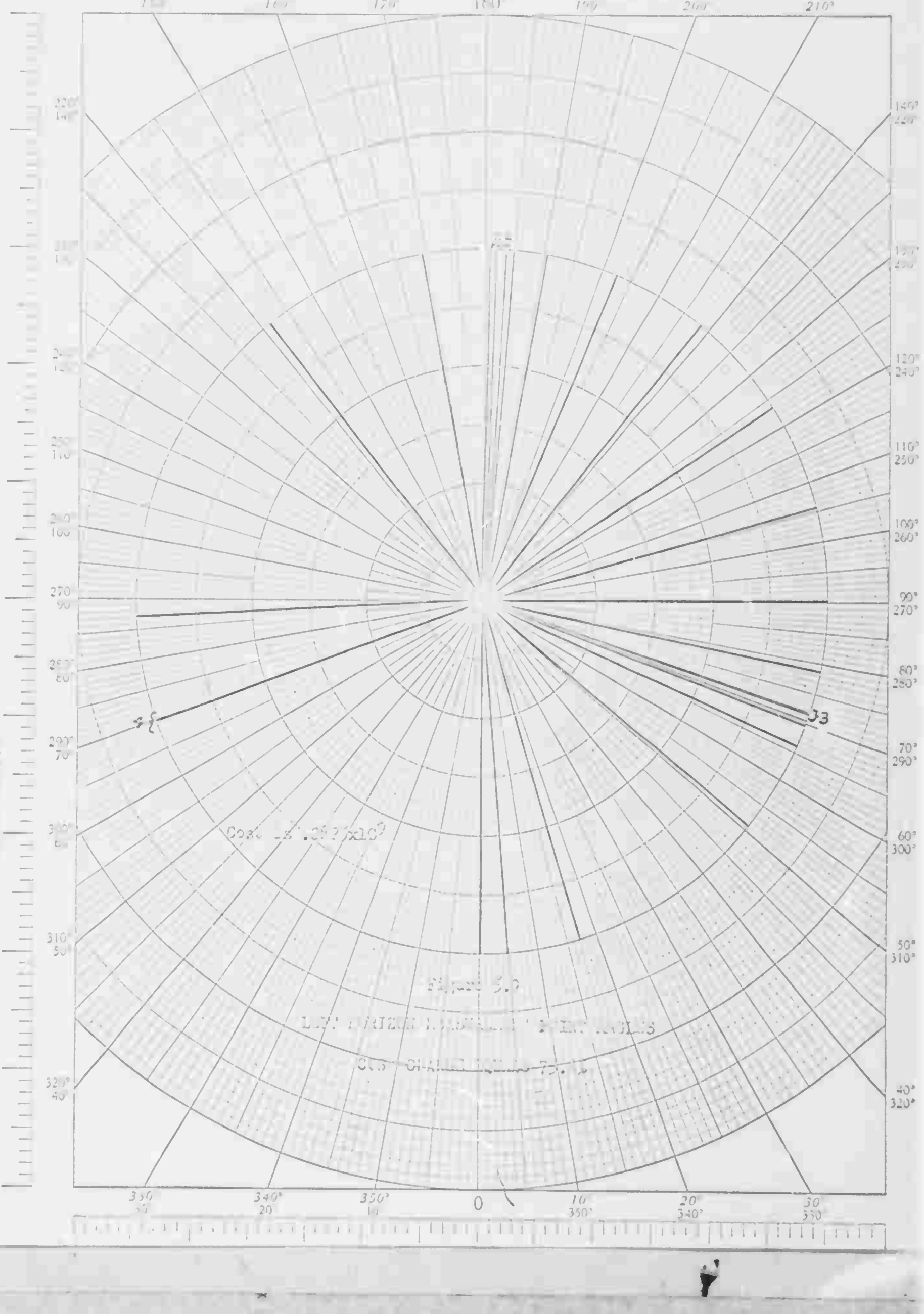


Figure 5.6

[illegible]

[illegible]


180
0
33
Figure 5.0

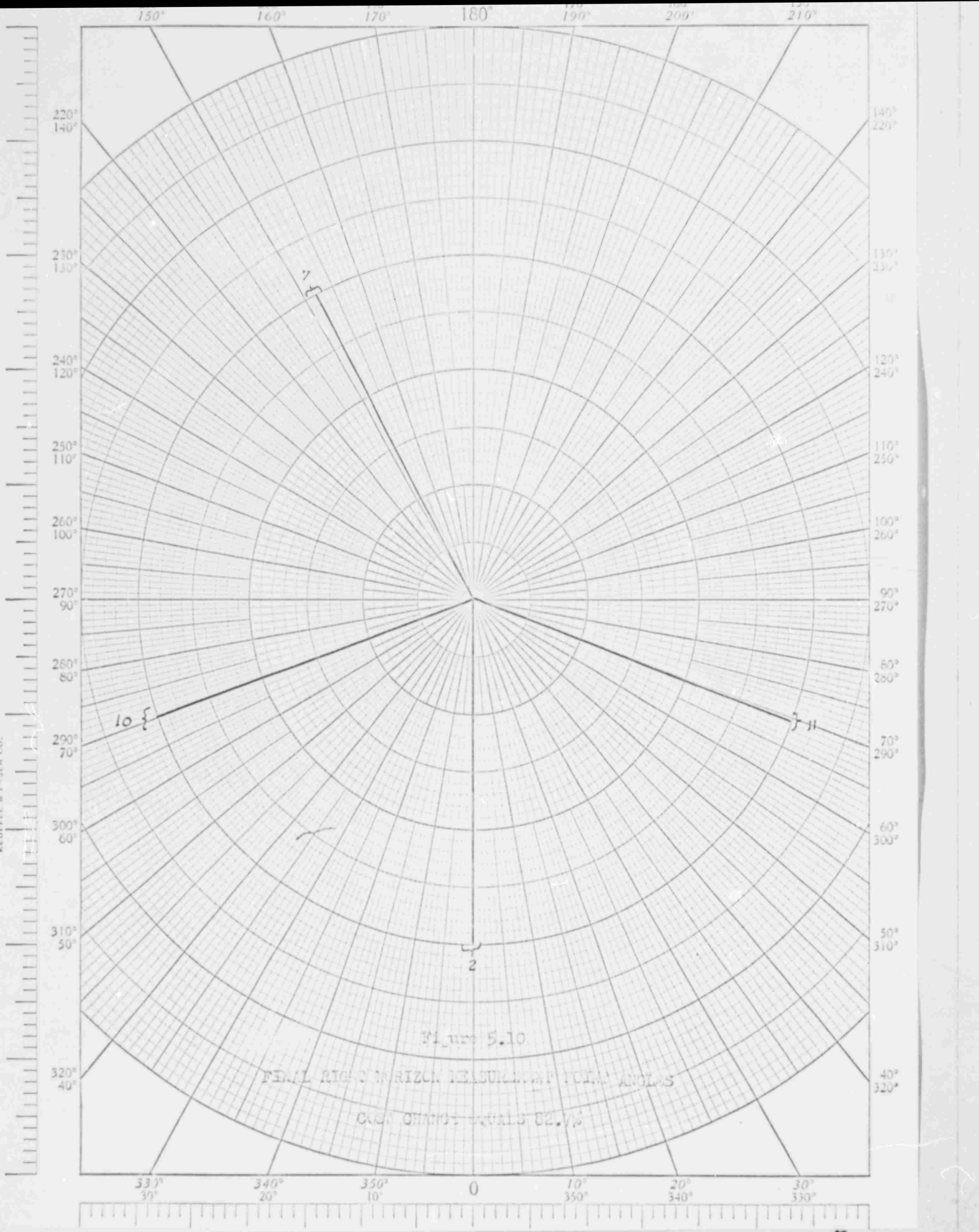


Figure 5.10

FINAL RIGHT HORIZON TESSELLATED POLAR ANGLES

COS CHORD EQUALS 82.5%

Figure 5.11

FINAL DEFLECTION MEASUREMENT POINTS

COSINE CHANGE EQUALS 81.43

KEUFFEL & ESSER CO.

POLAR CO-ORDINATE

46 4412

MADE IN U.S.A.

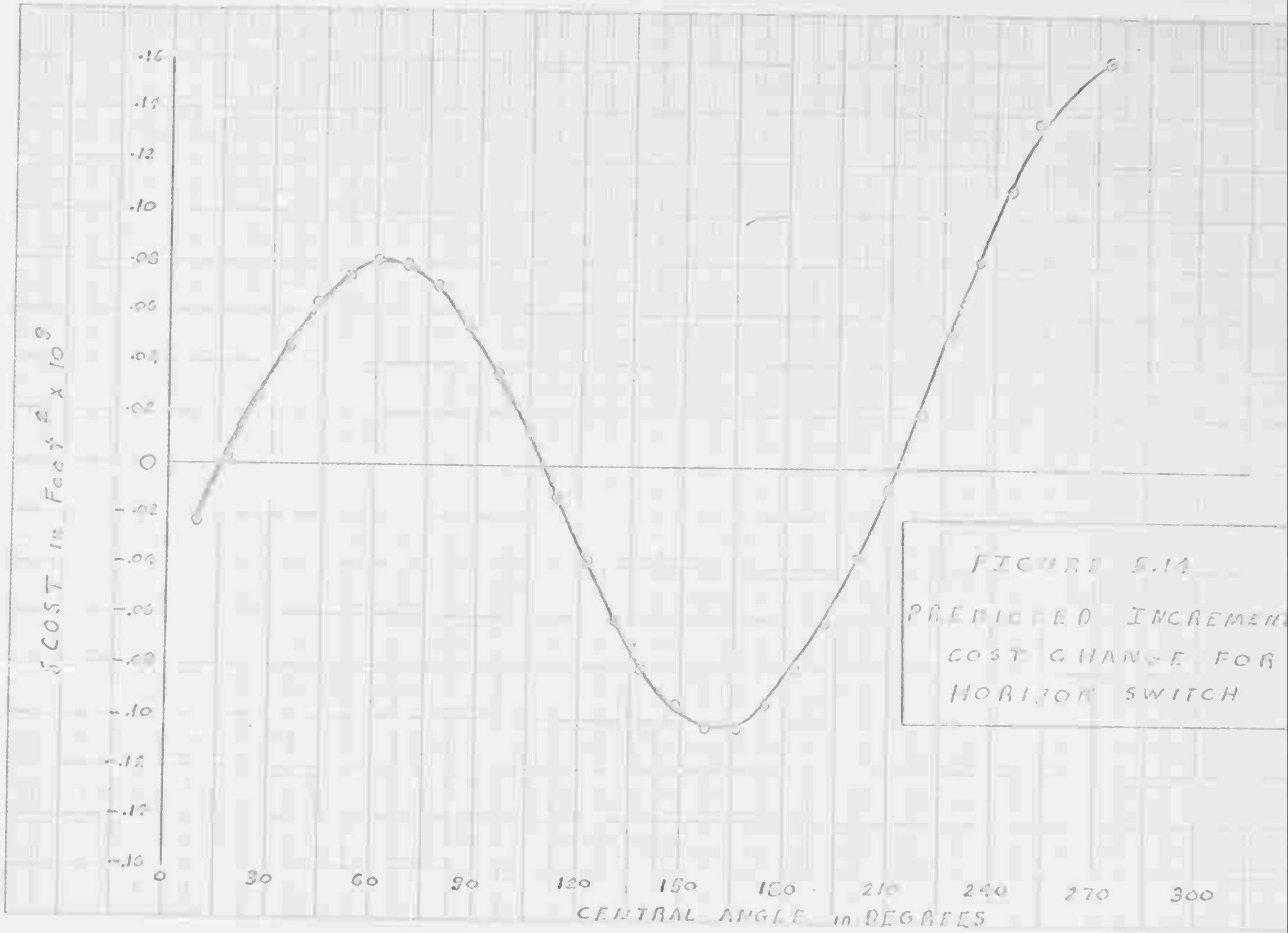


FIGURE 5.14
PREDICTED INCREMENTAL COST CHANGE FOR HORIZON SWITCH

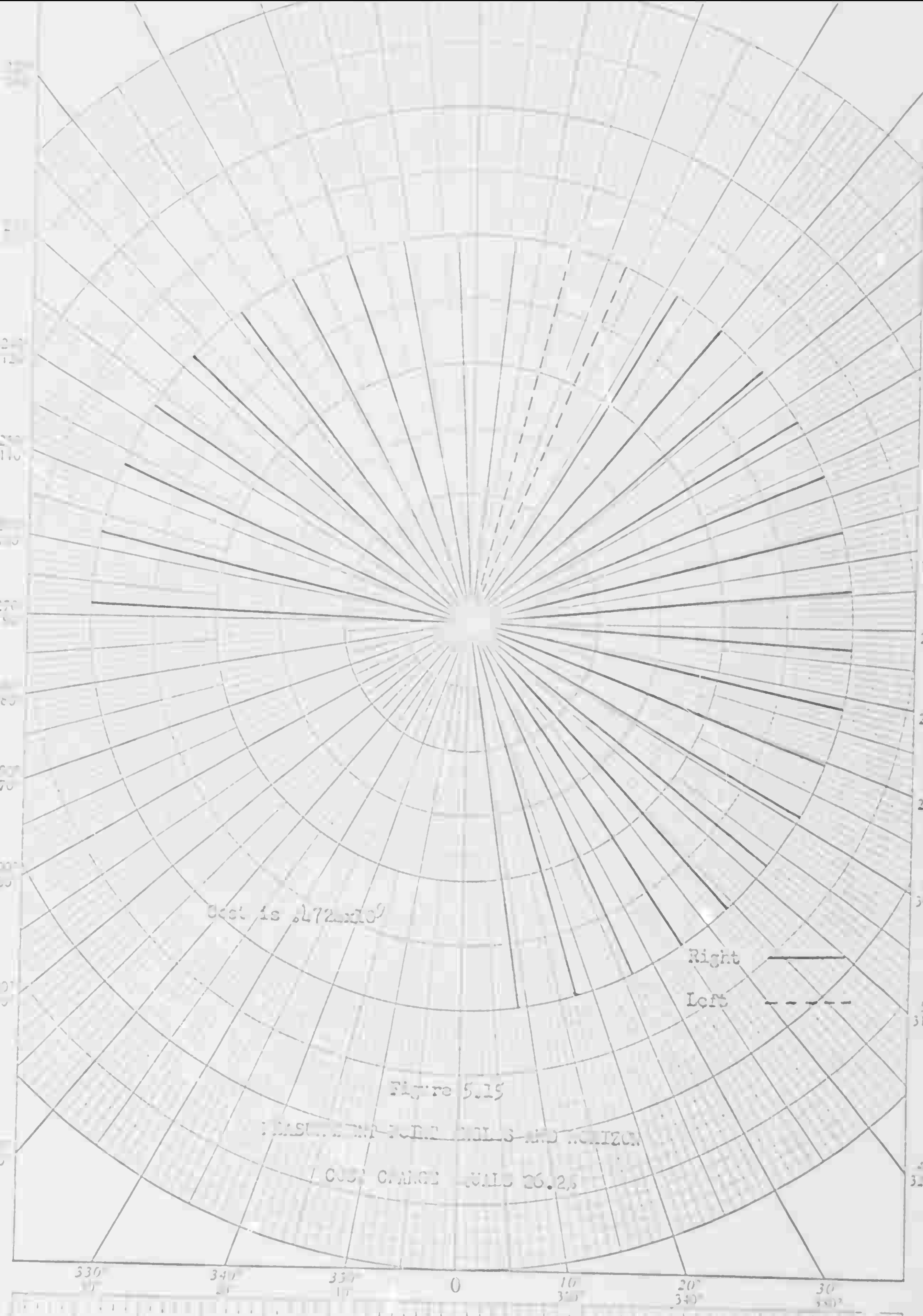
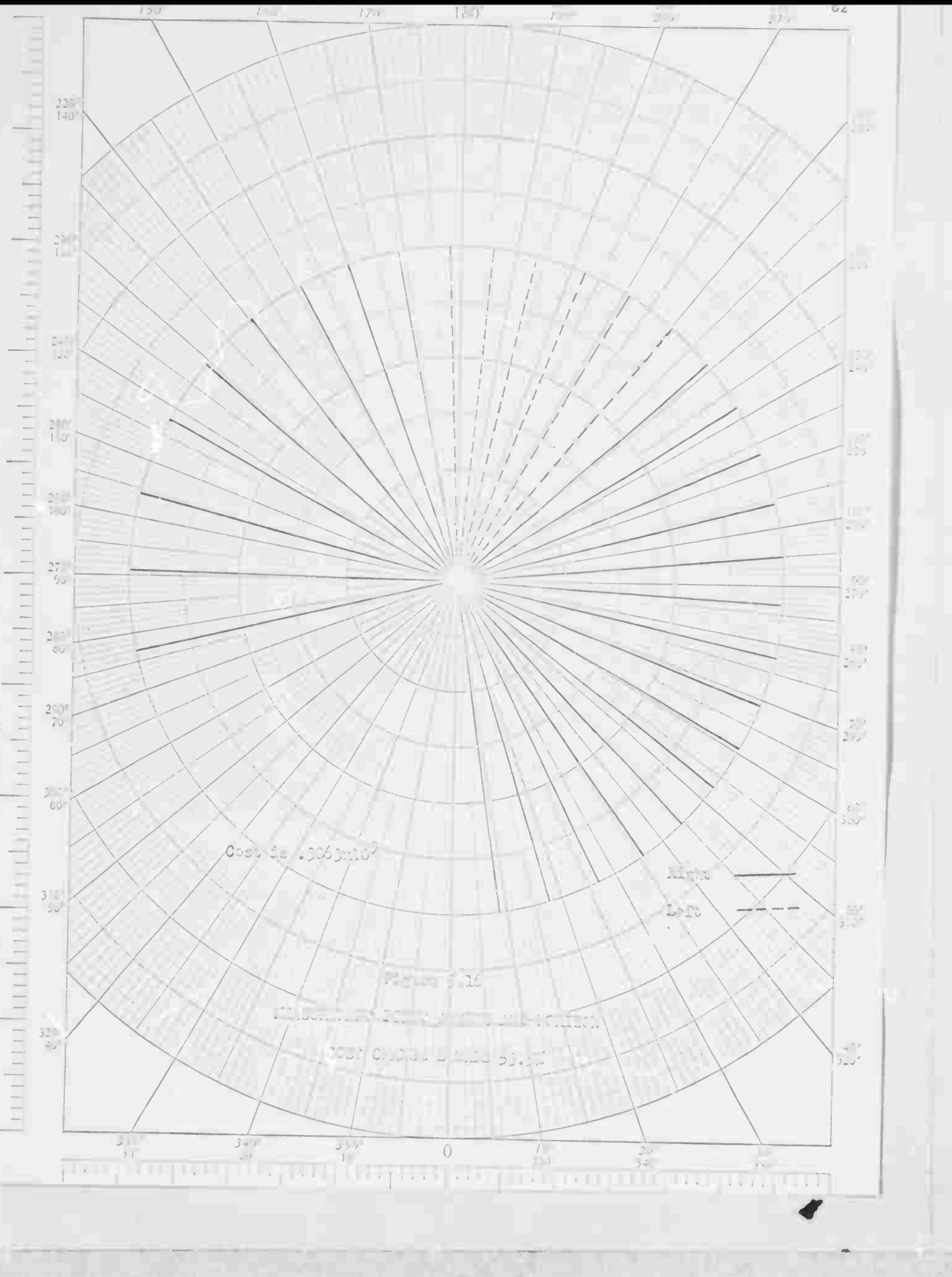
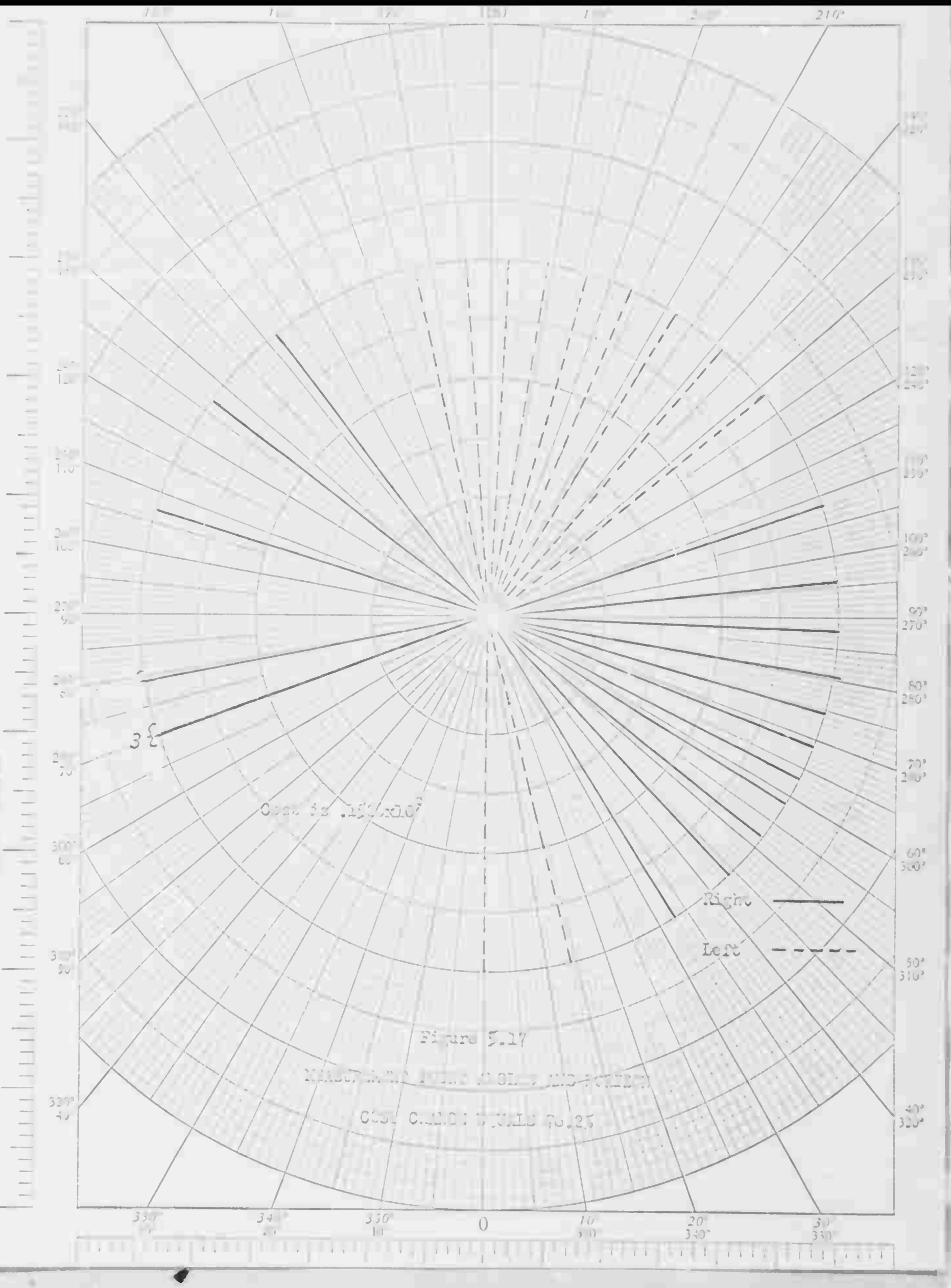

Right
Left
90°
270°
80°
280°
70°
290°
60°
300°
50°
310°
40°
320°
100°
260°
110°
120°
240°
0
10
20°
340°
30
330
340

Figure 5.15


Right
Left
0


Right
Left
Figure 5.17

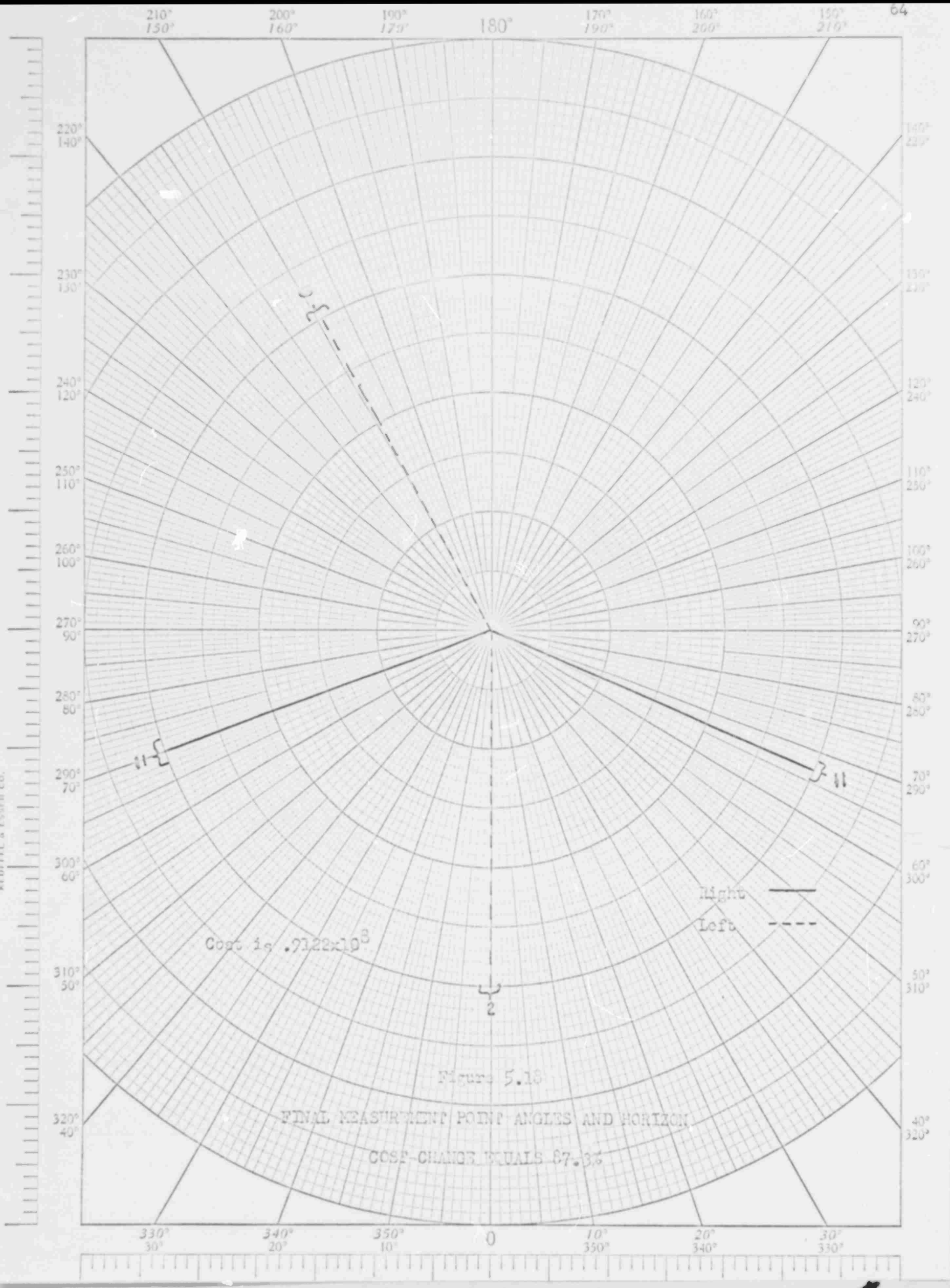


Figure 5.18

FINAL MEASUREMENT POINT ANGLES AND HORIZON

COSP CHANGE EQUALS 67.3%

time change takes on a different meaning, dependent on where it occurs along the trajectory.

The time-optimization problem can be coupled to a more sophisticated measurement-selection procedure as an extension for the second portion of the thesis. As discussed in Chapter 5, the steepest-descent technique is not always a reliable method in this discontinuous type of problem. The step size from one possible measurement to another must be small enough so that the linearizing assumptions are valid or false predictions will result. Denham and Speyer were conscious of these limitations and steepest-descent worked well in their study.

Different cost functions can be used to determine their effect on the clusters. Some possible schemes are mentioned in Chapter 2, such as a weighted average of position and velocity uncertainty. Instead of making many measurements in a short space of time, as is implied by the clusters of points, it might be more reasonable to track the angular elevation of a star for a certain amount of time. This would eliminate frequent changes of spacecraft altitude and result in a significant fuel saving. A cost function could be contrived which would compare the effect of tracking on terminal position uncertainty to the effect of making discrete measurements.

It is interesting to speculate on the physical reasons behind the clusters. It might be argued that a series of measurements early in the flight would tend to reduce the effect of the initial estimation error by improving the estimate before it propagates too far. Likewise, the measurements near the terminal point would tend to

reduce the position uncertainty of that point. The position of the middle two clusters is somewhat of a mystery, however. Also, the strength of the clusters, measured by the number of included points, and their convergence times are by no means identical. The values may be random in nature, but it is more likely that there are physical causes.

The results obtained are a strong argument for the practicality of changing the measurement times, in spite of the simplicity of the problem. It is reasonable to expect that a more complicated model would produce similar results.

APPENDIX A

MEASUREMENT VECTOR

The measurement vectors for a variety of techniques are developed in Reference 1. For this problem, the star-elevation measurement was chosen because of its superior accuracy in a planetary orbit. The local vertical co-ordinate system is used to coincide with the system adopted by Stern in developing his transition matrix.

As shown in Battin, each type of measurement is characterized by a measurement vector. The deviation in the quantity to be measured relates to the position deviation by the following formula, where $\underline{h}$ is this vector.

$$\delta q = \underline{h} \cdot \delta \underline{r} \qquad \text{(A-1)}$$

The procedure for determining $\underline{h}$ is the same for all measurements; the equation defining the quantity to be measured is developed and, from it, the perturbation equation. In this case, the measured quantity is the angle from the planet horizon to the line of sight to a star, as shown in Figure 1. Equation (A-2) defines the measured angle M.

$$\underline{n} \cdot \underline{s} = s \cos(M+B) \qquad \text{(A-2)}$$

where $\underline{n}$ is a unit vector in the direction of the star. Writing the

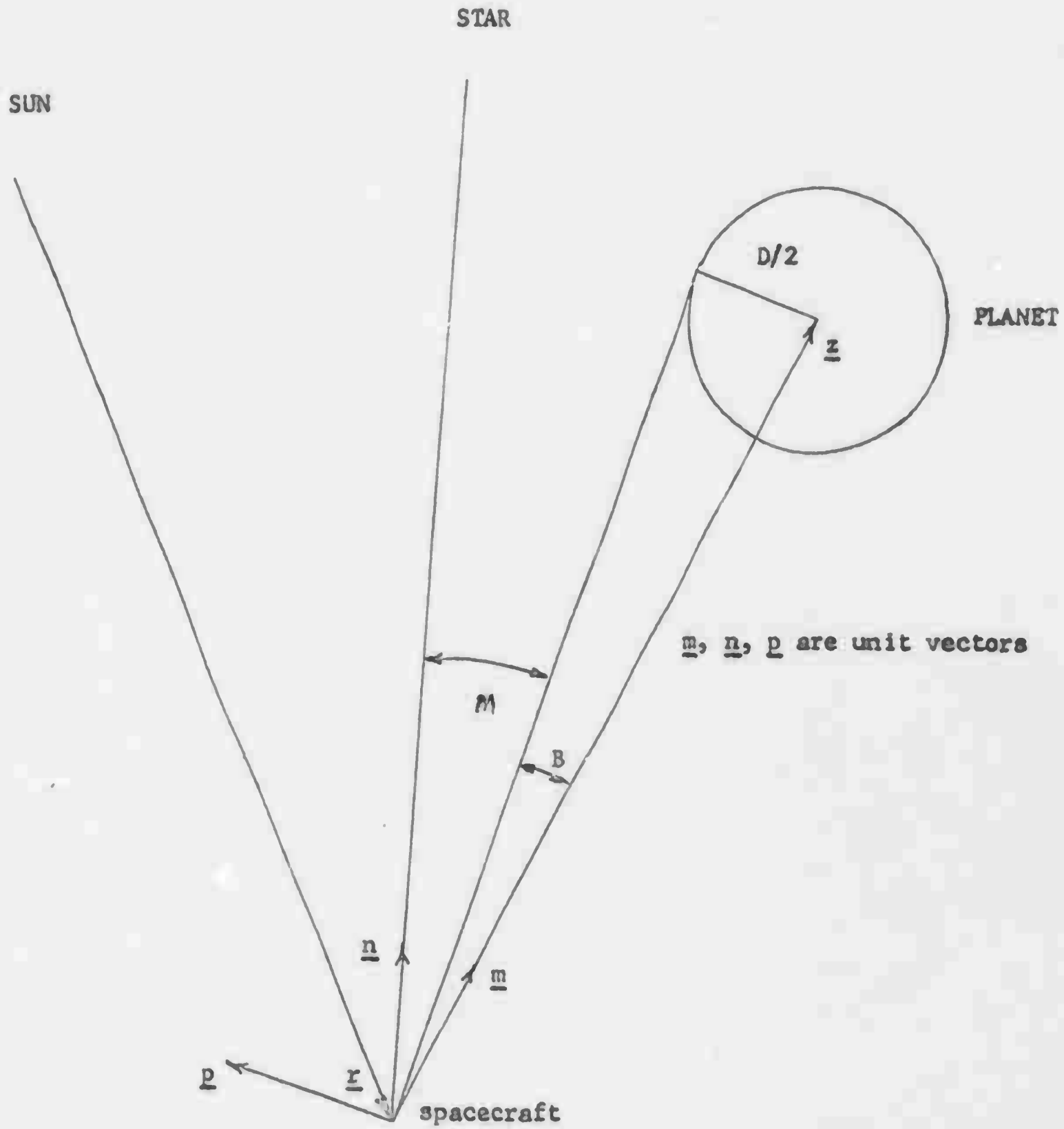


Figure 1

STAR ELEVATION ANGLE

perturbation equation:

$$\underline{n} \cdot \delta \underline{z} = -z \sin(M+B)(\delta M + \delta B) + \delta z \cos(M+B) \qquad \text{(A-3)}$$

A vector expression for the scalar δz will be useful.

$$\underline{z} \cdot \underline{z} = z^2$$

$$\delta \underline{z} \cdot \underline{z} + \underline{z} \cdot \delta \underline{z} = 2z \delta z$$

$$\frac{\underline{z} \cdot \delta \underline{z}}{z} = \delta z \qquad \text{(A-4)}$$

Substituting (A-4) into (A-3):

$$\frac{\underline{z} \cdot \delta \underline{z}}{z} \cos(M+B) - z \sin(M+B)(\delta M + \delta B) = \underline{n} \cdot \delta \underline{z} \qquad \text{(A-5)}$$

From Figure 2, it is obvious that $\delta \underline{z} = -\delta \underline{r}$.

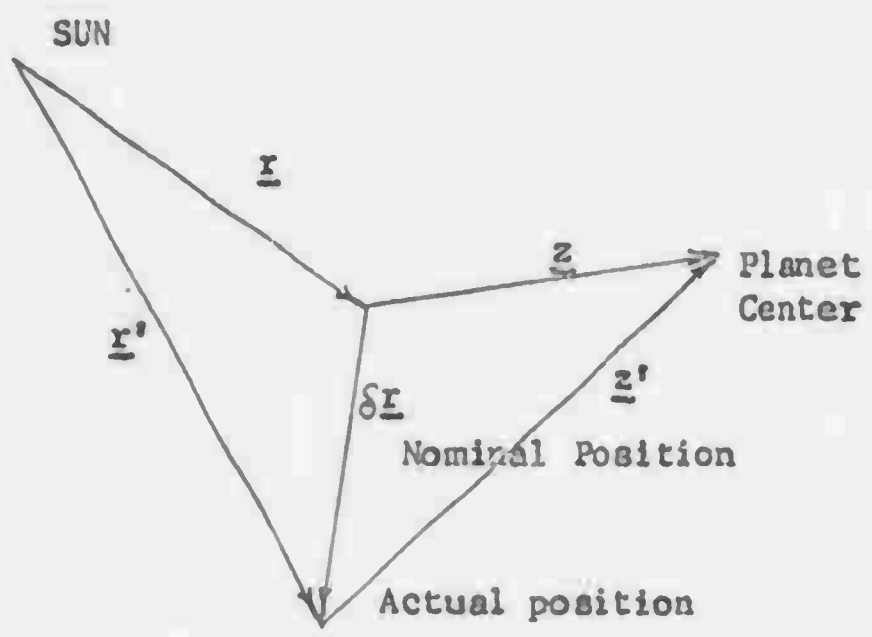


FIGURE 2

$$\underline{r} + \delta\underline{r} = \underline{r}'$$

$$\underline{z} - \delta\underline{r} = \underline{z}' = \underline{z} + \delta\underline{z} \quad \text{by definition}$$

$$\delta\underline{r} = -\delta\underline{z}$$

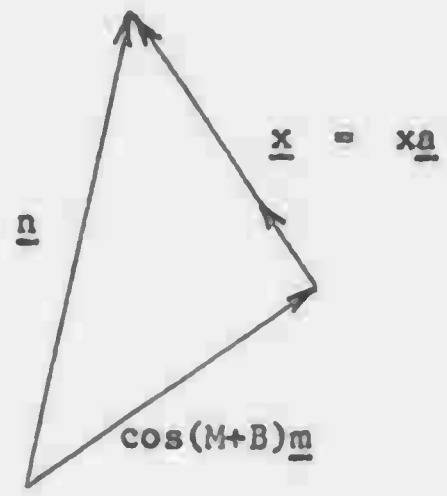
Rewriting (A-5) in terms of $\delta\underline{r}$:

$$z \sin(M+B)(\delta M + \delta B) = \left[\underline{n} - \underline{m}\cos(M+B)\right] \cdot \delta\underline{r} \qquad \text{(A-6)}$$

where $\underline{m}$ is the unit vector in the direction of planet center. From Figure 1:

$$\cos(M+B) = \underline{m} \cdot \underline{n} \qquad \text{the projection of } \underline{n} \text{ on } \underline{m}$$

Therefore, the bracketed quantity in (A-6) relates the two legs of a right triangle. Figure 3 shows that the magnitude of the resultant leg is sin(M+B).



$$n^2 - \cos^2(M+B) = x^2$$

$$1 - \cos^2(M+B) = \sin^2(M+B)$$

FIGURE 3

Rewriting (A-6) in terms of the vector $\underline{a}$:

$$\delta M + \delta B = \frac{\underline{a} \cdot \delta \underline{r}}{z} \qquad \text{(A-7)}$$

An expression for δB can be derived from Figure 1.

$$\sin B = \frac{D}{2z} \qquad \text{(A-8)}$$

Writing the perturbation of (A-8):

$$\cos B \, \delta B = - \frac{D}{2z^2} \delta z \qquad \text{(A-9)}$$

Substituting (A-4) and the results of Figure 2 in (A-9):

$$\delta B = \frac{-D}{2z^2 \cos B} \frac{\underline{z} \cdot \delta \underline{z}}{z}$$

$$\delta B = \frac{\underline{Dm}}{2z^2 \cos B} \cdot \delta \underline{r} \qquad \text{(A-10)}$$

Substituting (A-10) into (A-7):

$$\delta M = \frac{\underline{a} \cdot \delta \underline{r}}{z} - \frac{\underline{Dm} \cdot \delta \underline{r}}{2z^2 \cos B}$$

From Figure 1:

$$\sin B = \frac{D}{2z}$$

$$\delta M = \frac{1}{z}(\underline{a} - \tan \underline{Bm}) \cdot \delta \underline{r} \qquad \text{(A-11)}$$

It is shown in Figure 3 that $\underline{a}$ is perpendicular to $\underline{m}$. Figure 4 shows that the bracketed quantity in equation (A-11) defines the third leg of a right triangle.

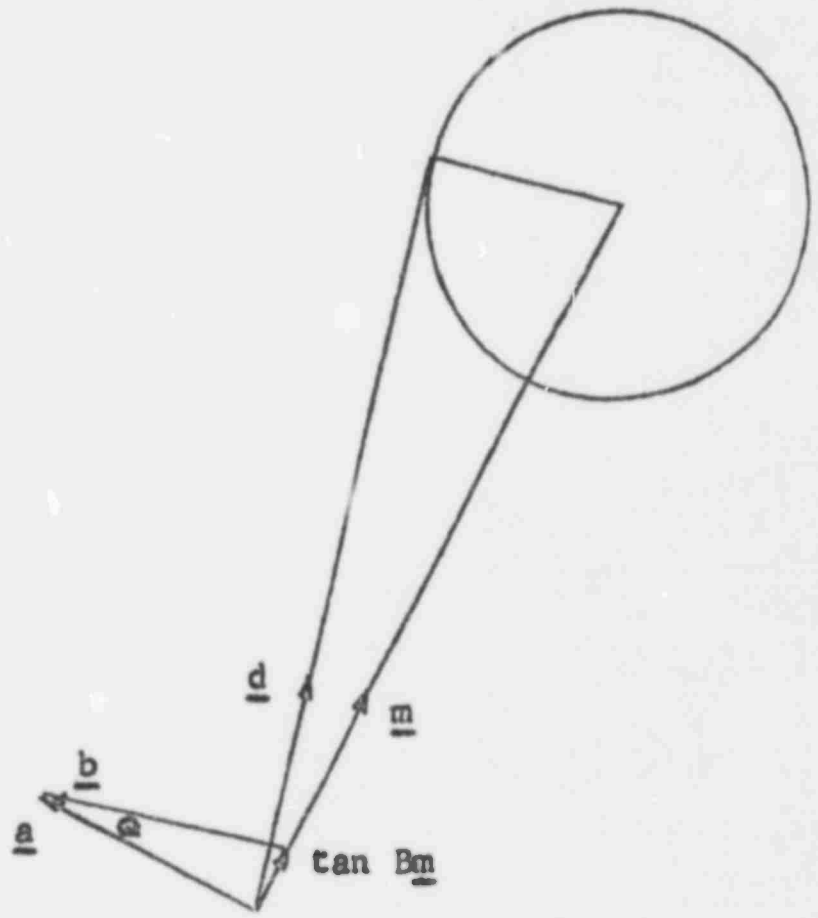


FIGURE 4

$$\tan Q = \frac{\tan B}{a} = \tan B$$

$$Q = B$$

$\underline{b}$ is perpendicular to $\underline{d}$, the unit vector in the direction of the planet edge.

Determining the magnitude of $\underline{b}$:

$$1 + \tan^2 B = b^2 = \sec^2 B$$

Defining the unit vector $\underline{p}$:

$$\underline{p} = \frac{\underline{b}}{\sec B}$$

Equation (A-11) can be rewritten:

$$\delta M = \frac{\underline{p} \cdot \delta \underline{r}}{z \cos B} \qquad \text{(A-12)}$$

Referring to equation (A-1), the vector which characterizes the star elevation measurement is:

$$\underline{h} = \frac{\underline{p}}{z \cos B} \qquad \text{(A-13)}$$

As noted in Figure 4, $\underline{p}$ is perpendicular to the planet edge and is therefore independent of the measured angle M. Hence, the expression for the deviation in M does not contain M explicitly. The $\underline{h}$ vector can be written directly from the orbital geometry.

The assumption of a circular orbit at a known altitude serves to completely determine the measurement vector. The expression is derived in Figure 5, in the local vertical co-ordinate system, rsz.

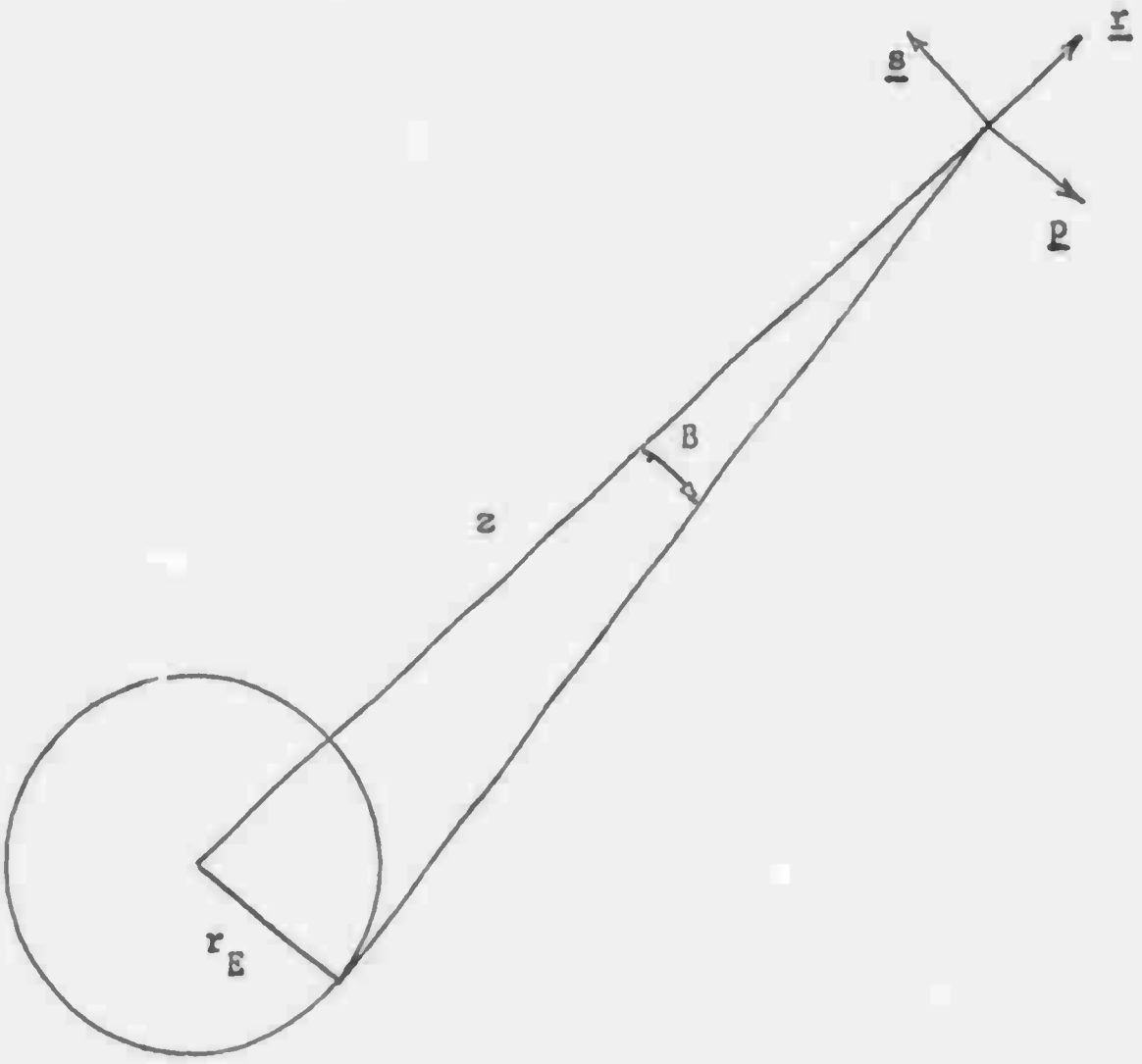


FIGURE 5

LOCAL VERTICAL CO-ORDINATE SYSTEM

$$\underline{p} = -\cos B\underline{s} + \sin B\underline{r}$$

$$\sin B = \frac{r_E}{z}$$

$$\cos B = \frac{(z^2 - r_E^2)^{\frac{1}{2}}}{z}$$

From (A-13).

$$\underline{h} = \frac{-\cos B\underline{s} + \sin B\underline{r}}{z \cos B}$$

$$\underline{h} = -\frac{1}{z}\underline{s} + \frac{1}{z}\tan B\underline{r}$$

$$\underline{h} = -\frac{1}{z}\underline{s} + \frac{r_E}{z(z^2 - r_E^2)^{\frac{1}{2}}}\underline{r} \qquad \text{(A-14)}$$

$$\underline{h} = \begin{bmatrix} h_r \\ h_s \end{bmatrix}$$

If the left horizon is used, the $\underline{s}$-component changes sign.

In order that it be compatible with the 4 X 4 state transition matrix, the measurement vector is modified to the 4-dimensional vector $\underline{g}$, whose first two elements are identical with $\underline{h}$.

$$\underline{g} = \begin{bmatrix} h_r \\ h_s \\ 0 \\ 0 \end{bmatrix} \qquad \text{(A-15)}$$

Equation (A-1) then becomes:

$$\delta q = \underline{g} \cdot \delta \underline{x} \qquad \text{(A-16)}$$

where $\delta \underline{x}$ is the complete state deviation vector.

$$\underline{X} = \begin{bmatrix} \underline{r} \\ \underline{v} \end{bmatrix} \tag{A-17}$$

APPENDIX B

ERROR CORRELATION MATRIX

The error vector $\underline{e}$ may be defined as the difference between the estimated and true values of the state deviation vector at any point in the trajectory.

$$\underline{e}_n = \delta\hat{\underline{x}}_n - \delta\underline{x}_n \qquad (B\text{-}1)$$

The covariance matrix of the error vector is of interest since it describes the uncertainty in the estimated position and velocity deviation of the spacecraft.

$$E_n = \overline{\underline{e}_n \underline{e}_n^T} \qquad (B\text{-}2)$$

Battin (Reference 2) develops a recursion formula for the error correlation matrix which will be presented in this section. It is then shown that the recursion formula is equivalent to a more convenient representation using the inverse of the covariance matrix. It is this latter form, referred to target co-ordinates, which is utilized in the computer program. Definition of the symbols to be used is necessary before proceeding. The following superscripts, applying to the quantities $\delta\underline{x}$, δq and $\underline{e}$, are defined:

$\sim$ observation

$'$ extrapolation

$\wedge$ estimation

The lack of a superscript on these symbols indicates their true values. The extrapolated vector is that resulting from the same vector at a previous time, if no new measurements intervene.

$$\delta \underline{x}'_n = C_{n,n-1}\, \delta \underline{x}_{n-1} \qquad \text{(B-3)}$$

Equation (B-3) defines the state-transition matrix, C. The extrapolated error matrix is encountered in the subsequent derivation and is defined as follows:

$$\underline{e}'_n = \delta \underline{\hat{x}}'_n - \delta \underline{x}'_n = C_{n,n-1}\, \delta \underline{\hat{x}}_{n-1} - C_{n,n-1}\, \delta \underline{x}_{n-1}$$

$$\underline{e}'_n = C_{n,n-1}\, \underline{e}_{n-1}$$

$$E'_n = \overline{\underline{e}'_n \underline{e}'^{T}_n} = C_{n,n-1}\, \overline{\underline{e}_{n-1} \underline{e}^{T}_{n-1}}\, C^{T}_{n,n-1}$$

$$E'_n = C_{n,n-1}\, E_{n-1}\, C^{T}_{n,n-1} \qquad \text{(B-4)}$$

An equation similar to (B-4) transfers any matrix from its local co-ordinate system to a reference system, target co-ordinates in this case.

$$E^{a}_n = C_{an}\, E_n\, C^{T}_{an} \qquad \text{(B-5)}$$

The same relation must hold true for the extrapolated matrices.

$$E'^{a}_n = C_{an}\, E'_n\, C^{T}_{an} \qquad \text{(B-6)}$$

Substituting from equations (B-4) and (B-5):

$$E'^{a}_n = C_{an} C_{n,n-1}\, E_{n-1}\, C^{T}_{n,n-1}\, C^{T}_{an}$$

$$E_n'^a = C_{a,n-1}\, E_{n-1}\, C_{a,n-1}^T$$

$$E_n'^a = E_{n-1}^a \qquad \text{(B-7)}$$

Thus, the estimation error covariance matrix at point n, just prior to the n^{th} measurement, equals the matrix just after the $(n-1)^{th}$ measurement when they are referred to the same co-ordinates. This is a clearer definition of the extrapolated covariance matrix than that which equation (B-4) provides.

A linear estimate of the state deviation vector $\underline{x}_n$ at time t_n is given in equation (B-8).

$$\delta \hat{\underline{x}}_n = \delta \hat{\underline{x}}_n' + \underline{w}_n(\delta \tilde{q}_n - \delta \hat{q}_n') \qquad \text{(B-8)}$$

The best estimate is the extrapolated estimate, defined in equation (B-3), plus the weighted difference between the observed and extrapolated measurement deviations. The extrapolated quantity, $\delta \hat{q}_n'$, defined as in equation (A-16), is what the measurement deviation is expected to be.

$$\delta \hat{q}_n' = \underline{g}_n^T \delta \hat{\underline{x}}_n' \qquad \text{(B-9)}$$

The vector $\underline{g}_n$ is the 4-dimensional measurement vector defined in Appendix A. The weighting vector, $\underline{w}_n$, in equation (B-6), is a function of the covariance matrix. The observed measurement deviation differs from its true value by the measurement error a, a random variable assumed to have zero mean value.

$$\delta \tilde{q}_n = \delta q_n + a \qquad \text{(B-10)}$$

Equation (B-8) is substituted into (B-1) to obtain an expression for the error vector in terms of known quantities.

$$\underline{e}_n = \delta\hat{\underline{x}}_n - \delta\underline{x}_n + \underline{w}_n\left[\underline{g}_n^T\,\delta\underline{x}_n + a - \underline{g}_n^T\,\delta\hat{\underline{x}}_n\right]$$

$$\underline{e}_n = \delta\hat{\underline{x}}_n' - \delta\underline{x}_n + \underline{w}_n\left[\underline{g}_n^T\,\delta\underline{x}_n + a - \underline{g}_n^T(\delta\underline{x}_n' + \underline{e}_n')\right] \qquad \text{(B-11)}$$

The vector $\delta\hat{\underline{x}}_n'$ is equivalent to $\delta\underline{x}_n$ since the actual state deviation at point n does not change when a new measurement is made. Therefore, in equation (B-11):

$$\delta\hat{\underline{x}}_n' - \delta\underline{x}_n = \delta\hat{\underline{x}}_n' - \delta\underline{x}_n' = \underline{e}_n'$$

$$\underline{g}_n^T\,\delta\underline{x}_n = \underline{g}_n^T\,\delta\underline{x}_n'$$

Rewriting equation (B-11):

$$\underline{e}_n = (I - \underline{w}_n\underline{g}_n^T)\underline{e}_n' + \underline{w}_n a \qquad \text{(B-12)}$$

From equation (B-2), the covariance matrix as a function of $\underline{w}_n$ is given in (B-13).

$$E_n = (I - \underline{w}_n\underline{g}_n^T)E_n'(I - \underline{g}_n\underline{w}_n^T) + \underline{w}_n\underline{w}_n^T\,\overline{a^2} \qquad \text{(B-13)}$$

In equation (B-13), the error vector and the measurement error are assumed independent. Then the cross terms $\bar{a}\,\bar{\underline{e}}_n'$ equal zero since the average measurement error is assumed zero. Another result of this assumption is the expression for the variance of the measurement error.

$$\sigma^2 = \overline{a^2} - \bar{a}^2 = \overline{a^2} \qquad \text{(B-14)}$$

Since the weighting vector, $\underline{w}_n$, is arbitrary, it can be chosen

to minimize the trace of the covariance matrix, thus minimizing both the position and velocity errors. Writing the perturbation equation of the trace of (B-13):

$$\delta \operatorname{tr}[E_n] = \operatorname{tr}[\delta E_n] = \operatorname{tr}\Big[(-\delta \underline{w}_n \underline{g}_n^T) E_n' (I - \underline{g}_n \underline{w}_n^T) \cdots$$

$$+ (I - \underline{w}_n \underline{g}_n^T) E_n' (-\underline{g}_n \delta \underline{w}_n^T) + \delta \underline{w}_n \underline{w}_n^T \sigma^{-2} + \underline{w}_n \delta \underline{w}_n^T \sigma^{-2} = 0 \qquad \text{(B-15)}$$

Since E_n' is symmetric by definition, the first two terms of equation (B-15) and the last two are the transpose of one another. The following relations hold for the trace:

$$\operatorname{tr}[A^T] = \operatorname{tr}[A]$$

$$\operatorname{tr}[A + B] = \operatorname{tr}[A + B^T]$$

Applying these relations to (B-15):

$$2 \operatorname{tr}\Big[(-\delta \underline{w}_n \underline{g}_n^T) E_n' (I - \underline{g}_n \underline{w}_n^T) + \underline{w}_n \underline{w}_n^T \sigma^2\Big] = 0$$

$$2 \operatorname{tr}\left\{\delta \underline{w}_n \left[\underline{w}_n^T \sigma^{-2} - \underline{g}_n^T E_n' (I - \underline{g}_n \underline{w}_n^T)\right]\right\} = 0 \qquad \text{(B-16)}$$

Since the weighting vector is arbitrary, the portion of (B-16) multiplied by $\delta \underline{w}_n$ must equal zero.

$$(\sigma^{-2} + \underline{g}_n^T E_n' \underline{g}_n) \underline{w}_n^T = \underline{g}_n^T E_n' \qquad \text{(B-17)}$$

The quantity $\underline{g}_n^T E_n' \underline{g}_n$ is a quadratic form, which makes it a scalar. Therefore the bracketed quantity in (B-17) is a scalar. The transpose of the optimum weighting vector then has the form:

$$\underline{w}_n^T = \frac{\underline{g}_n^T E_n'}{\sigma^{-2} + \underline{g}_n^T E_n' \underline{g}_n} \qquad \text{(B-18)}$$

Since the matrix E_n' is symmetric:

$$\underline{w}_n = \frac{E_n' \underline{g}_n}{\sigma^2 + \underline{g}_n^T E_n' \underline{g}_n} \tag{B-19}$$

The recursion formula for the covariance matrix is obtained by substituting (B-18) and (B-19) back into equation (B-13).

$$E_n = \left(I - \frac{E_n' \underline{g}_n \underline{g}_n^T}{\sigma^2 + \underline{g}_n^T E_n' \underline{g}_n}\right) E_n' \left(I - \frac{\underline{g}_n \underline{g}_n^T E_n'}{\sigma^2 + \underline{g}_n^T E_n' \underline{g}_n}\right) + \frac{E_n' \underline{g}_n \underline{g}_n^T E_n'^2}{(\sigma^2 + \underline{g}_n^T E_n' \underline{g}_n)^2}$$

$$E_n = E_n' - \frac{E_n' \underline{g}_n \underline{g}_n^T E_n'}{\sigma^2 + \underline{g}_n^T E_n' \underline{g}_n} - \frac{E_n' \underline{g}_n \underline{g}_n^T E_n'}{\sigma^2 + \underline{g}_n^T E_n' \underline{g}_n} + \frac{E_n' \underline{g}_n (\underline{g}_n^T E_n' \underline{g}_n) \underline{g}_n^T E_n'}{(\sigma^2 + \underline{g}_n^T E_n' \underline{g}_n)^2} \cdots$$

$$+ \frac{E_n' \underline{g}_n \underline{g}_n^T E_n'^2}{(\sigma^2 + \underline{g}_n^T E_n' \underline{g}_n)^2}$$

$$E_n = E_n' - 2\left[\frac{E_n' \underline{g}_n \underline{g}_n^T E_n'}{\sigma^2 + \underline{g}_n^T E_n' \underline{g}_n}\right] + \frac{(\underline{g}_n^T E_n' \underline{g}_n + \sigma^2)(E_n' \underline{g}_n \underline{g}_n^T E_n')}{(\underline{g}_n^T E_n' \underline{g}_n + \sigma^2)^2}$$

$$E_n = E_n' - \frac{E_n' \underline{g}_n \underline{g}_n^T E_n'}{\sigma^2 + \underline{g}_n^T E_n' \underline{g}_n} \tag{B-20}$$

The recursion formula for the estimation error covariance matrix is somewhat inconvenient for the steepest descent method of solution. The influence coefficient expression obtained by using (B-20) would be difficult to work with. A less complicated expression, involving the inverse of the E_n matrix, will be proved valid. The proposed formula is given in equation (B-21).

$$E_n^{-1} = (E_n')^{-1} + \frac{\underline{g}_n \underline{g}_n^T}{\sigma^2} \tag{B-21}$$

If equation (B-21) is true, then its product with (B-20) should equal the identity matrix.

$$I_4 \stackrel{?}{=} \left[E_n' - \frac{E_n' g_n g_n^T E_n'}{g_n^T E_n' g_n + \sigma^2}\right]\left[(E_n')^{-1} + \frac{g_n g_n^T}{\sigma^2}\right]$$

$$I_4 \stackrel{?}{=} I_4 - \frac{E_n' g_n g_n^T}{g_n^T E_n' g_n + \sigma^2} + \frac{E_n' g_n g_n^T}{\sigma^2} - \frac{E_n' g_n (g_n^T E_n' g_n) g_n^T}{2(g_n^T E_n' g_n + \sigma^2)}$$

$$I_4 \stackrel{?}{=} I_4 + \frac{-2E_n' g_n g_n^T + (g_n^T E_n' g_n + 2)E_n' g_n g_n^T - (g_n^T E_n' g_n)E_n' g_n g_n^T}{\sigma^2(g_n^T E_n' g_n + \sigma^2)}$$

$$I_4 = I_4$$

Equation (B-21) will be used instead of (B-20) because of its simpler form.

The estimation error covariance matrix at the target is needed for the cost function. Equation (B-5) is used to refer (B-21) to target co-ordinates.

$$E_n^a = C_{an} E_n C_{an}^T \qquad \text{(B-5)}$$

$$(E_n^a)^{-1} = (C_{an}^T)^{-1} E_n^{-1} C_{an}^{-1}$$

From (B-21):

$$(E_n^a)^{-1} = (C_{an}^T)^{-1}(E_n')^{-1} C_{an}^{-1} + \frac{(C_{an}^T)^{-1} g_n g_n^T C_{an}^{-1}}{\sigma^2}$$

$$(E_n^a)^{-1} = (E_n'^a)^{-1} + \frac{f_n f_n^T}{\sigma^2} \qquad \text{(B-22)}$$

The vector f_n is defined as the 4-dimensional measurement vector in target co-ordinates. Equation (B-7) provides a further simplifica-

tion.

$$(E_n{}^a)^{-1} = (E_{n-1}{}^a)^{-1} + \frac{\underline{f}_n \underline{f}_n{}^T}{\sigma^2} \qquad \text{(B-23)}$$

If there are N measurements to be made, equation (B-23) can be applied N times, where E_o is the initial covariance matrix.

$$(E_N{}^a)^{-1} = (E_o{}^a)^{-1} + \sum_{k=1}^{N} \frac{\underline{f}_k \underline{f}_k{}^T}{\sigma^2} \qquad \text{(B-24)}$$

From equation (B-7), since the N^{th} measurement is the last:

$$(E_a{}'^a)^{-1} = (E_N{}^a)^{-1} = (E_o{}^a)^{-1} + \sum_{k=1}^{N} \frac{\underline{f}_k \underline{f}_k{}^T}{\sigma^2} \qquad \text{(B-25)}$$

Henceforth, the left hand member of (B-25) will be referred to as either $E_a{}^{-1}$ or A.

$$E_a{}^{-1} = A = (E_o{}^a)^{-1} + \sum_{k=1}^{N} \frac{\underline{f}_k \underline{f}_k{}^T}{\sigma^2} \qquad \text{(B-26)}$$

APPENDIX C

STATE TRANSITION MATRIX

In Stern's thesis (Reference 8), a general formulation of the state transition matrix for a two-body conic was derived and is reprinted in equation (C-1). The development was carried out in the pqz or flight path co-ordinate system because of the relative simplicity of matrix operations. The pqz system is a member of a class known as reference trajectory co-ordinate systems, since its fundamental plane is the plane of the nominal orbit. The system rotates about its z-axis, which is defined perpendicular to the fundamental plane. The angular velocity of this rotation equals that of the vehicle's velocity vector. The q-axis lies parallel to the velocity vector and the p-axis forms the right-handed triad. Since, in a circular orbit, the velocity vector is always perpendicular to the position vector, the p-axis lies along the position vector. Thus, it is obvious that the flight-path co-ordinate system is identical to the more familiar local-vertical system. This is a fortunate result since the measurement vectors developed in Appendix A are more easily defined in the local vertical.

For the circular orbit case, $e = 0$ in Stern's formula and the eccentric and true anomalies are identical. In addition, the third and sixth rows and columns are eliminated since only the in-plane

$$C_{ji} = \begin{bmatrix} M_{ji} & N_{ji} \\ S_{ji} & T_{ji} \end{bmatrix}$$

M_{ji} — (c-1b)

N_{ji} — (c-1c)

S_{ji} — (c-1d)

T_{ji} — (c-1e)

(c.1a)

Symbols:

e	eccentricity
E	eccentric anomaly
$E_M =$	$\frac{1}{2}(E_j - E_i)$
$E_P =$	$\frac{1}{2}(E_j + E_i)$
n	mean angular motion

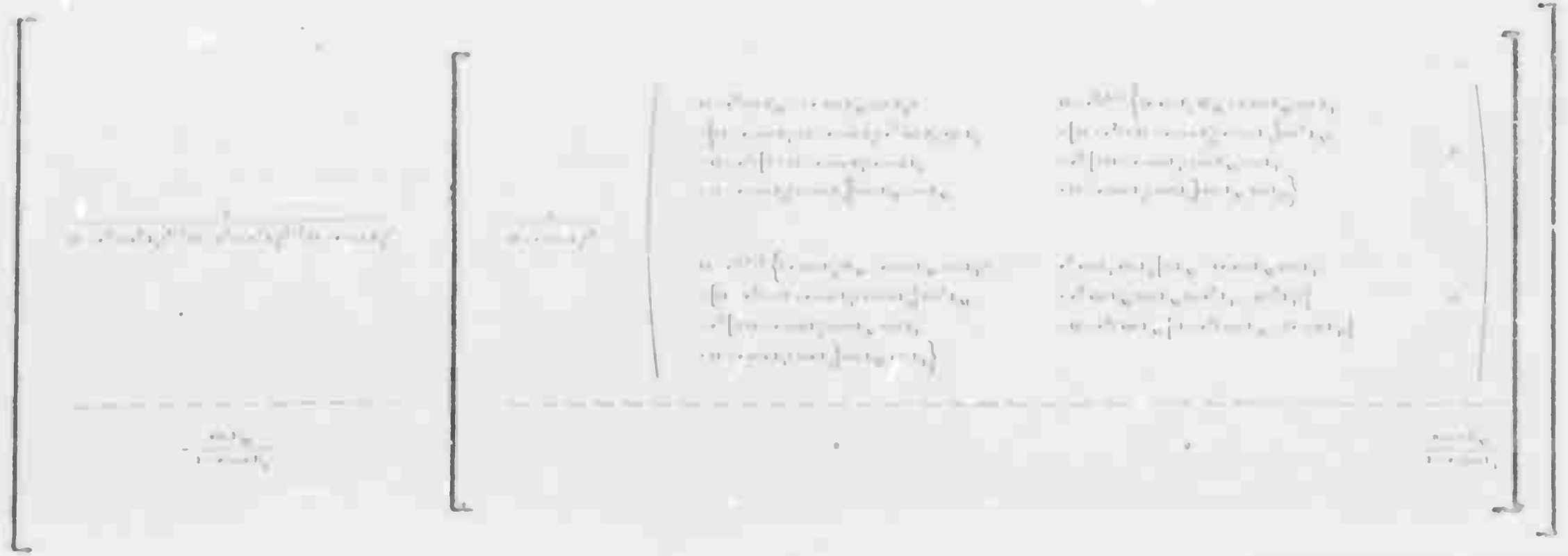
$$M_{j1} = \begin{bmatrix} & & \\ & \left[\quad\right] & \\ & & \left[\quad\right] \end{bmatrix} +$$

(c 12b)

$$N_{j1} = 2$$

(C-1c)

$$S_{ji} = \left[\quad \right] +$$



$$\pi_{j1} = 2$$

deviations from position are to be considered. The result is the 4 X 4 matrix in equation (C-2), where $f_M = \frac{1}{2}(f_j - f_i)$. The factor n is the mean angular motion, defined as 2 divided by the orbital period. Further simplification is possible since:

$$2 \sin\left(\frac{f_j - f_i}{2}\right) \cos\left(\frac{f_j - f_i}{2}\right) = \sin(f_j - f_i) \qquad \text{(C-3)}$$

Also, since the matrix will be used to refer all measurements to target co-ordinates, the j can be replaced by a. Equation (C-4) is the resulting 4 X 4 matrix C_{ai} .

The derivative of the state transition matrix with respect to time is needed in the determination of influence coefficients. This derivative will be denoted D_{ai} and is given in equation (C-5). Time is explicit in this formulation since nt = f .

A useful property of the state-transition matrix will be utilized to simplify computational procedures. As shown by Battin (Reference 1), the state-transition matrix is symplectic, that is, its inverse can be computed by the simple formula:

$$C^{-1} = -JC^{T}J \qquad \text{(C-6)}$$

$$\text{where } J = \begin{pmatrix} 0 & I \\ -I & 0 \end{pmatrix}$$

If the C matrix is partitioned and the rule applied, the inverse of C is shown to be a rearrangement of its elements.

$$C^{-1} = -J\begin{pmatrix} C_1 & C_2 \\ C_3 & C_4 \end{pmatrix}^{T} J$$

$$C_{ji} = \left[\begin{bmatrix} 1+2\sin^2 f_m & 2\sin f_m \cos f_m \\ -6f_m+4\sin f_m \cos f_m & 1-4\sin^2 f_m \end{bmatrix}^{\frac{1}{n}} \begin{bmatrix} 2\sin f_m \cos f_m & 4\sin^2 f_m \\ -4\sin^2 f_m & -6f_m+8\sin f_m \cos f_m \end{bmatrix} \right]$$

$$\left[\begin{bmatrix} 6f_m-2\sin f_m \cos f_m & 2\sin^2 f_m \\ -2\sin^2 f_m & -2\sin f_m \cos f_m \end{bmatrix}^{n} \begin{bmatrix} 1+2\sin^2 f_m & 6f_m-4\sin f_m \cos f_m \\ -2\sin f_m \cos f_m & 1-4\sin^2 f_m \end{bmatrix} \right]$$

$$C_{ai} = \begin{bmatrix} 1+2\sin^2\left(\frac{f_a-f_i}{2}\right) & \sin\left(f_a-f_i\right) & \frac{1}{n}\sin\left(f_a-f_i\right) & \frac{4}{n}\sin^2\left(\frac{f_a-f_i}{2}\right) \\ -3\left(f_a-f_i\right) & +2\sin\left(f_a-f_i\right) & 1-4\sin^2\left(\frac{f_a-f_i}{2}\right) & -\frac{4}{n}\sin^2\left(\frac{f_a-f_i}{2}\right) \\ 3n\left(f_a-f_i\right)-n\sin\left(f_a-f_i\right) & 2n\sin^2\left(\frac{f_a-f_i}{2}\right) & 1+2\sin^2\left(\frac{f_a-f_i}{2}\right) & 3\left(f_a-f_i\right)-2\sin\left(f_a-f_i\right) \\ -2n\sin^2\left(\frac{f_a-f_i}{2}\right) & -n\sin\left(f_a-f_i\right) & -\sin\left(f_a-f_i\right) & 1-4\sin^2\left(\frac{f_a-f_i}{2}\right) \end{bmatrix} - \frac{3}{n}\left(f_a-f_i\right)+\frac{4}{n}\sin\left(f_a-f_i\right)$$

$$D_{ai}\delta t_i = \begin{bmatrix} -n\sin(f_a-f_i) & -n\cos(f_a-f_i) & -\cos(f_a-f_i) & -2\sin(f_a-f_i) \\ 3n-2n\cos(f_a-f_i) & 2n\sin(f_a-f_i) & 2\sin(f_a-f_i) & 3-4\cos(f_a-f_i) \\ -3n^2+n^2\cos(f_a-f_i) & -n^2\sin(f_a-f_i) & -n\sin(f_a-f_i) & -3n+2n\cos(f_a-f_i) \\ n^2\sin(f_a-f_i) & n^2\cos(f_a-f_i) & n\cos(f_a-f_i) & 2n\sin(f_a-f_i) \end{bmatrix} \delta t_i$$

$$C^{-1} = \begin{pmatrix} 0 & -I \\ I & 0 \end{pmatrix} \begin{pmatrix} C_1^T & C_3^T \\ C_2^T & C_4^T \end{pmatrix} \begin{pmatrix} 0 & I \\ -I & 0 \end{pmatrix}$$

$$C^{-1} = \begin{pmatrix} -C_2^T & -C_4^T \\ C_1^T & C_3^T \end{pmatrix} \begin{pmatrix} 0 & I \\ -I & 0 \end{pmatrix}$$

$$C^{-1} = \begin{pmatrix} C_4^T & -C_2^T \\ -C_3^T & C_1^T \end{pmatrix} \qquad \text{(C-7)}$$

The transpose of the state-transition matrix is, of course, also symplectic. In the computation of influence coefficients, the derivatives of both the inverse and the inverse transpose are required. The fact that C and C^T are symplectic greatly simplifies these computations.

In the program, the state-transition matrix C, and its derivative D, are computed according to equations (C-4) and (C-5) respectively. The inverse and inverse transpose contain the same elements as C in different positions. Therefore, their derivatives will contain the same elements as D in these positions also. The easiest way to form these derivatives is by applying equation (C-7) to the elements of D and D^T. It is important to note that the results are not D^{-1} and $(D^T)^{-1}$, but rather the derivatives of C^{-1} and $(C^T)^{-1}$. The formula for the inverse of a symplectic matrix does not hold when applied to D and D^T, which are not symplectic. Equation (C-7) is merely the easiest programming method which results in the desired quantities: $\frac{d}{dt}(C^{-1})$ and $\frac{d}{dt}(C^T)^{-1}$.

APPENDIX D

INFLUENCE COEFFICIENTS - TIME CHANGE

The relation of the cost function to the independant variable is developed in Chapter 3. For this problem, the times of the decision points are varied and the relation has the following form:

$$\delta \text{Cost} = \sum_k \frac{\partial \text{Cost}}{\partial t_k} \delta t_k = \sum_k L_k \delta t_k \qquad \text{(D-1)}$$

The quantity L_k, the influence coefficient, determines the change in cost due to a change at the k^{th} decision point. The relation is accurate to the first order, assuming that the time change is small enough so that linearity is guaranteed. As explained in Chapter 2, the cost function is the mean-squared error in position estimation at the target.

$$\text{Cost} = \text{tr}\left[QE_a\right]$$

$$Q = \begin{bmatrix} I_2 & 0_2 \\ 0_2 & 0_2 \end{bmatrix} \qquad \text{(D-2)}$$

The correlation matrix of estimation errors, E_a, is premultiplied by Q so that only its first two diagonal elements are used to determine the cost function. For a different cost, Q would have a different

form, or might not be necessary. It is used only as a convenient means for expressing the cost function in equation form and is not essential to the subsequent derivation.

From equations (D-1) and (D-2), the expression for the k^{th} influence coefficient is developed as follows:

$$L_k = \frac{\partial}{\partial t_k} \operatorname{tr}\left[QE_a\right]$$

$$L_k = \operatorname{tr} \frac{\partial}{\partial t_k}\left[QE_a\right]$$

$$L_k = \operatorname{tr}\left[Q \frac{\partial E_a}{\partial t_k}\right] \qquad \text{(D-3)}$$

An expression for the error correlation matrix was determined in Appendix B.

$$E_a^{-1} = A = (E_o^a)^{-1} + \sum_{k=1}^{N} \frac{\underline{f}_k \underline{f}_k^T}{\sigma^2} \qquad \text{(B-24)}$$

E_o is the error matrix at target due to initial estimation errors, σ^2 is the variance of the measurement noise and $\underline{f}_k$ is the modified measurement vector expressed in target co-ordinates. Equation (B-24) is denoted A to simplify the mathematics.

$$E_a A = I \qquad \text{(D-4)}$$

The partial derivative of E_a can be written as follows:

$$\frac{\partial E_a}{\partial t_k} A + E_a \frac{\partial A}{\partial t_k} = 0$$

$$\frac{\partial E_a}{\partial t_k} = - E_a \frac{\partial A}{\partial t_k} A^{-1}$$

$$\frac{\partial E_a}{\partial t_k} = - E_a \frac{\partial \Lambda}{\partial t_k} E_a \qquad \text{(D-5)}$$

the last step following from (D-4).

Substituting equation (D-5) into (D-3):

$$L_k = -tr \left[Q E_a \frac{\partial A}{\partial t_k} E_a \right] \qquad \text{(D-6)}$$

An expression for the partial derivative of A is needed to complete the derivation. Since the measurement vectors are defined in the local vertical co-ordinate system for each of the decision points, the state-transition matrix must be used to transfer them to target co-ordinates. This was done in Appendix B, but will be considered here in a slightly different manner. As noted in Appendix A, the type of measurement used determines a unique $\underline{g}$ vector, the measurement vector in local vertical co-ordinates with two added zeroes. The star elevation angle will be used at each point, therefore the k^{th} measurement vector in the two co-ordinate systems are related by equation (D-7).

$$M = \underline{g}_k^T \underline{x}_k = \underline{f}_k^T \underline{x}_k^a \qquad \text{(D-7)}$$

M is the measured angle and $\underline{x}$ is the state vector, where the superscript a refers to the target co-ordinate system. The state-transition matrix is defined by equation (D-8).

$$\underline{x}_k^a = C_{ak} \underline{x}_k \qquad \text{(D-8)}$$

Rewriting (D-7):

$$\underline{g}_k^T \underline{x}_k = \underline{f}_k^T C_{ak} \underline{x}_k$$

$$\therefore \quad \underline{g}_k^T = \underline{f}_k^T C_{ak}$$

$$\underline{g}_k = C_{ak}^T \underline{f}_k$$

$$\underline{f}_k = (C_{ak}^T)^{-1} \underline{g}_k \qquad \text{(D-9)}$$

Each of the N identical measurement vectors are transferred to target co-ordinates by equation (D-9).

Referring to equation (B-5):

$$E_n^a = C_{an} E_n C_{an}^T \qquad \text{(B-5)}$$

Applying (B-5) to the initial error matrix, E_o^a, equation (B-24) can now be written as an explicit function of time.

$$E_a^{-1} = A = (C_{ao} E_o C_{ao}^T)^{-1} + \sum_{k=1}^{N} \frac{(C_{ak}^T)^{-1} \underline{g}_k \underline{g}_k^T C_{ak}^{-1}}{\sigma^2} \qquad \text{(D-10)}$$

The influence coefficient, equation (D-6), requires the partial derivative of A with respect to t_k.

$$\frac{\partial A}{\partial t_k} = \frac{1}{\sigma^2} \left[\frac{\partial (C_{ak}^T)^{-1}}{\partial t_k} \underline{g}_k \underline{g}_k^T C_{ak}^{-1} + (C_{ak}^T)^{-1} \underline{g}_k \underline{g}_k^T \frac{\partial C_{ak}^{-1}}{\partial t_k} \right] \qquad \text{(D-11)}$$

A simplified expression for the influence coefficient can now be obtained, using the following identities involving the trace.

$$\text{tr}[AB] = \text{tr}[BA]$$

$$\text{tr}[A + B] = \text{tr}[A + B^T]$$

From (D-6):

$$L_k = -\mathrm{tr}\left[QE_a \frac{\partial A}{\partial t_k} E_a\right]$$

$$L_k = -\mathrm{tr}\left[\frac{\partial A}{\partial t_k} E_a Q E_a\right]$$

$$L_k = -\frac{1}{\sigma^2}\,\mathrm{tr}\left[\left(\frac{(C_{ak}^T)^{-1}}{t_k}\,\xi_k\xi_k^T C_{ak}^{-1} + (C_{ak}^T)^{-1}\xi_k\xi_k^T\,\frac{C_{ak}^{-1}}{t_k}\right) E_a Q E_a\right]$$

$$L_k = -\frac{2}{\sigma^2}\,\mathrm{tr}\left[\frac{\partial (C_{ak}^T)^{-1}}{\partial t_k}\,\xi_k\xi_k^T\, C_{ak}^{-1}\, E_a Q E_a\right]$$

$$L_k = -\frac{2}{\sigma^2}\,\mathrm{tr}\left[\xi_k^T\, C_{ak}^{-1}\, E_a Q E_a\,\frac{\partial (C_{ak}^T)^{-1}}{\partial t_k}\,\xi_k\right] \qquad \text{(D-12)}$$

The expression in brackets in equation (D-12) is in the quadratic form, hence it is a scalar quantity and the trace notation can be dropped. The final expression for the k^{th} influence coefficient is given in (D-13).

$$L_k = -\frac{2}{\sigma^2}\,\xi_k^T\, C_{ak}^{-1}\, E_a Q E_a\,\frac{(C_{ak}^T)^{-1}}{t_k}\,\xi_k \qquad \text{(D-13)}$$

APPENDIX E

INFLUENCE COEFFICIENTS - HORIZON CHANGE

The influence coefficients associated with a possible change of horizon have the same function as those associated with the time change and the cost relation has a similar form:

$$\delta \text{ Cost} = \sum_k \frac{\partial \text{ Cost}}{\partial \underline{\ell}_k} \delta \underline{\ell}_k = \sum_k \underline{L}_k^T \delta \underline{\ell}_k \qquad \text{(E-1)}$$

The vector $\delta \underline{\ell}_k$ is defined in equation (E-2), where $\underline{\ell}_k$ is the measurement vector expressed in the local vertical system at the k^{th} point.

$$\underline{\ell}_k(\text{NEW}) = \underline{\ell}_k(\text{OLD}) + \delta \underline{\ell}_k \qquad \text{(E-2)}$$

The influence coefficient $\underline{L}_k$, in equation (E-1), is a column vector whose scalar product with $\delta \underline{\ell}_k$ is the cost change resulting from a change in horizon at the k^{th} point. As indicated in (E-1), the sum of the k cost changes should result in the total cost change, if the assumption of linearity is valid. The same cost function, position estimation error, is used and so the Q matrix, whose purpose was explained in Appendix D, is retained.

$$\text{Cost} = \text{tr}\left[QE_a\right] \qquad \text{(E-3)}$$

The expression for the cost change at the k^{th} point is found by

writing the perturbation equation of (E-3) for a change of horizon.

$$(\delta \text{ Cost})_k = \delta \text{ tr}\left[QE_a\right]$$

$$(\delta \text{ Cost})_k = \text{tr}\left\{\delta\left[QE_a\right]\right\}$$

$$(\delta \text{ Cost})_k = \text{tr}\left[Q\delta E_a\right] \qquad \text{(E-4)}$$

Equation (D-10) provides an expression in terms of known quantities for the inverse of the estimation error covariance matrix.

$$E_a^{-1} = A = (C_{ao}E_oC_{ao}^T)^{-1} + \sum_{k=1}^{N} \frac{(C_{ak}^T)^{-1}g_kg_k^TC_{ak}^{-1}}{\sigma^2} \qquad \text{(D-10)}$$

The perturbation of E_a can be written as a function of A, similar to the development in Appendix D.

$$E_aA = I$$

$$\delta E_aA + E_a\delta A = 0$$

$$\delta E_a = -E_a\delta A\, A^{-1}$$

$$\delta E_a = -E_a\delta A E_a \qquad \text{(E-5)}$$

Referring to (E-4), the k^{th} cost change now has the form:

$$(\delta \text{ Cost})_k = -\text{tr}\left[QE_a\delta A\, E_a\right] \qquad \text{(E-6)}$$

The trace of a product of matrices can be rearranged as follows:

$$\text{tr}\left[AB\right] = \text{tr}\left[BA\right] \qquad \text{(E-7)}$$

Applying (E-7) to (E-6), with "A" in (E-7) equal to E_a, results in

a more convenient expression.

$$(\delta \text{ Cost})_k = - \text{tr}\left[E_a Q E_a \delta A\right] \qquad \text{(E-8)}$$

The perturbation of A, at the k^{th} point, can be written directly from equation (D-10).

$$\delta A = \frac{1}{\sigma^2}\left[(C_{ak}^T)^{-1}\delta g_k g_k^T C_{ak}^{-1} + (C_{ak}^T)^{-1} g_k \delta g_k^T C_{ak}^{-1}\right] \qquad \text{(E-9)}$$

The k^{th} cost change can now be written as a function of known quantities from equations (E-8) and (E-9).

$$(\delta \text{ Cost})_k = - \frac{1}{\sigma^2}\text{tr}\left\{E_a Q E_a\left[(C_{ak}^T)^{-1}\delta g_k g_k^T C_{ak}^{-1} + (C_{ak}^T)^{-1} g_k \delta g_k^T C_{ak}^{-1}\right]\right\} \qquad \text{(E-10)}$$

Separating the trace expression in (E-10) into two parts and applying (E-7) with "A" equal to $E_a Q E_a$:

$$(\delta \text{ Cost})_k = - \frac{1}{\sigma^2}\left[\text{tr}\left\{E_a Q E_a\left[(C_{ak}^T)^{-1}\delta g_k g_k^T C_{ak}^{-1}\right]\right\} \cdots + \text{tr}\left\{\left[(C_{ak}^T)^{-1} g_k \delta g_k^T C_{ak}^{-1}\right] E_a Q E_a\right\}\right] \qquad \text{(E-11)}$$

Another identity involving the trace is found to be useful.

$$\text{tr}\left[A^T\right] = \text{tr}\left[A\right] \qquad \text{(E-12)}$$

An examination of equation (E-11) reveals the following identity:

$$\left[(C_{ak}^T)^{-1} g_k \delta g_k^T C_{ak}^{-1} E_a Q E_a\right]^T = E_a^T Q^T E_a^T (C_{ak}^T)^{-1}\delta g_k g_k^T C_{ak}^{-1} \qquad \text{(E-13)}$$

Since the E_a and Q matrices are both symmetric, the "T"'s on these matrices can be dropped. Equations (E-12) and (E-13) are then applied

to (E-11) to further reduce the cost change expression.

$$(\delta \text{ Cost})_k = -\frac{2}{\sigma^2} \text{tr} \left\{ E_a Q E_a \left[(C_{ak}^T)^{-1} \quad \xi_k \xi_k^T \, C_{ak}^{-1} \right] \right\} \qquad \text{(E-14)}$$

The elements of the trace can be rearranged by twice applying equation (E-7) where "A" in (E-7) is first equal to $E_a Q E_a$ and then $(C_{ak}^T)^{-1} \delta \xi_k$.

$$(\delta \text{ Cost})_k = -\frac{2}{\sigma^2} \text{tr} \left\{ \left[\xi_k^T \, C_{ak}^{-1} E_a Q E_a (C_{ak}^T)^{-1} \quad \xi_k \right] \right\} \qquad \text{(E-15)}$$

Equation (E-15) can be simplified by considering the following groupings of its elements:

$C_{ak}^{-1} \; E_a Q E_a (C_{ak}^T)^{-1}$ — a 4 X 4 matrix

$\xi_k^T \left[C_{ak}^{-1} E_a Q E_a (C_{ak}^T)^{-1} \right]$ — a row vector

Since $\delta \xi_k$ is a column vector, the bracketed expression in (E-15) is a scalar and the trace notation can be dropped.

$$(\delta \text{ Cost})_k = -\frac{2}{\sigma^2} \left[\xi_k^T \, C_{ak}^{-1} \; E_a Q E_a (C_{ak}^T)^{-1} \delta \xi_k \right] \qquad \text{(E-16)}$$

The influence vector may be identified by comparing equation (E-16) with (E-2).

$$L_k^T = -\frac{1}{\sigma^2} \left[\xi_k^T C_{ak}^{-1} E_a Q E_a (C_{ak}^T)^{-1} \right] \qquad \text{(E-17)}$$

In the time change portion of the problem, the time of the decision points could be changed in whichever direction resulted in a decrease in cost. There is no such control in changing the horizon since the $\delta \xi_k$ vector is pre-determined by the present horizon.

There are only two possible values for δg_k, depending on whether the proposed change is from the left to right horizon or vice-versa.

From Appendix A:

$$g_k \text{ (LEFT)} = \begin{bmatrix} \frac{r_E}{z(z^2 - r_E^2)^{\frac{1}{2}}} \\ \frac{1}{z} \\ 0 \\ 0 \end{bmatrix} \qquad \text{(A-14)}$$

$$g_k \text{ (RIGHT)} = \begin{bmatrix} \frac{r_E}{z(z^2 - r_E^2)^{\frac{1}{2}}} \\ -\frac{1}{z} \\ 0 \\ 0 \end{bmatrix} \qquad \text{(A-14)}$$

The new measurement vector is related to the old by equation (E-3). Therefore, the two δg_k vectors have the following form:

$$\delta g_k \text{(LEFT TO RIGHT)} = \begin{bmatrix} 0 \\ -\frac{2}{z} \\ 0 \\ 0 \end{bmatrix} \qquad \text{(E-18)}$$

$$\delta g_k \text{(RIGHT TO LEFT)} = \begin{bmatrix} 0 \\ \frac{2}{z} \\ 0 \\ 0 \end{bmatrix} \qquad \text{(E-19)}$$

As explained in Chapter 2, equation (E-20) is applied at each point and, if the resultant change in cost is negative, the proposed horizon change is made.

$$(\delta \text{ Cost})_k = \underline{L}_k^T \delta \underline{g}_k \qquad \text{(E-20)}$$

APPENDIX F

COMPUTER PROGRAM

The Fortran program used to test the theory covered in this thesis is quite complex and involves many subroutines. The main program flow charts are shown in Chapter 4. Also the logic for three subroutines is covered in Chapter 4. The main program carries the primary load for computing the new measurement time schedule, selecting the measurement horizon schedule, and performing the optimization. The subroutines compute various matrices, perform various matrix operations, create new measurement time and horizon schedules, and print various matrices.

The input data needed by the main program has the following sequence and units:

1) Variance of the measurement ($radians^2$)
2) Altitude of spacecraft (miles)
3) Radius of the earth (miles)
4) Final angle (degrees)
5) Gravitational constant ($feet^3/sec^2$)
6) Number of iterations
7) Initial measurement time schedule (seconds)
   a. Block one, right then left

b. Block two, initial schedule

8) Initial estimation error covariance matrix (feet^2 and feet^2/second^2)

9) Maximum time changes (seconds)

a. Right then left

10) Initial measurement horizon schedule (Block two only, right horizon--flag equals one, left horizon--flag equals zero)

The standard matrix operations are needed in the program. Also since some of the vectors and matrices depend upon the particular measurement time, special variations are needed. The operations needed, and the subroutines performing them are:

1) Dot product (MULTF and MULTG),

2) Diatic product (MULTA and MULTAA),

3) Matrix inversion (INVERT),

4) Matrix multiplication (MULTB, MULTC, MULTD, and MULTE),

5) Product of row vector and matrix (MULTI),

6) Matrix sums (SUMA, SUMB, and SUMC),

7) Trace (TRACE and TRACEB).

Various matrices, vectors, and the results of several matrix and vector equations are needed in the program. These are:

1) State transition matrix (TRANSM),

2) Inverse of state transition matrix (TRANMI),

3) Transpose inverse of state transition matrix (TRANIT),

4) Derivative of each element of the state transition matrix (DTRANM),

5) Matrix Q (AQM),

6) Initial estimation error covariance matrix in target coordinates equation(B-5) (ERRORI),

7) Target estimation error covariance matrix, equation (B-26) (ERRORM),

8) Horizon vector matrix (HORIZS),

9) Horizon change vector, L_k (LAMDA),

10) Equation F-1 (DERROR),

$$\frac{\frac{\partial (C_{ak}^T)^{-1}}{\partial t_k} \xi_k \xi_k^T C_{ak}^{-1}}{\sigma^2} + \frac{(C_{ak}^T)^{-1} \xi_k \xi_k^T \frac{\partial (C_{ak})^{-1}}{\partial t_k}}{\sigma^2} \qquad \text{(F-1)}$$

11) Test on time change (CHECK),

12) New measurement time schedule (LOGIC),

13) New measurement horizon schedule (JUMP).

Various matrices, vectors, lists, and headings are needed in the results. These are printed by PRINTA, PRINTB, PRINTC, PRINTD, and PRINTR.

Printouts will follow:

```
C       MAIN BLOCK ONE

*       LIST
*       LABEL
C       DEFINITION OF THE VARIABLES AND CONSTANTS
C       W IS THE ANGU AR VELOCITY OF THE SPACECRAFT
C       TO IS THE TIME OF ORBIT
C       FA IS THE FINAL ANGLE OR TARGET ANGLE
C       T(I) IS THE INITIAL TIMING SCHEDULE
C       EI IS THE INITIAL ESTIMATION ERROR COVARIENCE MATRIX
C       DELTMR AND DELTML ARE THE MAXIMUM TIME CHANGES
C       SIGMAS IS THE VARIANCE OF THE MEASUREMENT
C       AL IS THE ALTITUDE OF THE CRAFT
C       RE IS THE RADIUS OF THE EARTH
C       U IS THE GRAVITATIONAL CONSTANT
C       HR AND HL ARE THE MEASURMENT VECTORS
C       EAR AND EAL ARE THE FINAL CORRELATION MATRICES
        DIMENSION  DR(4,4),DL(4,4),DELTR(30),DELTL(30),EAL(4,4),
       1 EAR(4,4),EEL(30,4,4),EER(30,4,4),EI(4,4),ERRL(30,4,4),
       2 ERRR(30,4,4),HHL(4,4),HHL(4,4),HL(4),HR(4),Q(4,4),
       3 RR(30,4,4),SS(30,4,4),TAL(30),TALI(30),TAR(30),TARI(30),
       4 T(30),TR(30),TL(30)
        COMMON FA,W,SIGMAS,T,EI,DELTMR,DELTML
 1000   READ 1,SIGMAS,AL,RE,FA
    1   FORMAT(F20.10)
  111   READ 112,U
  112   FORMAT(E20.10)
        PRINT 800,SIGMAS
  800   FORMAT (1H0,10X34H THE VARIANCE OF THE MEASUREMENT =,F20.10)
        PRINT 801,AL
  801   FORMAT(1H0,10X33H THE ALTITUDE OF THE SPACECRAFT =,F20.10,
       16H MILES)
        PRINT 802, RE
  802   FORMAT(1H0,10X26H THE RADIUS OF THE EARTH =,F20.10,6H MILES)
        PRINT 803,FA
  803   FORMAT(1H0,10X18H THE FINAL ANGLE =,F20.10,8H DEGREES)
        PRINT 804,U
  804   FORMAT(1H0,10X31H THE GRAVITATIONAL CONSTANT U =,E20.10,
       130H FEET CUBED PER SECOND SQUARED)
        READ 999,NI
  999   FORMAT (I10)
        PRINT 998,NI
  998   FORMAT(1H0,10X28H THE NUMBER OF ITERATIONS IS,I6)
        READ 2,(TR(I),I=1,30)
        READ 2,(TL(I),I=1,30)
        PRINT 808
  808   FORMAT(1H5,10X36H THE INITIAL TIMING SCHEDULE FOR THE,
       1 17H RIGHT HORIZON IS)
        CALL PRINTA(TR)
        PRINT 888
  888   FORMAT(1H5,10X36H THE INITIAL TIMING SCHEDULE FOR THE,
       1 16H LEFT HORIZON IS)
```

```
      CALL PRINTA(TL)
    2 FORMAT (F20.10)
      READ 2000,((EI(J,K),J=1,4),K=1,4)
 2000 FORMAT (F25.8)
      PRINT 807
  807 FORMAT(1H5,10X28H THE INITIAL ERROR MATRIX IS)
      CALL PRINTB (EI)
      READ 2001, DELTMR,DELTML
 2001 FORMAT (F20.10)
      PRINT 2100,DELTMR
 2100 FORMAT(1H0,10X29H TIME INCREMENT RIGHT HORIZON,F20.10)
      PRINT 2101,DELTML
 2101 FORMAT(1H0,10X28H TIME INCREMENT LEFT HORIZON,F20.10)
      AL=AL*5280.
      RE=RE*5280.
      FA=FA*3.14159/180.
    3 TO=2.0*3.14159*((RE+AL)**1.5)/SQRTF(U)
    4 W=2.*3.14159/TO
      PRINT 805,TO
  805 FORMAT(1H0,10X20H THE TIME OF ORBIT =,E20.10,8H SECONDS)
      PRINT 806,W
  806 FORMAT(1H0,10X41H THE ANGULAR VELOCITY OF THE SPACECRAFT =,
     1E20.10,19H RADIANS PER SECOND)
    5 HR(1)=RE/((AL+RE)*SQRTF(((AL+RE)**2)-(RE**2)))
    6 HR(2)=-1./(AL+RE)
    7 HR(3)=0.
    8 HR(4)=0.
    9 HL(1)=+HR(1)
   10 HL(2)=-HR(2)
   11 HL(3)=0.
   12 HL(4)=0.
      PRINT 809
  809 FORMAT(1H5,10X28H THE RIGHT HORIZON VECTOR IS)
      PRINT 880,HR
      PRINT 810
  810 FORMAT(1H5,10X27H THE LEFT HORIZON VECTOR IS)
      PRINT 880, HL
  880 FORMAT (1H0,4E30.10)
      NII=0.0
   13 CALL ERRORM(EAR,HR,OLCOSR,TR)
   14 CALL ERRORM(EAL,HL,OLCOSL,TL)
      PRINT 811
  811 FORMAT(1H5,10X40H FINAL CORRELATION MATRIX, RIGHT HORIZON)
      CALL PRINTB (EAR)
      PRINT 812
  812 FORMAT(1H5,10X39H FINAL CORRELATION MATRIX, LEFT HORIZON)
      CALL PRINTB (EAL)
      TARIM=0.0
      TALIM=0.0
   15 DO 341 I=1,30,1
   16 CALL AQM(Q)
```

```
   17 CALL MULTB (DR,EAR,Q)
   18 CALL MULTB (DL,EAL,Q)
   19 CALL DERROR(ERRR,HR,I,TR)
   21 CALL DERROR(ERRL,HL,I,TL)
      CALL MULTE (RR,ERRR,DR,I)
  210 CALL MULTE (SS,ERRL,DL,I)
   22 CALL MULTC (EER,EAR,RR,I)
  220 CALL MULTC (EEL,EAL,SS,I)
   23 CALL TRACEB (TAR,EER,I)
   24 CALL TRACEB (TAL,EEL,I)
   25 TARI(I)=-TAR(I)*W
   26 TALI(I)=-TAL(I)*W
C     29 THRU 34 DETERMINE THE LARGEST INFLUENCE COEFFICIENT
C     FOR BOTH THE LEFT AND RIGHT HORIZON
C     TARIM,RIGHT AND TALIM,LEFT
   29 IF(ABSF(TARI(I))-ABSF(TARIM))311,33,33
  311 GO TO 30
   33 TARIM=TARI(I)
   30 IF(ABSF(TALI(I))-ABSF(TALIM))321,34,34
  321 GO TO 341
   34 TALIM=TALI(I)
  341 CONTINUE
      PRINT 261
  261 FORMAT(1H0,10X37H RIGHT HORIZON INFLUENCE COEFFICIENTS)
      DO 264 I=1,30,1
  264 PRINT 263,TARI(I),I
      PRINT 262
  262 FORMAT(1H0,10X36H LEFT HORIZON INFLUENCE COEFFICIENTS)
      DO 265 I=1,30,1
  265 PRINT 263,TALI(I),I
  263 FORMAT(1H0,10X12H COEFFICIENT,E20.10,5X12H TIME NUMBER,I4)
   36 PRINT 362
  362 FORMAT(1H5,10X23H DATA FOR RIGHT HORIZON)
  361 CRUD=DELTMR
   35 CALL CHECK(TARIM,TARI,OLCOSR,HR,TR,CRUD)
   37 CALL LOGIC(BIGA,TARIM,CRUD,TARI,OLCOSR,PCOSR,EAR,HR,TR)
   38 PRINT 382
  382 FORMAT(1H5,10X22H DATA FOR LEFT HORIZON)
  381 CRUD=DELTML
      CALL CHECK (TALIM,TALI,OLCOSL,HL,TL,CRUD)
   39 CALL LOGIC(BIGB,TALIM,CRUD,TALI,OLCOSL,PCOSL,EAL,HL,TL)
   40 NII=NII+1
   41 IF(NII-NI) 42,45,45
   42 PRINT 43,NII
   43 FORMAT(1H0,10X15H THE END OF THE,I5,10H ITERATION)
   44 GO TO 13
   45 PRINT 46
   46 FORMAT(1H0,10X8H THE END)
   47 GO TO 1000
      END
```

```
C       MAIN BLOCK TWO

*       LIST
*       LABEL
C       DEFINITION OF THE VARIABLES AND CONSTANTS
C       TO IS THE TIME OF ORBIT
C       W IS THE ANGU AR VELOCITY OF THE SPACECRAFT
C       FA IS THE FINAL ANGLE OR TARGET ANGLE
C       T(I) IS THE INITIAL TIMING SCHEDULE
C       EI IS THE INITIAL ESTIMATION ERROR COVARIENCE MATRIX
C       DELTMR AND DELTML ARE THE MAXIMUM TIME CHANGES
C       SIGMAS IS THE VARIANCE OF THE MEASUREMENT
C       AL IS THE ALTITUDE OF THE CRAFT
C       IHV(I) IS THE INITIAL HORIZON SCHEDULE
C       RE IS THE RADIUS OF THE EARTH
C       U IS THE GRAVITATIONAL CONSTANT
C       HR AND HL ARE THE MEASUREMENT VECTORS
C       EA IS THE FINAL CORRELATION MATRIX
        DIMENSION DELTR(30),DHLR(4),DHRL(4),EA(4,4),EI(4,4),
       1 EER(30,4,4),ERRR(30,4,4),HR(4),HL(4),IHV(30),
       2 HS(30,4),DELCOS(30),Q(4,4),RR(30,4,4),TAR(30), TARI(30),
       3 TLAMDA(30),T(30)
        COMMON FA,W,SIGMAS,T,EI,DELTMR,DELTML
 1000   READ 1,SIGMAS,AL,RE,FA
    1   FORMAT(F20.10)
  111   READ 112,U
  112   FORMAT(E20.10)
        PRINT 800,SIGMAS
  800   FORMAT (1H0,10X34H THE VARIANCE OF THE MEASUREMENT =,F20.10)
        PRINT 801,AL
  801   FORMAT(1H0,10X33H THE ALTITUDE OF THE SPACECRAFT =,F20.10,
       16H MILES)
        PRINT 802, RE
  802   FORMAT(1H0,10X26H THE RADIUS OF THE EARTH =,F20.10,6H MILES)
        PRINT 803,FA
  803   FORMAT(1H0,10X18H THE FINAL ANGLE =,F20.10,8H DEGREES)
        PRINT 804,U
  804   FORMAT(1H0,10X31H THE GRAVITATIONAL CONSTANT U =,E20.10,
       130H FEET CUBED PER SECOND SQUARED)
        READ 999,NI
  999   FORMAT (I10)
        PRINT 998,NI
  998   FORMAT(1H0,10X28H THE NUMBER OF ITERATIONS IS,I6)
        READ 2, (T(I),I=1,30)
    2   FORMAT (F20.10)
        PRINT 808
  808   FORMAT(1H5,10X31H THE INITIAL TIMING SCHEDULE IS)
        CALL PRINTA (T)
        READ 2000,((EI(J,K),J=1,4),K=1,4)
 2000   FORMAT (F25.8)
        PRINT 807
```

```
 807 FORMAT(1H5,10X28H THE INITIAL ERROR MATRIX IS)
     CALL PRINTB (EI)
     READ 2001, DELTMR,DELTML
2001 FORMAT (F20.10)
     PRINT 2100,DELTMR
2100 FORMAT(1H0,10X15H TIME INCREMENT,E20.10)
     READ 2002,(IHV(I),I=1,30)
2002 FORMAT(I5)
     PRINT 2003
2003 FORMAT(1H5,10X32H THE INITIAL HORIZON SCHEDULE IS)
     CALL PRINTD(IHV)
     AL=AL*5280.
     RE=RE*5280.
     FA=FA*3.14159/180.
   3 TO=2.0*3.14159*((RE+AL)**1.5)/SQRTF(U)
   4 W=2.*3.14159/TO
     PRINT 805,TO
 805 FORMAT(1H0,10X20H THE TIME OF ORBIT =,E20.10,8H SECONDS)
     PRINT 806,W
 806 FORMAT(1H0,10X41H THE ANGULAR VELOCITY OF THE SPACECRAFT =,
    1E20.10,19H RADIANS PER SECOND)
   5 HR(1)=RE/((AL+RE)*SQRTF(((AL+RE)**2)-(RE**2)))
   6 HR(2)=-1./(AL+RE)
   7 HR(3)=0.
   8 HR(4)=0.
   9 HL(1)=-HR(1)
  10 HL(2)=HR(2)
  11 HL(3)=0.
  12 HL(4)=0.
     PRINT 809
 809 FORMAT(1H5,10X28H THE RIGHT HORIZON VECTOR IS)
     PRINT 880,HR
     PRINT 810
 810 FORMAT(1H5,10X27H THE LEFT HORIZON VECTOR IS)
     PRINT 880, HL
     DHRL(1)=2.0*HL(1)
     DHRL(2)=0.0
     DHRL(3)=0.0
     DHRL(4)=0.0
     DHLR(1)=2.0*HR(1)
     DHLR(2)=0.0
     DHLR(3)=0.0
     DHLR(4)=0.0
     PRINT 820
 820 FORMAT(1H5,10X40H THE DELTA VECTOR FROM R TO L HORIZON IS)
     PRINT 880,DHRL
     PRINT 821
 821 FORMAT(1H5,10X40H THE DELTA VECTOR FROM L TO R HORIZON IS)
     PRINT 880,DHLR
 880 FORMAT(1H0,4E30.10)
     NII=0
```

```
   13 CALL HORIZS(IHV,HR,HL,HS)
      CALL ERRORM(EA,HS,OLCOSR)
      PRINT 811
  811 FORMAT(1H5,10X25H FINAL CORRELATION MATRIX)
      CALL PRINTB(EA)
      TARIM=0.0
   15 DO 341 I=1,30,1
   16 CALL AQM(Q)
   17 CALL MULTB(D,EA,Q)
   19 CALL DERROR(ERRR,HS,I)
   21 CALL MULTE(RR,ERRR,D,I)
   22 CALL MULTC(EER,EA,RR,I)
   23 CALL TRACEB (TAR,EER,I)
   25 TARI(I)=-TAR(I)*W
C     TARI(I) ARE THE INFLUENCE COEFFICIENTS
C     29 THRU 341 DETERMINE THE LARGEST INFLUENCE COEFFICIENT
   29 IF(ABSF(TARI(I))-ABSF(TARIM))311,33,33
  311 GO TO 341
   33 TARIM=TARI(I)
  341 CONTINUE
      DO 263 I=1,30,1
  263 PRINT 261,TARI(I),I
  261 FORMAT(2H0,10X23H INFLUENCE COEFFICIENT ,E20.10,
     1 5X12H TIME NUMBER,I3)
   35 CALL CHECK (TARIM,TARI,OLCOSR,HS)
  361 CRUD=DELTMR
   37 CALL LOGIC (BIGA,TARIM,CRUD,TARI,OLCOSR,PCOSR,EA,HS)
   38 CALL JUMP (TLAMDA,DHLR,DHRL,PCOSR,IHV,HR,HL)
      PUNCH 462,IHV
  462 FORMAT(I5)
   40 NII=NII+1
   41 IF(NII-NI) 42,45,45
   42 PRINT 43,NII
   43 FORMAT(1HO,10X15H THE END OF THE,I5,10H ITERATION)
   44 GO TO 13
   45 PRINT 46
   46 FORMAT(1HO,10X8H THE END)
   47 GO TO 1000
      END
```

```
      SUBROUTINE INVERT(N,QQ)

C     REFERENCE(7)

*     LIST
*     LABEL
C     SUBROUTINE INVERTS MATRIX BY SIMULTANEOUS DOUBLE PRECISION
C     ROW REDUCTION OF THE MATRIX TO IDEM AND IDEM TO THE INVERSE
D     DIMENSION QQ(4,4),Q(4,8)
      DO 10 I=1,4
      DO 5 J=1,4
      Q(I, J)=QQ(I, J)
      Q(I,J+4)=0.0
      Q(I,J+8)=0.0
    5 Q(I,J+12)=0.0
   10 Q(I,I+4)=1.0
      DO 30 I=1, N
      DO 14 J=I, N
      IF(ABSF(Q(I, I))-ABSF(Q(J, I))) 11, 14, 14
C     TEST FOR LARGEST ELEMENT IN COLUMN
   11 DO 12 K=1, N
D     S=Q(J, K)
D     Q(J, K)=Q(I, K)
D     Q(I, K)=S
D     S=Q(J,K+4)
D     Q(J,K+4)=Q(I,K+4)
D  12 Q(I,K+4)=S
C     TRANSFER ROW OF LARGEST ELEMENT TO FIRST ROW
   14 CONTINUE
D     DIV=Q(I,I)
      DO 15 J=1,N
D     Q(I,J)=Q(I,J)/DIV
D  15 Q(I,J+4)=Q(I,J+4)/DIV
C     DIVISION BY DIAGONAL ELEMENTS
      DO 30 J=1,N
      IF(I-J) 20,30,20
D  20 DIV=Q(J,I)
      DO 25 K=1,N
D     Q(J,K)=Q(J,K)-Q(I,K)*DIV
D  25 Q(J,K+4)=Q(J,K+4)-Q(I,K+4)*DIV
C     DIAGONALIZATION OF MATRIX
   30 CONTINUE
      DO 35 I=1,N
      DO 35 J=1,N
   35 QQ(I,J)=Q(I,J+4)
      RETURN
      END
```

```
      SUBROUTINE MULTA(HH,H)

*     LIST
*     LABEL
C     THIS SUBROUTINE PERFORMS THE MULTIPLICATION OF A ROW
C     VECTOR BY A COLUMN VECTOR
      DIMENSION HH(4,4), H(4)
 1111 DO 1114 I=1,4,1
 1112 DO 1114 J=1,4,1
 1113 HH(I,J)=0.0
 1114 HH(I,J)=H(I)*H(J)
      RETURN
      END
```

```
      SUBROUTINE MULTAA(HH,H,I)

*     LIST
*     LABEL
C     THIS SUBROUTINE PERFORMS THE MULTIPLICATION OF A ROW VECTOR
C     BY A COLUMN VECTOR, EACH VECTOR DEPENDS ON THE TIME I,
      DIMENSION HH(30,4,4),H(30,4)
    1 DO 4 J=1,4,1
    2 DO 4 K=1,4,1
    3 HH(I,J,K)=0.0
    4 HH(I,J,K)=H(I,J)*H(I,K)
      RETURN
      END
```

```
      SUBROUTINE MULTB(XX,XY,XZ)
*     LIST
*     LABEL
C     THIS SUBROUTINE MULTIPLIES TWO MATRICES

      DIMENSION XX(4,4), XY(4,4), XZ(4,4)
 2111 DO 2115 J=1,4,1
 2112 DO 2115 K=1,4,1
 2113 XX(J,K)=0.0
 2114 DO 2115 L=1,4,1
 2115 XX(J,K)=XX(J,K)+XY(J,L)*XZ(L,K)
      RETURN
      END
```

```
      SUBROUTINE MULTC(WX,WY,WZ,I)

*     LIST
*     LABEL
C     THIS SUBROUTINE MULTIPLIES TWO MATRICES
C     THE RESULT AND MULTIPLICAND INVOLVE THE  I TH TIME
      DIMENSION WX(30,4,4),WY(4,4),WZ(30,4,4)
 3111 DO 3115 J=1,4,1
 3112 DO 3115 K=1,4,1
 3113 WX(I,J,K)=0.0
 3114 DO 3115 L=1,4,1
 3115 WX(I,J,K)=WX(I,J,K)+WY(J,L)*WZ(I,L,K)
      RETURN
      END



      SUBROUTINE MULTD(UX,UY,UZ,I)

*     LIST
*     LABEL
C     THIS SUBROUTINE MULTIPLIES TWO MATRICES
C     THE RESULT,MULTIPLIER,AND MULTIPLICAND INVOLVE THE I TH TIME
      DIMENSION UX(30,4,4),UY(30,4,4),UZ(30,4,4)
 4111 DO 4115 J=1,4,1
 4112 DO 4115 K=1,4,1
 4113 UX(I,J,K)=0.0
 4114 DO 4115 L=1,4,1
 4115 UX(I,J,K)=UX(I,J,K)+UY(I,J,L)*UZ(I,L,K)
      RETURN
      END



      SUBROUTINE MULTE (RX,RY,RZ,I)

C     THIS SUBROUTINE PERFORMS THE MULTICLICATION OF TWO MATRICES
C     THE RESULT AND MULTIPLIER INVOLVE THE I  TH TIME
*     LIST
*     LABEL
      DIMENSION RX(30,4,4),RY(30,4,4),RZ(4,4)
  561 DO 565 J=1,4,1
  562 DO 565 K=1,4,1
  563 RX(I,J,K)=0.
  564 DO 565 L=1,4,1
  565 RX(I,J,K)=RX(I,J,K)+RY(I,J,L)*RZ(L,K)
      RETURN
      END
```

```
      SUBROUTINE MULTF(A,B,C)

*     LIST
*     LABEL
C     THIS SUBROUTINE PERFORMS THE DOT PRODUCT OF A VECTOR
      DIMENSION B(4),C(J.
    1 A=0.0
    2 DO 3 I=1,4,1
    3 A=A+B(I)*C(I)
      RETURN
      END

      SUBROUTINE MULTG(A,B,C,I)

*     LIST
*     LABEL
C     THIS SUBROUTINE PERFORMS THE DOT PRODUCT OF A VECTOR
C     WHERE THE I TH TIME IS INCLUDED
      DIMENSION A(30),B(30,4),C(4)
    1 A(I)=0.0
    2 DO 3 J=1,4,1
    3 A(I)=A(I)+B(I,J)*C(J)
      RETURN
      END

      SUBROUTINE MULTH(A,B,C)

*     LIST
*     LABEL
C     THIS SUBROUTINE MULTIPLIES A MATRIX BY A ROW VECTOR
C     TO GIVE A ROW VECTOR
    1 DO 4 I =1,4,1
    2 A(I)=0.0
    3 DO 4 J=1,4,1
    4 A(I)=A(I)+B(J)*C(J,I)
      RETURN
      END

      SUBROUTINE MULTI(A,B,C,I)

*     LIST
*     LABEL
C     THIS SUBROUTINE MULTIPLIES A MATRIX BY A ROW VECTOR
C     TO GIVE A ROW VECTOR WHERE THE I TH TIME IS NEEDED
      DIMENSION A(30,4),B(30,4),C(30,4,4)
    1 DO 4 J=1,4,1
    2 A(I,J)=0.0
    3 DO 4 K=1,4,1
    4 A(I,J)=A(I,J)+B(I,K)*C(I,K,J)
      RETURN
      END
```

```
      SUBROUTINE SUMA(VX,VY,VZ)

*     LIST
*     LABEL
C     THIS SUBROUTINE SUMS ALL OF THE MATRICES INVOLVING
C     THE I TH TIME AND DIVIDES THE RESULT BY VZ
      DIMENSION VX(4,4), VY(30,4,4)
 5111 DO 5117 J=1,4,1
 5112 DO 5117 K=1,4,1
 5113 VX(J,K)=0.0
 5114 DO 5116 I=1,30,1
 5116 VX(J,K) = VX(J,K)+VY(I,J,K)
 5117 VX(J,K)=VX(J,K)/VZ
      RETURN
      END
```

```
      SUBROUTINE SUMB(ZX,ZY,ZZ)

*     LIST
*     LABEL
C     THIS SUBROUTINE SUMS TWO MATRICES
      DIMENSION ZX(4,4),ZY(4,4),ZZ(4,4)
 7111 DO 7113 J=1,4,1
 7112 DO 7113 K=1,4,1
 7113 ZX(J,K)=ZY(J,K)+ZZ(J,K)
      RETURN
      END
```

```
      SUBROUTINE SUMC (SX,SY,SZ,SW,I)

*     LIST
*     LABEL
C     THIS SUBROUTINE SUMS TWO MATRICES INVOLVING THE
C     I TH TIME AND DIVIDES EACH ELEMENT BY SW
      DIMENSION SX(30,4,4),SY(30,4,4),SZ(30,4,4)
  911 DO 913 J=1,4,1
  912 DO 913 K=1,4,1
  913 SX(I,J,K)=(SY(I,J,K)+SZ(I,J,K))/SW
      RETURN
      END
```

```
      SUBROUTINE TRACE(TA,TY)

*     LIST
*     LABEL
C     THIS SUBROUTINE TAKES THE TRACE OF A MATRIX
      DIMENSION TY(4,4)
 6111 TA=0.0
 6112 DO 6113 J=1,4,1
 6113 TA=TA+TY(J,J)
 6114 PRINT 6115,TA
 6115 FORMAT(1H0,5X9H TRACE IS,E20.10)
      RETURN
      END
```

```
      SUBROUTINE TRACEB (TA,TY,I)

*     LIST
*     LABEL
C     THIS SUBROUTINE TAKES THE TRACE OF A MATRIX INVOLVING THE
C     I TH TIME
      DIMENSION TA(30),TY(30,4,4)
  781 TA=0.
  782 DO 786 J=1,4,1
  783 TA(I)=TA(I)+TY(I,J,J)
  786 CONTINUE
  784 PRINT 785, TA(I),I
  785 FORMAT(1H0,5X9H TRACE IS,E20.10,5X12H TIME NUMBER,I4)
      RETURN
      END
```

```
      SUBROUTINE TRANSM(TM,FI,I)

*     LIST
*     LABEL
C     THIS SUBROUTINE SETS UP THE STATE TRANSITION MATRIX
      DIMENSION TM(30,4,4),T(30),EI(4,4)
      COMMON FA,W,SIGMAS,T,EI,DELTMR,DELTML
      TM(I,1,1)=1.0+2.0*(SINF((FA-FI)/2.0))**2
      TM(I,1,2)=SINF(FA-FI)
      TM(I,1,3)=SINF(FA-FI)/W
      TM(I,1,4)=(4.0*(SINF((FA-FI)/2.0))**2)/W
      TM(I,2,1)=-3.0*(FA-FI)+2.0*SINF(FA-FI)
      TM(I,2,2)=1.0-4.0*(SINF((FA-FI)/2.0))**2
      TM(I,2,3)=-TM(I,1,4)
      TM(I,2,4)=-3.0*(FA-FI)/W+4.0*(SINF(FA-FI))/W
      TM(I,3,1)=3.0*W*(FA-FI)-W*SINF(FA-FI)
      TM(I,3,2)=2.0*W*(SINF((FA-FI)/2.0))**2.0
      TM(I,3,3)=TM(I,1,1)
      TM(I,3,4)=-TM(I,2,1)
      TM(I,4,1)=-TM(I,3,2)
      TM(I,4,2)=-W*TM(I,1,2)
      TM(I,4,3)=-TM(I,1,2)
      TM(I,4,4)=TM(I,2,2)
      RETURN
      END


      SUBROUTINE TRANMI(TMI,TM,I)

*     LIST
*     LABEL
C     THIS SUBROUTINE SETS UP THE INVERSE OF THE STATE TRANSITION
C     MATRIX  OR THE DERIVATIVE OF THE INVERSE OF THE STATE
C     TRANSITION MATRIX
      DIMENSION TM(30,4,4),TMI(30,4,4)
      TMI(I,1,1)=TM(I,3,3)
      TMI(I,1,2)=TM(I,4,3)
      TMI(I,1,3)=-TM(I,1,3)
      TMI(I,1,4)=-TM(I,2,3)
      TMI(I,2,1)=TM(I,3,4)
      TMI(I,2,2)=TM(I,4,4)
      TMI(I,2,3)=-TM(I,1,4)
      TMI(I,2,4)=-TM(I,2,4)
      TMI(I,3,1)=-TM(I,3,1)
      TMI(I,3,2)=-TM(I,4,1)
      TMI(I,3,3)=TM(I,1,1)
      TMI(I,3,4)=TM(I,2,1)
      TMI(I,4,1)=-TM(I,3,2)
      TMI(I,4,2)=-TM(I,4,2)
      TMI(I,4,3)=TM(I,1,2)
      TMI(I,4,4)=TM(I,2,2)
      RETURN
      END
```

```
      SUBROUTINE TRANIT (TMIT,TM,I)

*     LIST
*     LABEL
C     THIS SUBROUTINE SETS UP THE TRANPOSED INVERSE OF THE STAT
C     TRANSITION MATRIX OR THE DERIVATIVE OF THE TRANPOSED INVE
C     OF THE STATE TRANSITION MATRIX
      DIMENSION TM (30,4,4),TMIT (30,4,4)
      TMIT(I,1,1)=TM(I,3,3)
      TMIT(I,1,2)=TM(I,3,4)
      TMIT(I,1,3)=-TM(I,3,1)
      TMIT(I,1,4)=-TM(I,3,2)
      TMIT(I,2,1)=TM(I,4,3)
      TMIT(I,2,2)=TM(I,4,4)
      TMIT(I,2,3)=-TM(I,4,1)
      TMIT(I,2,4)=-TM(I,4,2)
      TMIT(I,3,1)=-TM(I,1,3)
      TMIT(I,3,2)=-TM(I,1,4)
      TMIT(I,3,3)=TM(I,1,1)
      TMIT(I,3,4)=TM(I,1,2)
      TMIT(I,4,2)=-TM(I,2,3)
      TMIT(I,4,2)=-TM(I,2,4)
      TMIT(I,4,3)=TM(I,2,1)
      TMIT(I,4,4)=TM(I,2,2)
      RETURN
      END

      SUBROUTINE DTRANM (DTM,FI,I)

*     LIST
*     LABEL
C     THIS SUBROUTINE SETS UP THE DERIVATIVE OF EACH ELEMENT
C     OF THE STATE TRANSITION MATRIX
      DIMENSION DTM(30,4,4),EI(4,4),T(30)
      COMMON FA,W,SIGMAS,T,EI,DELTMR,DELTML
      DTM(I,1,1)=-SINF(FA-FI)
      DTM(I,1,2)=-COSF(FA-FI)
      DTM(I,1,3)=-(COSF(FA-FI))/W
      DTM(I,1,4)=-2.*(SINF(FA-FI))/W
      DTM(I,2,1)=3.-2.COSF(FA-FI)
      DTM(I,2,2)=2.*SINF(FA-FI)
      DTM(I,2,3)=-DTM(I,1,4)
      DTM(I,2,4)=3./W-4.*(COSF(FA-FI))/W
      DTM(I,3,1)=-3.*W+W*COSF(FA-FI)
      DTM(I,3,2)=W*DTM(I,1,1)
      DTM(I,3,3)=DTM(I,1,1)
      DTM(I,3,4)=-DTM(I,2,1)
      DTM(I,4,1)=W*SINF(FA-FI)
      DTM(I,4,2)=W*COSF(FA-FI)
      DTM(I,4,3)=COSF(FA-FI)
      DTM(I,4,4)=DTM(I,2,2)
      RETURN
      END
```

```
      SUBROUTINE AQM(QM)

*     LIST
*     LABEL
C     THIS SUBROUTINE SETS UP THE Q MATRIX
      DIMENSION QM(4,4)
      DO 3 I=1,4,1
      DO 3 J=1,4,1
    3 QM(I,J)=0.0
      QM(1,1)=1.0
      QM(2,2)=1.0
      RETURN
      END
```

```
      SUBROUTINE ERRORI(EIE)

*     LIST
*     LABEL
C     THIS SUBROUTINE TRANSFERS THE INITIAL ERROR TO THE
C     TARGET CO-ORDINATE FRAME
      DIMENSION EIE(4,4),TM(4,4),TMT(4,4),A(4,4),
     1EI(4,4),T(30)
      COMMON FA,W,SIGMAS,T,EI,DELTMR,DELTML
      TM(1,1)=1.0+2.0(SINF(FA/2.0))**2
      TM(1,2)=SINF(FA)
      TM(1,3)=SINF(FA)/W
      TM(1,4)=4.0*(SINF(FA/2.0)**2)/W
      TM(2,1)=-3.0*FA+2.0*SINF(FA)
      TM(2,2)=1.0-4.0*(SINF(FA/2.0))**2
      TM(2,3)=-TM(1,4)
      TM(2,4)=(-3.0*FA+4.0*SINF(FA))/W
      TM(3,1)=3.0*W*FA-W*SINF(FA)
      TM(3,2)=2.0*W*(SINF(FA/2.0))**2
      TM(3,3)=TM(1,1)
      TM(3,4)=-TM(2,1)
      TM(4,1)=-TM(3,2)
      TM(4,2)=-W*TM(1,2)
      TM(4,3)=-TM(1,2)
      TM(4,4)=TM(2,2)
      TMT(1,1)=TM(1,1)
      TMT(1,2)=TM(2,1)
      TMT(1,3)=TM(3,1)
      TMT(1,4)=TM(4,1)
      TMT(2,1)=TM(1,2)
      TMT(2,2)=TM(2,2)
      TMT(2,3)=TM(3,2)
      TMT(2,4)=TM(4,2)
      TMT(3,1)=TM(1,3)
      TMT(3,2)=TM(2,3)
      TMT(3,3)=TM(3,3)
      TMT(3,4)=TM(4,3)
      TMT(4,1)=TM(1,4)
      TMT(4,2)=TM(2,4)
      TMT(4,3)=TM(3,4)
      TMT(4,4)=TM(4,4)
      CALL MULTB(A,EI,TMT)
      CALL MULTB(EIE,TM,A)
      RETURN
      END
```

```
      SUBROUTINE ERRORM(EA,H,COST,TT)

C     BLOCK ONE

*     LIST
*     LABEL
C     THIS SUBROUTINE COMPUTES THE TARGET ESTIMATION
C     ERROR COVARIANCE MATRIX AND THE COST
      DIMENSION C(30,4,4),CM(4,4),E(30,4,4),EA(4,4),EAC(4,4),
     1 EAI(4,4),EI(4,4),EII(4,4),EIE(4,4),H(4),HH(4,4),
     2 TM(30,4,4),TMI(30,4,4),TMIT(30,4,4),T(30),TT(30)
      COMMON FA,W,SIGMAS,T,EI,DELTMR,DELTML
  810 CALL ERRORI(EIE)
  816 CALL MULTA (HH,H)
  811 DO 818 I=1,30,1
  812 FI=W*TT(I)
  813 CALL TRANSM (TM,FI,I)
  814 CALL TRANMI (TMI,TM,I)
  815 CALL TRANIT (TMIT,TM,I)
  817 CALL MULTC (C,HH,TMI,I)
  818 CALL MULTD (E,TMIT,C,I)
  819 CALL SUMA (EAC,E,SIGMAS)
 8191 DO 8194 I=1,4,1
 8192 DO 8194 J=1,4,1
 8193 EII(I,J) = 0.0
 8194 EII(I,J)=EIE(I,J)
  820 CALL INVERT (4,EII)
  821 CALL SUMB (EAI,EII,EAC)
 8211 DO 8214 I=1,4,1
 8212 DO 8214 J=1,4,1
 8213 EA(I,J) = 0.0
 8214 EA(I,J) = EAI(I,J)
  822 CALL INVERT (4,EA)
  823 CALL AQM(QM)
  824 CALL MULTB(CM,QM,EA)
  825 CALL TRACE(COST,CM)
      RETURN
      END
```

```
      SUBROUTINE ERRORM (EA,H,COST)

C     BLOCK TWO

*     LIST
*     LABEL
C     THIS SUBROUTINE COMPUTES THE TARGET ESTIMATION
C     ERROR COVARIANCE MATRIX AND THE COST
      DIMENSION C(30,4,4),CM(4,4),E(30,4,4),EA(4,4),EAC(4,4),
     1 EAI(4,4),EI(4,4),EIE(4,4),EII(4,4),H(30(4),HH(30,4,4),
     2 TM(30,4,4),TMI(30,4,4),TMIT(30,4,4),T(30)
      COMMON FA,W,SIGMAS,T,EI,DELTMR,DELTML
  810 CALL ERRORI(EIE)
  811 DO 818 I=1,30,1
  812 FI=W*T(I)
  813 CALL TRANSM (TM,FI,I)
  814 CALL TRANMI (TMI,TM,I)
  815 CALL TRANIT (TMIT,TM,I)
  816 CALL MULTAA(HH,H,I)
  817 CALL MULTD(C,HH,TMI,I)
  818 CALL MULTD (E,TMIT,C,I)
  819 CALL SUMA (EAC,E,SIGMAS)
 8191 DO 8194 I=1,4,1
 8192 DO 8194 J=1,4,1
 8193 EII(I,J) = 0.0
 8194 EII(I,J)=EIE(I,J)
  820 CALL INVERT (4,EII)
  821 CALL SUMB (EAI,EII,EAC)
 8211 DO 8214 I=1,4,1
 8212 DO 8214 J=1,4,1
 8213 EA(I,J) = 0.0
 8214 EA(I,J) = EAI(I,J)
  822 CALL INVERT (4,EA)
  823 CALL AQM(QM)
  824 CALL MULTB(CM,QM,EA)
  825 CALL TRACE(COST,CM)
      RETURN
      END
```

```
      SUBROUTINE HORIZS(IHV,HR,HL,HS)

*     LIST
*     LABEL
      DIMENSION T(30),EI(4,4),IHV(30),HR(4),HL(4),HS(30,4)
      COMMON FA,W,SIGMAS,T,EI,DELTMR,DELTML
C     THIS SUBROUTINE SETS UP THE HORIZON VECTOR SCHEDULE
    1 DO 10 I=1,30,1
    2 IF (IHV(I)-1) 3,7,7
    3 DO 5 J=1,4,1
    4 HS(I,J)=0.0
    5 HS(I,J)=HL(J)
    6 GO TO 10
    7 DO 9 J=1,4,1
    8 HS(I,J)=0.0
    9 HS(I,J)=HR(J)
   10 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE LAMDA(TLAMDA,HS,I)

*     LIST
*     LABEL
C     THIS SUBROUTINE COMPUTES THE LAMDA VECTOR WHICH IS USED TO
C     COMPUTE THE INFLUENCE COEF. DUE TO A CHANGE IN HORIZON
      DIMENSION TM(30,4,4),TMI(30,4,4),TMIT(30,4,4),TLAMDA(30,4),
     1EA(4,4),QM(4,4),T(30),EI(4,4),HS(30,4),A(30,4,4),B(30,4,4),
     2C(30,4,4),D(30,4,4)
      COMMON  FA,W,SIGMAS,T,EI,DELTMR,DELTML
    4 CALL ERRORM(EA,HS,COST)
   89 FI=W*T(I)
    1 CALL TRANSM(TM,FI,I)
    2 CALL TRANMI(TMI,TM,I)
    3 CALL TRANIT(TMIT,TM,I)
    5 CALL AQM(QM)
    6 CALL MULTC(A,EA,TMIT,I)
    7 CALL MULTC(B,QM,A,I)
    8 CALL MULTC(C,EA,B,I)
    9 CALL MULTD(D,TMI,C,I)
   10 CALL MULTI(TLAMDA,HS,D,I)
   11 DO 12 J=1,4,1
   12 TLAMDA(I,J)=-2.0*TLAMDA(I,J)/SIGMAS
      RETURN
      END
```

```
      SUBROUTINE DERROR (ERR,H,I,TT)

C     BLOCK ONE

*     LIST
*     LABEL
      DIMENSION A(30,4,4),B(30,4,4),D(30,4,4),DTM(30,4,4),
     1 DTMI(30,4,4),DTMIT(30,4,4),EI(4,4),ERR(30,44),
     2 F(30,4,4),T(30),TM(30,4,4),TMI(30,4,4),TMIT(30,4,4),
     3 H(4),HH(4,4),TT(30)
      COMMON FA,W,SIGMAS,T,EI,DELTMR,DELTML
  981 FI=W*TT(I)
  982 CALL DTRANM(DTM,FI,I)
  983 CALL TRANMI (DTMI,DTM,I)
  984 CALL TRANIT (DTMIT,DTM,I)
  985 CALL TRANSM (TM,FI,I)
  986 CALL TRANMI(TMI,TM,I)
  987 CALL TRANIT (TMIT,TM,I)
  988 CALL MULTA (HH,H)
  989 CALL MULTC(A,HH,TMI,I)
  990 CALL MULTC (B,HH,DTMI,I)
  991 CALL MULTD (D,DTMIT,A,I)
  992 CALL MULTD (F,TMIT,B,I)
  993 CALL SUMC (ERR,D,F,SIGMAS,I)
      RETURN
      END
```

```
      SUBROUTINE DERROR(ERR,H,I)

C     BLOCK TWO

*     LIST
*     LABEL
      DIMENSION A(30,4,4),B(30,4,4),D(30,4,4),DTM(30,4,4),
     1 DTMI(30,4,4),DTMIT(30,4,4),EI(4,4),ERR(30,44),
     2 F(30,4,4),T(30),TM(30,4,4),TMI(30,4,4),TMIT(30,4,4),
     3 H(30,4),HH(30,4,4)
      COMMON FA,W,SIGMAS,T,EI,DELTMR,DELTML
  981 FI=W*T(I)
  982 CALL DTRANM(DTM,FI,I)
  983 CALL TRANMI (DTMI,DTM,I)
  984 CALL TRANIT (DTMIT,DTM,I)
  985 CALL TRANSM (TM,FI,I)
  986 CALL TRANMI(TMI,TM,I)
  987 CALL TRANIT (TMIT,TM,I)
  988 CALL MULTAA(HH,H,I)
  989 CALL MULTD(A,HH,TMI,I)
  990 CALL MULTD(B,HH,DTMI,I)
  991 CALL MULTD (D,DTMIT,A,I)
  992 CALL MULTD (F,TMIT,B,I)
  993 CALL SUMC (ERR,D,F,SIGMAS,I)
      RETURN
      END
```

```
      SUBROUTINE CHECK (TARIM,TARI,OLCOSR,HR,TT,CRUD)

C     BLOCK ONE

C     CHANGING ONLY ONE TIME SHOULD RESULT IA AN
C     ACCURATE PREDICTION OF CHANGE IN COST
C     WITHIN .10/0 OF ACTUAL CHANGE
*     LIST
*     LABEL
      DIMENSION  EAR(4,4),EI(4,4),F(30),
     1 TARI(30),T(30)
      COMMON FA,W,SIGMAS,T,EI,DELTMR,DELTML
      PRINT 16
   16 FORMAT(1H5,10X14H ACCURACY TEST)
    1 BIGA=ABSF(TARIM)/CRUD
    2 TT(15)=TT(15)-TARI(15)/BIGA
    3 CALL ERRORM (EAR,HR,PCOSR,TT)
      CALL PRINTB (EAR)
    4 COSTR=(TARI(15)**2)/BIGA
      PRINT 14,COSTR
   14 FORMAT(1H0,10X28H PREDICTED COST CHANGE  TEST,E20.8)
    5 ACOSTR=OLCOSR-PCOSR
      PRINT 15,ACOSTR
   15 FORMAT(1H0,10X25H ACTUAL COST CHANGE  TEST,E20.8)
    6 IF(ACOSTR) 7,7,8
    7 PRINT 71
   71 FORMAT(1H0,10X38H NEW COST GREATER THAN OLD COST   TEST)
      GO TO 131
    8 IF((COSTR-ACOSTR)/OLCOSR) 81,13,91
   81 BUTT=(ACOSTR-COSTR)/OLCOSR
    9 IF (BUTT-.001) 13,13,11
   91 RUTT=(COSTR-ACOSTR)/OLCOSR
   10 IF (RUTT-.001) 13,13,12
   11 PRINT 111,BUTT
  111 FORMAT(1H0,10X32H TEST   ACTUAL EXCEEDS PREDICTED,
     1 KX,F20.10,5X19H PERCENT DIFFERENCE)
      GO TO 131
   12 PRINT 121,RUTT
  121 FORMAT(1H0,10X32H TEST   PREDICTED EXCEEDS ACTUAL,
     1 5X,F20.10,5X19H PERCENT DIFFERENCE)
   13 PRINT 132
  132 FORMAT(1H0,10X27H SO FAR SO GOOD  TEST WORKS)
  131 TT(15)=TT(15)+TARI(15)/BIGA
      RETURN
      END
```

```
      SUBROUTINE CHECK (TARIM,TARI,OLCOSR,HS)

C     BLOCK TWO

C     CHANGING ONLY ONE TIME SHOULD RESULT IA AN
C     ACCURATE PREDICTION OF CHANGE IN COST
C     WITHIN .10/0 OF ACTUAL CHANGE
*     LIST
*     LABEL
      DIMENSION TARI(30),F(30),T(30),EI(4,4),EA(4,4),HS(30,4)
      COMMON FA,W,SIGMAS,T,EI,DELTMR,DELTML
      PRINT 16
   16 FORMAT(1H5,10X14H ACCURACY TEST)
    1 BIGA=ABSF(TARIM)/DELTMR
    2 T(15)=T(15)-TARI(15)/BIGA
    3 CALL ERRORM (EA,HS,PCOSR)
      CALL PRINTB (EA)
    4 COSTR=(TARI(15)**2)/BIGA
      PRINT 14,COSTR
   14 FORMAT(1H0,10X28H PREDICTED COST CHANGE  TEST,E20.8)
    5 ACOSTR=OLCOSR-PCOSR
      PRINT 15,ACOSTR
   15 FORMAT(1H0,10X25H ACTUAL COST CHANGE  TEST,E20.8)
    6 IF(ACOSTR) 7,7,8
    7 PRINT 71
   71 FORMAT(1H0,10X38H NEW COST GREATER THAN OLD COST   TEST)
      GO TO 131
    8 IF((COSTR-ACOSTR)/OLCOSR) 81,13,91
   81 BUTT=(ACOSTR-COSTR)/OLCOSR
    9 IF (BUTT-.001) 13,13,11
   91 RUTT=(COSTR-ACOSTR)/OLCOSR
   10 IF (RUTT-.001) 13,13,12
   11 PRINT 111,BUTT
  111 FORMAT(1H0,10X32H TEST   ACTUAL EXCEEDS PREDICTED,
     2 5X,F20.10,5X19H PERCENT DIFFERENCE)
      GO TO 131
   12 PRINT 121,RUTT
  121 FORMAT(1H0,10X32H TEST   PREDICTED EXCEEDS ACTUAL,
     1 5X,F20.10,5X19H PERCENT DIFFERENCE)
   13 PRINT 132
  132 FORMAT(1H0,10X27H SO FAR SO GOOD  TEST WORKS)
  131 T(15)=T(15)+TARI(15)/BIGA
      RETURN
      END
```

```
      SUBROUTINE LOGIC(BIGA,TARIM,CRUD,TARI,OLCOSR,PCOSR,
     1 EAR,HR,TT)

C     BLOCK ONE

C     DETERMINES NEW MEASUREMENT TIME SCHEDULE AND
C     NEW COST
*     LIST
*     LABEL
      DIMENSION F(30),EAR(4,4),EI(4,4),HR(4),FR(30),
     1 DELTR(30),T(30),TR(30),TT(30)
      COMMON FA,W,SIGMAS,T,EI,DELTMR,DELTML
    1 BIGA=ABSF(TARIM)/CRUD
      PRINT 26,CRUD
   26 FORMAT(1H0,10X19H MAX TIME INCREMENT,5X,F20.10)
      PRINT 27,BIGA
   27 FORMAT(1H0,10X13H SCALE FACTOR,5X,E20.8)
    2 DO 45 I=1,30,1
    3 DELTR(I)=TARI(I)/BIGA
    4 TR(I)=TT(I)-DELTR(I)
   41 IF(TR(I))42,45,43
   42 DELTR(I)=DELTR(I)+TR(I)
      TR(I)=0.0
      GO TO 45
   43 IF(TR(I)-(FA/W)) 45,45,44
   44 DELTR(I)=DELTR(I)+(TR(I)-(FA/W))
      TR(I)=FA/W
   45 CONTINUE
    5 DO 6 I=1,30,1
   61 F(I)=W*TR(I)
   62 F(I)=F(I)*180./3.14159
    6 TT(I)=TR(I)
    7 CALL ERRORM (EAR,HR,PCOSR,TT)
    8 COSTR=0.
    9 DO 10 I=1,30,1
   10 COSTR=COSTR+TARI(I)*DELTR(I)
   11 ACOSTR=OLCOSR-PCOSR
      PRINT 28,COSTR
   28 FORMAT(1H0,10X22H PREDICTED COST CHANGE,5X,E20.8)
      PRINT 29,ACOSTR
   29 FORMAT(1H0,10X19H ACTUAL COST CHANGE,5X,E20.8)
   12 IF (COSTR-ACOSTR) 131,15,141
  131 BOO=ACOSTR/COSTR
   13 IF (BOO-10.) 15,15,19
  141 FOO=COSTR/ACOSTR
   14 IF (FOO-10.) 15,15,20
   15 APERR=ACOSTR/OLCOSR
      PRINT 30,APERR
   30 FORMAT(1H0,10X15H PERCENT CHANGE,5X,F20.10)
      IF (APERR)111,333,555
  111 IF (.001+APERR)222,333,444
```

```
222 CRUD=CRUD*(.001/ABSF(APERR))
    GO TO 1
333 CRUD=CRUD/2.0
    GO TO 1
444 CRUD=CRUD*(1.0-ABSF(APERR)/.001)
    GO TO 1
555 IF (APERR-.001)21,18,18
 21 CRUD=CRUD*(.001/APERR)
    IF(CRUD-10000.)22,18,18
 22 GO TO 1
 18 PRINT 31
 31 FORMAT(1H0,10X18H NEW TIME SCHEDULE)
    CALL PRINTA(TT)
    PUNCH 311,TT
311 FORMAT(F20.10)
    PRINT 32
 32 FORMAT(1H0,10X19H NEW CENTRAL ANGLES)
181 CALL PRINTA (F)
    PRINT 33
 33 FORMAT(1H0,10X19H FINAL ERROR MATRIX)
    CALL PRINTB (EAR)
182 CALL PRINTR (OLCOSR,PCOSR,COSTR,ACOSTR)
    GO TO 25
 19 PRINT 191,800
191 FORMAT(1H0,10X40H ACTUAL GREATER THAN TEN TIMES PREDICTED
   1 F20.10)
    GO TO 25
 20 PRINT 201,FOO
201 FORMAT(1H0,10X40H PREDICTED GREATER THAN TEN TIMES ACTUAL
   1 F20.10)
 25 CONTINUE
    RETURN
    END
```

```
      SUBROUTINE LOGIC(BIGA,TARIM,CRUD,TARI,OLCOSR,PCOSR,EA,HS
C     BLOCK TWO
C     DETERMINES NEW MEASUREMENT TIME SCHEDULE AND
C     NEW COST
*     LIST
*     LABEL
      DIMENSION F(30),TR(30),FR(30),DELTR(30),TARI(30),
     1EA(4,4),HS(30,4),T(30),EI(4,4)
      COMMON FA,W,SIGMAS,T,EI,DELTMR,DELTML
    1 BIGA=ABSF(TARIM)/CRUD
      PRINT 26,CRUD
   26 FORMAT(1H0,10X19H MAX TIME INCREMENT,5X,F20.10)
      PRINT 27,BIGA
   27 FORMAT(1H0,10X13H SCALE FACTOR,5X,E20.8)
    2 DO 45 I=1,30,1
    3 DELTR(I)=TARI(I)/BIGA
    4 TR(I)=T(I)-DELTR(I)
   41 IF(TR(I))42,45,43
   42 DELTR(I)=DELTR(I)+TR(I)
      TR(I)=0.0
      GO TO 45
   43 IF(TR(I)-(FA/W)) 45,45,44
   44 DELTR(I)=DELTR(I)+(TR(I)-(FA/W))
      TR(I)=FA/W
   45 CONTINUE
    5 DO 6 I=1,30,1
   61 F(I)=W*TR(I)
   62 F(I)=F(I)*180./3.14159
    6 T(I)=TR(I)
    7 CALL ERRORM (EA,HS,PCOSR)
    8 COSTR=0.
    9 DO 10 I=1,30,1
   10 COSTR=COSTR+TARI(I)*DELTR(I)
   11 ACOSTR=OLCOSR-PCOSR
      PRINT 28,COSTR
   28 FORMAT(1H0,10X22H PREDICTED COST CHANGE,5X,E20.8)
      PRINT 29,ACOSTR
   29 FORMAT(1H0,10X19H ACTUAL COST CHANGE,5X,E20.8)
   12 IF (COSTR-ACOSTR) 131,15,141
  131 BOO=ACOSTR/COSTR
   13 IF (BOO-10.) 15,15,19
  141 FOO=COSTR/ACOSTR
   14 IF (FOO-10.) 15,15,20
   15 APERR=ACOSTR/OLCOSR
      PRINT 30,APERR
   30 FORMAT(1H0,10X15H PERCENT CHANGE,5X,F20.10)
      IF(APERR)111,333,555
  111 IF(.001+APERR)222,333,444
  222 CRUD=CRUD*(.001/ABSF(APERR))
```

```
      GO TO 1
  333 CRUD=CRUD/2.0
      GO TO 1
  444 CRUD=CRUD*(1.0-ABSF(APERR)/.001)
      GO TO 1
  555 IF(APERR-.001)21,18,18
   21 CRUD=CRUD*(.001/APERR)
      IF(CRUD-10000.)22,18,18
   22 GO TO 1
   18 PRINT 31
   31 FORMAT(1H0,10X18H NEW TIME SCHEDULE)
      CALL PRINTA (T)
      PUNCH 311,T
  311 FORMAT(F20.10)
      PRINT 32
   32 FORMAT(1H0,10X19H NEW CENTRAL ANGLES)
  181 CALL PRINTA (F)
      PRINT 33
   33 FORMAT(1H0,10X19H FINAL ERROR MATRIX)
      CALL PRINTB (EA)
  182 CALL PRINTR (OLCOSR,PCOSR,COSTR,ACOSTR)
      GO TO 25
   19 PRINT 191,BOO
  191 FORMAT(1H0,10X40H ACTUAL GREATER THAN TEN TIMES PREDICTED,
     1 F20.10)
      GO TO 25
   20 PRINT 201,FOO
  201 FORMAT(1H0,10X40H PREDICTED GREATER THAN TEN TIMES ACTUAL,
     1 F20.10)
   25 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE JUMP (TLAMDA,DHLR,DHRL,PCOSR,IHV,HR,HL)
*     LIST
*     LABEL
      DIMENSION IHV(30),DELCOS(30),TLAMDA(30,4),EA(4,4),
     1HS(30,4),DHRL(4),DHLR(4),HR(4),HL(4),T(30),EI(4,4)
      COMMON FA,W,SIGMAS,T,EI,DELTMR,DELTML
      PRED=0.0
      SCALE=0.0
      NUM=0
      CALL HORIZS(IHV,HR,HL,HS)
      DO 6 I=1,30,1
      CALL LAMDA(TLAMDA,HS,I)
      IF(IHV(I)-1) 1,2,2
    1 CALL MULTG (DELCOS,TLAMDA,DHLR,I)
      GO TO 3
    2 CALL MULTG (DELCOS,TLAMDA,DHRL,I)
    3 PRINT 81, I, DELCOS(I)
   81 FORMAT(1HO,10X20H COST CHANGE FOR THE,I4,6H POINT,E20.10)
      IF(DELCOS(I))4,6,6
    4 IF(DELCOS(I)-SCALE)5,6,6
    5 SCALE=DELCOS(I)
      NUM=I
    6 CONTINUE
      IF(IHV(NUM)-1)7,8,8
    7 IHV(NUM)=1
      GO TO 9
    8 IHV(NUM)=0
    9 PRED=-DELCOS(NUM)
      PRINT 100,PRED
  100 FORMAT(1HO,10X22H PREDICTED COST CHANGE,E20.10)
      CALL PRINTD (IHV)
      CALL HORIZS (IHV,HR,HL,HS)
      CALL ERRORM (EA,HS,HOR)
      ACT=PCOSR-HOR
      PRINT 91
   91 FORMAT (1H5,10X17H NEW ERROR MATRIX)
      CALL PRINTB(EA)
      IF (ACT) 10,10,12
   10 PRINT 11,ACT
   11 FORMAT(1HO,10X20H SOMETHING IS WRONG ,
     1 24H THE NEW COST IS GREATER,5X,E20.10)
     1E20.10)
      GO TO 19
   12 PRINT 200,ACT
  200 FORMAT(1HO,10X19H ACTUAL COST CHANGE,E20.10)
      DEL=PRED/ACT
      PRINT 14,DEL
   14 FORMAT(1HO,10X29H RATIO OF PREDICTED TO ACTUAL,F20.10)
   19 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE PRINTA(AA)
*     LIST
*     LABEL
C     THIS SUBROUTINE PRINTS OUT A LIST
      DIMENSION AA(30)
      PRINT 2,AA
    2 FORMAT(4E30.10)
      RETURN
      END


      SUBROUTINE PRINTB(BB)
*     LIST
*     LABEL
C     THIS SUBROUTINE PRINTS OUT A MATRIX
      DIMENSION BB(4,4)
      PRINT 3
    3 FORMAT(1H0,10H  I    J  ,20X10H  I    J  ,
     1 20X10H  I    J  ,20X10H  I    J  )
      L=1
      LL=2
      LLL=3
      LLLL=4
      DO 5 I=1,4,1
      PRINT 4,I,L,BB(I,L),I,LL,BB(I,LL),I,LLL,BB(I,LLL),
     1I,LLLL,BB(I,LLLL)
    4 FORMAT(1H0,4(I5,I5,E20.10))
    5 CONTINUE
      RETURN
      END


      SUBROUTINE PRINTC(BB,I)
*     LIST
*     LABEL
C     THIS SUBROUTINE PRINTS OUT ONE OF THE I TH MATRICES
      DIMENSION BB(30,4,4)
      PRINT 4
    4 FORMAT(1H0,5H  I  ,5H  J  ,5H  K  ,
     1 15X5H  I  ,5H  J  ,5H  K  ,15X5H  I  ,5H  J  ,5H  K  ,
     2 15X5H  I  ,5H  J  ,5H  K  )
      L=1
      LL=2
      LLL=3
      LLLL=4
      DO 8 J=1,4,1
      PRINT 7,I,J,L,BB(I,J,L),I,J,LL,BB(I,J,LL),I,J,LLL,
     2 BB(I,J,LLL),I,J,LLLL,BB(I,J,LLLL)
    7 FORMAT(1H0,4(3I5,E15.8))
    8 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE PRINTD(IHV)

*     LIST
*     LABEL
C     IHV(I) IS THE HORIZON VECTOR SCHEDULE
C     IF IHV(I) =1 WE ARE USING THE RIGHT HORIZON
C     IF IHV(I) =0 WE ARE USING THE LEFT HORIZON
      DIMENSION IHV(30)
      DO 717 I=1,30,1
  711 IF(IHV(I)-1) 712,715,715
  712 PRINT 713,I
  713 FORMAT(1H0,5X7H AT THE,I3,25H MEASUREMENT POINT WE ARE,
     1 22H USING THE LEFT HORIZON)
  714 GO TO 717
  715 PRINT 716,I
  716 FORMAT(1H0,5X7H AT THE,I3,25H MEASUREMENT POINT WE ARE,
     1 23H USING THE RIGHT HORIZON)
  717 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE PRINTR(A,B,C,D)

*     LIST
*     LABEL
      PRINT 1
    1 FORMAT(1H0,10X9H OLD COST,21X9H NEW COST,
     1 21X12H PRED CHANGE,18X14H ACTUAL CHANGE)
      PRINT 2,A,B,C,D
    2 FORMAT(4E30.10)
      RETURN
      END
```

REFERENCES

1. Battin, R. H., Astronautical Guidance, McGraw-Hill Book Company, Inc., New York, 1964.

2. Battin, R. H., MIT Course 16.46 Notes, 1964, (Unpublished).

3. Bryson, A. E. and Denham, W. F., A Steepest-Ascent Method For Solving Optimum Programming Problems, Raytheon Company Report BR-1303, Bedford, Mass., August 1961.

4. Bryson, A. E., Denham, W. F., Carroll, F. J. and Mikami, K., Determination of Lift or Drag Programs to Minimize Re-entry Heating, Journal of the Aerospace Sciences, vol. 29, pp. 420-430, April 1962.

5. Denham, W. F. and Speyer, J. L., Optimal Measurement and Velocity Correction Programs for Midcourse Guidance, AIAA Journal, vol. 2, pp. 896-907, May 1964.

6. Kelley, H. J., "Method of Gradients," Chap. 6 of Optimization Techniques, G. Leitmann, ed., Academic Press, New York, 1962.

7. Mitchell, E. D., "Guidance of Low-Thrust Interplanetary Vehicles," MIT Doctoral Thesis, June 1963.

8. Stern, R. G., "Interplanetary Midcourse Guidance Analysis," MIT Doctoral Thesis, June 1963.