James E. Keat
Analysis of Least-Squares Attitude Determination Routine DOAOP
Computer Sciences Corp. Rept. CSC/TM-77/6034 (Feb. 1977)

CSC/TM-77/6034

ANALYSIS OF LEAST-SQUARES ATTITUDE DETERMINATION ROUTINE DOAOP

Prepared For
NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
Goddard Space Flight Center
Greenbelt, Maryland

CONTRACT NAS 5-11999
Task Assignment 698

FEBRUARY 1977

CSC
COMPUTER SCIENCES CORPORATION

ANALYSIS OF LEAST-SQUARES ATTITUDE DETERMINATION ROUTINE DOAOP

Prepared for

GODDARD SPACE FLIGHT CENTER

By

COMPUTER SCIENCES CORPORATION

Under

Contract NAS 5-11999
Task Assignment 698

Prepared by:

J. Keat 2/22/77

J. Keat Date

Reviewed by:

R. A. Berg 2/22/77

R. A. Berg Date
Task Leader

Approved by:

R. M. Beard 2/22/77

R. M. Beard Date
Section Manager

ABSTRACT

This document describes routine DOAOP. DOAOP computes a least-squares estimate of the three-axis attitude of a spacecraft at a single time point, t_r. It requires as input (1) a data set containing the body frame components, $\hat{W}_i(t_r)$, of $i = 1, \ldots, n \geq 2$ observed unit vectors and (2) a data set of the geocentric inertial frame components, $\hat{V}_i(t_r)$, of these same unit vectors. The least-squares estimate will be generalized to a weighted least-squares one if the $\hat{W}_i(t_r)$ and/or $\hat{V}_i(t_r)$ are multiplied by weighting factors before being passed to DOAOP.

The main body of the document is divided into two parts: the first part discusses the basic attitude determination algorithm which is used in DOAOP; the second part discusses DOAOP itself, including the auxiliary computations and operations which have been implemented to support the basic algorithm. Appendix A of the document discusses a new, alternate algorithm for computing a least-squares estimate of spacecraft attitude and describes simulation tests which were performed recently using it.

TABLE OF CONTENTS

TABLE OF CONTENTS (Cont'd)

Appendix A (Cont'd)

LIST OF ILLUSTRATIONS

Figure

SECTION 1 - INTRODUCTION

This document describes a digital computer routine, DOAOP, which computes the three-axis attitude of spacecraft. DOAOP is an implementation of an attitude-estimation algorithm which was derived originally in Reference 1.

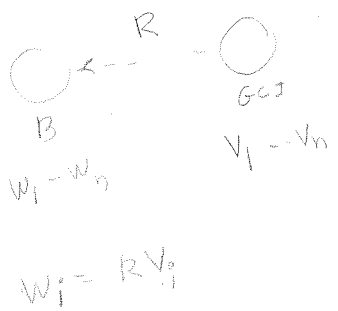
The main input required by DOAOP are (1) a data set containing the body frame components $\hat{W}_i(t_r)$ of $i = 1, \ldots, n \geq 2$ unit vectors and (2) a data set containing the geocentric inertial frame components, $\hat{V}_i(t_r)$, of these same unit vectors. The attitude which is computed by DOAOP will be a least-squares estimate of the spacecraft's attitude at the single time point t_r. This estimate, however, will be generalized to a weighted least-squares one if the $\hat{W}_i(t_r)$ and/or the $\hat{V}_i(t_r)$ are multiplied by weighting factors before being passed to DOAOP.

In practice, the $\hat{W}_i$ are observed vector components and the $\hat{V}_i$ are reference vector components. The $\hat{W}_i$ are obtained from onboard sensors, such as Sun sensors, magnetometers, or star trackers. The $\hat{V}_i$ are obtained independently of the $\hat{W}_i$. For a Sun observation, $\hat{V}_i$ normally is obtained using an ephemeris routine. For a magnetic field observation, $\hat{V}_i$ is obtained from orbit data and a magnetic field routine. For a star observation, $\hat{V}_i$ is obtained from a star catalog. When an onboard star tracker is employed, identifying the stars which are observed can be a major problem. Star identification and all other operations required in the generation of the $\hat{W}_i$ and $\hat{V}_i$ must be performed before passing their data sets to DOAOP.

As noted above, DOAOP requires $n \geq 2$ separate observation vectors at each time point, t_r, where attitude is to be computed. Spacecraft sensor systems which actually observe many observations simultaneously are rare. The requirement that the observations actually be obtained simultaneous obviously is eliminated if the spacecraft's attitude variation is negligible during the time spanned by the available set of observations. Also, spacecraft such as HEAO-A which

contain highly accurate gyros can, in effect, obtain simultaneous observations even though the attitude motion during the time spanned by the set of observations is nonnegligible. This is accomplished by using the gyro data to transform the vector components $\hat{W}_i(t_i)$; i = 1, ... , n into new components $\hat{W}_i(t_r)$. Except for transformation errors, the resulting $\hat{W}_i(t_r)$ are the components which would actually have been obtained if each observation i had been made at t_r instead of at t_i .

Section 2 of this document presents the mathematics of the basic attitude determination algorithm implemented in DOAOP. This section primarily is a repeat of material given in Reference 1 with some alterations in notation, mathematical details, and point of view. Section 3 discusses the DOAOP routine, including the auxiliary computations and operations which have been implemented to support the basic least-squares algorithm.



SECTION 2 - LEAST-SQUARES ATTITUDE DETERMINATION ALGORITHM

2.1 DERIVATION OF THE LEAST-SQUARES ATTITUDE DETERMINATION ALGORITHM

2.1.1 Development of the Least-Squares Attitude Gain Function g(R)

Let GCI be an inertially fixed reference frame and let B be a body-fixed frame in the spacecraft. Let R be the unknown 3 by 3 attitude matrix of frame B relative to frame GCI. That is, R transforms vector components from frame GCI resolution to frame B resolution. The objective is to determine R at a selected time t_r. It is assumed that observations of $n \geq 2$ distinct vectors have been obtained at the single time point t_r. Let the symbol $\hat{W}_i$ denote the observed components relative to frame B. Let $\hat{V}_i$ denote the reference components relative to frame GCI. It is emphasized that $\hat{W}_i$ and $\hat{V}_i$ (and all other vectors to be introduced subsequently) are not acutally vectors but 3 x 1 matrices that consist of the components of vectors; this definition of $\hat{W}_i$ and $\hat{V}_i$ will not preclude the use of the conventional vector cross and dot product notation later.

The problem is to devise an algorithm for estimating R from the $\hat{W}_i$ and $\hat{V}_i$. A weighted least-squares approach will be taken. The optimal estimate of R is defined to be the estimate which minimizes the following loss function $\ell(R)$:

$$\ell(R) = \frac{1}{2}\sum_{i=1}^{n} a_i \, \| \hat{W}_i - R \cdot \hat{V}_i \|^2 \qquad (2\text{-}1)$$

where $\| \ \|$ signifies the Euclidean norm. The a_i are optional weighting factors which can be assigned to the individual residuals.

Multiplying out the right side of Equation (2-1) yields three terms, two of which can be dropped, however, because they do not contain R and hence will not

affect the value of R which minimizes the loss function. Reversing the sign of the remaining term then yields a gain function which will be designated as g(R) :

$$g(R) = \sum_{i=1}^{n} a_i \hat{W}_i^T \cdot R \cdot \hat{V}_i \qquad (2\text{-}2)$$

where superscript T denotes transposition.

The problem now is to find the R which maximizes g(R) . Before addressing this problem, however, a slight simplification will be introduced into Equation (2-2) by eliminating the explicit appearance of the weighting factors a_i . This could be done by absorbing them into the $\hat{W}_i$ via the introduction of non-unit vectors $\vec{W}_i$.

$$\vec{W}_i = a_i \hat{W}_i \qquad (2\text{-}3a)$$

Alternatively, they could be absorbed into the $\hat{V}_i$ by introducing $\vec{V}_i$ vectors

$$\vec{V}_i = a_i \hat{V}_i \qquad (2\text{-}3b)$$

Finally, they could be absorbed into both the $\hat{W}_i$ and the $\hat{V}_i$, e.g.,

$$\begin{aligned} \vec{W}_i &= \sqrt{a_i}\, \hat{W}_i \\ \vec{V}_i &= \sqrt{a_i}\, \hat{V}_i \end{aligned} \qquad (2\text{-}3c)$$

For generality, the following equations will use both nonunit vectors $\vec{W}_i$ and $\vec{V}_i$. This does not, however, make the mathematics inapplicable to the weighting techniques shown in Equations (2-3a) and (2-3b). Equation (2-2) thus will be written as follows:

$$g(R) = \sum_{i=1}^{n} \vec{W}_i^T \cdot R \cdot \vec{V}_i \qquad (2\text{-}4)$$

2.1.2 Introduction of the Attitude Vector $\vec{Y}$

It is necessary at this point to introduce some of the attitude vector concepts of Reference 1. It is noted that the attitude least-squares problem being discussed here was posed first in Reference 2 in a slightly different form. (In Reference 2, the problem was stated from an alibi point of view whereas this document presents it in an alias form.) Several solutions to the problem later were summarized in Reference 3. These solutions all involved a direct determination of the optimal R and were derived using matrix methods. The work on the attitude least-squares problem in Reference 1 was motivated, at least in part, by Reference 2. The approach presented in Reference 1 was quite different from those in Reference 3. Reference 1 did not attack the problem through matrix methods. Instead, it uitlized a vector-like variable, $\vec{Y}$, which will be defined subsequently. Reference 1 was devoted primarily to developing an algebra for attitude representation using $\vec{Y}$ and, alternatively, using a similar variable, $\vec{Z}$. The least-squares problem was included primarily as an example of the application of these techniques.

The quantity $\vec{Y}$ sometimes is called the Gibbs vector. It is defined as

$$\vec{Y} = \hat{X} \tan \frac{\theta}{2} \qquad (2\text{-}5)$$

$\hat{X}$ and θ are the parameters of a rotation which would rotate the axes of frame GCI onto those of frame B. The unit vector $\hat{X}$ lies along the axis of this rotation. The parameter θ is the angle of the rotation. The polarity of $\hat{X}$ is chosen so that θ has the range $0 \leq \theta \leq \pi$. In the present work $\vec{Y}$ and $\hat{X}$ should not be regarded as vectors per se but as the components of vectors along the axes of the B or GCI frames. (Since $\vec{Y}$ and $\hat{X}$ both lie along the axis of the rotation, their components along the B frame axes will be identical to those along the GCI frame axes.)

The relationship between $\vec{Y}$ and R will now be developed. The following two equations, which define R as a function of $\hat{X}$ and θ, are well known expressions:

$$\underset{3\times3}{R} = c\theta\, I + [1 - c\theta]\, \hat{X} \cdot \hat{X}^T - s\theta\, \underset{\sim}{\hat{X}} \tag{2-6a}$$

$$R = \left[c^2 \frac{\theta}{2} - s^2 \frac{\theta}{2}\right] I + 2s^2 \frac{\theta}{2}\, \hat{X} \cdot \hat{X}^T - 2s \frac{\theta}{2}\, c \frac{\theta}{2}\, \underset{\sim}{\hat{X}} \tag{2-6b}$$

The symbols c and s above signify sine and cosine, respectively. I is the 3 x 3 identity matrix. The wavy underbar signifies the usual 3 x 3 skew-symmetric arrangement of vectors. Letting x_1, x_2, x_3 denote the components of $\hat{X}$, the full form of the third term in Equation (2-6a) thus is

$$s\theta\, \underset{\sim}{\hat{X}} = s\theta \begin{bmatrix} 0 & -x_3 & x_2 \\ x_3 & 0 & -x_1 \\ -x_2 & x_1 & 0 \end{bmatrix} \tag{2-7}$$

The expression $\hat{X} \cdot\cdot \hat{X}^T$ in Equation (2-6) signifies the outer product of $\hat{X}$ with itself. The full form of the second term in Equation (2-6a) thus is

$$[1 - c\theta]\hat{X} \cdot \hat{X}^T = [1 - c\theta]\begin{Bmatrix} x_1 \\ x_2 \\ x_3 \end{Bmatrix}\{x_1\ x_2\ x_3\}$$

$$= [1 - c\theta]\begin{bmatrix} x_1^2 & x_1x_2 & x_1x_3 \\ x_1x_2 & x_2^2 & x_2x_3 \\ x_1x_3 & x_2x_3 & x_3^2 \end{bmatrix} \qquad (2\text{-}8)$$

It will be noted at this point that the vector cross and dot products will be indicated in the usual manner in subsequent equations. In particular, the dot product will be indicated as merely $\vec{A} \cdot \vec{B}$, for example, rather than as $\vec{A}^T \cdot \vec{B}$.

By utilizing Equation (2-5) and performing some algebra, Equation (2-6b) can be transformed into

$$R = \frac{1}{[1 + \vec{Y} \cdot \vec{Y}]}\{[1 - \vec{Y} \cdot \vec{Y}]I + 2\vec{Y} \cdot \vec{Y}^T - 2\tilde{\vec{Y}}\} \qquad (2\text{-}9)$$

Equation (2-9) specifies R as a function solely of $\vec{Y}$ and thus is a significant expression.

2.1.3 Completion of the Derivation

To complete the derivation of the least-squares attitude algorithm of Reference 1, Equation (2-9) is first substituted into Equation (2-4), the least-squares

gain function expression. Let the resulting $\vec{Y}$-dependent function be designated as $g(\vec{Y})$. Thus,

$$g(\vec{Y}) = \sum_{i=1}^{n} \vec{W}_i^T \cdot \left[\frac{1}{\{1 + \vec{Y} \cdot \vec{Y}\}} \{[1 - \vec{Y} \cdot \vec{Y}] I + 2\vec{Y} \cdot \vec{Y}^T - 2\underset{\sim}{\vec{Y}}\} \right] \cdot \vec{V}_i \qquad (2\text{-}10)$$

Some simple manipulations yield

$$g(\vec{Y}) = \frac{1}{1 + \vec{Y} \cdot \vec{Y}} \sum_{i=1}^{n} \{[1 - \vec{Y} \cdot \vec{Y}][\vec{W}_i \cdot \vec{V}_i] + 2[\vec{W}_i \cdot \vec{Y}][\vec{Y} \cdot \vec{V}_i] + 2\vec{Y} \cdot [\vec{W}_i \times \vec{V}_i]\} \qquad (2\text{-}11)$$

Equation (2-11) and subsequent equations include product expressions such as $\vec{W}_i \cdot \vec{V}_i$, which involve vector components along two different coordinate frames. It is emphasized that no coordinate transformation is to be made before multiplying out these products.

The problem now is to determine the $\vec{Y}$ which maximizes $g(\vec{Y})$. To accomplish this, the gradient of $g(Y)$ is formed and this result is set to $\vec{0}$. If the resulting equation can be solved for $\vec{Y}$, it will establish the $\vec{Y}$ vectors which produce the stationary values of $g(\vec{Y})$. To aid in the differentiation, the following formulas are noted:

$$\frac{d}{d\vec{Y}} [1 + \vec{Y} \cdot \vec{Y}]^{-1} = \frac{-2\vec{Y}}{[1 + \vec{Y} \cdot \vec{Y}]^2} \qquad (2\text{-}12a)$$

$$\frac{d}{d\vec{Y}} [\vec{Y} \cdot \vec{Y}] = +2\vec{Y} \qquad (2\text{-}12b)$$

$$\frac{d}{d\vec{Y}} [\vec{W}_i \cdot \vec{Y}] [\vec{Y} \cdot \vec{V}_i] = [\vec{W}_i \cdot \vec{Y}] \vec{V}_i + [\vec{Y} \cdot \vec{V}_i] \vec{W}_i \tag{2-12c}$$

$$\frac{d}{d\vec{Y}} \vec{Y} [\vec{W}_i \times \vec{V}_i] = \vec{W}_i \times \vec{V}_i \tag{2-12d}$$

For convenience, the present work regards the derivative of a scalar with respect to a vector as being a column vector rather than a row vector. Differentiating Equation (2-11) with the aid of Equations (2-12), setting the result to zero, and performing some algebraic manipulation yields

$$\vec{0} = \frac{-4\vec{Y}}{[1 + \vec{Y} \cdot \vec{Y}]^2} \sum_{i=1}^{n} \{[\vec{W}_i \cdot \vec{V}_i] + [\vec{W}_i \cdot \vec{Y}] [\vec{Y} \cdot \vec{V}_i] + \vec{Y} \cdot [\vec{W}_i \times \vec{V}_i]\}$$

$$+ \frac{2}{[1 + \vec{Y} \cdot \vec{Y}]^2} [1 + \vec{Y} \cdot \vec{Y}] \sum_{i=1}^{n} \{[\vec{W}_i \cdot \vec{Y}] \vec{V}_i + [\vec{Y} \cdot \vec{V}_i] \vec{W}_i + \vec{W}_i \times \vec{V}_i\} \tag{2-13}$$

Making a final minor rearrangement now yields

$$\vec{Y} = \frac{0.5 [1 + \vec{Y} \cdot \vec{Y}] \sum_{i=1}^{n} \{[\vec{V}_i \cdot \vec{Y}] \vec{W}_i + [\vec{W}_i \cdot \vec{Y}] \vec{V}_i + \vec{W}_i \times \vec{V}_i\}}{\sum_{i=1}^{n} \{\vec{Y} \cdot [\vec{W}_i \times \vec{V}_i] + [\vec{V}_i \cdot \vec{Y}] [\vec{W}_i \cdot \vec{Y}] + \vec{W}_i \cdot \vec{V}_i\}} \tag{2-14}$$

Except for a few minor alterations in notation, Equation (2-14) is the equation given on page 18 of Reference 1. It also is the relation implemented in DOAOP.

As discussed in Section 3.3, DOAOP solves Equation (2-14) by iteration using an initial input $\vec{Y}_1$. In the numerous runs which have been made with DOAOP since its inception, the iteration technique reportedly has never failed to converge.

2.2 A SIMPLIFIED FORM OF THE LEAST-SQUARES ATTITUDE ALGORITHM

A significantly simplified form of Equation (2-14) was pointed out recently by P. Davenport. A derivation of this new form is presented in this section. The final equations of the new form are Equations (2-18), (2-22), and (2-26).

Consider first the summation term in the numerator of Equation (2-14). Writing the summations separately yields

$$\sum \vec{W}_i [\vec{V}_i \cdot \vec{Y}] + \sum \vec{V}_i [\vec{W}_i \cdot \vec{Y}] + \sum \vec{W}_i \times \vec{V}_i \qquad (2\text{-}15)$$

where, for simplicity, the range of the summations is omitted. A simple manipulation of the first two terms in Equation (2-15) yields

$$\left[\sum \vec{W}_i \cdot \vec{V}_i^T + \sum \vec{V}_i \cdot \vec{W}_i^T\right] \cdot \vec{Y} + \sum \vec{W}_i \times \vec{V}_i \qquad (2\text{-}16)$$

Let the third term in Equation (2-16) be designated as $\vec{Z}$

$$\vec{Z} = \sum \vec{W}_i \times \vec{V}_i \qquad (2\text{-}17)$$

and let the first term inside the brackets be designated as B .

$$B = \sum \vec{W}_i \cdot \vec{V}_i^T = [\vec{W}_1, \ldots, \vec{W}_n] [\vec{V}_1, \ldots, \vec{V}_n]^T \qquad (2\text{-}18)$$

The second term inside the brackets in Equation (2-16) is B^T

$$B^T = \sum \vec{V}_i \cdot \vec{W}_i^T \qquad (2\text{-}19)$$

Substituting Equations (2-17), (2-18), and (2-19) into Equation (2-16) yields

$$[B + B^T] \cdot \vec{Y} + \vec{Z} \tag{2-20}$$

This is the new form of the terms inside the summation in the numerator of Equation (2-14).

It is possible to compute $\vec{Z}$ directly from B, rather than with Equation (2-17). To derive the algorithm, let $\vec{Q}$ be an arbitrary vector and consider the product $[B^T - B] \cdot \vec{Q}$. Substituting Equations (2-18) and (2-19) into this product yields

$$\begin{aligned}
[B^T - B] \cdot \vec{Q} &= \sum \left[\vec{V}_i \cdot \vec{W}_i^T \right] \cdot \vec{Q} - \sum \left[\vec{W}_i \cdot \vec{V}_i^T \right] \cdot \vec{Q} \\
&= \sum \{ \vec{V}_i [\vec{W}_i \cdot \vec{Q}] - \vec{W}_i [\vec{V}_i \cdot \vec{Q}] \} \\
&= \sum \vec{Q} \times [\vec{V}_i \times \vec{W}_i] = \left\{ \sum \vec{W}_i \times \vec{V}_i \right\} \times \vec{Q} \\
&= \vec{Z} \times \vec{Q} = \underset{\sim}{\vec{Z}} \cdot \vec{Q}
\end{aligned} \tag{2-21}$$

Since $\vec{Q}$ is arbitrary, the result obtained above demonstrates that

$$\underset{\sim}{\vec{Z}} = B^T - B \tag{2-22}$$

Thus, $\vec{Z}$ can be computed with Equation (2-22) rather than with Equation (2-17). For any large number of observations, Equation (2-22) requires fewer numerical operations than the lengthy summation computations required by Equation (2-17). Forming the 3 x 1 vector $\vec{Z}$ from its 3 x 3 skew symmetric form $\underset{\sim}{\vec{Z}}$ is a minor step.

The following relation is noted at this point for use in a later paragraph:

$$\text{Trace } B = \sum \vec{W}_i \cdot \vec{V}_i \tag{2-23}$$

Equation (2-23) can be developed from the second portion of Equation (2-18) through use of the following two properties of the trace operator:

1. $$\text{Trace}\left[\sum_{\alpha=1}^{r} \underset{m\times m}{M_\alpha}\right] = \sum_{\alpha=1}^{r} \text{Trace } M_\alpha$$

   where the M_α are arbitrary square matrices and

2. $$\text{Trace}\left[\underset{m\times n}{P} \cdot \underset{n\times m}{N}\right] = \text{Trace}\,[N \cdot P]$$

   where N and P are also arbitrary except for dimensions.

The denominator in Equation (2-14) now will be considered. Simple manipulation converts it into

$$\vec{Y} \cdot \sum \vec{W}_i \times \vec{V}_i + \vec{Y}^T \cdot \left[\sum \vec{W}_i \cdot \vec{V}_i^T\right] \cdot \vec{Y} + \sum \vec{W}_i \cdot \vec{V}_i \tag{2-24}$$

Inserting Equations (2-17), (2-18), and (2-23) into (2-24) then yields

$$\vec{Y} \cdot \vec{Z} + \vec{Y}^T \cdot B \cdot \vec{Y} + \text{Trace } B \tag{2-25}$$

which is the desired simplified expression for the denominator of Equation (2-14).

To summarize, the simplified form of Equation (2-14) is

$$\vec{Y} = \frac{0.5\,[1 + \vec{Y} \cdot \vec{Y}]\,[\{B + B^T\} \cdot \vec{Y} + \vec{Z}]}{\vec{Y} \cdot \vec{Z} + \vec{Y}^T \cdot B \cdot \vec{Y} + \text{Trace } B} \tag{2-26}$$

In a routine which employs Equation (2-26), the matrix B would be computing using

$$B = \sum_{i=1}^{n} \vec{W}_i \cdot \vec{V}_i^T$$

which was listed earlier as Equation (2-18).

$\vec{Z}$ then would be computed using

$$\vec{Z} = B^T - B$$

which was listed earlier as Equation (2-22).

SECTION 3 - LEAST-SQUARES ATTITUDE DETERMINATION ROUTINE DOAOP

This section discusses the routine DOAOP which is an implementation of the least-squares attitude determination algorithm described in Section 2. Section 3.1 briefly discusses the application of DOAOP in a system which has been employed for attitude determination studies on HEAO-A. Section 3.2 then summarizes the main operations which are performed in DOAOP. That portion of DOAOP which actually performs the least-squares attitude computation is described in Section 3.3. Section 3.4 is a short section which denotes the computation of Euler angles α, β, δ from the least-squares attitude matrix R. These Euler angles are computed primarily for the convenience of the user of the routine. Finally, Section 3.5 discusses the computation of a preliminary attitude matrix R_o. This R_o matrix is developed using a two-observation technique. R_o is required as an input by that portion of DOAOP which performs the least-squares attitude computation.

3.1 APPLICATION OF DOAOP IN HEAO-A ATTITUDE DETERMINATION TEST SYSTEM

To illustrate the application of routine DOAOP as part of a more complete attitude determination system, Figure 3-1 presents a baseline diagram of the HEAO-A attitude acquisition test system. This system has been used in studies employing simulated HEAO-A data. In the HEAO-A application, the observation vectors, $\hat{W}_i$, which are needed by DOAOP are obtained solely from a star tracker. The gyro propagation technique is employed to obtain the data set of simultaneous observation vectors $\hat{W}_i(t_r)$ $i = 1, \ldots, n$ which are needed by DOAOP at each attitude determination time t_r. The system in Figure 3-1 does not include the capability of weighting the $\hat{W}_i$ or $\hat{V}_i$ vectors.

The two main operations performed by the system shown in Figure 3-1 are (1) the star identification operation, which is performed through the ACQID

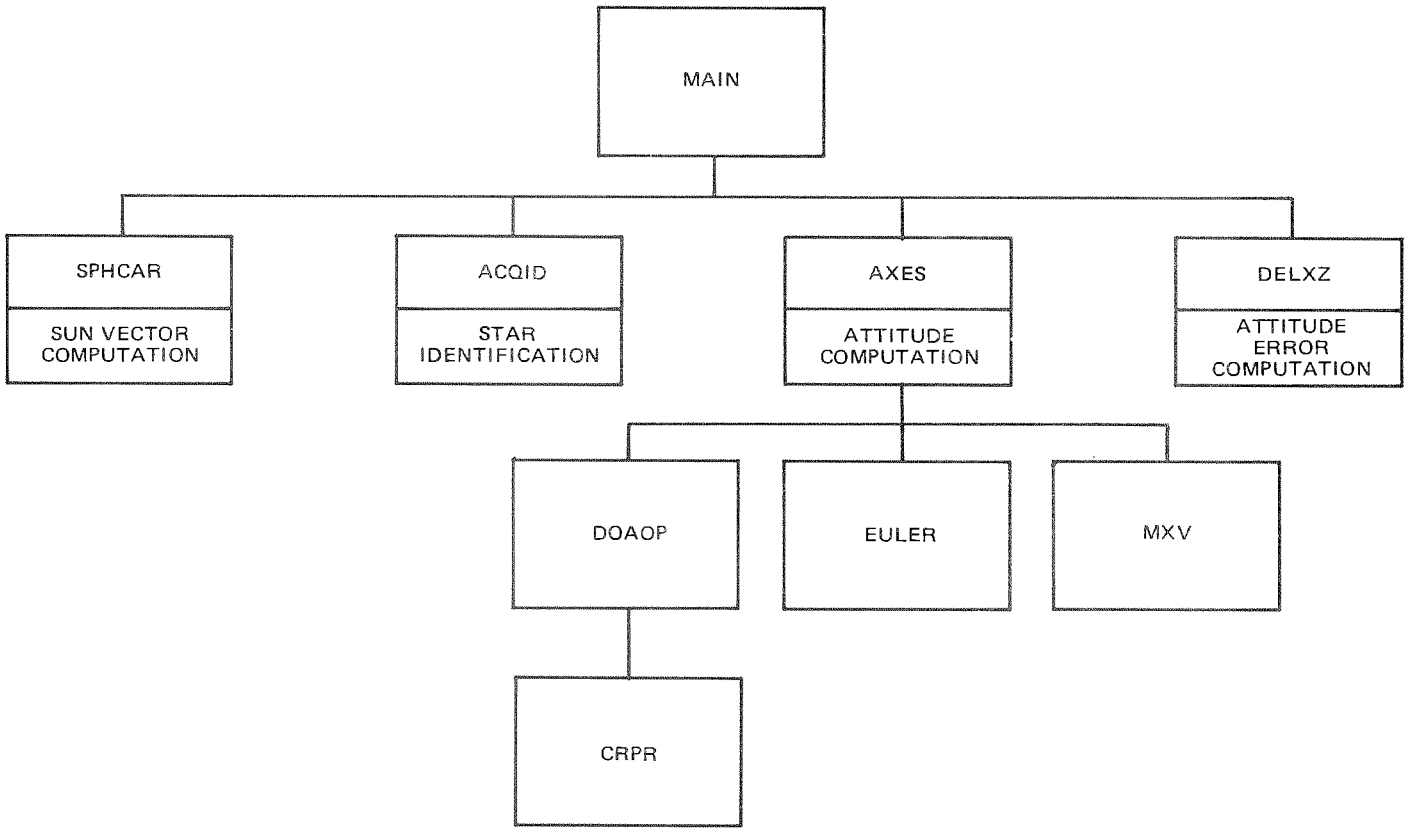


Figure 3-1. Baseline Diagram of HEAO-A Attitude Acquisition Test System Employing DOAOP

subsystem and (2) the least-squares attitude computation, which is performed through the AXES subsystem. As shown in Figure 3-1, DOAOP is called by AXES. MAIN is the driver routine. The main operations performed by MAIN are as follows (in sequential order):

1. Initialize parameter values
2. Read NAMELIST
3. Call SPHCAR to compute the unit Sun vector from its right ascension and declination values, which are input through NAMELIST
4. Read the observed star data set
5. Read the catalog star data set
6. Call ACQID to match the observed star vectors $\hat{W}_i$ to the catalog stars and generate the $\{\hat{W}_i\}$ and $\{\hat{V}_i\}$ data sets
7. Call AXES to compute spacecraft attitude
8. Call DELXZ to compute the deviation of the least squares attitude solution from a reference attitude
9. Exit

The unit Sun vector computed in SPHCAR is used solely in the star identification operation. Star identification is not within the scope of this report and thus is not discussed.

AXES is the driver for DOAOP. EULER, which also is called by AXES, computes the 3-1-3 Euler angles from the attitude matrix, R , generated by DOAOP. MXV (Figure 3-1) is a minor subroutine which merely multiplies a 3×3 matrix by a 3×1 vector. DOAOP itself calls one small subroutine, CRPR, which calculates the cross product of two vectors. In addition to acting as a driver in the system shown in Figure 3-1, AXES also computes the mean of the angular errors ϕ_i between $\hat{W}_i$ and $R \cdot \hat{V}_i$.

3.2 SYNOPSIS OF ROUTINE DOAOP

The main operations performed by DOAOP are as follows:

1. Compute a preliminary attitude matrix, R_o, using a two-observation technique
   a. Select the two observations, $\vec{W}_L$, $\vec{W}_K$, to be employed
   b. Use $\vec{W}_L$, $\vec{W}_K$ to compute R_o
2. Compute a more accurate attitude matrix, R, using all the observations $\vec{W}_1, \ldots, \vec{W}_n$
   a. Transform the reference vector components, $\vec{V}_i$, onto an intermediate coordinate frame, B_o, via $\vec{U}_i = R_o \cdot \vec{V}_i$
   b. Compute the attitude vector, $\vec{Y}$, relative to frame B_o, using the weighted least-squares algorithm discussed in Section 2
   c. Transform $\vec{Y}$ into an attitude matrix, P (this matrix still describes the attitude relative to frame B_o)
   d. Compute the final attitude matrix, R, with $R = P \cdot R_o$
3. Compute Euler angles α, β, δ from R

These three computations are discussed in detail in the following subsections. The description of the computation of R_o has been placed after the descriptions of the calculation of R and of the Euler angles. This reverse order was chosen because the least-squares calculation of R is of more present interest than is the computation of R_o. Also, the discussion of the computation of R_o is lengthy.

3.3 IMPLEMENTATION OF THE LEAST-SQUARES ALGORITHM IN DOAOP

After a preliminary estimate, R_o, of the spacecraft's attitude matrix, R, is obtained via the method discussed in Section 3.5, a more accurate estimate of R is computed using the iterative least-squares algorithm indicated previously as Equation (2-14). Figure 3-2 shows the implementation of this algorithm in DOAOP. With a few exceptions, the notation, computations, and flow on this diagram have been made similar to, or identical with, those of the coding. It is noted in particular that the symbols SMA, $\vec{Y}P$, RP, etc. denote single variables, not products.

In DOAOP, the $\vec{Y}$ vector used in the computations specifies spacecraft attitude relative to the coordinate frame defined by the preliminary attitude estimate, R_o, rather than relative to the geocentric inertial frame, GCI. To explain this, let B denote the spacecraft body-fixed frame and let B_o denote the initial estimate of frame B. The geometry is summarized in Figure 3-3.

To specify $\vec{Y}$ relative to frame B_o, it is necessary to transform the frame GCI vector components, $\vec{V}_i$, into frame B_o components, $\vec{U}_i$. This is the transformation

$$\vec{U}_i = R_o \cdot \vec{V}_i \; ; \; i = 1, \ldots, n \qquad (3\text{-}1)$$

shown in block A of Figure 3-2.

Equation (2-5) shows that $\vec{Y}$ is $\vec{0}$ when dealing with two coordinate frames which are congruent. The first estimate of $\vec{Y}$ therefore is $\vec{Y}_o = \vec{0}$. Equation (2-14) shows that the attitude estimation algorithm will be particularly

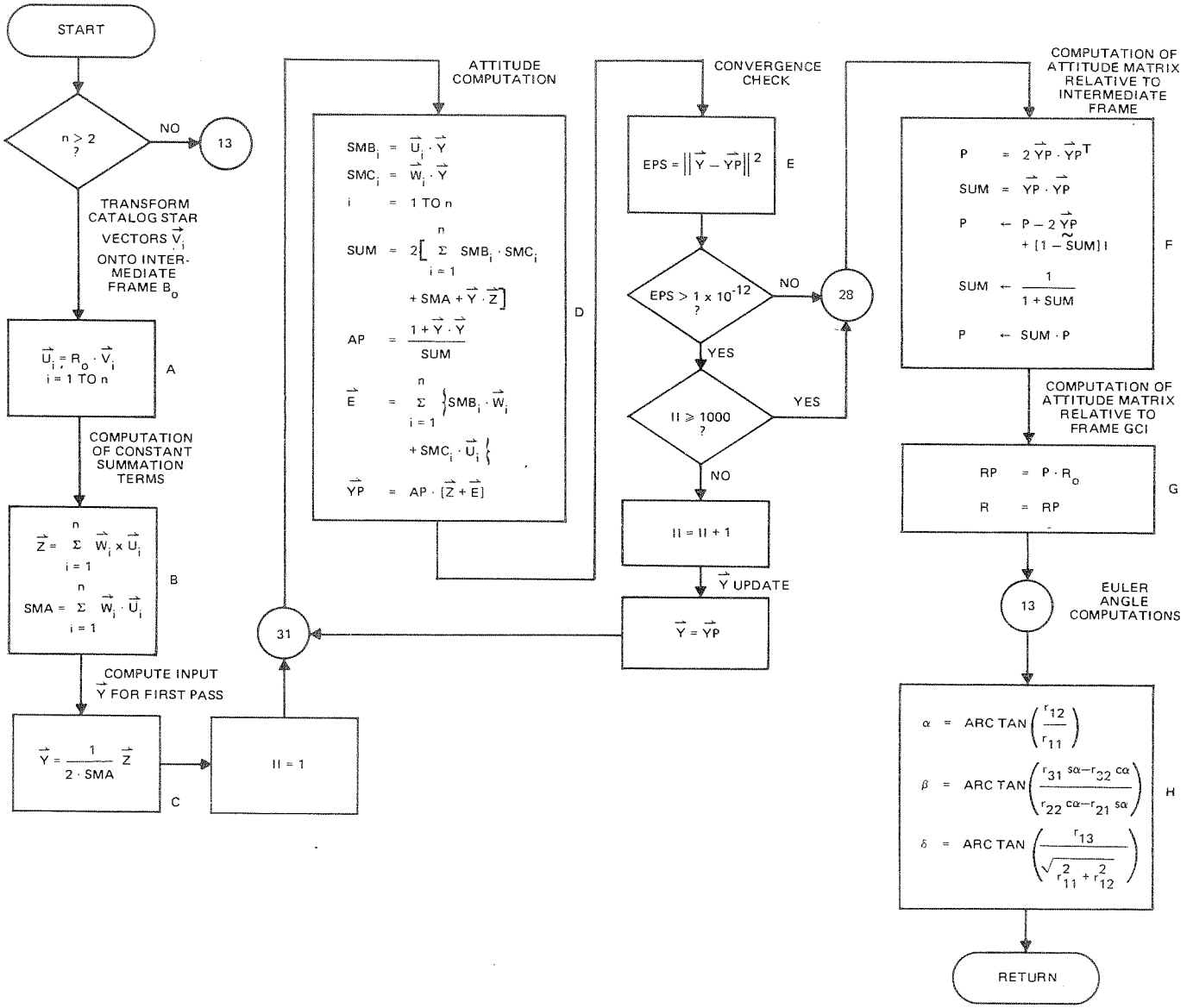


Figure 3-2. Weighted Least-Squares Attitude Computation in DOAOP

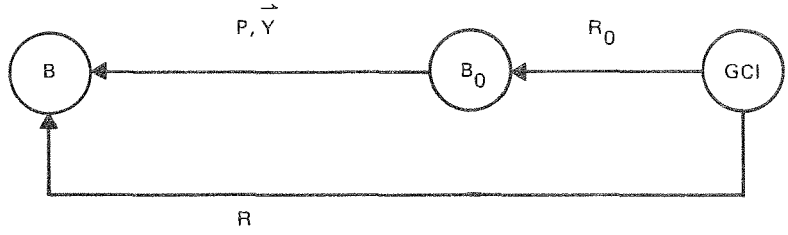


Figure 3-3. Flow Diagram Illustrating Coordinate Frames GCI, B_o, and B

simple for the first pass. That is, since $\vec{Y}_o = 0$, the first new attitude estimate, $\vec{Y}_1$, will be merely

$$\vec{Y}_1 = \frac{\sum \vec{W}_i \times \vec{U}_i}{.2 \sum \vec{W}_i \cdot \vec{U}_i} \tag{3-2}$$

This is the computation shown in block C of Figure 3-2. In subsequent attitude computation iterations, the full algorithm, Equation (2-14), must be employed because, in general, the input $\vec{Y}$ now will be nonzero.

The blocks on the second and third columns of Figure 3-2 make up the attitude iteration loop. The main computations here are performed in block D. This block is an implementation of Equation (2-14). A check of the equations in block D will show that they are analytically identical with Equation (2-14) except that the vector components, $\vec{V}_i$, of Equation (2-14) are replaced in block D by the transformed components $\vec{U}_i$.

The variable designated as $\vec{Y}$ in Figure 3-2 is the input to block C for the latest iteration pass; $\vec{Y}$ here is the attitude vector on the right-hand side of Equation (2-14). The term $\vec{YP}$ is the new attitude vector which is computed during

this pass; $\vec{YP}$ here is the term on the left-hand side of Equation (2-14). The program iterates by performing the block D computations repeatedly until convergence is achieved or until the maximum number of passes (1000) has been made without achieving convergence. The output, $\vec{YP}$, which is obtained in any one pass through block D, serves as the input, $\vec{Y}$, for the next pass.

The third column of Figure 3-2 shows the convergence check performed after each pass through block D. The convergence test employs a parameter EPS which is the square of the norm of the correction $(\vec{Y} - \vec{YP})$ generated in that pass. Thus,

$$EPS = ||\vec{Y} - \vec{YP}||^2 = (\vec{Y} - \vec{YP}) \cdot (\vec{Y} - \vec{YP}) \qquad (3\text{-}3)$$

The iteration is considered to have converged whenever EPS becomes less than 10^{-12}.

The first step after convergence is to transform the final attitude vector, $\vec{YP}$, into the corresponding attitude matrix, P . As shown in Figure 3-2, DOAOP also performs this operation if 1000 passes are made without attaining convergence. The $\vec{YP} \rightarrow P$ transformation is shown in block F of Figure 3-2. The equations listed in block F are analytically identical with Equation (2-9).

As noted earlier, P does not specify spacecraft attitude relative to the GCI frame. Instead, it specifies spacecraft attitude relative to frame B_o of the initial attitude estimate. Therefore, a transformation to yield attitude relative to frame GCI is needed. With the aid of Figure 3-3, the transformation is seen to be

$$R = P \cdot R_o \qquad (3\text{-}4)$$

which is the computation shown in block G of Figure 3-2.

3.4 COMPUTATION OF EULER ANGLES IN DOAOP

Before returning, DOAOP computes the values of a set of Euler angles α, β, δ. These angles provide the routine's user with a subjective understanding of the spacecraft's attitude. The equations for calculating α, β, δ are shown in block H of Figure 3-2. The variables r_{ij} in this block are the elements of R in the usual situation of $n > 2$, where n is the number of observations. In the degenerate case of $n = 2$, the r_{ij} are the elements of the preliminary attitude matrix R_o.

A check of the equations given in block H shows that the rotational sequence α, δ, β, when going from frame GCI to frame B, is as follows:

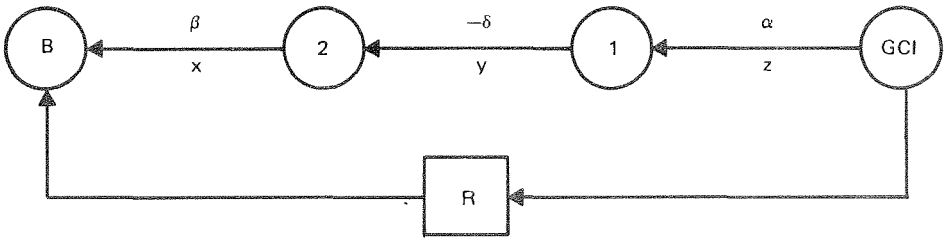


Figure 3-4. Flow Diagram Showing Euler Angles α, β, δ

Figure 3-5 illustrates the actual geometry. The validity of this interpretation of α, β, δ can be demonstrated by using Figure 3-4 to develop the matrix expression $R(\alpha, \beta, \delta)$

$$R \triangleq \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} = [\beta]_x [\delta]_y^T [\alpha]_z \qquad (3\text{-}5a)$$

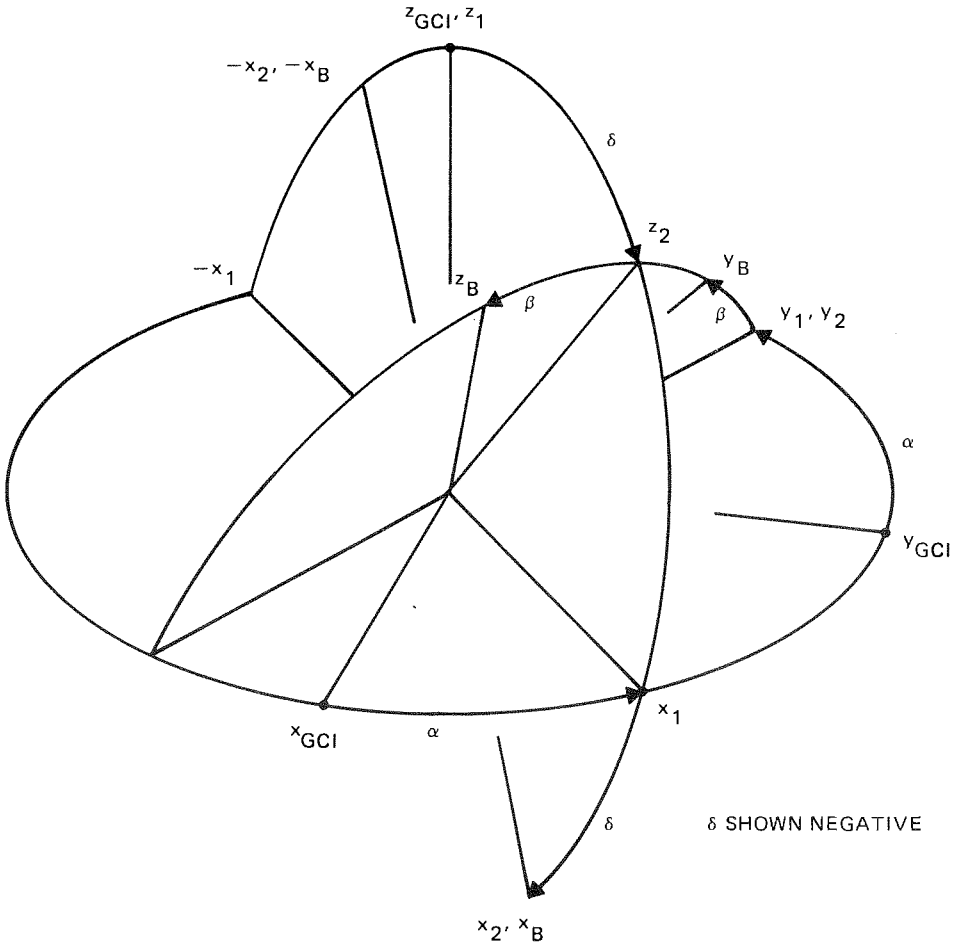


Figure 3-5. Geometry of Euler Angles α , β , δ

Multiplying out yields

$$R = \begin{array}{|c|c|c|} \hline c\alpha c\delta & s\alpha c\delta & s\delta \\ \hline -s\alpha c\beta - c\alpha s\delta s\beta & c\alpha c\beta - s\alpha s\delta s\beta & c\delta s\beta \\ \hline s\alpha s\beta - c\alpha s\delta c\beta & -c\alpha s\beta - s\alpha s\delta c\beta & c\delta c\beta \\ \hline \end{array} \qquad (3\text{-}5b)$$

Substitution of these r_{ij} $(\alpha, \beta, \delta,)$ expressions into the right-hand sides of the equations in block H will reduce these expressions to identities $\alpha = \alpha$ $\delta = \delta$, and $\beta = \beta$, thereby verifying the above interpretation of α, δ, and β.

The above Euler angles, α, β, δ, are particularly convenient for spacecraft which are spinning about their x-axis, because α and δ then are the right ascension and declination, respectively, of the spin axis, and β defines the phase of the spacecraft in its spin cycle. These angles, however, are not especially suited for the spinning mode of spacecraft such as HEAO-A which spin about their z-axis rather than about their x-axis.

3.5 PRELIMINARY ATTITUDE COMPUTATION IN DOAOP

As stated previously, the least-squares attitude determination algorithm used in DOAOP requires a preliminary attitude matrix, R_o, as an input. The computation of R_o is discussed in this section.

DOAOP computes R_o with a two-star attitude determination algorithm. The computation involves two distinct steps:

1. From the total of n available observations, select the two which will be used
2. Calculate R_o using these two observations

These two operations are discussed in the following subsections.

3.5.1 Selection of the Two Observations

The selection of the two observations will be considered first. The algorithm employed in DOAOP is summarized in Figure 3-6. The selection of the first observation (denoted here and in the coding as observation L) is shown in the first column of this figure. Perusal of this column shows that the first observation selected is the one whose reference vector $\vec{V}$ has the longest length; that is, the observation which is to be weighted the heaviest in the weighted least-squares attitude computation.

In most applications of DOAOP, the observations are not weighted. In this case, all the $\vec{V}_i$ nominally have unit lengths. At first glance, Figure 3-6 appears to indicate that the first vector in the block, $\vec{V}_i$, will be selected. However, this is not what actually happens. Instead, the unit vectors, $\vec{V}_i$, which are passed to DOAOP will usually differ slightly from unity as a result of numerical effects. In this situation, DOAOP will choose the $\vec{V}$ whose length accidently is the longest. For example, in a test run using the system shown in Figure 3-1, 25 observations (all of which supposedly had $\vec{V}_i$ vectors of unit length) were passed to DOAOP. The $\vec{V}_L$ which was picked by the routine was the 16th vector in the set. Supplementary computations verified that the length of this vector was slightly greater than unity and that it was longer than that of any other $\vec{V}_i$ in the set.

The selection of the second observation (denoted here and in the coding as observation K) is shown in the second column of Figure 3-6. When making this selection, the program computes the cosine $c\psi_{Li}$ of each of the angles ψ_{Li} between $\vec{V}_L$ and the remaining n - 1 reference vectors $\vec{V}_i$. Observation K is selected to be the one whose cosine, $c\psi_{LK}$, has the smallest algebraic value. In other words, $\vec{V}_K$ is the vector whose separation angle from $\vec{V}_L$ is the closest to π. For example, in the run noted in the previous paragraph, the separation angle between $\vec{V}_L$ and $\vec{V}_K$ was 174.25^o.

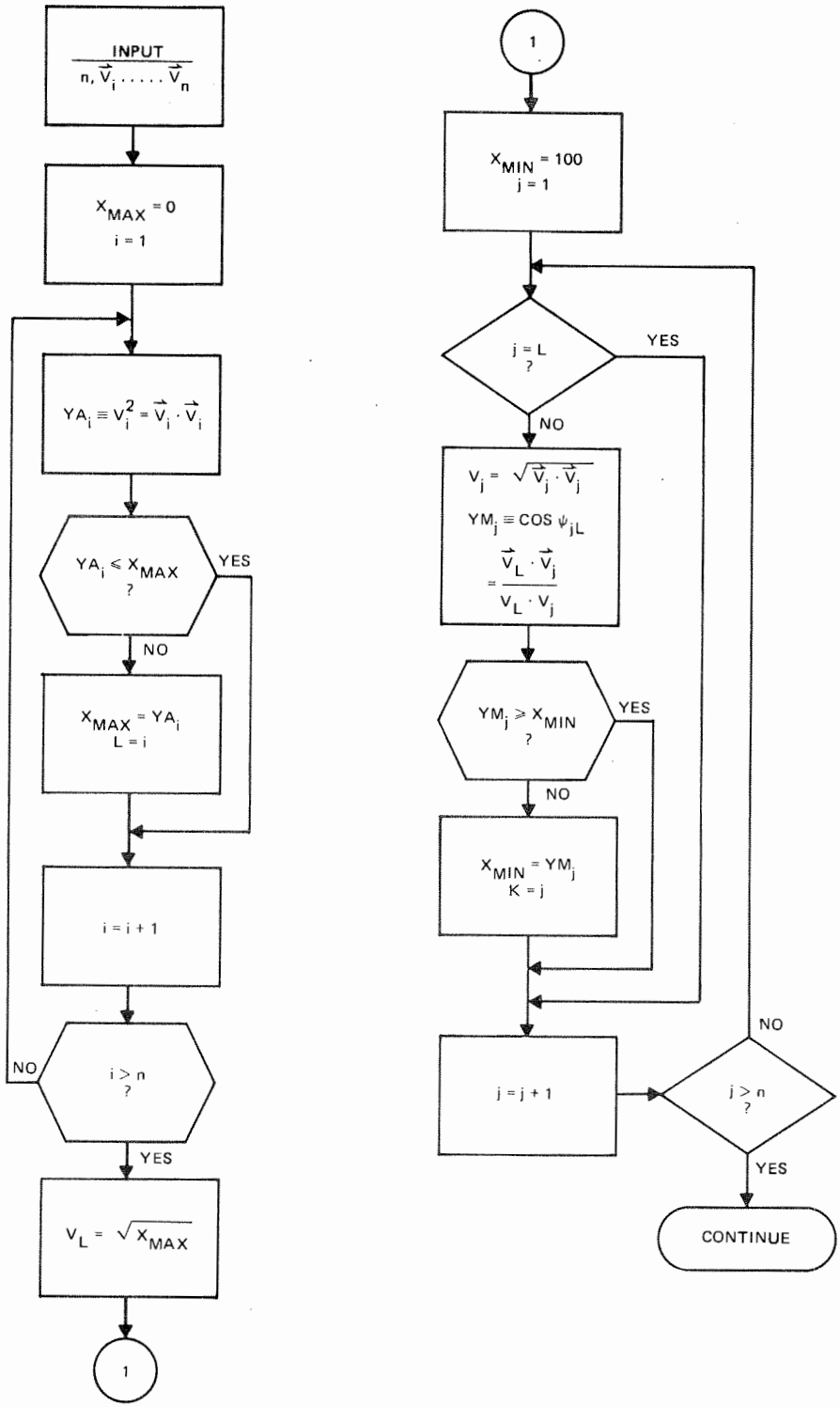


Figure 3-6. Algorithm for Selecting the Two Observations for the Preliminary Attitude Computation

Selection of the $\vec{V}_K$ closest to π radians from $\vec{V}_L$ apparently is an unintended imperfection in DOAOP. It is well known that the two-vector technique for computing attitude is most accurate if the two vectors are $\pi/2$ radians apart. Attitude cannot be computed at all if the two vectors are separated by π radians.

3.5.2 Two-Observation Attitude Determination Algorithm

Having selected $\vec{V}_L$ and $\vec{V}_K$ and having obtained the corresponding observation vectors $\vec{W}_L$ and $\vec{W}_K$, the next step is to compute the preliminary attitude matrix, R_o. The R_o computations which are used in DOAOP are summarized in Figure 3-7. For simplicity, most of the straightforward operations of normalizing nonunit vectors and/or of computing their lengths have been excluded from Figure 3-7. Most of the symbols in Figure 3-7, particularly UP_i, are similar to or identical with those of the coding. The vectors $\hat{U}_i$ in Figure 3-7 are not the same vectors as the $\hat{U}_i$ used in Section 3.3.

This discussion takes the point of view that the algorithm for computing the preliminary attitude matrix, R_o, can be divided logically into steps A, B, C, and D, as shown in Figure 3-7. Step D is regarded as the basic attitude determination algorithm. The function of steps A, B, and C is merely to generate the vectors $\hat{UP}_j$, $\hat{U}_j$; $j = 1, 2, 3$, which are used as inputs to block D. Disregarding the exact nature of the $\hat{UP}_j$, $\hat{U}_j$ vectors which are employed as input to block D, the algorithm in this block is a well-known and commonly used technique for computing attitude from a pair of observation vectors. It apparently was first reported in Reference 4.

The $\hat{UP}_j$ are a new set of observation vectors. They are still resolved on the spacecraft body frame B. Similarly, the $\hat{U}_j$ are a new set of reference vectors; they still are resolved on the GCI frame. Unlike the actual observation and reference vector pairs, $\vec{W}_L$, $\vec{W}_K$; $\vec{V}_L$, $\vec{V}_K$, the three $\hat{UP}_j$ vectors are orthonomal, as are the three $\hat{U}_j$ vectors. That is, $[\hat{UP}_i\ \hat{UP}_2\ \hat{UP}_3]$ is an orthogonal matrix, as is $[\hat{U}_1\ \hat{U}_2\ \hat{U}_3]$. The algorithm used in block D requires

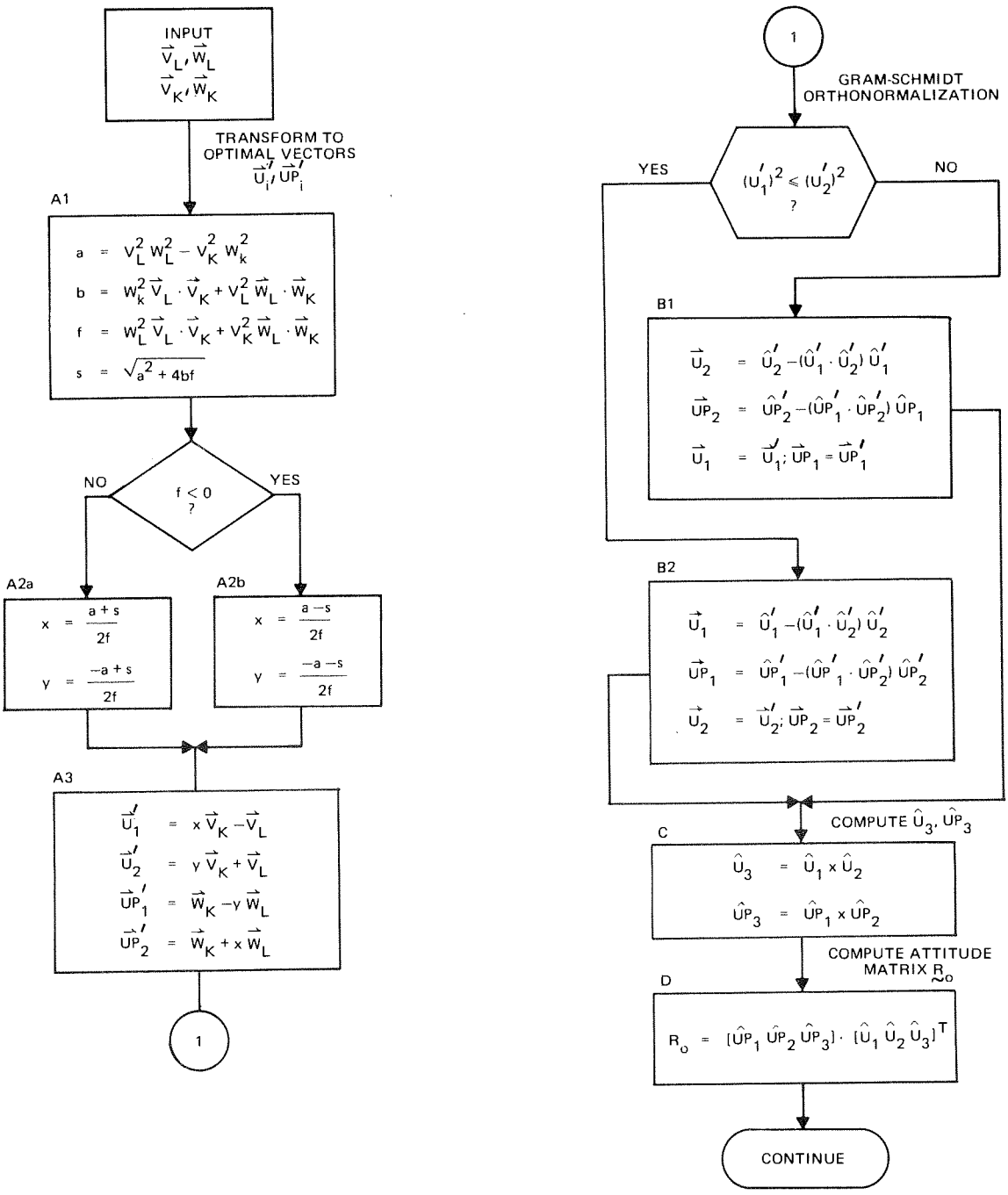


Figure 3-7. Computation of Preliminary Attitude Matrix R_o

matrices which are orthogonal, and this is the reason for transforming to the $\hat{UP}_j$ and $\hat{U}_j$ vectors.

As is shown in block C, $\hat{U}_3$ and $\hat{UP}_3$ are computed by forming the crossproducts of $\hat{U}_1$, $\hat{U}_2$ and of $\hat{UP}_1$, $\hat{UP}_2$, respectively. The vectors $\hat{U}_1$, $\hat{U}_2$, $\hat{UP}_1$, and $\hat{UP}_2$ are obtained by performing the orthonormalizing operations shown in blocks A and B on $\vec{V}_K$, $\vec{V}_L$, $\vec{W}_K$, $\vec{W}_L$.

There is a continuum of ways in which $\vec{V}_K$, $\vec{V}_L$, $\vec{W}_K$, $\vec{W}_L$ could be transformed into orthonormal triads $\hat{U}_j$ and $\hat{UP}_j$. As will be discussed briefly later, the technique used in DOAOP is the optimal way; that is, the overall algorithm shown in Figure 3-7 maximizes the gain function g(R) of Equation (2-4) in the two vector (n = 2) case and hence is a weighted least-squares approach.

The remainder of this subsection will discuss the portion of the algorithm denoted here as step D. The orthonormalizing operations, steps A and B, will be discussed in the following subsection.

As noted above, the $\hat{UP}_j$ are transformed observation vectors which are resolved on frame B. Similarly, the U_j are the corresponding transformed reference vectors, resolved on frame GCI. For any single pair of vectors, $\hat{UP}_j$ and $\hat{U}_j$, there is an equation of the form

$$\underset{3\times1}{\hat{UP}_j} = \underset{3\times3}{R_o} \cdot \underset{3\times1}{\hat{U}_j} + \vec{\epsilon}_j \qquad (3\text{-}6)$$

where R_o is the unknown attitude matrix, and $\vec{\epsilon}_j$ is an unknown error vector which results mainly from inaccuracy in the observation/computation of $\hat{UP}_j$. Since $j = 1, 2, 3$, there are three equations of this form. They can be combined into the following matrix format.

$$[\hat{UP}_1\ \hat{UP}_2\ \hat{UP}_3] = R_o \cdot [\hat{U}_1\ \hat{U}_2\ \hat{U}_3] + [\vec{\epsilon}_1\ \vec{\epsilon}_2\ \vec{\epsilon}_3] \qquad (3\text{-}7)$$

Omitting the error matrix and solving for R_o yields

$$R_o = [\hat{UP}_1 \; \hat{UP}_2 \; \hat{UP}_3] \cdot [\hat{U}_1 \; \hat{U}_2 \; \hat{U}_3]^{-1} \tag{3-8}$$

The matrices on the right-hand side of Equation (3-8) are orthogonal. The equation can thus be simplified to

$$R_o = [\hat{UP}_1 \; \hat{UP}_2 \; \hat{UP}_3] \cdot [\hat{U}_1 \; \hat{U}_2 \; \hat{U}_3]^{T} \tag{3-9}$$

which is the algorithm shown in block D of Figure 3-7.

Analytically, it is possible to use $\vec{W}_L$, $\vec{W}_K$, $\vec{V}_L$, $\vec{V}_K$ directly in the attitude computation without an intermediate generation of the orthonormal vector triads, $\hat{U}_i$, $\hat{UP}_i$. This approach employs

$$R_o = [\vec{W}_K \; \vec{W}_L \; \vec{W}_M] \cdot [\vec{V}_K \; \vec{V}_L \; \vec{V}_M]^{-1} \tag{3-10}$$

which is of the same form as Equation (3-8). The new vectors $\vec{W}_M$ and $\vec{V}_M$ are obtained with the simple block C approach

$$\vec{W}_M = \vec{W}_K \times \vec{W}_L \tag{3-11a}$$

$$\vec{V}_M = \vec{V}_K \times \vec{V}_L \tag{3-11b}$$

or else they are an independent vector pair selected from the remaining n - 2 vector pairs $\{\vec{W}_\alpha, \vec{V}_\alpha\}$. With this approach, each $\vec{W}_j$ must be the same length as the corresponding $\vec{V}_j$.

Techniques which transform to orthonormal vector triads are usually considered to be superior to the approach in Equation (3-10) which works directly with

nonorthonormal triads. One of the disadvantages of the technique in Equation (3-10) is that errors in $\vec{W}_K$, $\vec{W}_L$, $\vec{V}_K$, $\vec{V}_L$ tend to yield an R_o which is not orthogonal. To be more specific, it can be shown that Equation (3-10) cannot yield an orthogonal R_o unless the angle ψ_V between $\vec{V}_L$ and $\vec{V}_K$ is identical with the angle ψ_W between $\vec{W}_L$ and $\vec{W}_K$; errors in determining $\vec{V}_L$, $\vec{V}_K$, $\vec{W}_L$, $\vec{W}_K$, however, can prevent the condition $\psi_W = \psi_V$ from being encountered exactly. Methods which transform to orthonormal triads before computing R_o are certain to generate an R_o which is orthogonal.

3.5.3 Orthogonalization of the Observation and Reference Vectors

In DOAOP, the orthogonalization operation is divided into two steps (Figure 3-7). In step A, $\vec{V}_K$, $\vec{V}_L$, $\vec{W}_K$, and $\vec{W}_L$ are transformed into intermediate vectors $\vec{U}'_1$, $\vec{U}'_2$, $\vec{UP}'_1$, and $\vec{UP}'_2$. In general, these four intermediate vectors all will have nonunit length. However, $\vec{U}'_1$ is perpendicular to $\vec{U}'_2$ and $\vec{UP}'_1$ is perpendicular to $\vec{UP}'_2$. In step B, these intermediate vectors $\vec{U}'_1$, $\vec{U}'_2$, $\vec{UP}'_1$, $\vec{UP}'_2$ are transformed into the final vectors $\hat{U}_1$, $\hat{U}_2$, $\hat{UP}_1$, $\hat{UP}_2$, which are employed to compute R_o in steps C and D. As noted earlier, the four final vectors are of unit length, with $\hat{U}_1$ perpendicular to $\hat{U}_2$, and $\hat{UP}_1$ perpendicular to $\hat{UP}_2$. Step B is merely an implementation of the usual Gram-Schmidt orthonormalization algorithm.

Steps A and B are redundant. That is, it is possible analytically to omit step B; this would require merely normalizing $\vec{U}'_1$, $\vec{U}'_2$, $\vec{UP}'_1$, and $\vec{UP}'_2$--a trivial operation which is performed anyway. Alternatively, step A could be omitted instead; this procedure would involve applying the Gram-Schmidt operation first to $\vec{V}_L$, $\vec{V}_K$ and then, independently, to $\vec{W}_L$, $\vec{W}_K$.

The main orthogonalization step is step A. An analysis summarized in Reference 5, pages 23 through 26, indicated that the algorithm formed by steps A, C, and D constitutes an optimal weighted least-squares solution to the attitude determination problem in the case of two observation vectors. Therefore, in the

least-squares sense, the step A approach apparently is the best of all possible algorithms which could be used to transform $\vec{V}_K$, $\vec{V}_L$, $\vec{W}_K$, $\vec{W}_L$ into an orthonormal observation vector pair $\hat{UP}'_1$, $\hat{UP}'_2$, and an orthonormal reference vector pair $\hat{U}'_1$, $\hat{U}'_2$. The Gram-Schmidt operation, step B, has been included in DOAOP only to trim out any nonorthogonality which may be induced in the step A results by numerical inaccuracies.

3.5.3.1 Gram-Schmidt Orthonormalization Step

The Gram-Schmidt step, step B, will be discussed before step A. Only the transformation from $\vec{U}'_1$, $\vec{U}'_2$ to $\hat{U}_1$, $\hat{U}_2$ in step B will be described specifically. The other transformation in step B, from $\vec{UP}'_1$, $\vec{UP}'_2$ to $\vec{UP}_1$, $\vec{UP}_2$, is mathematically identical with this one.

The first operation which is performed on $\vec{U}'_1$, $\vec{U}'_2$ in step B is to establish their lengths U'_1, U'_2, and their corresponding unit vectors $\hat{U}'_1$, $\hat{U}'_2$. These are near-trivial computations which are not shown in Figure 3-7.

DOAOP next determines which vector, $\vec{U}'_1$ or $\vec{U}'_2$, is the longer. The longer of the two vectors is unaltered. That is, the projection operation which the Gram-Schmidt procedure entails is performed on the shorter vector. For example, if $U'_1 > U'_2$, the program uses

$$\hat{U}_1 = \hat{U}'_1 \tag{3-12}$$

In this case, $\hat{U}'_2$ is projected to yield a new vector, $\vec{U}_2$, which is perpendicular to $\hat{U}'_1$. The geometry is illustrated in Figure 3-8, which shows that

$$\hat{U}'_2 = \vec{U}_2 + (\hat{U}'_1 \cdot \hat{U}'_2)\, \hat{U}_1 \tag{3-13}$$

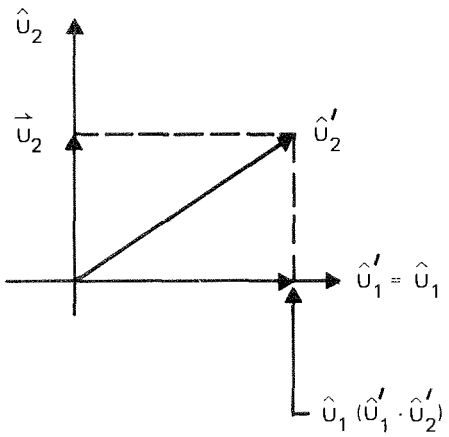


Figure 3-8. Gram-Schmidt Orthonormalization

Thus,

$$\vec{U}_2 = \hat{U}'_2 - (\hat{U}'_1 \cdot \hat{U}'_2)\hat{U}_1 \tag{3-14}$$

which is the computation shown in block B2 of Figure 3-7. The final operation is to normalize $\vec{U}'_2$ by the usual method

$$\hat{U}_2 = \frac{\vec{U}_2}{\sqrt{\vec{U}_2 \cdot \vec{U}_2}} \tag{3-15}$$

If $U'_2 \geq U'_1$, the computations are identical with those noted above except that the 1 and 2 subscripts are reversed.

3.5.3.2 Main Orthogonalization Step

The discussion of the main orthogonalization step, step A, will not delve into its least-squares optimality. Reference 5 presents a study which evidently demonstrates that it does constitute a weighted least-squares solution to the two-observation attitude determination problem when used in conjunction with

steps C and D. The discussion herein shows simply that (1) $\vec{U}'_1$ and $\vec{UP}'_1$ actually are orthogonal to $\vec{U}'_2$ and $\vec{UP}'_2$, respectively, as desired and that (2) the step A orthogonalization operation does not induce any inherent error into the attitude matrix, R_o, to be computed in step D.

The analysis herein will start with the following transformation equations, which are a generalization of those shown in block A3 of Figure 3-7:

$$\vec{U}'_1 = x_1\vec{V}_K + x_2\vec{V}_L \tag{3-16a}$$

$$\vec{U}'_2 = y_1\vec{V}_K + y_2\vec{V}_L \tag{3-16b}$$

$$\vec{UP}'_1 = p_1\vec{W}_K + p_2\vec{W}_L \tag{3-16c}$$

$$\vec{UP}'_2 = q_1\vec{W}_K + q_2\vec{W}_L \tag{3-16d}$$

It is evident that $\vec{U}'_1$ and $\vec{U}'_2$ will be in the plane formed by $\vec{V}_K$ and $\vec{V}_L$; to be called the V-plane. Similarly, $\vec{UP}'_1$ and $\vec{UP}'_2$ will be in the plane formed by $\vec{W}_K$ and $\vec{W}_L$; to be called the W-plane. The eight coefficients x, ..., q_2 are to be chosen to satisfy the constraints of the problem. This must be done in such a way that the step A algorithm is generated exactly.

The constraints which the transformation must satisfy are of two types. First, because $\vec{U}'_1$ is supposed to be perpendicular to $\vec{U}'_2$, and $\vec{UP}'_1$ is supposed to be perpendicular to $\vec{UP}'_2$, it is necessary that

$$\vec{U}'_1 \cdot \vec{U}'_2 = 0 \tag{3-17a}$$

$$\vec{UP}'_1 \cdot \vec{UP}'_2 = 0 \tag{3-17b}$$

Introducing Equations (3-16) into Equation (3-17) yields

$$x_1 y_1 v_K^2 + [x_1 y_2 + y_1 x_2] \vec{V}_K \cdot \vec{V}_L + x_2 y_2 v_L^2 = 0 \qquad (3\text{-}18a)$$

$$p_1 q_1 w_K^2 + [p_1 q_2 + q_1 p_2] \vec{W}_K \cdot \vec{W}_L + p_2 q_2 w_L^2 = 0 \qquad (3\text{-}18b)$$

which are the first constraint equations that must be satisfied by the transformation. The parameters v_L, v_K, w_L, w_K above signify the lengths of $\vec{V}_L$, $\vec{V}_K$, $\vec{W}_L$, and $\vec{W}_K$ respectively.

The present work regards the constraint on Equations (3-16) to be that the transformation must not induce an inherent error into the resulting computed attitude R_o. That is, when the vectors $\vec{V}_L$, $\vec{V}_K$, $\vec{W}_L$, $\vec{W}_K$ are errorfree, the resulting solution for R_o obtained with the Figure 3-7 approach should be analytically equivalent to that obtained by the more straightforward method (Equations (3-10) and (3-11)), which does not employ orthogonalization and which is analytically perfect in the errorfree $\vec{V}_L$---$\vec{W}_K$ case. In this context, $\vec{V}_L$---$\vec{W}_K$ can be considered errorfree if

$$\hat{V}_K \cdot \hat{V}_L = \hat{W}_K \cdot \hat{W}_L \qquad (3\text{-}19)$$

because this enables $\hat{V}_K$ and $\hat{V}_L$ to be rotated onto $\hat{W}_K$ and $\hat{W}_L$, respectively, by the same rotation

$$[\hat{W}_K \; \hat{W}_L] = R_o [\hat{V}_K \; \hat{V}_L] \qquad (3\text{-}20)$$

(For the remainder of this document it is convenient to take an alibi point of view and regard R_o as a rotation. The various vectors can be regarded best

as actual vectors or as vector components on GCI frame.) When Equation (3-19) is satisfied, the transformations in Equations (3-16) should be such that $\hat{U}'_1$ and $\hat{U}'_2$ are rotated onto $\hat{UP}'_1$ and $\hat{UP}'_2$ respectively by this same rotation R_o.

$$[\hat{UP}'_1 \; \hat{UP}'_2] = R_o \cdot [\hat{U}'_1 \; \hat{U}'_2] \tag{3-21}$$

Figure 3-9 is useful for comprehending the geometry of the rotation. R_o can be regarded as being decomposed nonuniquely into two sequential parts: (1) a rotation which rotates the V-plane onto the W-plane, and (2) an azimuth rotation about the normal $\hat{n}_W$ to the W-plane. It is certain that Equations (3-16), when used in conjunction with block D, will still rotate the V-plane onto the W-plane as desired, because $\hat{U}'_1$, $\hat{U}'_2$ and $\hat{UP}'_1$, $\hat{UP}'_2$ lie on the V and W planes, respectively. Use of coefficients $x_1, \ldots, q_2$ which yield vectors $\hat{UP}'_1, \ldots, \hat{U}'_2$ that do not satisfy Equation (3-21), however, would result in the aximuth angle about $\hat{n}_W$ being established incorrectly.

The constraint imposed by the requirement discussed above now will be placed in a usable mathematical form. First, let Equations (3-16) be manipulated into the form:

$$[\hat{U}'_1 \; \hat{U}'_2] = [\hat{V}_K \; \hat{V}_L] \begin{bmatrix} \frac{x_1 v_K}{U_1} & \frac{y_1 v_K}{U_2} \\ \frac{x_2 v_L}{U_1} & \frac{y_2 v_L}{U_2} \end{bmatrix} \tag{3-22a}$$

$$[\hat{UP}'_1 \; \hat{UP}'_2] = [\hat{W}_K \; \hat{W}_L] \begin{bmatrix} \frac{p_1 w_K}{UP_1} & \frac{q_1 w_K}{UP_2} \\ \frac{p_2 w_L}{UP_2} & \frac{q_2 w_L}{UP_2} \end{bmatrix} \tag{3-22b}$$

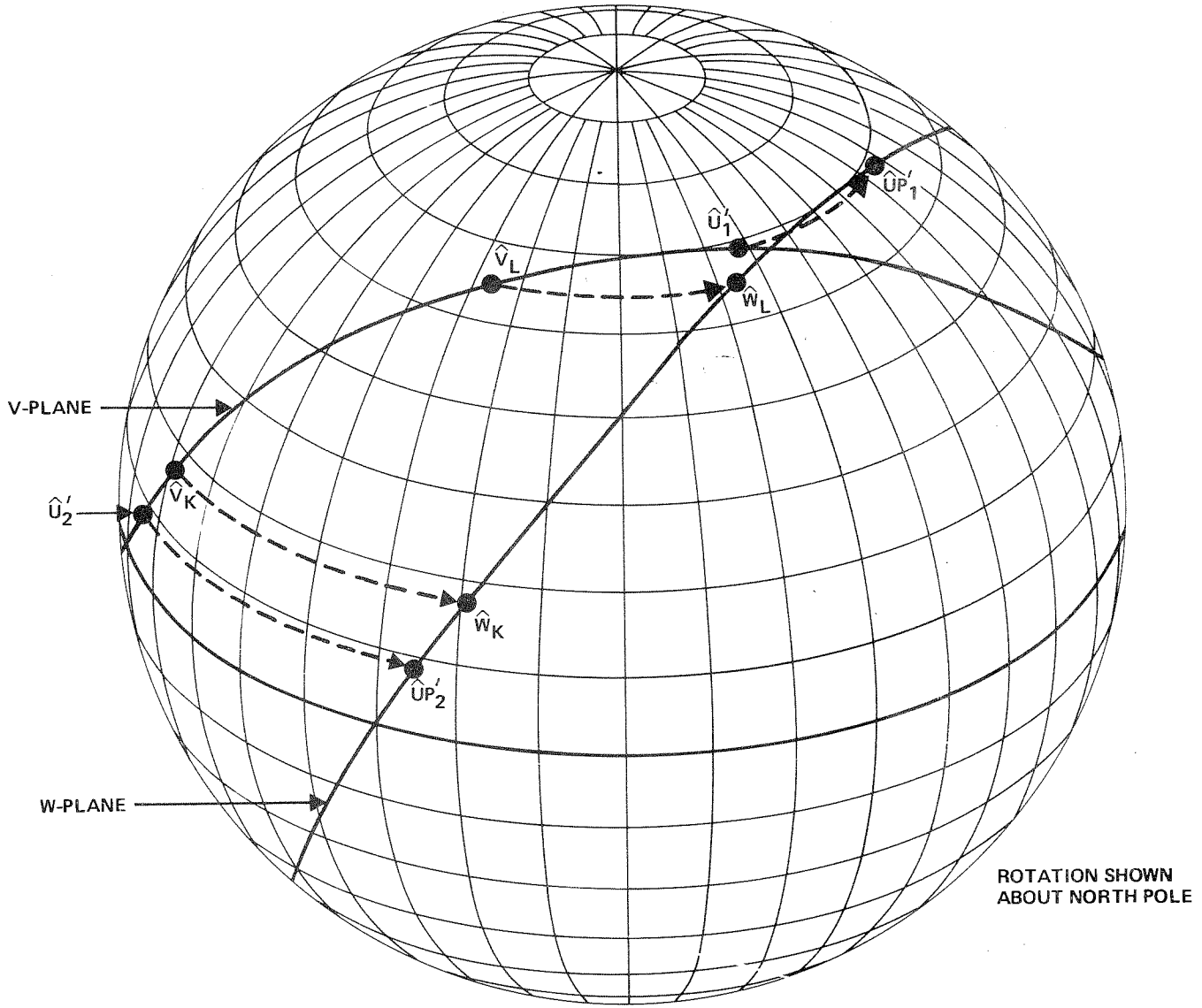


Figure 3-9. Alibi Description of the Two-Vector Attitude Determination Problem

where

$$U_1 = [x_1^2 v_K^2 + 2x_1 x_2 v_K v_L \hat{V}_K \cdot \hat{V}_L + x_2^2 v_L^2]^{.5} \tag{3-23a}$$

$$U_2 = [y_1^2 v_K^2 + 2y_1 y_2 v_K v_L \hat{V}_K \cdot \hat{V}_L + y_2^2 v_L^2]^{.5} \tag{3-23b}$$

$$UP_1 = [p_1^2 w_K^2 + 2p_1 p_2 w_K w_L \hat{W}_K \cdot \hat{W}_L + p_2^2 w_L^2]^{.5} \tag{3-23c}$$

$$UP_2 = [q_1^2 w_K^2 + 2q_1 q_2 w_K w_L \hat{W}_K \cdot \hat{W}_L + q_2^2 w_L^2]^{.5} \tag{3-23d}$$

Now assume the validity of Equation (3-19) and substitute Equation (3-20) into Equation (3-22b), and Equations (3-22a and b) into Equation (3-21). From this result, it can be shown that $x_1, \ldots, q_2$ must be chosen such that the following relations are satisfied

$$\frac{x_1 v_K}{U_1} = \frac{p_1 w_K}{UP_1} \tag{3-24a}$$

$$\frac{x_2 v_L}{U_1} = \frac{p_2 w_L}{UP_1} \tag{3-24b}$$

$$\frac{y_1 v_K}{U_2} = \frac{q_1 w_K}{UP_2} \tag{3-24c}$$

$$\frac{y_2 v_L}{U_2} = \frac{q_2 w_L}{UP_2} \tag{3-24d}$$

Equations (3-24) constitute a second set of constraint equations. It should be noted, however, that this set is not entirely independent of the two constraints specified by Equations (3-18). In fact, geometrical considerations based on Figure 3-9 indicate that only one of the above relations is fully independent. Equations (3-24) are cumbersome due to the complicated expressions for the UP_i and U_i terms. Two simpler equations are obtained by dividing Equation (3-24a) by Equation (3-24b) and Equation (3-24c) by Equation (3-24d).

$$\frac{x_1 v_K}{x_2 v_L} = \frac{p_1 w_K}{p_2 w_L} \tag{3-25a}$$

$$\frac{y_1 v_K}{y_2 v_L} = \frac{q_1 w_K}{q_2 w_L} \tag{3-25b}$$

Also, combining Equation (3-25a) and Equation (3-25b) yields the requirement

$$x_1 y_2 p_2 q_1 = x_2 y_1 p_1 q_2 \tag{3-26}$$

The next step is to separate $x_1, \ldots, q_2$ into (1) a set of independent parameters whose values can be selected arbitrarily and (2) a set of dependent parameters whose values are to be established via constraint equations. Using Figure 3-9 as a guide and referring back to Equations (3-16), the development is as follows:

1. $\vec{U}_1'$ can be placed anywhere on the V-plane. Hence, x_1 and x_2 are arbitrary except that both cannot be zero

2. $\vec{U}_2'$ must be orthogonal to $\vec{U}_1'$. Thus, one of the coefficients, (e.g., y_2) of Equation (3-16b) can be given any arbitrary nonzero value and the other, y_1, should be established by the orthogonality constraint equations.

3. $\vec{UP}'_1$ is constrained to lie at the proper azimuth angle in the W-plane. This means that one of the parameters, e.g., p_1, of Equation (3-16c) must be established through a constraint equation. The value of the other, p_2, cannot be zero, but otherwise is arbitrary except for sign. This sign restriction is necessary to guarantee that $\vec{UP}'_1$ can lie in that half of the W-plane which is required by the rotation.

4. $\vec{UP}'_2$ must be orthogonal to $\vec{UP}'_1$. Hence, the value of one of the coefficients, e.g., q_2, of Equation (3-16d) should be established through the orthogonality constraint equations. The value of the other, q_1, cannot be zero but otherwise is arbitrary except for sign. This sign restriction is necessary to guarantee that $\vec{UP}'_2$ can lie in that half of the W-plane which is required by the rotation.

The values of the arbitrary parameters, x_1, x_2, y_2, p_2, and q_1, now will be selected. Keeping in mind that the present derivation is intended to yield the step A algorithm implemented in DOAOP, the choices are

$$x_1 = q_2 \qquad x_2 = -1 \qquad y_2 = 1 \qquad p_2 = -\sigma_1 y_1 \qquad q_1 = \sigma_2 \tag{3-27}$$

where $\sigma_1 = \pm 1$

$\sigma_2 = \pm 1$

Substituting Equations (3-27) into (3-16) and making some minor notational changes (deleting the subscripts and replacing q by x) yields

$$\vec{U}'_1 = x\vec{V}_K - \vec{V}_L \tag{3-28a}$$

$$\vec{U}'_2 = y\vec{V}_K + \vec{V}_L \tag{3-28b}$$

$$\vec{UP}'_1 = p\vec{W}_K - \sigma_1 y\vec{W}_L \tag{3-28c}$$

$$\vec{UP}'_2 = \sigma_2\vec{W}_K + x\vec{W}_L \tag{3-28d}$$

The orthogonality constraint equations, Equations (3-18), now are

$$xy\, v_K^2 + [x - y]\,\vec{V}_K \cdot \vec{V}_L - v_L^2 = 0 \tag{3-29a}$$

$$p\sigma_2\, w_K^2 + [px - \sigma_1\sigma_2 y]\,\vec{W}_K \cdot \vec{W}_L - \sigma_1 xy\, w_L^2 = 0 \tag{3-29b}$$

and Equations (3-24) through (3-26) are

$$\frac{x\, v_K}{U_1} = \frac{p\, w_K}{UP_1} \tag{3-30a}$$

$$\frac{v_L}{U_1} = \frac{\sigma_1\, y\, w_L}{UP_1} \tag{3-30b}$$

$$\frac{y\, v_K}{U_2} = \frac{\sigma_2\, w_K}{UP_2} \tag{3-30c}$$

$$\frac{v_L}{U_2} = \frac{x\, W_L}{UP_2} \tag{3-30d}$$

$$\frac{x\, v_K}{V_L} = \frac{\sigma_1\, pw_K}{y w_L} \tag{3-31a}$$

$$\frac{y\, v_K}{v_L} = \frac{\sigma_2\, w_K}{x\, w_L} \tag{3-31b}$$

$$p = \sigma_1 \sigma_2 \tag{3-32}$$

The problem now is to utilize the above equations to establish x, y, p, and the signs of σ_1 and σ_2.

Equation (3-32) shows that p must be $+1$ or -1. Equation (3-30d) shows that $x > 0$. Equation (3-30a) shows that x and p must have the same sign. Therefore, $p = +1$.

It now follows from Equation (3-32) that σ_1 and σ_2 must have the same sign. Therefore, the following work will utilize $\sigma = \sigma_1 = \sigma_2 = \pm 1$.

Equations (3-29) now will be manipulated to obtain separate quadratic equations in x and y. After inserting $p = +1$ and $\sigma = \sigma_1 = \sigma_2$, the result is

$$f x^2 - a x - b = 0 \tag{3-33a}$$

$$f y^2 + a y - b = 0 \tag{3-33b}$$

where

$$f = v_K^2\, \vec{W}_K \cdot \vec{W}_L + \sigma w_L^2\, \vec{V}_K \cdot \vec{V}_L \tag{3-34a}$$

$$a = \sigma \left[w_L^2\, v_L^2 - w_K^2\, v_K^2 \right] \tag{3-34b}$$

$$b = v_L^2 \vec{W}_K \cdot \vec{W}_L + \sigma w_K^2 \vec{V}_K \cdot \vec{V}_L \qquad (3\text{-}34c)$$

The unorthodox symbols f , a , b above were taken from the coding. Equations (3-33) can be solved for x and y

$$x = \frac{1}{2f} [a \pm s] \qquad (3\text{-}35a)$$

$$y = \frac{1}{2f} [-a \pm s] \qquad (3\text{-}35b)$$

where

$$s = \sqrt{a^2 + 4bf} \qquad (3\text{-}35c)$$

Equations (3-31a or b) show that

$$xy = \frac{w_K v_L}{w_L v_K} \qquad (3\text{-}36)$$

Because $x > 0$, Equation (3-36) shows that $y > 0$.

Equations (3-35a and b) now will be employed to form xy . If it is assumed that the same sign is used with the s term in both equations, the result is

$$xy = \frac{b}{f} = \frac{v_L^2 \vec{W}_K \cdot \vec{W}_L + \sigma w_K^2 \vec{V}_K \cdot \vec{V}_L}{v_K^2 \vec{W}_K \cdot \vec{W}_L + \sigma w_L^2 \vec{V}_K \cdot \vec{V}_L} \qquad (3\text{-}37)$$

The use of opposite signs with the s term in the two equations does not yield a result that reduces to anything simple, and thus it is evidently not acceptable. Assuming $\hat{V}_K \cdot \hat{V}_L = \hat{W}_K \cdot \hat{W}_L$ enables Equation (3-37) to be manipulated into the form

$$xy = \frac{w_K v_L}{w_L v_K} \frac{[v_L w_L + w_K v_K \sigma]}{[v_L w_L \sigma + v_K w_K]} \tag{3-38}$$

Comparison of Equations (3-36) and (3-38) shows that $\sigma = +1$ is required.

Use of $\sigma = +1$ in Equations (3-34) yields

$$f = v_K^2 \vec{W}_K \cdot \vec{W}_L + w_L^2 \vec{V}_K \cdot \vec{V}_L \tag{3-39a}$$

$$a = w_L^2 v_L^2 - w_K^2 v_K^2 \tag{3-39b}$$

$$b = v_L^2 \vec{W}_K \cdot \vec{W}_L + w_K^2 \vec{V}_K \cdot \vec{V}_L \tag{3-39c}$$

which are the relations implemented in DOAOP (step A1 of Figure 3-7). Also, use of $\sigma = p = +1$ in Equations (3-28) yields

$$\vec{U}_1' = x \vec{V}_K - \vec{V}_L \tag{3-40a}$$

$$\vec{U}_2' = y \vec{V}_K + \vec{V}_L \tag{3-40b}$$

$$\vec{UP}_1' = \vec{W}_K - y\vec{W}_L \tag{3-40c}$$

$$UP_1' = W_K + xW_L \qquad (3\text{-}40d)$$

which are the relations implemented in DOAOP (step A3 of Figure 3-7).

The only remaining problem is to determine the sign to be employed with the s terms in Equations (3-35). It was noted earlier that $x > 0$ and $y > 0$. Equations (3-39a and c) show that f and b can be either positive or negative, depending on the separation angles ψ_V and ψ_W. However, it is certain that $\psi_V \approx \psi_W$ and therefore f and b normally will have the same sign. Thus, $f\,b > 0$. From Equation (3-35c) it can be seen that $s > |a|$. To satisfy the $x > 0$, $y > 0$ requirement, it thus is required that

$$x = \begin{cases} \frac{1}{2f}[a + s] & \text{if } f > 0 \\ \frac{1}{2f}[a - s] & \text{if } f < 0 \end{cases}$$

and (3-41)

$$y = \begin{cases} \frac{1}{2f}[-a + s] & \text{if } f > 0 \\ \frac{1}{2f}[-a - s] & \text{if } f < 0 \end{cases}$$

Equations (3-41) are identical with those shown in step A2 of Figure 3-7.

The study of the step A operation now is completed. The work verifies that the algorithm does produce two orthogonal pairs of vectors without inducing inherent errors in the attitude R_o, to be computed in steps C and D. As noted earlier, no attempt was made here to verify that the resulting R_o will be the weighted least-squares solution as was claimed in Reference 5. Generating a weighted least-squares solution depends on the proper selection of the values of the

parameters x_1 , x_2 , y_2 , p_2 , q_1 . The current study regarded the values of these parameters as arbitrary and selected them for concurrence with the values used in the actual program.

The step A algorithm blows up if $f = 0$. The $f = 0$ condition occurs when the vector pairs already are orthogonal; that is, when $\vec{V}_L \cdot \vec{V}_K = 0$ and $\vec{W}_L \cdot \vec{W}_K = 0$. Singularity at this condition is unfortunate, since it is the optimal condition for attitude determination. If step A is employed in a revised version of DOAOP, it would be desirable to bypass the step whenever this condition is encountered or approached.

APPENDIX A- THE q-METHOD: A NEW LEAST-SQUARES ATTITUDE DETERMINATION ALGORITHM

A.1 INTRODUCTION

The distinctive feature of the attitude determination algorithm which is implemented in DOAOP is the use of the vector-like variable $\vec{Y}$ to specify attitude. $\vec{Y}$ is the attitude state variable used in deriving the algorithm, and it is the quantity which is computed directly by the algorithm. For this reason, Appendix A refers to this technique as the Y-method.

The Y-method is not the only possible algorithm for solving the least-squares attitude determination problem. References 3 and 5, for example, discuss R-methods, presenting algorithms which employ the attitude matrix R in much the same way that the Y-method uses the attitude vector $\vec{Y}$. In addition, a new least-squares algorithm which employs the attitude quaternion $\bar{q}$ has been devised recently by P. Davenport. After a period of simulation testing, it was incorporated into the HEAO attitude support system for operational use. This new method is called the q-method in Appendix A.

The purpose of this appendix is to describe the q-method. Most of the information presented is either based on or taken almost directly from unpublished material provided by P. Davenport

A.2 SYNOPSIS OF THE q-METHOD

This subsection summarizes the main features of the q-method and provides a first look at the implementation of the method in a computer routine. Subsequent subsections present a derivation of the algorithm and discuss its mathematics in detail.

The two main operations which are performed in a computer implementation of the q-method are as follows:

1. Computation of the elements of the symmetric 4 x 4 matrix, which is designated in this appendix as matrix K

2. Computation of the normalized eigenvector $\bar{q}_1$ which pertains to the largest positive eigenvalue, λ_1, of K

$\bar{q}_1$ is the optimal attitude estimate; that is, the attitude quaternion which minimizes the weighted least-squares loss function.

The algorithm for computing K is as follows:

1. $$\underset{3\times 3}{B} = \sum_{i=1}^{n} \vec{W}_i \cdot \vec{V}_i^T$$

2. $$S = B^T + B$$

3(a) $$\underset{\sim}{\vec{Z}} = B^T - B$$

(b) Obtain $\vec{Z}$ from the 3 x 3 skew symmetric matrix $\underset{\sim}{\vec{Z}}$

4. σ = Trace B

5. $$\underset{4\times 4}{K} = \left[\begin{array}{c|c} \underset{3\times 3}{S - I\sigma} & \underset{3\times 1}{\vec{Z}} \\ \hline \underset{1\times 3}{\vec{Z}^T} & \underset{1\times 1}{\sigma} \end{array}\right]$$

Many algorithms exist for solving the eigenvector problem after K is computed. As noted earlier, only the eigenvector, $\bar{q}_1$, which is associated with the largest eigenvalue, λ_1, of K is needed. The other three eigenvectors $\bar{q}_2$,

$\bar{q}_3$, $\bar{q}_4$, are not required, nor are any of the four eigenvalues. The so-called power method or matrix iteration approach is one technique for computing $\bar{q}_1$, and the analysis in this Appendix on the problem of extracting $\bar{q}_1$ from K is devoted entirely to the power method. The power method is a well-known technique for computing the largest eigenvalue and its eigenvector. It is described in References 6 through 9 and in most texts on vibrations.

Because $\bar{q}_1$ is the only quantity which must be established in the present aplication, the power method here can consist merely of sequential passes p through the equation

$$\bar{q}^p = K' \bar{q}^{p-1} \tag{A-1}$$

The operation is continued until a convergence criteria is satisfied or until a specified number of passes has been made without attaining convergence. The significance of the prime on K in Equation (A-1) will be discussed later. The input $\bar{q}$ for any pass is the normalized output of the preceding pass.

Except for one special situation that can be ignored here (namely, when the a priori input $\bar{q}^o$ is orthogonal to $\bar{q}_1$), the technique discussed above will converge to $\bar{q}_1$ if $|\lambda_1| > |\lambda_2|, |\lambda_3|, |\lambda_4|$. (Note that the eigenvalues are numbered such that $\lambda_1 \geq \lambda_2 \geq \lambda_3 \geq \lambda_4$.) It is shown later that the condition $|\lambda_1| > |\lambda_3| \geq |\lambda_2|$ will always be encountered. This later work shows that $|\lambda_4| \geq |\lambda_3| \geq |\lambda_2|$ and, more significantly, that $|\lambda_1| \geq |\lambda_4|$. The condition $|\lambda_1| = |\lambda_4|$ will be encountered if and only if (1) there are only two star observations, or (2) there are more than two observations, but they all must be in a common plane. Convergence will not occur if $|\lambda_1| = |\lambda_4|$. Also, convergence will be very slow if the ratio $|\lambda_4/\lambda_1|$ is close to its upper limit of unity. A modification to the basic power method to handle this potential

convergence problem is therefore necessary. A simple technique for alleviating the convergence problem is to use the following matrix K' rather than K

$$K' = K - sI$$

where s is a suitably chosen negative constant. It can be shown that the term -sI will (1) shift all the eigenvalues of K by s , ($\lambda'_k = \lambda_k - s$) thereby assuring convergence and improving the convergence rate and (2) will not alter the eigenvectors.

For a 4 x 4 matrix, the convergence characteristics of the power method can be shown to be determined by the transient responses of three error modes. The optimal value s* of s is normally considered to be the value which maximizes the speed of response of the slowest of these modes. It is possible to prove that this s* is

$$s^* = \frac{1}{2}[\lambda_2 + \lambda_4] \qquad \text{(A-2)}$$

Equation (A-2) is given tn References 7 and 10. It shifts the eigenvalues to the point where $|\lambda'_4| = |\lambda'_2|$.

Determination of s* presents a difficulty because λ_2 and λ_4 are not known a priori. An exact analytic solution for s* , however, can be generated in the current application. The steps in this algorithm are as follows:

1. Compute $C = B^T B$
2. Compute the coefficients of the cubic polynomial in γ formed by expanding Det $[C - \gamma I] = 0$

3. Obtain the middle eigenvalue, γ_2 , of C with the exact analytic solution of the above polynomial

4. $s^* = -\sqrt{\gamma_2}$

The details of steps 2 and 3 are given in Section A.5.3.3.

Suboptimal algorithms for s can also be employed. One approach uses

$$s = -k \sum_{i=1}^{n} \frac{w_i}{v_i} \vec{V}_i \cdot \vec{V}_i = -k \sum_{i=1}^{n} w_i v_i \qquad \text{(A-3)}$$

where w_i and v_i are the lengths of $\vec{W}_i$ and $\vec{V}_i$, respectively, and k is a selected constant. Two separate analytical studies have indicated the optimum value of k to be 1/3 and 1/2, respectively. In the case in which the vectors are unweighted (of unit lengths), Equation (A-3) reduces to

$$s = -k\, n \qquad \text{(A-4)}$$

The eigenvalue shifting technique noted above is not the only approach for assuring or accelerating convergence. One simple method which can be used to supplement it consists merely of raising K' to a selected power, m , prior to performing the power method iterations. This method will accelerate the convergence rate without altering the final solution $\bar{q}_1$.

A.3 DERIVATION OF THE q-METHOD ALGORITHM

A.3.1 Derivation of the Least-Squares Gain Equation as a Function of $\bar{q}$

The derivation starts with the weighted least-squares gain function which was listed earlier as Equation (2-4)

$$g(R) = \sum_{i=1}^{n} \vec{W}_i^T \cdot R \cdot \vec{V}_i \qquad \text{(A-5)}$$

The following 3 x n matrices will be employed:

$$W = [\vec{W}_1 \text{---} \vec{W}_n] \tag{A-6a}$$

$$V = [\vec{V}_1 \text{---} \vec{V}_n] \tag{A-6b}$$

By forming the n x n matrix $W^T \cdot R \cdot V$ and multiplying it out in terms of the $\vec{W}_i$ and $\vec{V}_i$, it can be demonstrated that Equation (A-5) can be written as

$$g(R) = \text{Trace } [W^T R\ V] \tag{A-7}$$

Using a property of the trace operator which was listed in Section 2.2, Equation (A-7) can be rearranged into

$$g(R) = \text{Trace } [R\ V\ W^T] \tag{A-8}$$

The following 3 x 3 matrix B now will be introduced

$$B = W \cdot V^T = \sum_{i=1}^{n} \vec{W}_i \cdot \vec{V}_i^T \tag{A-9}$$

B (above) is the same matrix B which was used in Section 2.2. Substituting Equation (A-9) into (A-8) yields

$$g(R) = \text{Trace } R\ B^T \tag{A-10}$$

The next step is to introduce the 4 x 1 column vector $\bar{q}$ which consists of the Euler symmetric parameters.

$$\underset{4 \times 1}{\bar{q}} = \begin{Bmatrix} \underset{3 \times 1}{\vec{Q}} \\ \underset{1 \times 1}{q} \end{Bmatrix} = \begin{Bmatrix} \hat{X} \sin\frac{\theta}{2} \\ \cos\frac{\theta}{2} \end{Bmatrix} \tag{A-11}$$

$\bar{q}$ henceforth will be referred to as the attitude quaternion. It specifies the orientation of frame B relative to frame GCI. The terms $\hat{X}$ and θ are the rotation parameters which were defined in Section 2.1.2.

Substitution of Equation (A-11) into Equation (2-6b) enables R to be written as a function of $\vec{Q}$ and q :

$$R = [q^2 - \vec{Q} \cdot \vec{Q}]I + 2\vec{Q} \cdot \vec{Q}^T - 2q\underset{\sim}{\vec{Q}} \tag{A-12}$$

Equation (A-12) is a well-known relation.

Substitute Equation (A-12) into (A-10), and let the resulting gain function be designated $g(\bar{q})$. A few manipulations yield

$$g(\bar{q}) = [q^2 - \vec{Q} \cdot \vec{Q}]\sigma + 2 \text{ Trace } [\vec{Q} \cdot \vec{Q}^T \cdot B^T] - 2q \text{ Trace } [\underset{\sim}{\vec{Q}} \cdot B^T] \tag{A-13}$$

where, for simplicity, the following new variable σ has been introduced

$$\sigma = \text{Trace } B = \sum_{i=1}^{n} \vec{W}_i \cdot \vec{V}_i \tag{A-14}$$

The second portion of Equation (A-14) was presented earlier as Equation (2-23).

The next step is to simplify the second and third terms on the right side of Equation (A-13). Working first on the second term yields

$$2\ \text{Trace}\left[\vec{\tilde{Q}}\cdot\vec{\tilde{Q}}^{T}\cdot B^{T}\right] = 2\ \text{Trace}\left[\vec{\tilde{Q}}^{T}\cdot B^{T}\cdot\vec{\tilde{Q}}\right] \tag{A-15}$$

$$= 2\,\vec{\tilde{Q}}^{T}\cdot B^{T}\cdot\vec{\tilde{Q}} = Q^{T}\cdot S\cdot Q$$

where

$$S = B + B^{T} \tag{A-16}$$

The third term on the right side of Equation (A-13) is more difficult. By expressing Trace $\left[\vec{\tilde{Q}}\cdot B^{T}\right]$ in scalar form, it can be shown that

$$\text{Trace}\left[\vec{\tilde{Q}}\cdot B^{T}\right] = -\vec{Q}\cdot\vec{Z} \tag{A-17}$$

where

$$\vec{Z} = \begin{Bmatrix} b_{23} - b_{32} \\ b_{31} - b_{13} \\ b_{12} - b_{21} \end{Bmatrix} \tag{A-18}$$

The b scalars above are the elements of B . To identify the vector designated $\vec{Z}$, let its skew symmetric form $\vec{\tilde{Z}}$ be written out in full

$$\vec{\tilde{Z}} = \left[\begin{array}{c|c|c} 0 & b_{21} - b_{12} & b_{31} - b_{13} \\ \hline b_{12} - b_{21} & 0 & b_{32} - b_{23} \\ \hline b_{13} - b_{31} & b_{23} - b_{32} & 0 \end{array}\right] \tag{A-19}$$

Equation (A-19) shows that

$$\vec{\underset{\sim}{Z}} = B^T - B \tag{A-20}$$

Comparison of Equation (A-20) with Equation (2-22) shows that the vector $\vec{Z}$ introduced in Equation (A-17) is the same vector $\vec{Z}$ employed in Section 2.2. That is

$$\vec{Z} = \sum_{i=1}^{n} \vec{W}_i \times \vec{V}_i \tag{A-21}$$

Next, substitute Equations (A-15) and (A-17) into (A-13)

$$g(\bar{q}) = [q^2 - \vec{Q} \cdot \vec{Q}]\sigma + \vec{Q}^T \cdot S \cdot \vec{Q} + 2q\, \vec{Z} \cdot \vec{Q} \tag{A-22}$$

A simple rearrangement yields

$$g(\bar{q}) = \left\{ \vec{Q}^T \; q \right\} \left[\begin{array}{c|c} S - I\sigma & \vec{Z} \\ \hline \vec{Z}^T & \sigma \end{array} \right] \left\{ \begin{array}{c} \vec{Q} \\ \hline q \end{array} \right\} \tag{A-23}$$

thus

$$g(\bar{q}) = \bar{q}^T \cdot K \cdot \bar{q} \tag{A-24a}$$

where

$$K_{4\times4} = \left[\begin{array}{c|c} S - I\sigma & \vec{Z} \\ \hline \vec{Z}^T & \sigma \end{array}\right] \qquad \text{(A-24b)}$$

Equations (A-24) specify the least-squares gain criteria as a function of the attitude quaternion $\bar{q}$. The problem of solving this expression to obtain an algorithm for the optimal attitude estimate is considered in the following subsections. Equation (A-24a) is a very convenient form, since it is a quadratic function of $\bar{q}$. The 4×4 matrix K is symmetric and its elements are constants. The set of equations which are needed to establish the elements of K are Equations (A-9), (A-14), (A-16), and (A-20).

Some general information about K and its eigenvalues and eigenvectors can be presented at this point. Let the eigenvalues of K be designated as λ_1, λ_2, λ_3, λ_4. Since $K = K^T$, all λ_k will be real (Reference 11, Theorem 4.6). For convenience, it is assumed henceforth that the λ_k are ordered such that $\lambda_1 \geq \lambda_2 \geq \lambda_3 \geq \lambda_4$.

Since $K = K^T$, a set of four orthonormal eigenvectors (designated here $\bar{q}_1 \ldots, \bar{q}_4$) can be found for K (Reference 11, Theorem 4.7); and K can be diagonalized as follows (Reference 11, Corollary 4.8)

$$\Lambda = Q^T K Q \qquad \text{(A-25a)}$$

where

$$\underset{4 \times 4}{\Lambda} = \text{Diag}\,(\lambda_1, \lambda_2, \lambda_3, \lambda_4) \qquad \text{(A-25b)}$$

and

$$\underset{4 \times 4}{Q} = [\bar{q}_1 \; \bar{q}_2 \; \bar{q}_3 \; \bar{q}_4] \tag{A-25c}$$

Using Equation (A-14), (A-16), and (A-24), it can be shown that

$$\text{Trace } K = 0 \tag{A-26}$$

From this, it follows (Equation (A-25) and Reference 11, Corollary 4.3) that

$$\lambda_1 + \lambda_2 + \lambda_3 + \lambda_4 = 0 \tag{A-27}$$

Because all λ_k cannot be zero, it is certain therefore that $\lambda_1 > 0$ and $\lambda_4 < 0$ and thus that K is indefinite.

A.3.2 Determination of the Least-Squares Attitude Solution

This subsection considers the problem of determining the quaternion $\bar{q}$ which maximizes the least-squares gain function

$$g(\bar{q}) = \bar{q}^T \cdot K \cdot \bar{q} \tag{A-28}$$

where K is a real, symmetric matrix;

$$K = K^T \tag{A-29}$$

and $\bar{q}$ is subject to the constraint

$$\bar{q}^T \cdot \bar{q} = 1 \tag{A-30}$$

The constraint can be handled by the Lagrange multiplier approach. That is, we seek to maximize

$$g'(\bar{q}) = \bar{q}^T \cdot K \cdot \bar{q} - \lambda [\bar{q}^T \cdot \bar{q} - 1] \tag{A-31}$$

where λ is the Lagrange multiplier. Differentiating Equation (A-31) with respect to $\bar{q}$ and setting the result to $\bar{0}$ in the usual manner yields

$$\lambda \bar{q} = K \cdot \bar{q} \tag{A-32}$$

Thus, the attitude quaternions which produce stationary values of Equation (A-28) are the eigenvectors of K, and the Lagrange multipliers are the corresponding eigenvalues. Equation (A-32) thus can be written as

$$\lambda_k \bar{q}_k = K \cdot \bar{q}_k \; ; \; k = 1, 2, 3, 4 \tag{A-33}$$

where the $\bar{q}_k$ are the eigenvectors of K and the λ_k are the eigenvalues. It has been shown thus far that the eigenvectors $\bar{q}_k$ produce stationary values of $g(\bar{q})$. To show which one of the four will produce the largest g let Equation (A-33) be substituted into Equation (A-28) to yield

$$g(\bar{q}_k) = \bar{q}_k^T \cdot [\lambda_k \bar{q}_k] = \lambda_k \tag{A-34}$$

Hence, the four stationary g values, in decreasing order of value, are

$$\begin{aligned} g_1 &= \lambda_1 \\ g_2 &= \lambda_2 \\ g_3 &= \lambda_3 \\ g_4 &= \lambda_4 \end{aligned} \tag{A-35}$$

The eigenvector which maximizes g , and which thus is the optimal attitude estimate, is $\bar{q}_1$, the eigenvector which pertains to the most positive eigenvalue λ_1 .

It is noted that the extremum problem posed by Equations (A-28), (A-29), and (A-30) also is solved in Reference 6, pages 117 through 118, using a more cumbersome approach.

A.4 STUDY OF THE MAGNITUDES OF THE EIGENVALUES OF K

The material in the preceding subsection has demonstrated that the four eigenvalues, λ_k , of K are real and that $\lambda_1 + \lambda_2 + \lambda_3 + \lambda_4 = 0$. By definition, $\lambda_1 \geq \lambda_2 \geq \lambda_3 \geq \lambda_4$, and it follows therefore that $\lambda_1 > 0$ and $\lambda_4 < 0$. Since λ_1 and λ_4 have opposite signs, the condition $\lambda_1 > \lambda_4$ thus does not necessarily signify that $|\lambda_1| > |\lambda_4|$.

This subsection presents more information on the relative values and magnitudes of the λ_k . This information is useful for investigation of the main remaining task in the q-method development: namely, selection of an algorithm for determining the eigenvector $\bar{q}_1$ of K . The question of whether or not the condition $|\lambda_1| > |\lambda_j|$; $j = 2, 3, 4$ will be encountered in all situations is of particular concern. This is because the power method of extracting eigenvalues

and eigenvectors is a prime candidate for computing $\bar{q}_1$, and it is known that the power method will converge to $\bar{q}_1$ only if this condition is satisfied.

It can be shown that the λ_K are related to three scalars $d_1 \geq d_2 \geq d_3 \geq 0$ by the following equations:

$$\begin{aligned} \lambda_1 &= d_1 + d_2 + d_3 \\ \lambda_2 &= d_1 - d_2 - d_3 \\ \lambda_3 &= -d_1 + d_2 - d_3 \\ \lambda_4 &= -d_1 - d_2 + d_3 \end{aligned} \qquad \text{(A-36)}$$

The desired information concerning relative values of the λ_k and $|\lambda_k|$ can be obtained from Equations (A-36).

The three d_j elements above are the positive square roots $(d_j = \sqrt{\gamma_j})$ of the eigenvalues γ_j of the matrix $C = B^T B$. The derivation of Equation (A-36) utilizes material in the R-method of solving the least-squares attitude problem. The derivation is lengthy, and to avoid a major interruption in the present material it has been relegated to Section A.6 of this appendix.

The facts which can be deduced from Equations (A-36) and the relations $d_1 \geq d_2 \geq d_3 \geq 0$ are as follows:

1. $\lambda_1 > 0$
2. λ_2 can be > 0, 0, or < 0
3. $\lambda_3 \leq 0$ with equality encountered only if $d_3 = 0$ and $d_1 = d_2$
4. $\lambda_4 < 0$
5. $\lambda_1 > \lambda_2, \lambda_3, \lambda_4$
6. $\lambda_2 \geq \lambda_3$ with equality encountered only if $d_1 = d_2$

7. $\lambda_2 \geq \lambda_4$ with equality encountered only if $d_1 = d_2 = d_3$

8. $\lambda_3 \geq \lambda_4$ with equality encountered only if $d_2 = d_3$

9. $|\lambda_1| > |\lambda_2|$

10. $|\lambda_1| > |\lambda_3|$

11. $|\lambda_1| \geq |\lambda_4|$ with equality encountered only if $d_3 = 0$

12. $|\lambda_2| \leq |\lambda_3|$ with equality encountered only if (a) $d_3 = 0$ or (b) $d_1 = d_2$

13. $|\lambda_2| \leq |\lambda_4|$ with equality encountered only if $d_1 = d_2 = d_3$

14. $|\lambda_3| \leq |\lambda_4|$ with equality encountered only if $d_2 = d_3$

In determining the above equalities and inequalities, the case $d_2 = d_3 = 0$ was not considered. This condition is encountered only if all $\vec{V}_i$ are colinear. Thus it is a case for which attitude cannot be computed. It yields $\lambda_1 = \lambda_2 = -\lambda_3 = -\lambda_4$.

Item 5 verifies that the least-squares attitude solution is unique. Items 9 to 11, however, are the most significant ones because they verify that the power method will converge to $\bar{q}_1$ unless $d_3 = 0$. The condition $d_3 = 0$ is encountered (1) when there are only two reference vectors, $\vec{V}_i$, or (2) when there are more than two $\vec{V}_i$, but they all lie on a common plane. Because the $\vec{V}_i$ all lie close to a common plane in HEAO-A, it follows that the power method will converge slowly in this application unless special provision to accelerate convergence is made.

A.5 DETERMINATION OF EIGENVECTOR $\bar{q}_1$ OF K BY THE POWER METHOD

A.5.1 Description of the Approach

As noted previously, the normalized eigenvector $\bar{q}_1$ which pertains to the largest eigenvalue λ_1 of K is the optimal attitude estimate. The analysis

to date on the problem of extracting $\bar{q}_1$ has been devoted mainly to the power method, a well-known technique which is described in References 6 through 9.

Because eigenvector $\bar{q}_1$ is the only variable which must be obtained in the present problem, the power method here can consist merely of sequential passes p through the equation

$$\bar{q}^p = K' \quad \bar{q}^{p-1} \tag{A-37}$$

The significance of the prime on K above is discussed later. The input $\bar{q}$ for any pass is the normalized output of the previous pass.

The power method requires a check to determine when convergence has occurred. The recommended technique of testing for convergence utilizes the quaternion properties of the $\bar{q}^p$. With this method, the quaternion $\bar{r} = [\bar{q}^{p-1}]^{-1} \bar{q}^p$ is computed at the end of each pass p using quaternion multiplication. $\bar{r}$ specifies the small rotation which would rotate the attitude frame existing at the start of pass p onto that seen at the end of pass p. The angle, Θ, of this rotation then is computed from the vector portion, $\vec{R}$, of $\bar{r}$ via $\Theta = 2 \arcsin \sqrt{\vec{R} \cdot \vec{R}}$. The approximation $\Theta = 2 \sqrt{\vec{R} \cdot \vec{R}}$ is acceptable here, since Θ is very small. Convergence is deemed to have occurred when Θ is smaller than a selected input constant.

A.5.2 Convergence

A mathematical study of the conditions under which convergence to $\bar{q}_1$ can be expected and of the convergence rate has been relegated to Section A.5.4 to avoid an undesirable interruption in the present material. Only the most pertinent conclusions regarding convergence and convergence rate will be noted at this point.

The eigenvalue of K whose absolute value is greater than that of all other eigenvalues is called the dominant eigenvalue λ_D, and the eigenvalue of the

next largest absolute value is called the second dominant eigenvalue λ_{D2}. Assuming that λ_D is unique (that is, that $|\lambda_D| > |\lambda_{D2}|$), the power method always will converge to the eigenvector $\bar{q}_D$ which pertains to λ_D. (There is one exception here, and it occurs when the a priori input vector $\bar{q}^o$ is orthogonal to $\bar{q}_D$.) This result is proved in References 6 and 7 and in Section A.5.5. Section A.4 showed that in the present application, λ_1 is the dominant eigenvalue and λ_4 is the second dominant eigenvalue. Thus, except for one special situation described below, the conditions $|\lambda_1| > |\lambda_4| > |\lambda_2|$, $|\lambda_3|$ will be encountered and convergence to $\bar{q}_1$ will be attained.

The case where $\lambda_D = -\lambda_{D2}$ presents a problem. This is the special situation which was noted above. Convergence will not occur in this case (page 40 Reference 7). This case is encountered in the present problem with $d_3 = 0$ and thus $\lambda_4 = -\lambda_1$. A technique for surmounting the difficulty is discussed in the next subsection.

It is demonstrated in the references and in Section A.5.4 that the most significant factor affecting the rate of convergence is the ratio $|\lambda_{D2}|/|\lambda_D|$. The smaller this ratio, the faster the convergence rate. The ratio λ_4/λ_1 is of prime concern in the present problem.

A.5.3 Improvement of Convergence by Eigenvalue Shifting

A.5.3.1 Introduction

A simple technique for alleviating convergence problems involves using the following matrix K' in place of K

$$K' = K - sI \tag{A-38}$$

where s is a suitably chosen negative scalar. It will be shown below that the term -sI will (1) shift all the eigenvalues of K by s (that is, $\lambda'_k = \lambda_k - s$),

thereby assuring convergence and improving the convergence rate, and (2) will not alter the eigenvectors $\bar{q}_k$. This approach is recommended in Reference 7.

The above two claims concerning Equation (A-38) are easily proved. The proof starts with the basic eigenvalue equation which was listed earlier as Equation (A-32).

$$\lambda_k \bar{q}_k = K \cdot \bar{q}_k \tag{A-32}$$

where λ_k signifies any of the four eigenvalues and $\bar{q}_k$ is the corresponding eigenvector. Subtracting $s\bar{q}_k$ from both sides produces

$$[\lambda_k - s]\bar{q}_k = [K - sI]\bar{q}_k \tag{A-39}$$

Introducing new notation yields

$$\lambda'_k \bar{q}_k = K'\bar{q}_k \tag{A-40}$$

where

$$K' = K - sI \tag{A-41}$$

$$\lambda'_k = \lambda_k - s \tag{A-42}$$

Equation (A-40) constitutes a new eigenvalue problem. The new eigenvalues λ'_k are related to the eigenvalues, λ_k, of the original matrix K, as indicated in Equation (A-42); and the new eigenvectors are identical with those, $\bar{q}_k$, of K.

A.5.3.2 Criteria for Selecting the Shift s

Figure A-1 is a sample plot of the variation of the eigenvalues λ'_k ; k = 1 to 4 and their absolute ratios $|r'_j| = |\lambda'_j|/|\lambda'_1|$; j = 2 to 4 with the eigenvalue shift parameter s . Only negative s values are shown on the figure, since positive values are not of current interest. The initial (s = 0) eigenvalue values, λ_k , were chosen to be $\lambda_1 = 1.0$, $\lambda_2 = .3$, $\lambda_2 = -.4$, $\lambda_3 = -.9$. These values are not intended necessarily to be typical of those encountered on HEAO-A. The plots were obtained merely by the use of Equation (A-42).

Figure A-1 shows the positive eigenvalues λ'_1 and λ'_2 increasing linearly as s is made more and more negative. The negative eigenvalues λ'_3 and λ'_4 are driven linearly toward zero, they reach zero at $s = \lambda_3$ and $s = \lambda_4$ respectively, and thereafter increase linearly. As a result, $|r'_2|$ increases in size monotonically toward unity while $|r'_3|$ and $|r'_4|$ are driven to zero, after which they also increase monotonically toward unity. Mathematically at least, λ_2 can be negative rather than positive. In such a case, the $|r'_2|$ plot would resemble the $|r'_3|$ and $|r'_4|$ plots on Figure A-1.

Figure A-1 shows that λ'_2 will replace λ'_4 as the second dominant eigenvalue if s is made more negative than the value indicated as s* . By use of Equation (A-42) s* can be shown to be

$$s^* = .5\,[\lambda_2 + \lambda_4] \qquad \text{(A-43)}$$

This result concurs with the results given in References 7 and 10. The parameter s* can also be specified as a function of the elements d_j by substitution of Equation (A-36) into Equation (A-43) to yield

$$s^* = -d_2 \qquad \text{(A-44)}$$

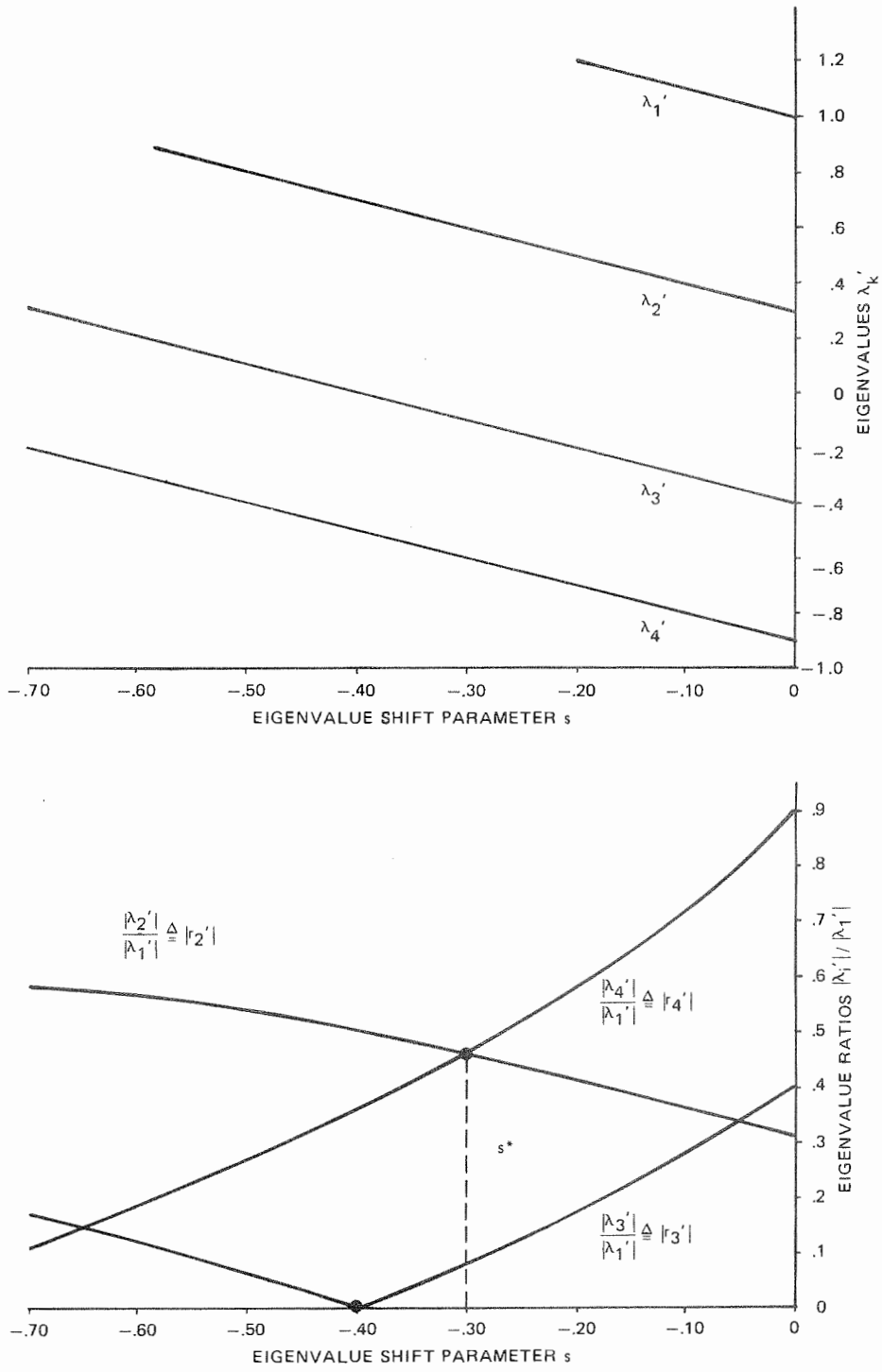


Figure A-1. Typical Variation of Eigenvalues and Their Ratios With s

The parameter s^* generally is considered to be the optimal value of s, since it minimizes the magnitude of the largest $|r'|$ ratio. The question of the optimality of s^* as defined is considered in Section A.5.4, where it is shown that s^* is not necessarily the s which minimizes the number of iterations needed for convergence in any specific problem. The remainder of this appendix, however, will regard s^* as the optimum s.

A.5.3.3 Implementation of the Optimal Shift s^*

Implementation of the optimal shift s^* in a computer routine is usually not possible, since the λ_k are not known a priori. In the present attitude problem however, it is possible to compute s^* exactly. The technique involves use of Equation (A-44).

It will be recalled that in Section A.4 the claim was made that

$$d_j = + \sqrt{\gamma_j} \qquad j = 1, 2, 3 \tag{A-45}$$

where the γ_j are the eigenvalues of the 3×3 matrix C

$$C = B^T B \tag{A-46}$$

and

$$\underset{3\times3}{B} = \sum_{i=1}^{n} \vec{W}_i \cdot \vec{V}_i^T \tag{A-47}$$

It follows that d_2, and thus s^*, can be obtained easily from the eigenvalue γ_2 of C. An algorithm for computing γ_2 can be developed by expanding

$$\det |C - \gamma I| = 0 \tag{A-48}$$

to yield

$$\gamma^3 + m_1\gamma^2 + m_2\gamma + m_3 = 0 \tag{A-49}$$

where

$$m_1 = -\left[c_{11} + c_{22} + c_{33}\right] \tag{A-50a}$$

$$m_2 = \left[c_{11}\,c_{22} + c_{11}\,c_{33} + c_{22}\,c_{33} - c_{12}^2 - c_{13}^2 - c_{23}^2\right] \tag{A-50b}$$

$$m_3 = \left[c_{11}\,c_{23}^2 + c_{22}\,c_{13}^2 + c_{33}\,c_{12}^2 - c_{11}\,c_{22}\,c_{33} - 2c_{12}\,c_{13}\,c_{23}\right] \tag{A-50c}$$

The scalars c above are the elements of C. Equation (A-49) can be solved for γ_1, γ_2, γ_3 using the analytical solution for a cubic polynomial. A suitable form of this solution is expressed by the following set of equations

$$a_1 = m_2 - \frac{1}{3}\,m_1^2 \tag{A-51a}$$

$$b_1 = \frac{2}{27}\,m_1^3 - \frac{m_1\,m_2}{3} + m_3 \tag{A-51b}$$

$$f_1 = \sqrt{\frac{a_1}{-3}} \tag{A-51c}$$

$$\phi = \text{Arc cos}\left(\frac{3b_1}{2\,a_1 f_1}\right) \quad 0 \le \phi \le 2\pi \tag{A-52}$$

$$\gamma_1 = \frac{-m_1}{3} + 2f_1 \cos\frac{\phi}{3} \tag{A-53a}$$

$$\gamma_2 = \frac{-m_1}{3} - f_1\left[\cos\frac{\phi}{3} - \sqrt{3}\sin\frac{\phi}{3}\right] \tag{A-53b}$$

$$\gamma_3 = \frac{-m_1}{3} - f_1\left[\cos\frac{\phi}{3} + \sqrt{3}\sin\frac{\phi}{3}\right] \tag{A-53c}$$

Equations (A-47), (A-46), (A-50) to (A-52), (A-53b), (A-45), and (A-44) taken in that order, constitute the full algorithm for computation of s^* .

The above equations should always yield

$$a_1 < 0 \tag{A-54a}$$

$$\left(\frac{b_1}{2}\right)^2 + \left(\frac{a_1}{3}\right)^3 \leq 0 \tag{A-54b}$$

$$\gamma_1 \geq \gamma_2 \geq \gamma_3 \geq 0 \tag{A-54c}$$

The scalars a_1 and b_1 above are the coefficients of the equation

$$x^3 + a_1 x + b_1 = 0 \tag{A-55a}$$

which is obtained by inserting the transformation

$$\gamma = x - \frac{m_1}{3} \tag{A-55b}$$

into Equation (A-49).

The above technique, in conjunction with Equations (A-36), enables all the eigenvalues $\lambda_1 \ldots \lambda_4$ of K to be established. A potential alternative to the power method in the present problem involves using λ_1, as computed above, in the equation

$$[K - I\lambda_1] \cdot \bar{q}_1 = \bar{0} \tag{A-56}$$

which could be solved algebraically for $\bar{q}_1$.

A.5.3.4 A Suboptimal Technique for Computing the Shift

This section derives a suboptimal algorithm which has also been proposed for computing the eigenvalue shift s. The derivation starts with the matrix B which was employed previously.

$$B = \sum_{i=1}^{n} \vec{W}_i \cdot \vec{V}_i^T \tag{A-57}$$

Now note the following approximate relation

$$\vec{W}_i \approx \frac{w_i}{v_i} R \cdot \vec{V}_i \tag{A-58}$$

where w_i and v_i are the lengths of $\vec{W}_i$ and $\vec{V}_i$, respectively. The above relation omits system errors, particularly those due to the star tracker. Substituting Equation (A-58) into (A-57) produces

$$B \approx \sum \frac{w_i}{v_i} R \cdot \vec{V}_i \cdot \vec{V}_i^T \tag{A-59}$$

Moving R outside the summation yields

$$B \approx R \cdot A \tag{A-60}$$

where A is a newly defined 3 x 3 matrix which is

$$A = \sum \frac{w_i}{v_i} \vec{V}_i \cdot \vec{V}_i^T \tag{A-61}$$

The trace of A is

$$\text{Trace } A = \sum \frac{w_i}{v_i} \vec{V}_i \cdot \vec{V}_i = \sum w_i v_i \tag{A-62a}$$

In the special case where the observation and reference vectors are unweighted, this reduces to

$$\text{Trace } A = n \tag{A-62b}$$

where n is the number of observations. Equations (A-62) will be employed to compute trace A in the final s algorithm.

Let the eigenvalues of A be designed as α_1 , α_2 , α_3 . It is trivial to show from Equation (A-61) that $A = A^T$. Therefore, α_1 , α_2 , α_3 will be real. Let the subscripts be ordered such that $\alpha_1 \geq \alpha_2 \geq \alpha_3$. By forming the product $\bar{x}^T \cdot A \cdot \bar{x}$, where $\bar{x}$ is an arbitrary vector of dimension 3 and A is defined by Equation (A-61), it can be shown that A is at least positive semi-definite. Therefore $\alpha_1 \geq \alpha_2 \geq \alpha_3 \geq 0$.

Recall the matrix C which was introduced earlier

$$C = B^T B \tag{A-63}$$

and recall from Section A.4 that the eigenvalues of C are γ_1 , γ_2 , γ_3 where $\gamma_j = d_j^2$, and $d_1 \geq d_2 \geq d_3 \geq 0$. Substitution of Equation (A-60) into (A-63) yields

$$C \approx A^T R^T R A = A^2 \tag{A-64}$$

Applying a well known result in matrix theory (Reference 12) to Equation (A-64) yields $\gamma_j \approx \alpha_j^2$ and hence $\alpha_j^2 \approx d_j^2$. Since $\alpha_j \geq 0$ and $d_j \geq 0$, it follows that $\alpha_j \approx d_j$. Using the relation

$$\text{Trace } A = \alpha_1 + \alpha_2 + \alpha_3 \tag{A-65}$$

it therefore follows that

$$\text{Trace } A \approx d_1 + d_2 + d_3 = \lambda_1 \tag{A-66}$$

where the second part of Equation (A-66) was obtained from Equation (A-36). Applying a previously mentioned relation, $\lambda_1 + \lambda_2 + \lambda_3 + \lambda_4 = 0$, to Equation (A-66) now yields

$$-.5 \text{ Trace } A \approx .5 \left[\lambda_2 + \lambda_3 + \lambda_4\right] \tag{A-67}$$

The optimum eigenvalue shift was shown earlier to be $s^* = .5 \left[\lambda_2 + \lambda_4\right]$. From Equation (A-67), it is obvious that use of the relation

$$s = -\frac{1}{2} \text{Trace } A \tag{A-68}$$

will provide a very rough approximation to the optimal shift. This equation, however, will usually cause an overshift because of the undesirable λ_3 term in Equation (A-67). The relation

$$s = -\frac{1}{3} \text{Trace } A \tag{A-69}$$

often should be better. Equation (A-69) utilizes the approximation $\lambda_3 = .5 \left[\lambda_2 + \lambda_4\right]$, while Equation (A-68) employs the approximation $\lambda_3 = 0$.

In summary, the operations required by the suboptimal eigenvalue shift method are as follows:

1. Compute Trace A with Equation (A-62a) or, when applicable, (A-62b)
2. Compute the shift s with Equation (A-68) or (A-69)

A.5.4 Analysis of Convergence and Convergence Rate

This subsection considers the convergence and convergence rate of the power method at a more detailed mathematical level than has been done to this point. The work is devoted mainly to the derivation of Equation (A-79), which specifies

the relationship between eigenvalue ratios, initial conditions, and convergence properties. Material from References 6 and 7 has been used in the study.

Recall that λ_k and $\bar{q}_k$, k = 1 to 4, are the eigenvalues and eigenvectors of K. Therefore, they satisfy the equation

$$\lambda_k \bar{q}_k = K \cdot \bar{q}_k \tag{A-70}$$

Through manipulation of Equation (A-70), it is possible to obtain the more general relation

$$\lambda_k^p \bar{q}_k = K^p \cdot \bar{q}_k \tag{A-71}$$

where superscript p is an arbitrary positive integer signifying that λ_k and K are to be raised to the pth power.

In the present work, the $\bar{q}_k$ have unit length by definition. Since K is real and symmetric, it follows that

$$\bar{q}_k^T \cdot \bar{q}_m = \delta_{km} \tag{A-72}$$

where δ_{km} is the Kronecker delta. The multiplication which is used in Equation (A-72) and in later equations of this subsection is the usual Euclidean inner (dot) product.

Since the four $\bar{q}_k$ are linearly independent (orthonormal, to be more precise, as indicated in Equation (A-72)), they can be regarded as a set of basis vectors in a real Euclidean vector space R^4. It thus is certain that any vector $\bar{q}$ in

R^4 can be expressed as a linear combination of the $\bar{q}_k$. Letting $\bar{q}^o$ be the a priori attitude estimate which is employed in an application of the power method, therefore

$$\bar{q}^o = c_1 \bar{q}_1 + c_2 \bar{q}_2 + c_3 \bar{q}_3 + c_4 \bar{q}_4 \tag{A-73}$$

The c's above are constants which depend on $\bar{q}^o$ and the $\bar{q}_k$; the precise relation can be obtained easily from Equations (A-72) and (A-73)

$$c_k = \bar{q}_k^T \cdot \bar{q}^o \tag{A-74}$$

The first pass through the power method equation yields

$$\bar{q}^1 = K \cdot \bar{q}^o \tag{A-75}$$

Ignoring, for simplicity, the normalizing operation which (in the current application of the power method) is performed at the end of each pass, successive applications of Equation (A-75) yield

$$\bar{q}^p = K^p \cdot \bar{q}^o \tag{A-76}$$

where $\bar{q}^p$ is the resultant at the end of the pth pass. (In the present notation, superscript p, when used with scalars and matrices, signifies that these quantities are to be raised to the pth power except where noted otherwise in the text. Superscript p is used with vectors, however, only as a label to designate pass or iteration number.)

Next, premultiply Equation (A-73) by K^p. Application of Equations (A-74) and (A-71) and use of the obviously valid assumption that the dominant eigenvalue, λ_1, is not zero yields

$$\bar{q}^p = \lambda_1^p \left[c_1 \bar{q}_1 + c_2 \left(\frac{\lambda_2}{\lambda_1}\right)^p \bar{q}_2 + c_3 \left(\frac{\lambda_3}{\lambda_1}\right)^p \bar{q}_3 + c_4 \left(\frac{\lambda_4}{\lambda_1}\right)^p \bar{q}_4 \right] \qquad \text{(A-77)}$$

Equation (A-77) is the basic result of the present derivation. It can be made applicable to the shifted-eigenvalue application by merely adding primes to the λ_k.

$$\bar{q}^p = [\lambda_1']^p \left[c_1 \bar{q}_1 + c_2 \left(\frac{\lambda_2'}{\lambda_1'}\right)^p \bar{q}_2 + c_3 \left(\frac{\lambda_3'}{\lambda_1'}\right)^p \bar{q}_3 + c_4 \left(\frac{\lambda_4'}{\lambda_1'}\right)^p \bar{q}_4 \right] \qquad \text{(A-78)}$$

Recall that, by definition, $\lambda_1 \geq \lambda_2 \geq \lambda_3 \geq \lambda_4$. It can be shown that when the star observation set contains at least two noncolinear vectors, the eigenvalue shifting technique ($\lambda_k' = \lambda_k - s$) can always yield $|\lambda_1'| > |\lambda_2'|, |\lambda_3'|, |\lambda_4'|$. In this case, Equation (A-78) shows that

$$\lim_{p \to \infty} \bar{q}^p = [\lambda_1']^p c_1 \bar{q}_1 \qquad \text{(A-79)}$$

if $c_1 \neq 0$. This, in effect, means convergence toward $\bar{q}_1$ since the computed $\bar{q}^p$ always can be normalized to unity. If $c_1 = 0$ the power method obviously will not converge to $\bar{q}_1$. Equation (A-74) shows that the condition $c_1 = 0$ will be encountered only if $\bar{q}^o$ is accidently chosen orthogonal to $\bar{q}_1$.

Assuming $c_1 \neq 0$, Equation (A-78) can be expressed in the form

$$\bar{q}^p = [\lambda'_1]^p \, c_1 \, [\bar{q}_1 + \sum_{j=2}^{4} \epsilon_j^{(p)} \, \bar{q}_j] \tag{A-80a}$$

where

$$\epsilon_j^{(p)} = \left(\frac{c_j}{c_1}\right)\left(\frac{\lambda'_j}{\lambda'_1}\right)^p \tag{A-80b}$$

The superscript p is used in $\epsilon_j^{(p)}$ above only as a label.

The three $\epsilon_j^{(p)} \bar{q}_j$ terms in Equations (A-80) can be regarded as transient response modes. Equation (A-80a) shows that the rate of convergence of the power method is determined entirely by the rates at which the amplitudes $\epsilon_j^{(p)}$ of these three modes attenuate toward zero, and Equation (A-80b) shows that these three attenuation rates are determined solely by the eigenvalue ratios λ'_j/λ'_1 . The λ'_j/λ'_1 ratios, however, are not the only phenomena which determine the number of iterations required to converge (that is, to reach a selected converge criteria) in any specific application. Equations (A-74) and (A-79b) show that the selected initial condition $\bar{q}^o$ affects the necessary number of iterations via the initial mode amplitudes $\epsilon_j^{(o)} = c_j/c_1$.

It will be recalled that the optimal value s* of the eigenvalue shift parameter s shifted the eigenvalues to the point where $\lambda_2 = -\lambda_4$. Equations (A-80) show that this approach causes the mode 2 and mode 4 responses to decay at the same rate. By doing this, it maximizes the speed of response of the slowest of the three transient modes.

In any specific application, however, s* is not necessarily the s which will yield convergence in the minimum number of iterations. This phenomena was

seen in the simulation runs discussed in Section A.7 and was not understood prior to the present analysis. Figure A-2 illustrates the phenomena. As shown in Figure A-2, the initial value, $\epsilon_4^{(o)} \triangleq c_4/c_1$, of Mode 4 is much larger than $\epsilon_2^{(o)} \triangleq c_2/c_1$ of Mode 2. For simplicity, the initial value $\epsilon_3^{(o)}$ of Mode 3 is assumed negligible. As a result, overshifting s beyond s* to produce a very rapid decay of Mode 4 has the potentiality of producing convergence in a lesser number of iterations than would be obtained by use of s*. Iteration N_c in the sketch is, roughly, the crossover iteration number. That is, the $s < s^*$ approach will produce faster convergence than the $s = s^*$ one if it can achieve convergence in fewer than N_c iterations. Otherwise, the $s = s^*$ method is the faster one.

Equations (A-80) show that the transient mode values $\epsilon_j^{(o)}$ will switch signs on each successive iteration if λ_j'/λ_1' is negative. This is the case with Modes 3 and 4 in the present application (no attempt was made to portray the effect in Figure A.2). Thus, large deviations in the computed $\bar{q}^p$ values will be seen on each successive iteration whenever the amplitudes of Modes 3 or 4 are significant compared to that of Mode 2. This phenomenon was encountered frequently in the simulation runs discussed in Section A.7 and was not understood prior to the present analysis. The phenomenon has some potential for disturbing the usual convergence criteria methods which utilize the $\bar{q}$ results obtained on successive iterations.

A.5.5 Improvement of Convergence Rate by Raising K' to a Power

The eigenvalue shifting technique discussed in Section A.5.3 is not the only approach for enhancing the convergence rate of the power method. A simple approach that can be used to supplement the eigenvalue shifting technique consists of raising K' to a power, i.e., m, and performing the power method computations on $[K']^m$ rather than on K'. This approach was recommended for the present High Energy Astronomy Observatory (HEAO) application by B. Gambhir. It is noted briefly and favorably in References 9 and 10.

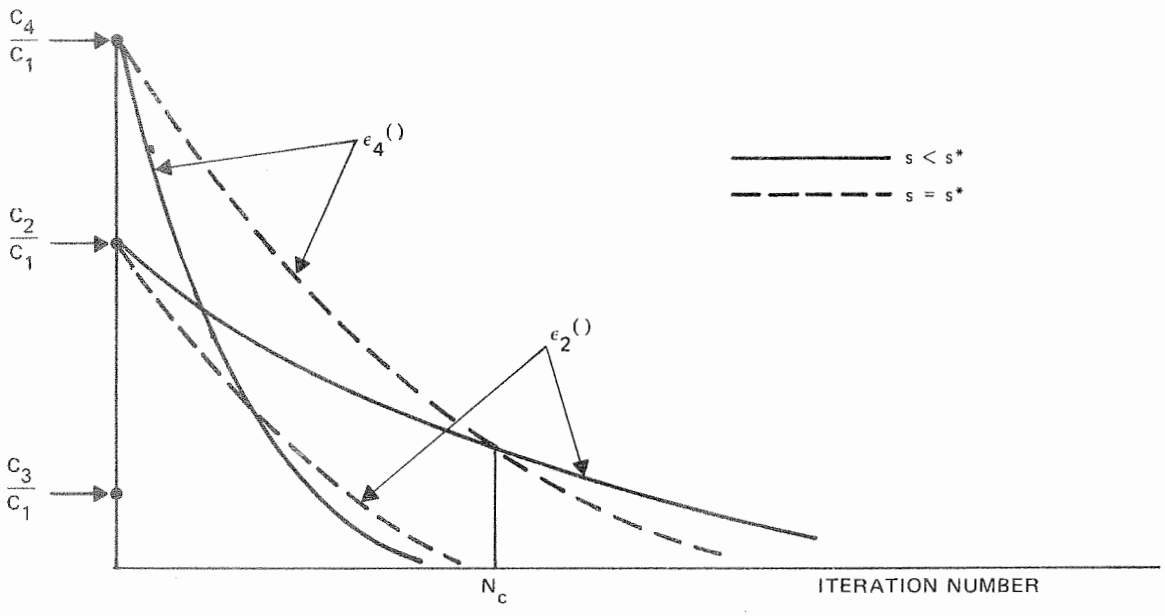


Figure A-2. Sample Transient Response

It is possible to show that when the power method computations are performed on K'^{m}, the dynamic response equations corresponding to Equation (A-80a) for the simple K' case are

$$\bar{q}^{p} = \lambda'^{mp}_{1} c_1 \left[\bar{q}_1 + \sum_{j=2}^{4} \epsilon_j^{(p)} \bar{q}_j\right] \tag{A-81a}$$

where now

$$\epsilon_j^{(p)} = \left(\frac{c_j}{c_1}\right)\left(\frac{\lambda'_j}{\lambda'_1}\right)^{mp} \tag{A-81b}$$

This method's potential for reducing the required number of iterations is evident, since the eigenvalue ratios are raised to the power mp rather than merely to the power p. If applied to an actual application, such as HEAO, some study would be needed to establish a criteria for selecting m.

A.5.6 An Alternate Algorithm for Computing $\bar{q}_1$

The technique discussed in the preceding subsections for extracting the first eigenvector $\bar{q}_1$ from K' treats K' as a general real symmetric matrix and $\bar{q}_1$ as a general real eigenvector. That is, the technique does not make direct use of the facts that $\bar{q}_1$ is a rotation quaternion and that K' can be logically separated into the submatrices indicated in Equation (A-24b). The present subsection summarizes the main features of a method recommended by P. Davenport which does utilize these special features of $\bar{q}_1$ and K'.

As in the DOAOP routine, the approach utilizes an intermediate body coordinate frame B_o obtained from an a priori attitude estimate R_o; the coordinate frame relationships shown earlier in Figure 3-3 are applicable here. In a preliminary operation, the algorithm transforms the matrix B of Equation (A-9)

via $B \leftarrow B \cdot R_o^T$. This operation, in effect, transforms the catalog vectors $\vec{V}_i$ onto Frame B_o resolution. As a result, the quaternion $\bar{q}$ which is established by the basic algorithm defines the orientation of the spacecraft body frame B relative to B_o . The operations which are necessary to transform $\bar{q}$ into the desired attitude matrix, R , of Frame B relative to Frame GCI are straightforward and will not be delineated here.

Since $\bar{q}$ is a rotation quaternion, it is representable in the form shown earlier as Equation (A-11).

$$\bar{q} = \begin{Bmatrix} \vec{Q} \\ q \end{Bmatrix} = \begin{Bmatrix} \hat{X} \sin \frac{\theta}{2} \\ \cos \frac{\theta}{2} \end{Bmatrix} \qquad \text{(A-11)}$$

Assuming that the a priori attitude estimate R_o is reasonably accurate, the rotation angle θ will be small, thus $\cos(\theta/2) \approx 1$.

The algorithm is an iterative one much like the power method. At each iteration p it computes an estimate λ'_p of the dominant eigenvalue using a specialized form of the Rayleigh quotient approximation (Reference 7). The basic Rayleigh equation is

$$\lambda'_p \approx \frac{\bar{q}_{p-1}^T \cdot K' \cdot \bar{q}_{p-1}}{\bar{q}_{p-1}^T \cdot \bar{q}_{p-1}} \qquad \text{(A-82)}$$

The p's above designate pass or iteration number. For convenience, the present subsection presents the p's as subscripts rather than as superscripts, as is done elsewhere in the report. Equation (A-82) computes the true eigenvalue exactly if $\bar{q}_{p-1}$ is the exact eigenvector.

Let the $\bar{q}_{p-1}$ in Equation (A-82) be broken into components $\vec{Q}_{p-1}$ and q_{p-1} as indicated in Equation (A-11) and let K' be broken into the submatrices specified in Equation (A-24b). The result, after multiplying out, is

$$\lambda'_p = \frac{\vec{Q}^T_{p-1} \cdot S' \vec{Q}_{p-1} + 2 q_{p-1} \vec{Z}^T \cdot \vec{Q}^T_{p-1} + \sigma' q^2_{p-1}}{\left[\vec{Q}^T_{p-1} \cdot \vec{Q}_{p-1} + q^2_{p-1}\right]} \tag{A-83}$$

where for simplicity we have used $S' = S - I(\sigma + s)$ and $\sigma' = \sigma - s$

Recall now that $\vec{Q}/q = \vec{Y}$ where $\vec{Y}$ is the Gibbs vector used in DOAOP. Equation (A-83) can be expressed in terms of $\vec{Y}$ by simple manipulation

$$\lambda'_p = \frac{\vec{Y}^T_{p-1} \cdot S' \vec{Y}_{p-1} + 2 \vec{Z}^T \cdot \vec{Y}_{p-1} + \sigma'}{1 + \vec{Y}^T_{p-1} \cdot \vec{Y}_{p-1}} \tag{A-84}$$

Equation (A-84) is the equation actually used to compute λ'_p in the algorithm.

Consider next the basic equation of the power method

$$\lambda'_p \bar{q}_p \approx K' \bar{q}_p \tag{A-85}$$

with exact equality being encountered when λ'_p and $\bar{q}_p$ are an exact eigenvalue and its associated eigenvector. The $\bar{q}_p$ on the right side of Equation (A-85) will be replaced by $\bar{q}_{p-1}$ in order to develop an algorithm which employs iteration. Separating $\bar{q}_{p-1}$ and K' into their components as was done before and multiplying out yields

$$\vec{Q}_p = \frac{1}{\lambda'_p} \left[S \cdot \vec{Q}_{p-1} + \vec{Z} q_{p-1}\right] \tag{A-86a}$$

$$q_p = \frac{1}{\lambda'_p} \left[\vec{Z}^T \cdot \vec{Q}_{p-1} + \sigma' q_{p-1} \right] \tag{A-86b}$$

Manipulation of Equation (A-86a) produces

$$\vec{Y}_p = \frac{1}{\lambda'_p} \left(\frac{q_{p-1}}{q_p} \right) \left[S' \cdot \vec{Y}_{p-1} + \vec{Z} \right] \tag{A-87}$$

As noted earlier, $\cos(\theta/2) \approx 1$ because of the transformation to Frame B_o resolution and $q_{p-1} \approx q_p \approx 1$. Thus, q_{p-1}/q_p is very close to unity and Equation (A-87) can be approximated by

$$\vec{Y}_p = \frac{1}{\lambda'_p} \left[S' \cdot \vec{Y}_{p-1} + \vec{Z} \right] \tag{A-88}$$

which is the equation actually employed in the algorithm.

In summary, the algorithm consists basically of Equations (A-84) and (A-88). These are to be employed iteratively, as in the usual power method.

The algorithm provided by Equations (A-84) and (A-88) now will be compared with the simplified DOAOP algorithm which was presented earlier in Section 2.2 and was summarized as Equation (2-26). In order to make the comparison, Equation (A-84) is first substituted into Equation (A-88). The definitions of S' and σ' are then introduced

$$S' = B + B^T - I \left[\sigma + s \right] \tag{A-89a}$$

$$\sigma' = \sigma - s \tag{A-89b}$$

where

$$\sigma = \text{Trace } B \tag{A-89c}$$

and some minor manipulations are made to yield

$$\vec{Y}_p = \frac{.5\,[1 + \vec{Y}_{p-1} \cdot \vec{Y}_{p-1}]\,[\{B + B^T - I\,(\sigma + s)\} \cdot \vec{Y}_{p-1} + \vec{Z}]}{\vec{Y}_{p-1} \cdot \vec{Z} + \vec{Y}_{p-1}^T \cdot [B - .5\,I\,(\sigma + s)] \cdot \vec{Y}_{p-1} + .5\,[\sigma - s]} \tag{A-90}$$

The similarity between Equation (2-26) and Equation (A-90) is immediately evident. In fact, if the eigenvalue shift parameter s is given a value of $-\sigma$, Equation (A-90) reduces to Equation (2-26) exactly. Therefore, the algorithm (Equations (A-84) and (A-88)) discussed in this subsection can be regarded as a generalization of the algorithm of Section 2.2 (and of the algorithm used in DOAOP) to include a variable value of s. Conversely, the DOAOP algorithm can be regarded as a special $s = -\sigma$ implementation of Equations (A-84) and (A-88).

A.6 PERTINENT MATERIAL ON THE R-METHOD

The main purpose of this section is to derive Equations (A-36), upon which the preceding study of the power technique convergence in the q-method application depended heavily. The only way this derivation can be performed involves going through the main steps in the development of the R-method of solving the least-squares attitude problem. Material on the R-method was presented previously in References 3 and 5.

The development can start with Equation (A-10)

$$g(R) = \text{Trace } R\,B^T = \text{Trace } B^T R \tag{A-10}$$

The problem is to determine (directly) the matrix R which maximizes g . Since R is an attitude matrix, it is subject to the constraints

$$R^T R = I \tag{A-91}$$

$$\text{Det } R = +1 \tag{A-92}$$

If Equation (A-91) is satisfied, it is certain that Det R = ±1 . Thus, the effect of the constraint specified by Equation (A-92) would be to eliminate some solutions for R which otherwise would be acceptable.

The constraint imposed by Equation (A-91) can be handled by the usual Lagrange multiplier approach. In the present problem, the Lagrange multipliers can be arranged in a symmetric 3 × 3 matrix H and incorporated into the least-squares gain function, Equation (A-6), as follows

$$g'(R) = \text{Trace } B^T R - .5 \text{ Trace } [H\{R^T R - I\}] \tag{A-93}$$

Equation (A-93) is different from the usual static optimization problem, since a matrix R, rather than a vector, is to be optimized. However, the present problem can be handled by an analogous technique; a gradient matrix method. The necessary gradient matrix expressions can be obtained from Reference 13 or can be developed using index notation. The expressions are

$$\frac{\partial}{\partial R} \text{Trace } RB^T = B \tag{A-94}$$

$$\frac{\partial}{\partial R} \text{Trace } [H \{R^T R - 1\}] = 2 RH \tag{A-95}$$

When developing (A-95), use was made of the previously mentioned restriction $H = H^T$.

Differentiating Equation (A-93) with respect to R, employing Equations (A-94) and (A-95), and setting the result to zero yields

$$B = RH \tag{A-96}$$

The problem posed by Equation (A-96) is to factor B into the product of (1) an orthogonal matrix R of Determinant + 1 and (2) a symmetric matrix H. The resulting matrices R will yield stationary values of $g(R)$ in Equation (A-10).

The gain function g can be expressed as a function of H. The derivation consists merely of premultiplying Equations (A-96) by R^T and transposing to obtain

$$H = B^T R \tag{A-97}$$

Comparison of Equations (A-10) and (A-97) shows that

$$g = \text{Trace } H \tag{A-98}$$

Equation (A-98) will be employed later.

The problem which will be pursued now is to develop a technique for establishing H. For this work, it will be convenient to introduce the following new 3×3 symmetric matrix C.

$$C = B^T B = VW^T WV^T = \sum_{i=1}^{n} \sum_{j=1}^{n} [\vec{W}_i^T \cdot \vec{W}_j] \vec{V}_i \cdot \vec{V}_j^T \tag{A-99}$$

The second and third portions of Equation (A-99) follow from the definition of B provided by Equation (A-9). Substitution of Equation (A-96) into (A-99) and utilizing $H = H^T$ yields

$$C = H^2 \tag{A-100}$$

The eigenvalues of C are all real, since $C = C^T$ (Reference 11, Theorem 4.6). Let the eigenvalues of C be designated as γ_j and let them be numbered such that $\gamma_1 \geq \gamma_2 \geq \gamma_3$. The matrix C will be nonnegative definite, since it is factorable into $B^T B$ (Reference 11, Theorem 4.18). Therefore $\gamma_1 \geq \gamma_2 \geq \gamma_3 \geq 0$ (Reference 11, Definition 4.9). Rank C will be equal to Rank B (Reference 11, Theorem 3.15 and Problem 3.12). Therefore, the number of zero eigenvalues of C is (3 - Rank B).

Since $C = C^T$, orthogonal matrices U exist which will diagonalize C with a congruence transformation, even if the γ_j are not all distinct (Reference 11, Corollary 4.8). The columns $\hat{u}_j$ of U are the normalized eigenvectors of C, and the nonzero elements of the resulting diagonal matrix Γ are the eigenvalues γ_j. Thus

$$\Gamma = \begin{bmatrix} \gamma_1 & 0 & 0 \\ 0 & \gamma_2 & 0 \\ 0 & 0 & \gamma_3 \end{bmatrix} = U^T \cdot C \cdot U \tag{A-101a}$$

where

$$U = [\hat{u}_1 \; \hat{u}_2 \; \hat{u}_3] \tag{A-101b}$$

and

$$U^T \cdot U = I \tag{A-101c}$$

Equation (A-101a) can be inverted to yield

$$C = U \cdot \Gamma \cdot U^T \tag{A-102}$$

The following new diagonal matrix D now will be defined

$$D = \begin{bmatrix} \mu_1 d_1 & 0 & 0 \\ 0 & \mu_2 d_2 & 0 \\ 0 & 0 & \mu_3 d_3 \end{bmatrix} \tag{A-103}$$

where

$$d_j = +\sqrt{\gamma_j} \tag{A-104a}$$

and

$$\mu_j = \pm 1 \tag{A-104b}$$

The motivation for introducing D should soon become apparent. The d_j are nonnegative by definition. Since $\gamma_1 \geq \gamma_2 \geq \gamma_3 \geq 0$, it is obvious that $d_1 \geq d_2 \geq d_3 \geq 0$.

There are eight possible triads $\{\mu_1\ \mu_2\ \mu_3\}$. Thus, when $d_3 \neq 0$, there are eight possible D matrices. Let the triads be numbered as follows:

$$\{\mu_1\mu_2\mu_3\}_1 = \{1,\ 1,\ 1\} \qquad \{\mu_1\mu_2\mu_3\}_5 = \{1,\ 1,\ -1\}$$

$$\{\mu_1\mu_2\mu_3\}_2 = \{1,\ -1,\ -1\} \qquad \{\mu_1\mu_2\mu_3\}_6 = \{1,\ -1,\ 1\}$$

$$\{\mu_1\mu_2\mu_3\}_3 = \{-1,\ 1,\ -1\} \qquad \{\mu_1\mu_2\mu_3\}_7 = \{-1,\ 1,\ 1\}$$

$$\{\mu_1\mu_2\mu_3\}_4 = \{-1,\ -1,\ 1\} \qquad \{\mu_1\mu_2\mu_3\}_8 = \{-1,\ -1,\ -1\}$$

Comparison of Equations (A-101a) with (A-103) and (A-104) shows that

$$D^2 = \Gamma \tag{A-106}$$

for all D matrices. Using Equations (A-100), (A-102), and (A-106), it is trivial to show that

$$H^2 = U \cdot D^2 \cdot U^T \tag{A-107}$$

Equation (A-107) can be solved for H. This solution must satisfy the supplementary requirement $H = H^T$. The result is

$$H = U \cdot D \cdot U^T \tag{A-108}$$

The validity of Equation (A-108) can be verified by multiplying H by itself and employing $U^T U = I$ to reproduce Equation (A-107).

Equation (A-108) establishes H as a function of the eigenvalues γ_j and eigenvectors $\hat{v}_j$ of $C = B^T B$. However, it yields up to eight possible H matrices

corresponding to the eight possible $\{\mu_j\}$ sets of Equation (A-105). The problems now are to establish (1) which ones can be eliminated because they yield attitude matrices R with determinant of -1 and (2) which of the remaining ones yields the largest value of the least-squares gain function g .

Taking the determinant of Equation (A-108), employing (A-103), and utilizing the relation Det U = ±1 yields

$$\text{Det } H = \text{Det } D = d_1 d_2 d_3 \mu_1 \mu_2 \mu_3 \tag{A-109}$$

Taking the determinant of Equation (A-97) and inserting (A-109) produces

$$d_1 d_2 d_3 \mu_1 \mu_2 \mu_3 = \text{Det } B \text{ Det } R \tag{A-110}$$

For Det R = +1 , it therefore is necessary that

$$d_1 d_2 d_3 \mu_1 \mu_2 \mu_3 = \text{Det } B \tag{A-111}$$

A supplementary analysis which will not be delineated here indicated that the condition Det B < 0 will never be encountered. (The analysis used the A matrix of Section A.5.3.4 and, in particular, Equations (A-60) and (A-61). Thus, Det B ≥ 0 . It is not difficult to show that all d_j are nonzero when Det B ≠ 0 . Therefore, Equation (A-111) demonstrates that for the usual case of Det B > 0 , the condition Det R = +1 necessitates that (1) all μ_j be positive or (2) two μ_j be negative and the remaining one be positive. This requirement thus eliminates the $\{\mu\}$ triads numbered 5 through 8 in Equations (A-106). Triads 1 to 4 yield Det R = +1 and thus are acceptable.

The development in the paragraph above is not applicable when Det B = 0 . The previously mentioned study which employed the A matrix of Section A.5.3.4

indicated that the condition Det B = 0 will be encountered only if (1) all reference vectors $\vec{V}_i$ are colinear (Rank B = 1) or (2) all $\vec{V}_i$ lie in a common plane (Rank B = 2). It can be shown that the first of these cases implies $d_1 > d_2 = d_3 = 0$. This case is of no interest here, since it is known that attitude cannot be computed uniquely in this condition. The second case, however, is of interest, since it includes the common condition where only two observation vectors are available; it is known that attitude can be established uniquely in this situation. It is possible to verify that the condition Rank B = 2 implies $d_1 \geq d_2 > d_3 = 0$. When $d_3 = 0$, the sign of μ_3 is meaningless. Inspection of Equations (A-105) should show that when $d_3 = 0$, the only four distinct $\{\mu_j\}$ sets are the ones numbered 1 through 4. A supplementary study, which also will not be detailed here, verified that it is possible to obtain a separate solution for R, with determinant of +1, for each of these four $\{\mu_j\}$ sets in the $d_3 = 0$ case.

To determine which one of the four "acceptable" H matrices (those which pertain to $\{\mu_j\}$ sets 1 to 4) yields the largest value of the least-squares gain function g, take the trace of Equation (A-108) and utilize (A-103)

$$\begin{aligned} \text{Trace } H = \text{Trace } [UDU^T] = \text{Trace } [DU^TU] = \text{Trace } D \\ = \mu_1 d_1 + \mu_2 d_2 + \mu_3 d_3 \end{aligned} \tag{A-112}$$

Comparison of Equations (A-98) and (A-112) shows that

$$g = \mu_1 d_1 + \mu_2 d_2 + \mu_3 d_3 \tag{A-113}$$

Since there are four acceptable $\{\mu_j\}$ triads (Triads 1 to 4 of Equations (A-105)), Equation (A-113) shows that there are four stationary gain values g_k. Equations (A-105) and (A-113) show that these are

$$g_1 = d_1 + d_2 + d_3 \tag{A-114a}$$

$$g_2 = d_1 - d_2 - d_3 \tag{A-114b}$$

$$g_3 = -d_1 + d_2 - d_3 \tag{A-114c}$$

$$g_4 = -d_1 - d_2 + d_3 \tag{A-114d}$$

Since $d_1 \geq d_2 \geq d_3 \geq 0$, Equations (A-114) show that $g_1 \geq g_2 \geq g_3 \geq g_4$. Therefore, the condition $\mu_1 = \mu_2 = \mu_3 = +1$ is the one which produces the largest value of the least-squares gain function, and thus is the condition which should be used in Equation (A-103) to establish D.

The derivation of the algorithm for computing R for the case $d_3 \neq 0$ now can be completed. (The $d_3 = 0$ case requires special procedures which are not discussed here.) Suitable equations for R can be obtained by either substituting Equation (A-108) into (A-97) and solving for R

$$R = [B^T]^{-1} UDU^T \tag{A-115}$$

or, optionally, by substituting (A-108) into (A-96) and solving for R

$$R = B U D^{-1} U^T \tag{A-116}$$

To summarize, with the R-method the attitude matrix can be computed using either Equations (A-115) or (A-116). The matrix B in these two equations is specified by Equation (A-9). The matrix U is made up, columnwise, of the orthonormal eigenvectors $\hat{u}_j$ of $B^T B$. The non-zero elements d_j of the diagonal matrix D are the positive square roots of the eigenvalues γ_j of $B^T B$. Use of negative signs with two of the d_j elements when forming D yields an attitude matrix with determinant of +1 which stationizes, but does not maximize, the least-squares gain function g; the values of g for these three cases are indicated by Equations (A-114 b through d).

The four g_i equations, Equations (A-114), are of more immediate interest than is the overall R-method of attitude computation. These equations indicate the four stationary values g_k of the least-squares gain function g. They are produced by four attitude matrices R_k. It was demonstrated in Section A.3 that g also is stationized by the four normalized eigenvectors $\bar{q}_k$ of the matrix K. The pertinent equations were Equations (A-35). The $\bar{q}_k$ are the attitude quaternions, and the stationary values, g_k, of g are the corresponding eigenvalues, λ_k, of K. The stationary values g_k produced by the R_k are identical to the corresponding stationary values g_k produced by the $\bar{q}_k$ because the R_k and the corresponding $\bar{q}_k$ are merely different parameterizations of the same attitude. From Equations (A-35) and (A-114), thus

$$
\begin{aligned}
g_1 &= \lambda_1 = d_1 + d_2 + d_3 \\
g_2 &= \lambda_2 = d_1 - d_2 - d_3 \\
g_3 &= \lambda_3 = -d_1 + d_2 - d_3 \\
g_4 &= \lambda_4 = -d_1 - d_2 + d_3
\end{aligned}
\qquad \text{(A-117)}
$$

These are the relations listed earlier as Equation (A-36) and employed in the subsequent studies of the feasibility of determining the dominant eigenvector of K by use of the power method.

A.7 SIMULATION TESTS OF THE q-METHOD--INTRODUCTION

A computer routine (SNAPLS) of the q-method was coded and several series of test runs were made. The purposes of the runs were to verify the basic validity of the q-method of attitude determination, to evaluate its performance in the HEAO-A application, and to determine whether or not it should be included as part of HEAO-A's operational attitude support system.

In the first version of SNAPLS, the attitude eigenvector $\bar{q}_1$ of K' was computed by the power method approach discussed in Section A.5. The eigenvalue shift ($\lambda' = \lambda - s$) technique was employed to enhance the convergence rate. The shift parameter s was calculated via the suboptimal technique of subsection A.5.3.4. The so-called K'^m method, discussed in subsection A.5.5, in which convergence rate is further enhanced by raising K' to a power m before performing the matrix iterations was not used. Only the direct method of solving the eigenvector problem was implemented. That is, the alternative technique described in subsection A.5.6, in which $\vec{Y}$ is computed rather than $\bar{q}_1$ was not employed.

The test runs used simulated star observations similar to those which will be obtained in the spinning mode of HEAO-A. The simulated observations all lay close to a common plane, the spacecraft's spin plane. Figure A-3 portrays the geometry. For convenience, the spacecraft spin axis, $\hat{z}_B$, was chosen to coincide with the celestial north pole in the runs. With this $\hat{z}_B$ attitude, the projected location of any observed star, (e.g., star i), on the spacecraft spin plane can be specified by the right ascension angle, α_i, of the star, and the location of the star above or below the spin plane is the declination δ_i.

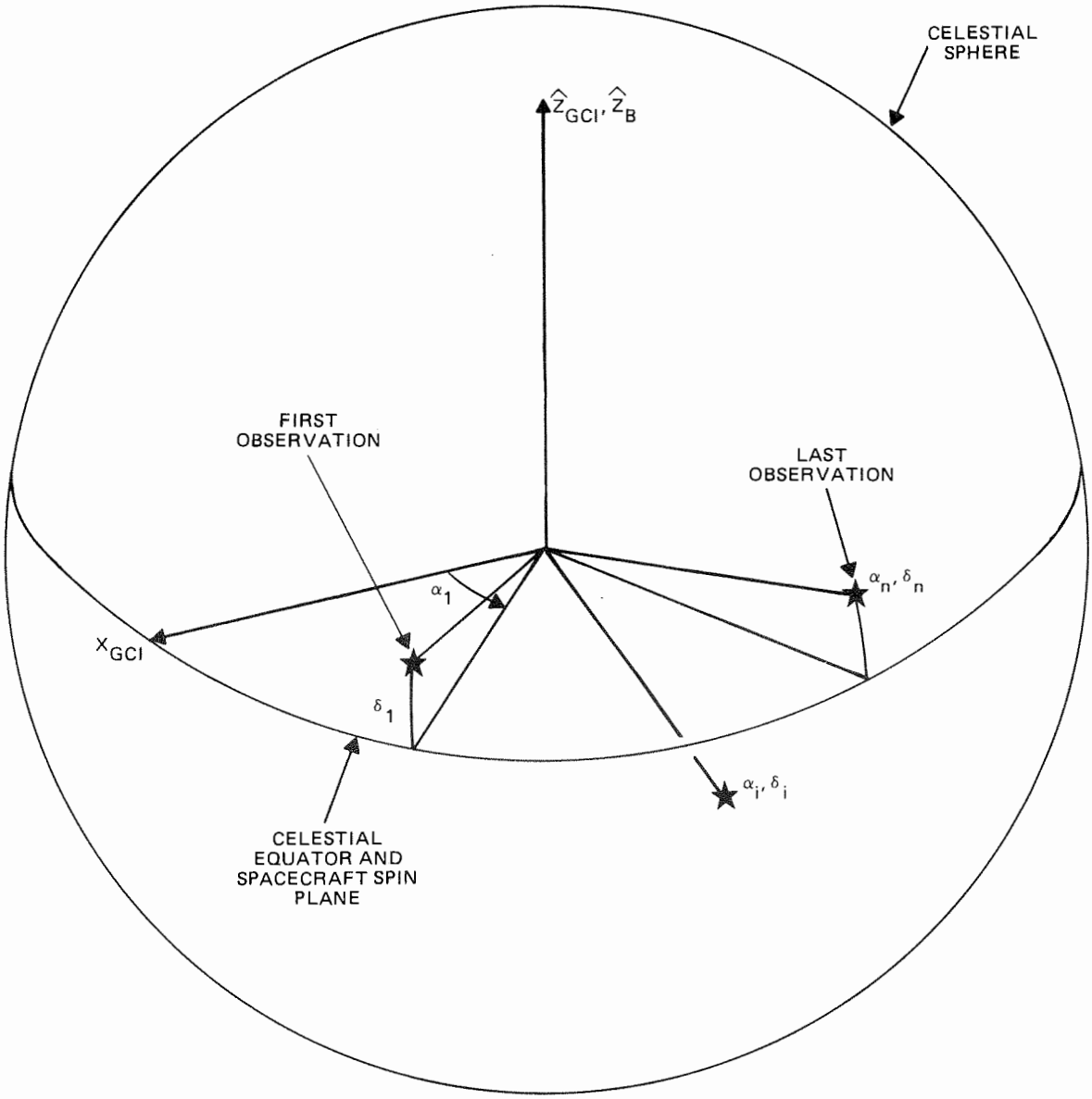


Figure A-3. Geometry of Simulated Star Observations

Runs were made with perfect star observations and also with noisy observations. In the latter runs, no attempt was made to duplicate precisely the noise characteristics expected on HEAO. Specifically, the noise and inaccuracy which will result, in the actual HEAO system, when the star observations are rotated via imperfect gyro data were not modeled directly.

A.7.1 Summary

The first runs made using the power method were all plagued by extremely slow convergence. In these early runs, the maximum number of iterations at first was limited to 50. This limit, however, proved to be far too low; the indications were that typically a thousand or so iterations might be needed in the HEAO application.

In order to obtain a better understanding of the convergence rate difficulties, a series of runs was made in which the power method was replaced by a packaged subroutine which computed all four eigenvalues and eigenvectors of K. This subroutine was EIGRS, which is part of the International Mathematical and Statistical Library (IMSL). With the EIGRS approach, the symmetric matrix K is first reduced to a symmetric tridiagonal matrix T by means of the Householder method. The eigenvalues and eigenvectors of T then are computed using the QL method. The eigenvalues of K are identical to those of T. The eigenvectors of K are computed, in the final operation performed by the routine, by backtransforming those of T. One of the minor advantages of EIGRS is that, unlike the power method, EIGRS does not require the external generation of an initial attitude estimate $\bar{q}^o$.

The version of SNAPLS which included EIGRS in place of the power method computed attitude correctly in the test runs and encountered no noticeable problems. To save time in developing the operational attitude support software for HEAO-A, this version of SNAPLS was incorporated into the operational system. That is, it was considered preferable to code the least-squares

altitude estimation portion of the operational system immediately with this version of SNAPLS, rather than to delay the coding with further attempts to solve the power method's convergence rate problem.

Later simulation runs and analyses verified that the slowness of convergence of the power method in the early runs was caused by the fact that the simulated stars were not spread out over a large region of the celestial sphere. Instead they were bunched in a small clump. The selected maximum separation angle in declination, $\Delta\delta_{MX} = \delta_{MX} - \delta_{MM}$, between observations generally was 8 degrees to 19 degrees in the runs. The 19-degree value is considerably larger than the 8-degree limit which will actually be encountered on HEAO-A. However, the selected maximum separation angle in right ascension, $\Delta\alpha_{MX} = \alpha_n - \alpha_1$, was only 2 degrees to 20 degrees. This is much narrower than the values, generally about 20 degrees to 360 degrees, which normally can be anticipated in the operational HEAO-A problem.

Data presented in the next subsection shows that when the observed stars all lie close to a common plane but are spread out over a large separation angle in this plane ($\Delta\delta_{MX}$ small, $\Delta\alpha_{MX}$ large) the values of the eigenvalues λ_i of K are such that $|r_2|$, $|r_3| << 1$ and $|r_4| \approx 1$. (The present section uses the notation $r_j = \lambda_j/\lambda_1$ and $r'_j = \lambda'_j/\lambda'_1$). The eigenvalue shifting approach ($\lambda'_i = \lambda_i - s$) can be quite effective in enhancing convergence in this case. That is, it can yield $|r'_2|$, $|r'_3|$, $|r'_4| << 1$. However, when the observed stars all lie in a small clump ($\Delta\delta_{MX}$ small, $\Delta\alpha_{MX}$ small), the condition $|r_2|$, $|r_3|$, $|r_4| \approx 1$ will be encountered. In this case, the effectiveness of the eigenvalue shifting method is very limited because it is not possible to make all three ratios $|r'_j| << 1$. This provides an explanation for the slowness of power method convergence seen in the simulation tests.

It is currently believed that use of the K'^m approach of Section A.5.5 would enable the power method to converge accurately in a reasonable number of iterations, even under geometrical conditions as adverse as those of the early

simulation runs. Simulation runs to test this belief, however, were not made in the present study.

There is considerable evidence that the rate of convergence encountered in the simulation runs was significantly slower than that of DOAOP. DOAOP always converges in far fewer than 1000 iterations even when dealing with clump sizes of only a few degrees. The question, however, was not explored in the present work.

A.7.2 Discussion of the Final Series of Runs

This subsection discusses the final series of simulation runs made in the study. The primary purpose of this series was to check the effect of maximum planar separation angle $\Delta\alpha_{MX}$ on the convergence properties of the power method. Both the power method and the EIGRS method were used. The main function of EIGRS here was to establish the eigenvalues λ_1 to λ_4. Knowledge of λ_1 to λ_4 is essential for analysis of results produced by the power method, since the power method's convergence properties depend heavily upon them.

Twelve runs were made in the series. Only the last seven, however, are described here. All inputs were identical for the seven runs except for the spacing in right ascension, $\Delta\alpha_i = \alpha_{i-1} - \alpha_i$, between observations. Ten observations were simulated, their declination values δ_i being as follows: 0, 1, 2, 3, 4, 2, 0, -2, -3, -4 degrees. All the observations were made error-free and were given the same least-squares weighting factor $a_i = 1.0$. The observation spacing in right ascension RA is delineated in the following tabulation.

Run	Separation in RA between each observation (DEG)	Separation in RA between first and last observation $\Delta\alpha_{MX}$ (DEG)
6	1	9
7	2	18
8	4	36
9	8	72
10	16	144
11	32	288
12	40	360

The number of iterations actually required by the power method for convergence and the final error in attitude are shown in Table A-1. In these runs, convergence occurred, by definition, when $\theta_p \leq \theta_{MIN}$ where θ_p is the angular rotation during iteration p and θ_{MIN} was set to $.6 \times 10^{-4}$ degrees. The convergence numbers in Table A-1 have a tolerance of roughly ±5. The variable s^* is the optimal shift value which yields $\lambda_2' = -\lambda_4'$. The values of s^* in runs 6 to 12, respectively, were -.037, -.109, -.400, -1.47, -4.39, -4.50. The variable $s = -3.33$ above is a suboptimal value which was computed using Equation (A-4), with the gain factor k set to 1/3.

Table A-1 provides sample values of the relationship between $\Delta\alpha_{MX}$ and N_c. The results cannot be accepted as universal, however, because N_c is dependent on θ_{MIN} and also on the initial attitude estimate $\bar{q}^o$. For the input conditions which were used, convergence was achieved in a reasonably small number of iterations when $\Delta\alpha_{MX} > 70^o$. For $\Delta\alpha_{MX} < 70^o$ the suboptimal eigenvalue-shift method produced faster convergence than the optimal method, at the expense of a larger final attitude error. Increasing $\Delta\alpha_{MX}$ beyond 90^o evidently does not alter the convergence response significantly.

Table A-2 shows the initial conditions $\epsilon_j^{(o)}$, $j = 2, 3, 4$ of the three transient response modes discussed in Section A.5.4. These $\epsilon_j^{(o)}$ values were calculated using Equations (A-74), (A-80b), and the eigenvectors $\bar{q}_k$ generated by EIGRS. The $\epsilon_j^{(o)}$ results are heavily dependent on the relation between the initial attitude estimate $\bar{q}^o$ and the true attitude $\bar{q}_1$. However, they are also influenced significantly by the observation geometry, as Table A-2 demonstrates.

The remaining data to be presented in this subsection were obtained from the eigenvalues λ_i which were computed by EIGRS. Table A-3 shows the values of the three d_j.

Table A-1. Number of Iterations for Convergence

RUN	$\Delta\alpha_{MX}$ (DEGREES)	s = s*		s = −3.33	
		ITERATIONS TO CONVERGE (N_c)	FINAL ATTITUDE ERROR (DEGREES)	ITERATIONS TO CONVERGE (N_c)	FINAL ATTITUDE ERROR (DEGREES)
6	9	1020	.29 E−4	340	.88 E−2
7	18	420	.35 E−4	220	.34 E−2
8	36	130	.21 E−4	90	.11 E−2
9	72	40	.62 E−5	30	.29 E−3
10	144	10	.47 E−4	20	.51 E−6
11	288	10	.47 E−4	20	.54 E−6
12	360	10	.30 E−4	20	.51 E−6

Table A-2. Initial Values of the Transient Response Modes

RUN	$\Delta\alpha$ MX (DEGREES)	$100\,\epsilon_2^{o}$	$100\,\epsilon_3^{o}$	$100\,\epsilon_4^{(o)}$
6	9	−.073	−1.280	.200
7	18	.156	1.110	−.654
8	36	.316	.945	−.831
9	72	.594	.703	−.914
10	144	.853	−.083	−.974
11	288	.831	.018	−.996
12	360	.593	−.638	−.961

Table A-3. Results of d_1, d_2, and d_3

RUN	$\Delta\alpha$ MX (DEGREES)	d_1	d_2	d_3
6	9	9.956	.0367	.00722
7	18	9.881	.1089	.00972
8	36	9.589	.4004	.01278
9	72	8.522	1.468	.01015
10	144	5.612	4.380	.00830
11	288	5.607	4.393	.00073
12	360	5.496	4.500	.00380

The d_j are the positive square roots of the eigenvalues of $B^T B$ and they are of fundamental importance as indicators of the basic attitude determination geometry. Table A-3 shows that the condition $d_3 << d_1$ was encountered in all runs. This is because all the star observations lay close to a common plane. Table A-3 also shows that the condition $d_2 << d_1$ was encountered when $\Delta\alpha_{MX}$ was small, but that d_2 became comparable in size to d_1 for $\Delta\alpha_{MX} \geq 90^o$. This result tends to validate an earlier assertion that a single observation or multiple colinear observations will yield $d_2 = d_3 = 0$.

The eigenvalues λ_i of K and their ratios $r_j = \lambda_j/\lambda_1$ are shown in Table A-4. This table shows that the expected result $\lambda_1 > \lambda_2 > \lambda_3 > \lambda_4$ was encountered in all runs. However, for $\Delta\alpha_{MX} < 18$ degrees the deviations between the four absolute values $|\lambda_i|$ are small. Increasing $\Delta\alpha_{MAX}$ causes λ_2 and λ_3 to move in toward the origin but produces a negligible effect of λ_1 and λ_4. Each of the three transient response modes decays in proportion to $[r_j]^p$, where p is the iteration number and is to be interpreted as a power. Table A-4 therefore indicates that the basic power method (without eigenvalue shifting) would not converge adequately for any of the runs. For small $\Delta\alpha_{MX}$, convergence of all three modes would be unreasonably slow. For $\Delta\alpha_{MX}$ of 70 degrees or more, modes 2 and 3 would decay with sufficient rapidity, but the rate of mode 4 would not be improved at all. As an example of the slowness of convergence which Table A-5 indicates, for an r_j of .9980, approximately 1150 iterations would be needed to attenuate a modal amplitude ϵ_j to 10 percent of its initial value.

The improvement in convergence rate which can be affected by the eigenvalue shifting technique is shown in Table A-5. These results tend to confirm the results shown in Table A-1 which were obtained with the power method in the simulation runs. When $\alpha_{MAX} > 70$ degrees or so, Table A-5 indicates that convergence should have been rapid for both the $s = s^*$ and the $s = -3.33$ cases, since all three ratios $|r_j'|$ are much less than unity. When $\Delta\alpha_{MX}$ is

small, however, the table indicates that convergence should have been slow because r'_2 was close to unity in both cases.

Table A-4. Eigenvalues λ_i and Their Ratios

RUN	$\Delta\alpha$ MX (DEGREES)	λ_1	λ_2	$-\lambda_3$	$-\lambda_4$	r_2	$-r_3$	$-r_4$
6	9	10.0000	9.912	9.926	9.986	.9912	.9926	.9986
7	18	10.0000	9.763	9.782	9.981	.9763	.9782	.9981
8	36	10.0000	9.179	9.199	9.979	.9179	.9199	.9979
9	72	10.0000	7.044	7.065	9.980	.7044	.7065	.9980
10	144	10.0000	1.223	1.240	9.983	.1223	.1240	.9983
11	288	10.0000	1.213	1.215	9.9985	.1213	.1215	.99985
12	360	10.0000	.992	1.000	9.992	.0992	.1000	.9992

Table A-5. Ratios of Shifted Eigenvalues

RUN	$\Delta\alpha_{MX}$ (DEGREES)	$s = s^*$		$s = -3.33$		
		$r_2' = -r_4'$	$-r_3'$	$-r_2'$	r_3'	$-r_4'$
6	9	.991	−.9854	.993	−.494	.499
7	18	.976	−.957	.982	−.483	.499
8	36	.921	−.846	.938	−.440	.498
9	72	.742	−.488	.778	−.280	.498
10	144	.390	.218	.341	.157	.499
11	288	.390	.221	.341	.159	.500
12	360	.379	.241	.324	.175	.499

GLOSSARY

a	See Figure 3-7, Block A1
a_i	Weighting factor. See Equations (2-1), (2-2) and (2-3)
A	See Equation (A-61)
a_1	See Equations (A-51a), (A-55a)
B	See Equation (2-18)
B	Body-fixed reference frame in spacecraft
B_o	Frame aligned with the preliminary estimate of B. See Figure 3-3
b	See Figure 3-7, Block A1
b_1	See Equations (A-51b), (A-55a)
c	Cosine
c_k	$\bar{q}_k^T \cdot \bar{q}^o$
C	$B^T B$
D	Diagonal matrix of the eigenvalues of H
d_j	Absolute value of eigenvalue j of H; $d_j = \sqrt{\gamma_j}$
f_1	See Equation (A-51c)
f	See Figure 3-7, Block A1
GCI	Geocentric inertial coordinate frame
g( )	Least-squares gain function
H	Symmetric matrix of Lagrange multipliers. See Equation (A-72)
I	3 × 3 Identity matrix

K	See Equations (A-24)
K'	$K - sI$
$\ell(\ )$	Least-squares loss function
m_j	See Equations (A-49), (A-50)
N_c	Number of iterations required for convergence
n	Number of observations to be used in the least-squares attitude computation
P	Attitude matrix indicating the orientation of frame B relative to frame B_o. See Figure 3-3
p, p_1, p_2 q_1, q_2	Coefficients used in the study of the main orthogonalization operation in the preliminary attitude computation; see Equations (3-16) and (3-28)
Q	Matrix whose columns are the eigenvectors of K
$\vec{Q}$	The vector part of quaternion $\bar{q}$
q	The scalar part of quaternion $\bar{q}$
$\bar{q}$	Attitude quaternion
$\bar{q}_k$	Eigenvector k of K
$\bar{q}_1$	Attitude quaternion and first eigenvector of K
$\bar{q}^o$	Initial estimate of $\bar{q}_1$
$\bar{q}_1^p$	Estimate of $\bar{q}_1$ obtained from pth pass through the power method equation
R	Attitude matrix indicating the orientation of frame B relative to frame GI
R_o	The initial estimate of R. See Figure 3-3
r_{ij}	The elements of R

r_j	λ_j/λ_1 ; j = 2, 3, 4
r'_j	λ'_j/λ'_1 ; j = 2, 3, 4
S	$B^T + B$
s	Negative scalar employed to shift the eigenvalues λ_k of K
s	Sine
s	See Figure 3-7, Block A1
s*	The optimum value of s
t_i	The actual time at which Observation i was performed
t_r	The time at which attitude is to be computed
U	Matrix comprised of the eigenvectors $\hat{u}_j$ of C
$\hat{u}_j$	The jth eigenvector of C
$\vec{U}_i$	$R_o \cdot \vec{V}_i$
$\vec{U}_i$	The transformed reference vectors in the preliminary attitude computation. See Figure 3-7
$\vec{U}'_i$	The transformed reference vectors generated by the main orthonormalization operation in the preliminary attitude computation. See Figure 3-7
$\vec{UP}_i$	The transformed observation vectors in the preliminary attitude computation. See Figure 3-7
$\vec{UP}'_i$	The transformed observation vectors generated by the main orthonormalization operation in the preliminary attitude computation. See Figure 3-7
V	3 × n matrix comprised of the reference vectors $\vec{V}_i$
$\hat{V}_i$	Unweighted (i.e., unit) reference vector (frame GI resolution)
$\vec{V}_i$	Weighted reference vector (frame GI resolution)
$\vec{V}_L$	First reference vector selected for the preliminary attitude computation

$\vec{V}_K$	Second reference vector selected for the preliminary attitude computation
v_i	The length of reference vector $\vec{V}_i$
W	3 × n matrix comprised of the observation vectors $\vec{W}_i$
$\hat{W}_i$	Unweighted (i.e. unit) observation vector (frame B resolution)
$\vec{W}_i$	Weighted observation vector (frame B resolution)
w_i	The length of observation vector $\vec{W}_i$
$\hat{X}$	Unit vector along the axis of the rotation which rotates frame GCI onto frame B
x	See Figure 3-7, Block A2
x	See Equation (A-55)
x_1, x_2	See Equations (3-16)
$\vec{Y}$	The attitude vector. See Equation (2-5)
$\vec{Y}P$	The predicted $\vec{Y}$ during DOAOP's iterative attitude computation. See Figure 3-2, Block D
$\vec{Y}_1$	The first $\vec{Y}$ vector which is passed to the iterative attitude computation loop of DOAOP. See Figure 3-2, Block C
y	See Figure 3-7, Block A2
y_1, y_2	See Equations (3-16)
$\vec{Z}$	See Equations (2-17) and (2-22)
α, β, δ	Euler angles. See Figures 3-4 and 3-5
α_j	Eigenvalue j of A
$\Delta\alpha_{mx}$	Separation angle, in spacecraft spin plane, between first and last observed star
$\epsilon_j^{(p)}$	Amplitude of transient response mode j at end of iteration p

Γ	Diagonal matrix of the eigenvalues γ_i of C
γ_j	The jth eigenvalue of C
$\vec{\epsilon}_j$	Observation error vector. See Equation (3-6)
θ	The angular magnitude of the rotation which rotates frame GCI onto frame B
Λ	Diagonal matrix of the eigenvalues λ_k of K
λ_k	The kth eigenvalue of K
λ'_k	The kth eigenvalue of K' ; $\lambda'_k = \lambda_k - s$
μ_j	± 1. See Equation (A-82)
σ , σ_1 , σ_2	± 1
σ	Trace B
ϕ	See Equation (A-52)
ψ_{Li}	Angle between $\hat{V}_L$ and $\hat{V}_i$
ψ_V	Angle between $\hat{V}_L$ and $\hat{V}_K$
ψ_W	Angle between $\hat{W}_L$ and $\hat{W}_K$

Special Notations

$\lvert\ \rvert$	Absolute value
$\hat{(\ )}$	3 × 1 Unit vector
$\vec{(\ )}$	3 × 1 Nonunit vector
$(\ )$	3 × 3 Matrix
$[\ ]$	Matrix, not necessarily 3 × 3
$\underset{\sim}{\vec{(\ )}}$	3 × 3 skew symmetric matrix arrangement of a vector
$\lVert\cdot\rVert$	Euclidean norm

REFERENCES

1. Goddard Space Flight Center, NASA TN D-4696, A Vector Approach To The Algebra of Rotations With Applications, P. Davenport, August 1968

2. Wahba, G., A Least Squares Estimate of Satellite Attitude, Problem 65-1, SIAM Review, July 7(3):409, July 1965

3. Stuelpnagel, J., et al, "A Least Squares Estimate of Satellite Attitude," SIAM Review, July 1966, vol. VIII, no. 3, pp. 384-386

4. Black, H. D., "A Passive System for Determining the Attitude of a Satellite," AIAA Journal, July 1964, vol. II, no. 7, pp 1350-1351

5. Goddard Space Flight Center, X-514-71-312, Attitude Determination and Sensor Alignment Via Weighted Least Squares Affine Transformations, P. Davenport, August 1971

6. Schwarz, H., H. Rutishauser, E. Stiefel, Numerical Analysis of Symmetric Matrices, Prentice-Hall, Inc., 1973, pp. 187-193 and 114-117

7. Gourlay, A., and G. Watson, Computational Methods for Matrix Eigenproblems, John Wiley and Sons, Inc., 1973, pp 38-43 and 45-46

8. Householder, A., The Theory of Matrices in Numerical Analysis, Dover Publications Inc., 1964, Chapter 7

9. Faddeeva, V., Computational Methods of Linear Algebra, Dover Publications, Inc., 1959, Sections 29 and 30

10. Householder, A., Principles of Numerical Analysis, Dover Publications Inc., 1953, pages 153 and 154

11. Wiberg, D., State Space and Linear Systems, Schaum's Outline Series, McGraw-Hill Book Company, 1971

12. Ogato, Katsuhiko, State Space Analysis of Control Systems, Prentice-Hall, Inc., 1967, page 292, Problem B-5-6

13. Athans, M., The Matrix Maximum Principle, Information and Control, vol. 11, nos. 5 and 6, November-December, 1967