NASA CONTRACTOR REPORT

NASA CR-1654

NASA CR-16

C. 1

ANALYSIS OF LIMITED MEMORY ESTIMATORS AND THEIR APPLICATION TO SPACECRAFT ATTITUDE DETERMINATION

by Edwin C. Foudriat

Prepared by
MARQUETTE UNIVERSITY
Milwaukee, Wis.
for Langley Research Center

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION • WASHINGTON, D. C. • AUGUST 1970

1. Report No. NASA CR-1654	2. Government Accession No.	3.
4. Title and Subtitle ANALYSIS OF LIMITED MEMORY ESTIMATORS AND THEIR APPLICATION TO SPACECRAFT ATTITUDE DETERMINATION		5. Report Date August 1970
		6. Performing Organization Code
7. Author(s) Edwin C. Foudriat		8. Performing Organization Report No.
		10. Work Unit No. 125-17-07-06-23
9. Performing Organization Name and Address Marquette University Milwaukee, Wisconsin		11. Contract or Grant No. NGR-50-001-009
		13. Type of Report and Period Covered Contractor Report
12. Sponsoring Agency Name and Address National Aeronautics and Space Administration Washington, D. C. 20546		
		14. Sponsoring Agency Code

15. Supplementary Notes

16. Abstract

A comparison study of the problem of estimating the attitude of a spin stabilized satellite in an earth's field environment has been made employing four types of "optimal" sequential filters. These include the nonlinear Kalman filter, and the Schmidt gain, noise added, and least-squares nonlinear limited memory filter systems. The limited memory concept is applicable to this problem since random disturbances on the satellite are negligible.

The study indicates that all limited memory types give improved accuracy estimates over the Kalman filter. For the Schmidt gain and noise added, the magnitude of the additional terms included to improve filter performance are a compromise between good accuracy and convergence, since the convergence for these filters is degraded from the Kalman filter. In addition, the noise added system requires trial and error to determine the magnitude of the noise covariance terms needed.

The least-squares limited memory filter is made practical by using a set of approximate equations in order to obtain rapid extrapolation of the state and state transition equations. This system which employs more than one measurement in processing is shown to improve both convergence and accuracy over the Kalman filter.

17. Key Words (Suggested by Author(s)) Spin-stablized spacecraft Attitude determination State parameter estimation Kalman filters Non-linear limited memory filters	18. Distribution Statement Unclassified – Unlimited		
19. Security Classif. (of this report) Unclassified	20. Security Classif. (of this page) Unclassified	21. No. of Pages 84	22. Price* $3.00

TABLE OF CONTENTS

ANALYSIS OF LIMITED MEMORY ESTIMATORS AND THEIR APPLICATION
TO SPACECRAFT ATTITUDE DETERMINATION

by

Edwin C. Foudriat

INTRODUCTION

The concept of limited memory has been proposed as a method by which the insensitivity problem in Kalman filtering applications can be alleviated. [1,2,3,4] Insensitivity results after a large number of measurements when the noise disturbance to the states are insignificant. The effect of the number of measurements and state noise can be shown to be counteractive, that is, each measurement reduces the covariance matrix while the effect of noise is to increase the state uncertainty and hence the covariance matrix. If the noise is insignificant then the covariance matrix, after a large number of measurements becomes vanishingly small and the filter becomes insensitive to errors.

The vanishing of the covariance matrix would not, in itself, result in an unsuitable estimation of the states. However, in the usual, non-linear, real-world estimation problem, the system modeled in the computer is generally simplified since many of the states and forces of the real system neglected or approximated. The result of the insensitivity is that these neglected state and forces may create errors between the real and modeled systems and if the filter is insensitive, these errors can exceed the desired estimation accuracy.

The determination of spacecraft attitude is an excellent example of the insensitivity-modeling inaccuracy problem. First, high accuracy is desired over long periods of time. Second, disturbances in the form of earth and solar environmental torques are small. However, these torques with the exception of micrometeorites, should not be considered random disturbances and hence, should not be modeled as noise. Since the effect of micrometeorites on spacecraft motion is considerably less than other torque sources, it is usually neglected. Hence, the noise disturbance to the state are negligible. Finally,

the equations for the modeling magnetic field, aerodynamic, and gravity gradient torques are not exact resulting in errors between the real-world spacecraft and the computer model. As a result, the altitude determination problem when Kalman estimation techniques are used is a classic example of the insensitivity problem and is particularly suited to the use of limited memory techniques.

The study discusses the general problem of state estimation and develops four methods for achieving limited memory. These include the Schmidt gain [1,2] which adds an additional term to the covariance weighting matrix, the addition of noise effects to the covariance matrix (without equivalent noise added to the states), the limited memory, maximum likelihood estimation (M.L.E.) concept of Jazwinski,[3] and a nonlinear version of the least-squares limited memory concept as developed in Appendix IV. A similar concept for linear systems was developed independently by Lee[4].

Using a special set of time-averaged equations[5] it was found that three of the limited memory methods were computationally feasible (the M.L.E. concept was eliminated) for the attitude determination problem. These were programmed for computer analysis so that comparison between the concepts could be made. The report presents the computer program and the comparative results of the three limited memory concepts for a typical spacecraft subjected to earth environment torques.

LIST OF SYMBOLS

$\underline{a}_r$ = rth slit unit normal vector

$\underline{B}$ = earth's magnetic field vector in angular momentum coordinates, gauss

b = $\dfrac{L - L_z}{L_z}$

$\underline{b}$ = $T \underline{s}_i$ = transformed star vector

b_1, b_2, b_3 = components of $\underline{b}$

C_2 = constant (eq. 12)

$\underline{c}$ = $E \underline{b}$ = tranformed star vector

c_1, c_2, c_3 = components of $\underline{c}$

E = Euler angle transformation (angular momentum to body axes)

E_{ij} = ijth term of E

$f(\underline{x})$ = general function of $\underline{x}$

2

G = Jacobian matrix $\frac{\partial g_i}{\partial x_j}$

G_{ij} = ijth term of G

$\underline{g}(\underline{x},t)$ = differential function (see eq. 1)

$\underline{H}(\underline{x},t)$ = measurement model equations

I = identity matrix

$I[q,p]$ = interval of time from q to p

i = satellite orbit inclination

K = spacecraft eddy current damping coefficient, #-ft-sec/gauss^2

K_1, K_2 = earth magnetic field constants

K_s = Schmidt gain factor, (eq. 9)

k = times of occurrance of measurements

L = spacecraft x- and y-body axis inertias, slug-ft^2

L_z = spacecraft z-body axis inertia, slug-ft^2

M_z = spacecraft z-axis magnetic dipole constant, #-ft/gauss

m = magnitude of measured star

m_o = +1 magnitude

n = number of states

P = covariance matrix of $\hat{\underline{x}}$

$Pr(\underline{x})$ = probability density function of $\underline{x}$

$Pr_{\underline{v}_k}$ = probability density function of kth measurement noise

p = most recent measurement time considered in limited memory

Q = system noise covariance

q = earliest measurement time considered in limited memory

R = measurement noise covariance

r = magnitude of angular momentum

$\underline{s}_i$ = ith star unit normal vector

T = inertial-to-angular momentum transformation

T_{ij} = ijth term of T

t_m = measured time, sec

t_t = true time, sec

3

V $\quad = \nabla_{\underline{x}(p)} \underline{H}(\underline{x},k) \big|_{\hat{\underline{x}}_o(k)}$ = gradient of measurement vector

$\hat{V}$ $\quad = V R^{-1/2}(k)$

$\underline{v}$ $\quad$ = measurement noise

W $\quad$ = diagonal matrix of additive noise (see eq. 33)

$\underline{w}$ $\quad$ = system noise

wt $\quad$ = magnitude of comparative function (see eq. 32)

$\underline{x}$ $\quad$ = augmented state vector

$\hat{\underline{x}}_o$ $\quad$ = initial (before addition of newly processed data) estimate of state

$\hat{\underline{x}}$ $\quad$ = improved estimate of state

$\underline{y}$ $\quad$ = measurement

α
β $\quad = \begin{cases} \text{spacecraft orbit position angles related} \\ \text{to earth's magnetic field, deg.} \end{cases}$

$\delta\hat{\underline{x}}$ $\quad$ = deviation in $\hat{\underline{x}}$

η $\quad$ = noise in time measurement, sec.

θ $\quad$ = cone angle (eq. 18), deg.

ν $\quad$ = argument of latitude, deg.

ξ $\quad$ = angle defining direction of angular momentum (eq. 18), deg.

σ_o $\quad$ = standard deviation of +1 magnitude star

σ_η $\quad$ = standard deviation of measured star

τ $\quad$ = angle defining direction of angular momentum (eq. 18), deg.

$\Phi(t,k)$ $\quad$ = state transition matrix

ϕ $\quad$ = spin angle (eq. 18), deg.

ψ $\quad$ = precession angle (eq. 18), deg.

Ω $\quad$ = longitude of ascending node, deg.

ω_o $\quad$ = spacecraft orbital rate

LIMITED MEMORY SEQUENTIAL FILTER SYSTEM

Sequential Filtering System

The model for the dynamic system and the measurement equations are given by

$$\dot{\underline{x}} = \underline{g}(\underline{x},t) + \underline{w}$$

$$\underline{y}(k) = \underline{H}(\underline{x}(k),k) + \underline{v}(k) \tag{1}$$

$$k = 1, \ldots, p$$

where $\underline{x}$ = augmented state vector model of the system

$\underline{y}(k)$ = measurement equation

$\underline{H}(\underline{x})$ = measurement model

$\underline{w}$ = system noise

$\underline{\dot{v}}(k)$ = measurement noise

It is assumed that the noise is white, gaussian with

$$E[\underline{w}] = E[\underline{v}(k)] = 0$$

$$E[\underline{v}\underline{v}^T] = R(k)\ \delta(t-k) \tag{2}$$

$$E[\underline{w}\underline{w}^T] = Q(\tau)\ \delta(t-\tau)$$

In many attitude determination problem cases the additive noise, $\underline{w}$, is zero or at least the neglected torques should not be represented as gaussian white noise.

The weighted least-squares solution* for an improved estimate for $\hat{\underline{x}}(p)$ given an initial estimate $\hat{\underline{x}}_o(p)$ is

$$\hat{\underline{x}}(p) = \hat{\underline{x}}_o(p) + \left[\sum_{k=q}^{p} \hat{V}(k)\hat{V}^T(k)\right]^{-1} \cdot$$

$$\left[\sum_{k=q}^{p} \hat{V}(k)R^{-1/2}(k)\ (\underline{y}(k) - \underline{H}(\hat{\underline{x}}_o,k))\right] \tag{3}$$

---

*See Appendix IV for detailed derivation.

where

$$\hat{V}(k) = \nabla_{\underline{x}(p)} H(\underline{x}(k),k) R^{-1/2}(k)\Big|_{\hat{\underline{x}}_0(k)} \tag{4}$$

Equation (3) assumes that only the measurements $k \in I[q,p]$ are employed, and hence, can be classified as a limited memory system.

To obtain a sequential filter algorithm, it is necessary to formulate a method for including new measurements as they are taken and, if one is to maintain a limited memory concept, for eliminating old measurements. The most common method uses the standard matrix inversion lemma[6,7] along with the assumption that only the new measurement is used to approximate the error term of equation (3). The filter formula adding a new measurement at $k = p+1$ based upon the nonlinear version by Cox,[8] becomes

$$\hat{\underline{x}}(p+1) = \hat{\underline{x}}_0(p+1) + P_{p+1,q} \hat{V}(p+1) R^{-1}(p+1) \cdot$$
$$(\underline{y}(p+1) - \underline{H}(\hat{\underline{x}}_0(p+1),p+1)) \tag{5}$$

As noted, the second summation term in equation (3) has been truncated to contain only the last term. The covariance matrix, (the first summation term in equation (3)) becomes

$$P_{p+1,q} = \left[\sum_{k=q}^{p+1} \hat{V}(k)\hat{V}^T(k) \right]^{-1}_{\hat{\underline{x}}_0(p+1)} \tag{6}$$

Equation (6) can be determined from

$$P_{p+1,q} = \left[P_{p,q}^{-1}(p+1) + \hat{V}(p+1) \hat{V}^T(p+1) \right]^{-1}$$
$$= P_{p,q} - P_{p,q}\hat{V}(I + \hat{V}^T P_{p,q}\hat{V})^{-1}\hat{V}^T P_{p,q} \tag{7}$$

where $P_{p,q}$ is "theoretically" evaluated using the initial estimate, $\hat{\underline{x}}_0(p+1)$. In practice $P_{p,q}$ is accumulated by subsequent application of equation (7) and extrapolation using the P matrix Riccati equation or equivalent, and hence may not represent the true summation inverse as shown

6

in equations (6) and (7). However, in a number of practical applications
the technique used above has lead to a successful although "sub-optimal"
nonlinear filter.

To complete the algorithm the system requires a method for extrapola-
ting the estimate and the covariance matrix to the new measurement point,
that is, calculating $\hat{\underline{x}}_o(p+1)$ and $P_{p,q}(p+1)$ given $\hat{\underline{x}}_o(p)$ and $P_{p,q}(p)$. For
the nonlinear case (again an approximation from the linear case) the
equations developed by Cox[8] and others appear to be adequate. They are

$$\dot{\underline{x}}(p) = \underline{g}(\hat{\underline{x}}(p),p)$$

$$P_{p,q}(p+1) = \Phi(p+1,p) P_{p,q}(p) \Phi^T(p+1,p) \quad\quad (8)$$
$$+ Q(p)$$

The equations are based upon approximating the term, $E[\underline{g}(\underline{x}(p))]$ by the
first term in the Taylor series expansion and by assuming that terms in
the probability density function of higher order than the second central
moment (covariance) are negligible. The term Q represents the result of
the noise $\underline{w}$ in the system equations.

Equations (5), (7) and (8) are the nonlinear Kalman filter system.

Limited Memory Concepts

In general the nonlinear Kalman filter shown above has been used
successfully in a number of aerospace applications. However, when the
disturbing noise, $\underline{w}$, is small or when the neglected terms within the model
do not have a character which can be readily approximated by white noise
the filter has given inaccurate results. This is mainly due to the fact
that the covariance matrix terms become small so that the filter becomes in-
sensitive to small errors. Concepts for alleviating this problem have
become known as limited memory filters.

Limited memory filters have been studied mainly by Schmidt,[1,2]
Jazwinski,[3] and Lee.[4] These three concepts will be outlined in this
section.

The limited memory concept proposed by Schmidt adds a term to the
equation which attaches more weight to the most recent measurement. For
the scalar measurement the second term in equation (8) becomes

$$\delta\hat{\underline{x}} = \left[P_{p,q} \,\hat{\underline{v}}(\hat{\underline{v}}^T P_{p,q}\underline{v} + 1)^{-1} + \frac{\hat{\underline{v}}K_s}{\hat{\underline{v}}^T\hat{\underline{v}}}(\hat{\underline{v}}^T P_{p,q}\underline{v} + 1)^{-1}\right] \cdot$$

$$R^{-1/2} (y - H(\underline{x})) \tag{9}$$

where the gain factor K_s is used to weight the present measurement. Note that the first term in the brackets in equation (9) is an alternate method for writing the measurement weighting term of equation (5).

The effect of weighting factor K_s can be seen by multiplying equation (9) by $\underline{v}^T$ to give

$$\cdot \underline{v}^T\delta\underline{x} = \left(\frac{\hat{\underline{v}}^T P_{p,q}\,\hat{\underline{v}} + K_s}{\hat{\underline{v}}^T P_{p,q}\,\hat{\underline{v}} + 1}\right) (y(p+1) - H(\underline{x}(p+1))) \tag{10}$$

Since $\underline{v}^T\delta\underline{x}$ is the first term in the Taylor series expansion of the measurement error, then equation (10) can be related to the amount of the measurement error incorporated in the new estimate. With $K_s = 1$ the weighting is very large; with $K_s << 1$ and noting that for small covariance matrix terms, $\hat{\underline{v}}^T P_{p,q}\,\hat{\underline{v}} << 1$ then the weighting on the present measurement becomes small.

A conceptually similar weighting procedure can be evolved by assuming that sufficient noise is added to the covariance matrix between measurement points p and p + 1 such that $P_{p,q}(p+1)$ satisfies the relationship

$$\hat{\underline{v}}^T P_{p,q}\,\hat{\underline{v}} >> 1$$

Using this relationship, and equation (5) with the alternate form of the Kalman gain, the term $\underline{v}^T\delta\underline{x}$ becomes identical to condition in equation (10) with $K_s = 1$. Hence, the use of the Schmidt gain coefficient is conceptually equivalent to the addition of noise in that present measurement is given additional emphasis in formulating the new estimate of the state.*

While the two different forms of limited memory have similar effects, a difference between them exists. The weighted measurement using K_s does not affect directly the values of the covariance matrix in that the weighting is added to the new estimate equation directly. On the other hand the addition of noise directly to the covariance matrix does affect its future values.

---

*An alternate approach to demonstrating the effect of noise is to examine the solution matrix of the Ricatti differential equation.

The limited memory concept developed by Jazwinski is formulated from the maximum likelihood technique and Bayes estimation.[9] Jazwinski[3] proves that

$$Pr(\underline{x} \mid |Y_{p,q+1}|) = C_2 \frac{Pr(\underline{x}|Y_{p,1})}{Pr(\underline{x}|Y_{q,1})} \tag{12}$$

where $Y_{m,n}$ indicates the sequence of measurements from m to n and the term on the left indicates the density function dependent upon the sequence only and not upon the apriori estimate $Pr(\underline{x}(o))$. Hence $C_2 = f(\underline{x})$. Note that equation (12) can be obtained directly from Bayes rule for the additive noise filter problem, equation (1), and the assumption that $\underline{w}$ is zero as

$$Pr(\underline{x} \mid Y_{p,1}) = \frac{Pr(\underline{x}, Y_{p,1})}{Pr(Y_{p,1})}$$

$$= \frac{Pr(x_0) \prod\limits_{i=1}^{p} Pr_{v_k}(\underline{y}(k) - \underline{H}(\underline{x}(k),k))}{Pr(Y(1), \ldots, Y(p))} \tag{13}$$

with the constraint equation $\dot{\underline{x}} = \underline{g}(\underline{x},t)$. By formulating both numerator and denominator, equation (12) becomes

$$Pr(\underline{x} \mid |Y_{p,q+1}|) = \frac{\prod\limits_{i=q+1}^{P} Pr_{v_k}(\underline{y} - \underline{H}(\underline{x}))}{Pr(Y_{p,1} | Y_{q,1})} \tag{14}$$

The limited memory system of Jazwinski requires the operation of two simultaneous Kalman filters to satisfy the estimation and as a result a "batch-type" process is necessary. Consequently, the numerical solution on a digital computer for an attitude determination problem might be as involved as the least-squares batch program used for NAS1-6010[10] and hence undesirable from a computer usage standpoint.

Another method for limited memory is based upon the least-squares estimation procedure. The linear case was originally developed by Lee[4]. The nonlinear procedure, an extension of the least squares approach in Appendix IV, can

be evolved from equation (3) by considering the two portions of the second term separately. The covariance term can be treated by repeated application of equation (7) where

$$P_{p+1,q+1} = [P_{p+1,q}^{-1} - \hat{V}(q)\hat{V}^T(q)]^{-1}$$

which becomes

$$P_{p+1,q+1} = P_{p+1,q} + P_{p+1,q}\hat{V}(I - \hat{V}^T P_{p+1,q}\hat{V})^{-1}\hat{V}^T P_{p+1,q} \quad (15)$$

using the matrix inversion lemma.

Equation (15) can be evaluated at $t = p$ by noting that equation (4) can be written (for the scalar measurement) as

$$R^{1/2}(k)\hat{\underline{V}}(k) = \nabla_{\underline{x}(p)} H(\underline{x}(k),k)$$

$$= \begin{pmatrix} \sum_i \dfrac{\partial H}{\partial x_i(k)} \dfrac{\partial x_i(k)}{\partial x_1(p)} \\ \cdot \\ \cdot \\ \cdot \\ \sum_i \dfrac{\partial H}{\partial x_i(k)} \dfrac{\partial x_i(k)}{\partial x_n(p)} \end{pmatrix}$$

$$= \nabla_{\underline{x}(k)}^T H(\underline{x}(k),k)\ \Phi(k,p) \quad (16)$$

where $\Phi(k,p)$ is the state transition matrix contained in equation (8).

Note that in evaluating equation (16) at $k = q$ it is necessary to integrate both state and state transition backwards from $t = p+1$. In this process it becomes feasible to evaluate measurement error portion, that is, the second term in equation (3)

$$\sum_{k=q+1}^{p+1} \hat{V}(k)\ R^{-1/2}(k)\ (\underline{y}(k) - \underline{H}(\hat{\underline{x}}_o(k),k)) \quad (17)$$

at all points within the interval $I \in [q,p+1]$ at which the time corresponds to a measurement point. These terms can be used to reinforce the error and hence conceivably obtain improved convergence and accuracy.

In this section the general sequential filter system and four concepts for limited memory; additive noise, Schmidt gain weighted measurement, maximum likelihood estimation using limited statistics with no apriori estimate, and least-squares addition and subtraction of data points have been outlined. These concepts differ substantially in their effects on the various functions within the filter. As noted previously, the noise addition affects only the covariance matrix which after a time period should stabilize to a specific value mostly dependent upon added magnitude of noise assuming the measurement sequence is repeated. Conversely, the covariance matrix for the Schmidt gain concept should continue to decrease since the weighting does not affect this factor. Both of these techniques rely upon processing the present measurement error only.

The maximum likelihood limited memory reflects the optimum filter and the covariance should indicate the accuracy of the estimate. As noted, this procedure is relatively difficult to program requiring two Kalman filters and a "batch-type" process even for the linear case. For the nonlinear problem a process similar to the NASI-6010[10] least-squares estimate might result and hence be impractical.

The limited memory least-square technique employs a concept, which like noise addition, maintains after stabilization a relatively fixed covariance matrix. This should reflect to some degree the accuracy of the unbiased estimate. The technique has the additional feature that past data points within the span can be included if desired to augment accuracy. It requires extrapolating the present estimate and its state transition matrix to the end points of each memory span. If data within the memory span is used the state and state transition matrix at each measure-ment point must be evaluated. This is only practical when a rapid integra-tion solution for the system of state variables is available. This latter condition exists for the spinning body attitude determination problem by use of the time-averaged perturbation equations.[5]

In the remainder of the report the three sequential filtering tech-niques, noise addition, weighted measurements and least-squares sequential limited memory will be developed for the attitude determination problem and test results discussed.

APPLICATION OF LIMITED MEMORY SEQUENTIAL FILTER TO
THE ATTITUDE DETERMINATION PROBLEM

In the above section four limited memory concepts were formulated. The maximum likelihood concept appeared to be impractical and the least-square concept was dependent upon a rapid integration procedure for extrapolating the estimate over the memory interval. The following section of the report develops the specific equations for the attitude determination problem and presents the computer programs necessary to simulate the three limit memory concepts. In the first section the vehicle equations of motion and the auxiliary equations necessary to generate the earth's field environment for a spinning satellite are formulated. These are followed by the measurement equation. The resultant computer programs necessary to obtain the star sighting and the limited memory sequential estimation are then formulated.

Vehicle Equations of Motion

The spinning body dynamics assumed for the problem are described by a set of time-averaged perturbation equations. The body is assumed to be symmetric and influenced by earth magnetic field torques only. The six state equations are

$$\dot{\phi} = \frac{r}{L} + \frac{K B_y B_z (1 + b \cos^2\theta) \cot \tau}{L}$$

$$+ \frac{B_x M_z \cos \theta \cot \tau}{r} - \frac{B_z M_z \cos \theta}{r}$$

$$\dot{\psi} = \frac{b r \cos \theta}{L} + \frac{B_z M_z}{r} \qquad * \qquad\qquad (18)$$

$$\dot{\theta} = - \frac{b K \sin 2\theta (B^2 + B_z^2)}{4L}$$

$$\dot{\tau} = \frac{K B_x B_z (1 + b \cos^2\theta)}{L} - \frac{B_y M_z \cos \theta}{r}$$

---

*Note that the term in $\theta(o)$ has been neglected.

12

$$\dot{\xi} = \frac{K\,B_y\,B_z\,(1 + b\cos^2\theta)}{L\sin\tau} + \frac{B_x\,M_z\,\cos\theta}{r\sin\tau}$$

$$\dot{r} = \frac{r\,K(B_z^2 - B^2)(1 + b\cos^2\theta)}{L}$$

where

ϕ = spin angle, first rotation-about angular momentum $\bar{z}$ axis

θ = cone angle, second rotation-about once displaced momentum x axis

ψ = precession angle, third rotation about once displaced z axis

τ = angle defining inclination of angular momentum direction relative to inertial Z-axis

ξ = angle between X inertial axis and Z-$\bar{z}$ plane

r = magnitude of angular momentum

L = spacecraft x- and y-axes inertia (assuming symmetric body)

L_z = spacecraft z-axis inertia

$b = \dfrac{L - L_z}{L_z}$

K = spacecraft eddy current damping constant

M_z = spacecraft z-axis magnetic dipole constant

$\underline{B}$ = earth's magnetic field vector in angular momentum coordinate system

$B^2 = \underline{B}^T\underline{B}$

The derivation of equations (18) can be found in ref. 5. A unique feature of the equations is the fact that all short period cyclic variations have been eliminated by time averaging. Since the remaining state variables θ, τ, ξ, and r change extremely slowly* it is feasible to integrate the equations using extremely long time intervals. This feature permits the successful implementation of the least-squares limited memory concept.

The accuracy of these set of equations is also indicated in ref. 5. In that report a typical comparison shows the states remain within a few seconds of arc of the exact equations over periods of time as long as 1000 sec. Thus these equations are capable of representing, with a high

---

*This fact is demonstrated in the simulation results.

degree of accuracy, the true attitude motion of a symmetric spin stabilized vehicle in an earth's field environment. Reference 5 gives a method for obtaining a similar set of equations for the nonsymmetric vehicle and the inclusion of gravity gradient torque effects.

In addition to the six states certain of the spacecraft parameters may be uncertain before the flight or change slightly during flight. Hence b, K, and M_z can be used in an augmented state system and estimated also.

A dipole magnetic field model has been used assuming that the space-craft is in a circular orbit. With this model the inertial components of the magnetic field become

$$B_X = K_1 \sin 2\alpha \cos \beta$$
$$B_Y = K_1 \sin 2\alpha \sin \beta \qquad (19)$$
$$B_Z = K_1 (K_2 - \cos 2\alpha)$$

The constants K_1 and K_2 are functions of the spacecraft orbital altitude. The angles α and β are functions of the orbital elements* and are given by

$$\sin \alpha = \sin \nu \sin i$$
$$\tan \beta = \frac{\cos i \sin \nu}{\cos \nu} \qquad (20)$$

With a circular orbit

$$\nu = \nu_0 + \omega_0 t \qquad (21)$$

The relationships for the magnetic field are derived in ref. 11.

The inertial components of magnetic field are transformed to the angular momentum axes by

$$T = \begin{pmatrix} \cos \tau \cos \xi & \cos \tau \sin \xi & -\sin \tau \\ -\sin \xi & \cos \xi & 0 \\ \sin \tau \cos \xi & \sin \tau \sin \xi & \cos \tau \end{pmatrix} \qquad (22)$$

---

*The longitude of the ascending node, Ω, can be assumed to be zero.

In addition to the inertial-to-angular momentum axes transformation, the Euler angle (angular momentum-to-body axis) transformation is required. This is

$$E = \begin{bmatrix} c\phi c\psi - c\theta s\phi s\psi & s\phi c\psi + c\theta c\phi s\psi & s\theta s\psi \\ -c\phi s\psi - c\theta s\phi c\psi & -s\phi s\psi + c\theta c\phi c\psi & s\theta c\psi \\ s\theta s\phi & -s\theta c\phi & c\theta \end{bmatrix} \qquad (23)$$

Equations (18) - (23) define the motion of the spinning spacecraft in an earth's magnetic field environment.

Measurement System

A measurement is taken when the slit plane of a body mounted telescope and the star are coincident.[12] The time of occurrence of this event is recorded and it is this time which is employed to determine the precise vehicle attitude.

The geometric condition of the measurement is satisfied when

$$H(\underline{x}(t),t) = \underline{a}_r^T E T \underline{s}_i = 0 \qquad (24)$$

where $\underline{a}_r$ = r th slit normal vector

$\underline{s}_i$ = i th star vector

Besides the measurement equation, the measurement gradient, $\nabla_{x(p)} H(\underline{x}(k),k)$ is required in equations (4) and (16). The gradient function $\nabla_{x(k)} H(\underline{x}(k),k)$ is

$$\frac{\partial H(k)}{\partial x_1(k)} = \underline{a}_r^T \frac{\partial E}{\partial \phi} T \underline{s}_i$$

$$\frac{\partial H(k)}{\partial x_2(k)} = \underline{a}_r^T \frac{\partial E}{\partial \psi} T \underline{s}_i$$

$$\qquad (25)$$

$$\frac{\partial H(k)}{\partial x_3(k)} = \underline{a}_r^T \frac{\partial E}{\partial \theta} T \underline{s}_i$$

$$\frac{\partial H(k)}{\partial x_4(k)} = \underline{a}_r^T E \frac{\partial T}{\partial \tau} \underline{s}_i$$

15

$$\frac{\partial H(k)}{\partial x_5(k)} = \underline{a}_r^T E \frac{\partial T}{\partial \xi} \underline{s}_i$$

$$\frac{\partial H(k)}{\partial x_i(k)} = 0 \quad i = 6, \ldots, n$$

The state-transition matrix as indicated in equations (8) and (16) is needed both for the measurement equation and to update the covariance matrix. It can be obtained in differential equation forms as

$$\dot{\phi}_{ij} = \frac{\partial \dot{x}_i(k)}{\partial x_j(p)} = \sum_k \frac{\partial g_i}{\partial x_k} \frac{\partial x_k}{\partial x_j} = \sum_k G_{ik} \phi_{kj} \tag{26}$$

or in matrix form

$$\dot{\phi} = G\phi \qquad \phi(0) = I$$

The actual terms used in equations (25) and (26) for the attitude determination problem of equations (18) - (23) are given in Appendix I. The results of equations (25) and (26) can be used to obtain $\nabla_{\underline{x}(p)} H(\underline{x}(k),k)$.

In addition to the gradient, the covariance of the additive noise in the measurement equation is required. The actual measurement equation (24) does not contain time explicitly but time is the quantity recorded at each measurement event. While this factor does not affect the measurement equation (24) since the states at the specific time instant are required, the additive noise term in the measurement equation is not readily available. An approximation to the additive noise can be made by considering

$$H(\underline{x}(t_+)) \approx H(\underline{x}(t_m)) + \frac{dH}{dt}\Big|_{t_m} \eta \tag{27}$$

where t_m = measured time

t_+ = true time

η = random time error

16

Noting that $\frac{\partial H}{\partial t} = 0$ so that

$$\frac{dH}{dt} = \nabla_{\underline{x}} H^T \dot{\underline{x}}$$

the additive noise term is $\nabla_{\underline{x}} H^T \dot{\underline{x}} \Big|_{t_m} \eta$

Assuming $E[\eta] = 0$ and uncorrelated with $\hat{\underline{x}}$, then the measurement noise covariance R_k is given by

$$R(k) = E\{(\nabla_{\underline{x}} H^T \dot{\underline{x}} \Big|_{t_m})\eta)^2\} \tag{28}$$

Assuming $\dot{\theta}$, $\dot{\tau}$, $\dot{\xi}$, and $\dot{p}$ are negligible in comparison to $\dot{\phi}$ and $\dot{\psi}$ and that the first terms in equation (18) predominate then

$$R(k) = E\{(\underline{a}_r^T \frac{\partial E}{\partial \phi} T \underline{s}_i + \underline{a}_r^T \frac{\partial E}{\partial \psi} T \underline{s}_i k \cos^2\theta)^2 \frac{r^2}{L^2} \Big|_{\underline{x}(k)} \eta^2\} \tag{29}$$

If one further assumes that the random variables $\hat{\underline{x}}(t_m)$ can be replaced by their present estimates and the covariances are negligible in comparison to the means then

$$R(k) = (\underline{a}_r^T \frac{\partial E}{\partial \phi} T \underline{s}_i + \underline{a}_r^T \frac{\partial E}{\partial \psi} T \underline{s}_i k \cos^2\theta)^2 \frac{r^2}{L^2} \Big|_{\hat{\underline{x}}_o} \sigma_\eta^2(k) \tag{30}$$

While it is true that the measurements noise in above form are correlated the use of $R(k)$ as the weighting matrix is more desirable than attempting to process an alternate form of the measurement equation.

The value used for σ_n is determined from the work of Ostroff and Romancyzk,[13] who show both analytically and experimentally that the standard derivation σ for estimating the center of the star corresponds to the equation

$$\sigma_\eta = \sigma_o \, 10^{\left(\frac{m - m_o}{5.02}\right)} \tag{31}$$

Assuming a +1 magnitude star has angular error with a 10 $\widehat{\text{sec}}$. standard deviation, equation (31) is used to generate the standard deviation used in equation (30).

Computer Simulation

A computer simulation using equations (10) - (26) and (30) - (31) has
been developed for an IBM 7040 in order to study the effectiveness of the
limited memory procedures for precise attitude determination. The simula-
tion consists of two programs, the first to generate star sightings and
the second to employ the various nonlinear limited memory estimation
concepts in determining the attitude of typical spacecraft.

The computer program flow diagram for the instrument simulation,
that is, the generation of star sighting data is shown in Fig. I. The
program consists of two main branches. The first integrates the equations
of motion to null the error between the slit plane and the star vector
in order to satisfy the measurement, equation (24). This branch interlaces
the actual differential equations along with the angle transforms since
the angular momentum transform data is required for both the differential
equations and the measurement equations. In addition the star vector
misalignment data in the body axes can be used to obtain a simple but
accurate estimation of the rotation angular error - - hence, the time
interval required if the null condition is not satisfied. To preserve a
sighting accuracy of I $\widehat{sec}$., the states, the differential equations for $\dot{\psi}$
and $\dot{\phi}$, and the measurement times are carried out in double precision.

The second branch is used when the measurement equation is satisfied
to within I $\widehat{sec}$. A gaussian random noise generator is used to add a zero
mean time error proportional to the star magnitude,

$$t_m = t_t + \sigma_n$$

where σ_n is the time error noise selected using equation (31). The data
are then recorded on tape, including the star and slit, the true and measured
time, the six state variables, and the vector components of the instrument
axis (assumed to be y-body axis) in inertial space. The program can
generate data over any period but does not have the capability to add or
drop stars.

A computer listing of the program is shown in Appendix II. Note that
the comment cards in the listing correspond blocks on Fig. I to simplify
understanding of the program.

18

DATA INPUT,CONSTANTS,INITIAL STATES(VAL),CONTROL
NUMBERS,ETC., STAR ANGLES(SIGMA,GAMA),
MAGNITUDES(SMAG),SLIT ANGLES(ALAMD,ALPH1), ETC.

TAPE INITIALIZATION

PARAMETER INITIALIZATION
    SLIT VECTOR COMPONENTS
    STAR VECTOR COMPONENTS

INERTIAL-ANGULAR MOMENTUM AXES
    TRANSFORM

NO

TEST FOR INTEGRATION TIME INTERVAL
    COMPLETED

YES

EULER TRANSFORM
    (ANGULAR MOMENTUM-BODY AXES)

STAR VECTOR TRANSFORMED TO
    BODY AXES

RANDOM NOISE GENERATOR
    SUBROUTINE CALL

YES

TEST FOR ORTHOGNAL SLIT AND
    STAR VECTORS

NO

COMPUTE MEASUREMENT TIME

MAGNETIC FIELD INTENSITY
    COMPONENTS

OPTICAL AXES COMPONENTS
    IN INERTIAL SPACE

DIFFERENTIAL EQUATIONS

WRITE DATA ON TAPE

NO

TEST FOR INTEGRATION TIME INTERVAL
    COMPLETED

YES

INDEX TO NEXT STAR
    AND/OR SLIT

ANGULAR ERROR IN SPIN AND
    PRECESSION PLANE

TEST FOR STAR SIGHTING COUNT

NO

INTEGRATION TIME INETERVAL TO
    NULL ERROR

YES

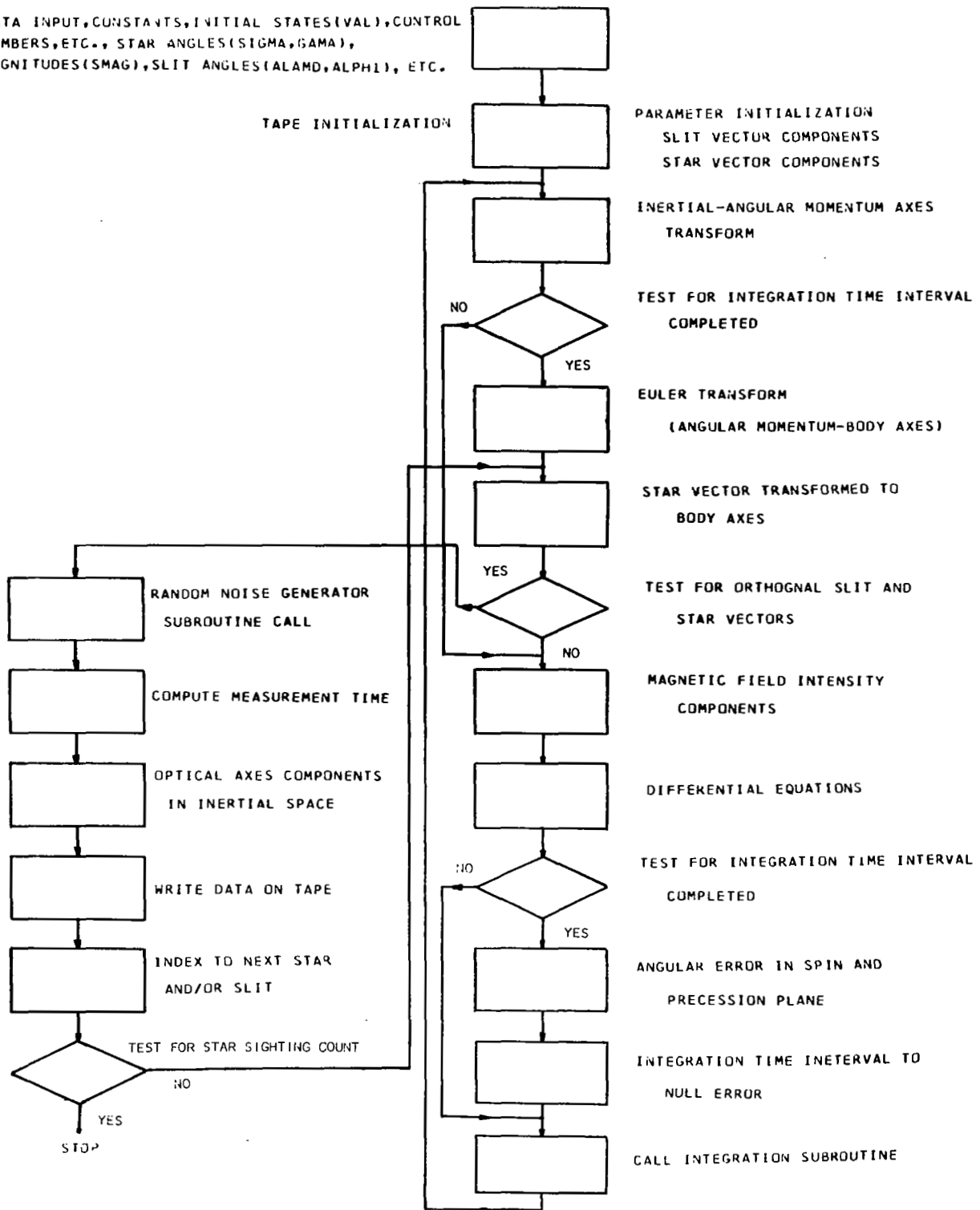STOP

CALL INTEGRATION SUBROUTINE

FIGURE I   PROGRAM FOR DETERMINATION OF STAR SIGHTINGS AND PREPARATION OF
           TAPES FOR LIMITED MEMORY FILTER

19

The computer flow diagram for the second program, the limited memory estimation system, is shown in Fig. 2. This program is relatively straightforward in that it follows a fairly standard pattern for sequential filtering with a few exceptions. The first, is the method for storing star data in the memory and the method by which new data is added and old deleted. Due to the fact that the IBM 7040 was limited to 16,000 words of storage, it was decided to read a segment of stars into the memory from tape and process these stars. When the sequential filter needed a new star not in the memory then those stars no longer needed in the filter are eliminated, the set of stars in the memory but still required by the filter are shifted and new stars read in. The tape is then reindexed back to the proper point.

The second is the method by which stars can be selected for processing within the next data cycle for the least-squares limited memory system. Two alternatives which both use random selection are available. The first uses an ordered sequence, and the second selects fixed step sizes and then selects a particular star within a band of 5 - 10 stars about that fixed point. This latter technique leads to a somewhat more uniform selection of stars across filter memory length.

The integration of the state and the solution of the state transition matrix use a fourth order Runge Kutta routine. The state transition matrix is used to extrapolate the covariance matrix but not exactly as indicated in equation (8). Instead the $P^{1/2}$ concept as reported in ref. 14 is used. This latter procedure eliminates the need for double precision in the calculation of P while at the same time assuring a positive definite covariance matrix.

The processing of star data for a particular measurement update is straightforward. The newest star is processed first as in the Kalman filter and then if the least-squares limited memory procedure is being used any additional previously measured stars are processed until all selected stars within the memory window have been included.

Once all star points within the window have been processed the new covariance matrix is calculated. Ref. 1 presented a method for processing both the addition and subtraction to the covariance matrix simultaneously. Due to the fact that this solution may cause difficulties because of large

20

Left column (top to bottom):

4. CALCULATE GRADIENT OF H(X(K)) WITH RESPECT TO X(K) AND MEASUREMENT COVARIANCE INVERSE

5. UPDATE GRADIENT AND MEASUREMENT USING STATE TRANSITION MATRIX

6. ACCUMULATE WEIGHTED ERROR

TEST FOR FIRST(NEW) OR LAST (OLDEST) STAR IN WORKING CHART (MEMORY WINDOW)

NO / YES

STORE GRADIENT DATA FOR FIRST AND LAST STAR TO BE USED IN P**1/2 UPDATE

NO

TEST FOR LAST STAR

YES

CALCULATE NEW P**1/2 MATRIX

CALCULATE NEW ESTIMATE

TEST FOR STAR POINTS AT WHICH TO PERFORM ACCURACY ANALYSIS AND DATA PRINT

NO / YES

READ TRUE STATES AND DIRECTION OF INSTRUMENT AXIS VECTOR

COMPARE STATE AND INSTRUMENT AXIS POINTING TO OBTAIN ERRORS

WRITE EVALUATION OF ESTIMATION ACCURACY

INDEX FOR NEW DATA POINT AND TEST FOR RUN COMPLETION OR NONCONVERGENCE

Right column (top to bottom):

DATA INITIALIZATION FOR EACH RUN.

READ STAR CHART FROM TAPE

SLIT,STAR,AND CONSTANT PARAMETER INITIALIZATION

TEST FOR EXCEEDING STAR CHART IN MEMORY

NO / YES

UPDATE STAR CHART TO ELIMINATE OLD DATA AND READ IN NEW

SELECT BY RANDOM NUMBERS THE INTERMEDIATE STARS

ASSEMBLE WORKING STAR CHART(STARS TO BE USED TO OBTAIN NEW ESTIMATE)

UPDATE STATE AND CALULATE STATE TRANSITION MATRIX TO NEW STAR TIME

CALCULATE COVARIANCE MATRIX(P**1/2) AND ADD NOISE IF ANY

TEST FOR VALID STAR (SINCE NEGATIVE STARS ARE NOT ALLOWED)

NO / YES

INTEGRATE STATE AND STATE TRANSITION MATRIX TO OLD STAR SIGHTING TIMES

MEASUREMENT ANALYSIS
1. DETERMINE SLIT AND STAR

2. TRANSFORM STAR VECTOR FROM INERTIAL TO BODY AXES

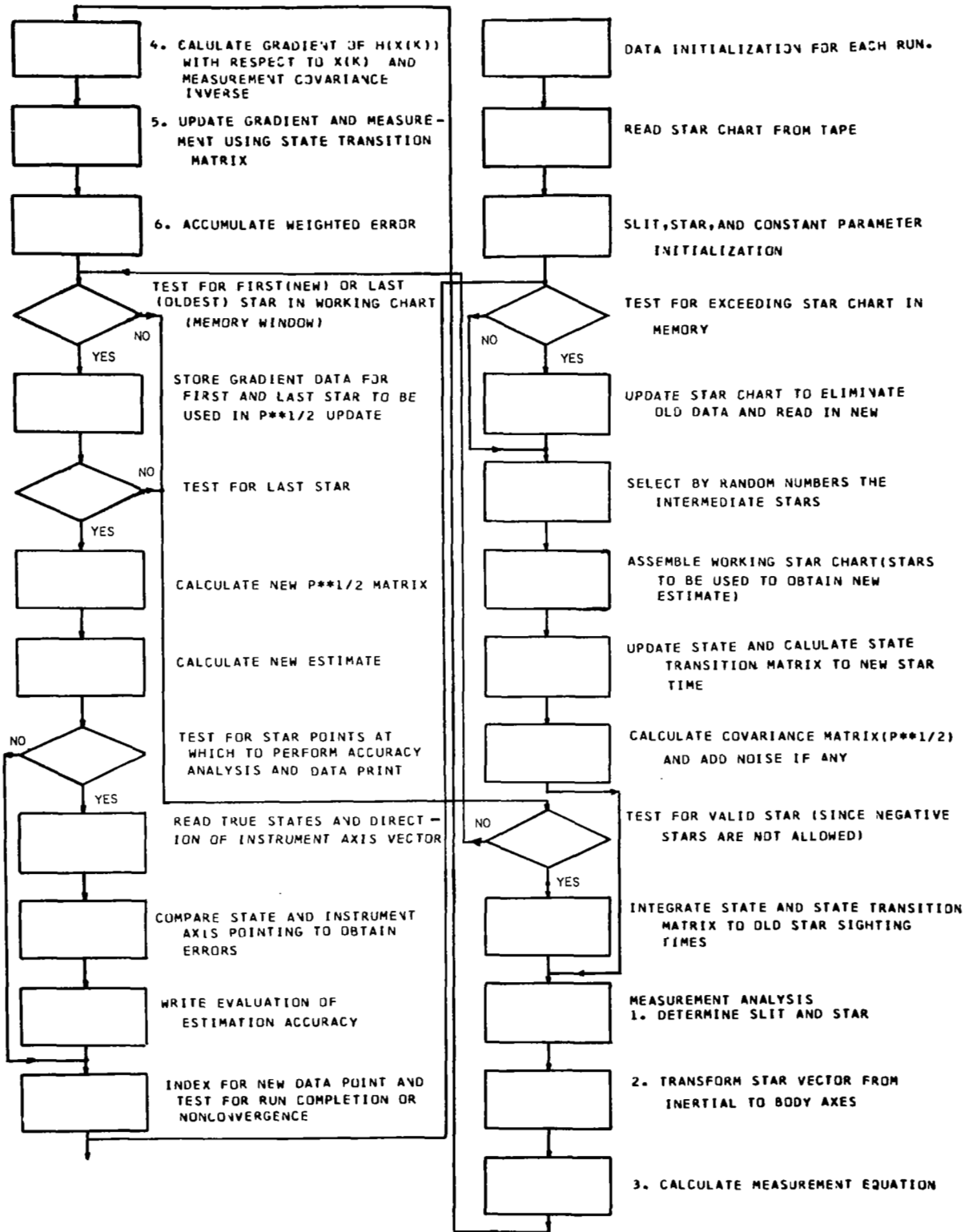3. CALCULATE MEASUREMENT EQUATION

FIGURE 2   LIMITED MEMORY FILTER PROGRAM

subtractive changes in P, the computer augments P in two steps, first
adding the new data and then subtracting the old data. Before the subtrac-
tion to P is accomplished its effect is tested by the equation

$$\hat{v}^T \ P_{p,q} \ \hat{v} \ < \ wt \tag{32}$$

If this condition is not satisfied $P^{1/2}$ is not augmented by deletion of the
old data.

Using the new P matrix, the new estimate is obtained and at specific
points, the accuracy of the estimate is compared to that of the exact
solution.

The estimation program with instructions on how to control the options
is listed in Appendix III. The comment statements in the program correspond
to the blocks in Fig. 2.

SIMULATION RESULTS

The two computer programs described in the previous section have been
used to study the various concepts of sequential and limited memory filters
for the attitude determination problem of a spin-stabilized vehicle in an
earth-field torque environment. The specific capabilities of each system
to obtain the correct estimate in the case of measurements corrupted by
noise is presented.

The results of the star generation program are shown in Figs. 3-5 for
the cases with eddy current only and eddy current plus magnetic dipole
torques. The nominal spacecraft has the following conditions

$$\phi(o) \ = \ \psi(o) \ = \ 0 \ \text{deg.}$$
$$\theta(o) \ = \ 1.5 \ \text{deg.}$$
$$\tau(o) \ = \ -87.5 \ \text{deg.}$$
$$\xi(o) \ = \ 85 \ \text{deg.}$$
$$r(o) \ = \ 20. \ \text{\#-ft-sec}$$
$$L \ = \ 56.68 \ \text{slug. ft.}^2$$
$$L_z \ = \ 65.62 \ \text{slug. ft.}^2$$
$$K \ = \ .1427 \times 10^{-4} \ \text{\#-ft-s/gauss}^2$$
$$M_z \ = \ .5105 \times 10^{-5} \ \text{\#-ft/gauss}$$

The orbital characteristics which affect the magnetic field are identical to those given in Ref. 11. With the assumed conditions the spin axis (angular momentum direction) is within 10° of the inertial X-Z plane and within 15° of a nominal sun synchronous orbit with an orbital inclination i = 97.38°.

The placement of the stars shown in Table I would be similar to those in a typical low altitude orbit where earth blockage might eliminate stars within a 140° portion of a revolution for a vehicle with its nominal spin axis normal to its orbital plane.

Table I

Star	Mag.	Az.	Elevation
1	+1.5	10°	0°
2	+2.5	-5°	70°
3	+1.0	10°	165°
4	+3.0	0°	225°

The star magnitudes were used to weigh the accuracy of the sighting data according to equation (31) with a +1.0 nominal magnitude star assumed to have a $\sigma = 10 \overset{\frown}{sec}$. The estimation program assumed that all stars were +1.0 magnitude.

Figs. 3-5 show the effect of the eddy current and magnetic dipole torques. As noted in equation (18) the magnetic dipole torques do not affect cone angle and angular momentum. Conversely the results of Fig. 3 indicate that the eddy current torques have negligible affect on the motion of the angular momentum axes in comparison to the magnetic dipole effects. Hence the ability to estimate M_z, the magnetic dipole, will depend on the ability to determine the angles τ and ξ accurately, whereas the eddy current coefficient, K, will be primarily dependent upon the accuracy of r. Although K effects θ, the change in cone angle is in the order of 10^{-5} degs. over the half orbit. One can conclude from Figs. 3-5 that the torque coefficients, M_z and K, can be adequately determined if τ and ξ can be estimated with an accuracy the order of .01° and r to within .0001 #-ft-s.

In the following three sections the results for the convergence and estimation accuracy for a typical spacecraft initial condition are presented.

FIGURE 3 EFFECTS OF MAGNETIC TORQUES ON ORIENTATION OF ANGULAR MOMENTUM.
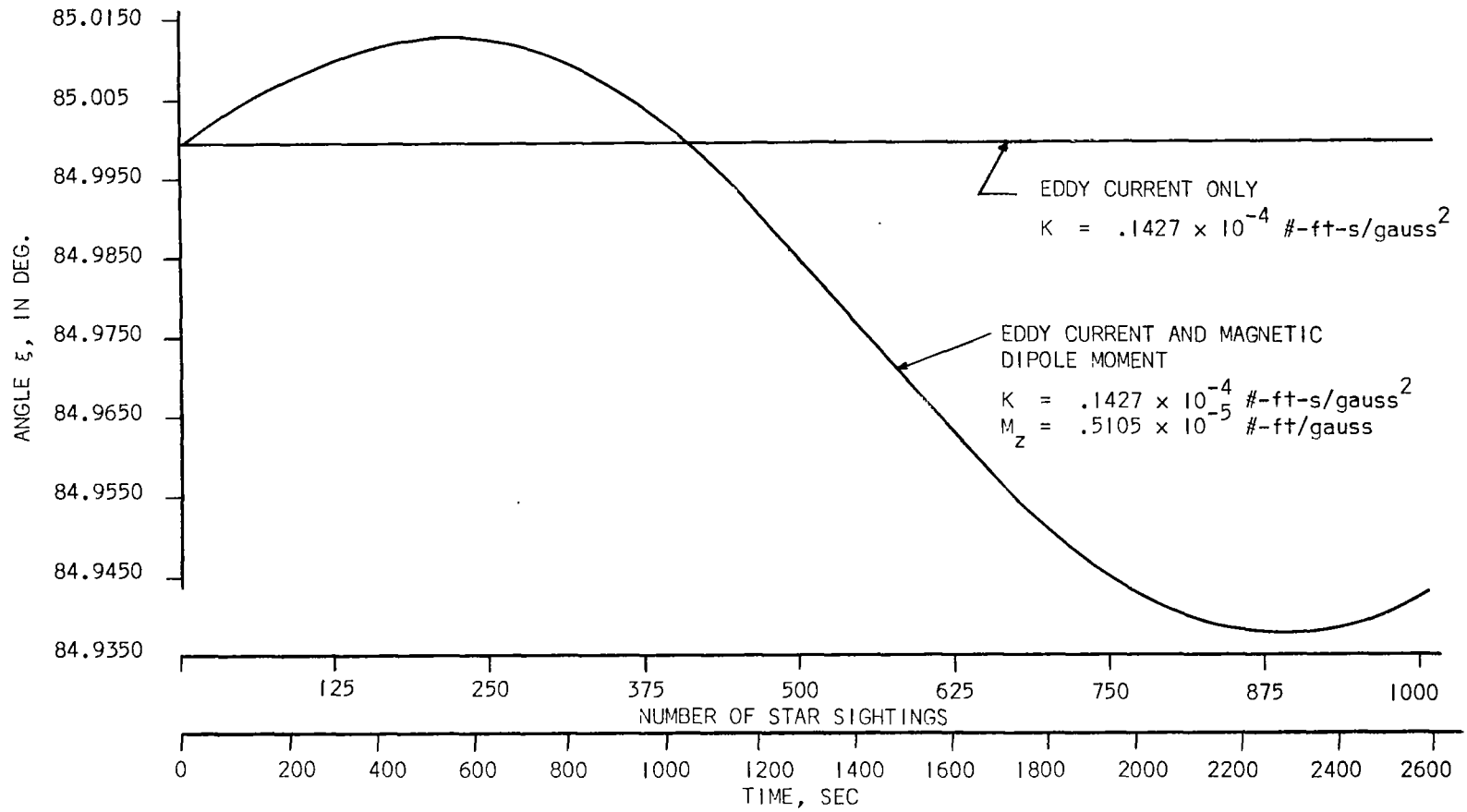
a) ANGLE ξ

EDDY CURRENT ONLY

$K = .1427 \times 10^{-4}$ #-ft-s/gauss^2

EDDY CURRENT AND MAGNETIC DIPOLE MOMENT

$K = .1427 \times 10^{-4}$ #-ft-s/gauss^2
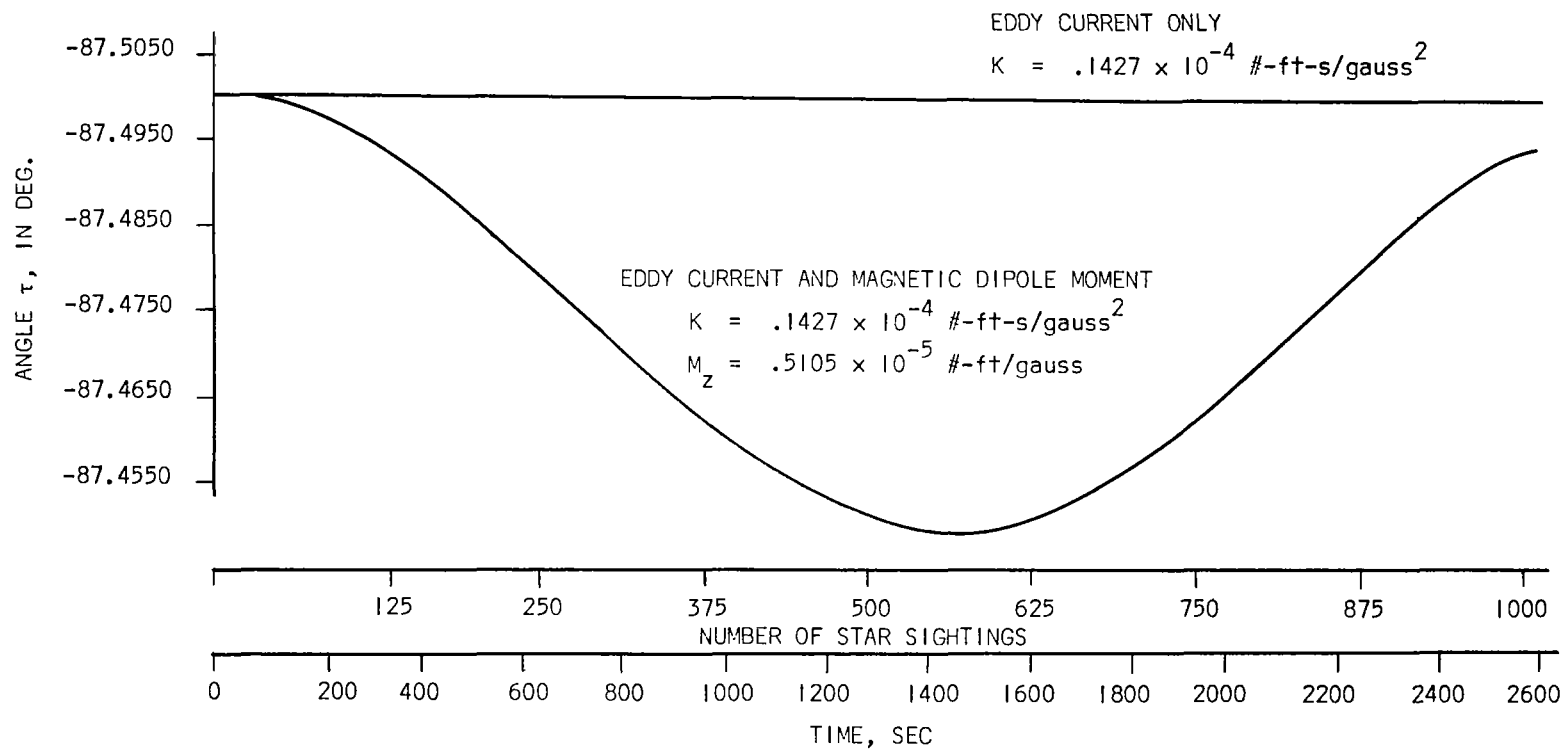$M_z = .5105 \times 10^{-5}$ #-ft/gauss

FIGURE 3 (CONCLUDED)

b) ANGLE τ

FIGURE 4 EFFECTS OF MAGNETIC TORQUE ON ANGULAR MOMENTUM MAGNITUDE.

26
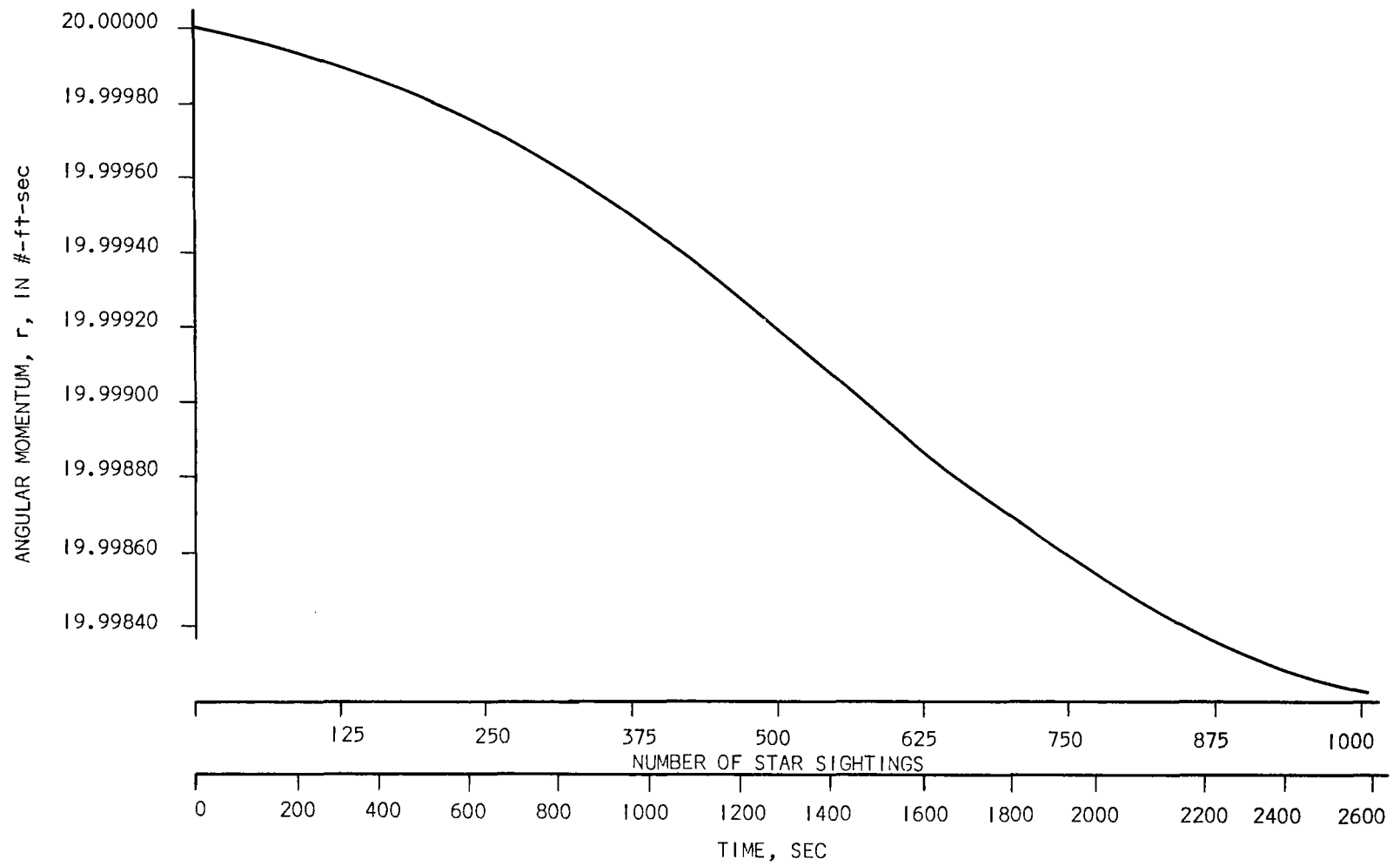
FIGURE 5 EFFECTS OF MAGNETIC TORQUE ON CONE ANGLE.
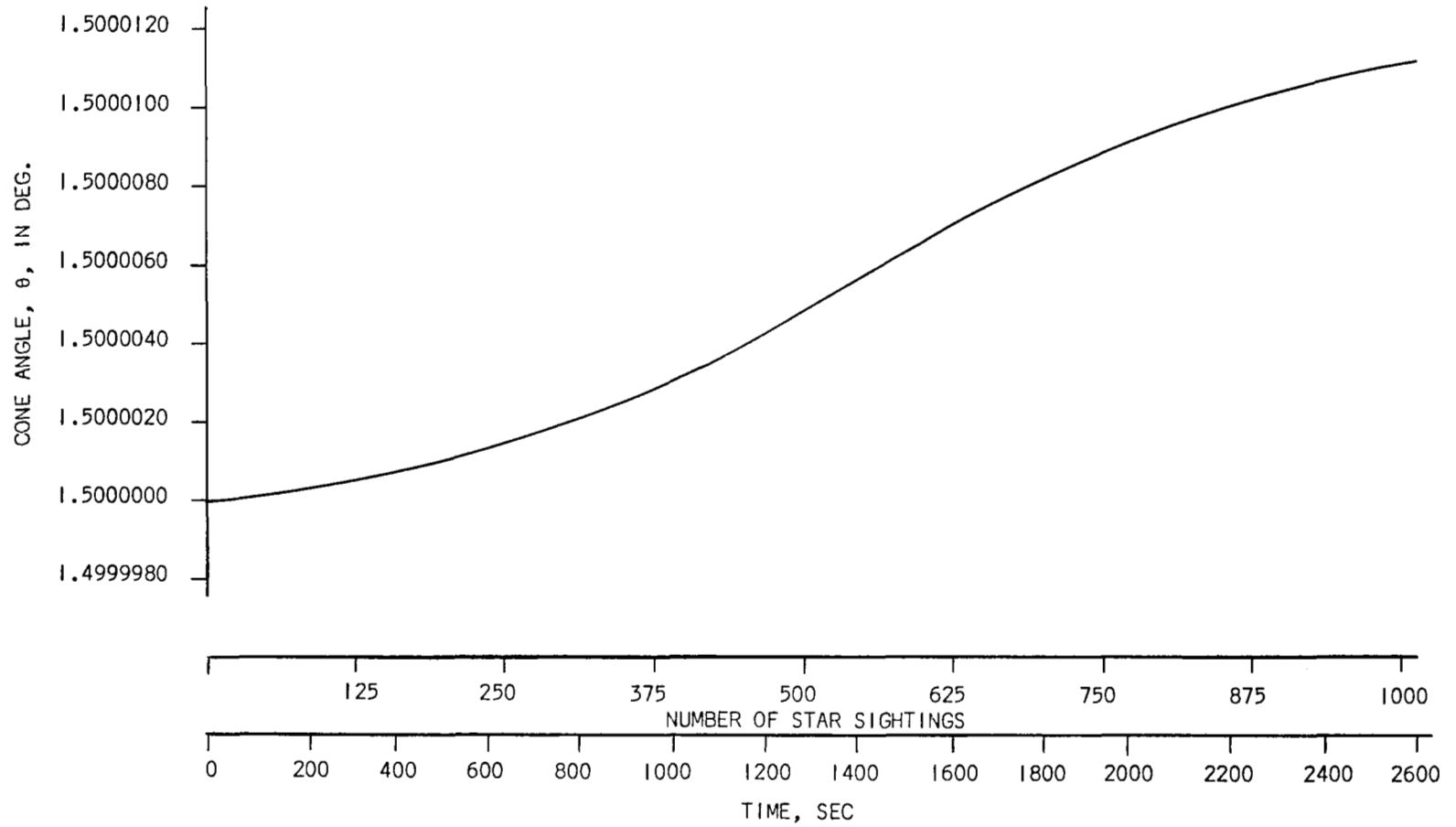
The various cases use identical noise added to the star time measurement in order that the capabilities of each system may be compared. The initial conditions are:

$$\phi(o) = 1°$$
$$\psi(o) = -1°$$
$$\theta(o) = 2°$$
$$\tau(o) = -80°$$
$$\xi(o) = 85°$$
$$r(o) = 21 \text{ #-ft-sec.}$$
$$K = .135 \times 10^{-4} \text{ #-ft-sec/gauss}^2$$

In both the nominal spacecraft and the estimation simulations the magnetic dipole, M_z, was zero.

The initial covariance matrix for each run was given as:

$$P = \begin{bmatrix} .004 & -.0008 & 0 & 0 & 0 & 0 & 0 \\ -.0008 & .004 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & .004 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & .004 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & .004 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & .16 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & .4 \times 10^{-7} \end{bmatrix}$$

While the capabilities of one particular form of filter might be emphasized if other initial conditions had been taken or might be changed somewhat if another sighting error sequence had been used, the comparison of filter capabilities based upon identical runs seems to be a reasonably valid procedure, at least initially to indicate the more gross effects of each filter system.

Limited Memory Filter Using Schmidt Gain

This section discusses the results of the attitude determination study using the Schmidt gain limited memory concept by presenting the convergence and accuracy obtained for relevant states of the attitude determination problem using the above conditions and various gain values, K_s.

Fig. 6 shows the effect of change in Schmidt gain on the accuracy of the estimate of vehicle orientation in the inertial X-Z plane. This factor is used since the nominal spin plane is within 10° of the X-Z plane so that

its accuracy is equivalent to the estimation of the spin orientation. This fact was also indicated numerically by the high degree of correlation between the X-Z plane error and the term, $(\phi - \hat{\phi} + \psi - \hat{\psi})$, which for small cone angles, θ, is also the nominal spin plane accuracy.

Fig. 6 illustrates both the speed at which the estimate of spin error converges and maintenance of accuracy for the various gain values. The standard nonlinear Kalman filter, $K_s = 0$, converges most rapidly with the larger values of K_s converging more slowly. However, once convergence has taken place, (between 100 and 200 stars depending on gain) then the affects of insensitivity can be noted. For example, the $K_s = 0$ error slowly increase after about 200 stars, relative to the $K_s = .01$. After 350 stars all Schmidt gains accuracies are better than the Kalman filter. However in all cases the Schmidt gain shows a degree of insensitivity since the estimate error remains biased in that all errors remain positive.

Similar results appear in some of the other state variables as illustrated in Figs. 7-10. In the case of angular momentum, Fig. 7, the value for Schmidt gain, $K_s = .01$, converges more rapidly than either larger values or $K_s = 0$. As previously all values of gain eventually exceed the accuracy of the $K_s = 0$ system. In considering the angular momentum direction angle, τ, shown in Fig. 8, all values of gain K_s improve both convergence and accuracy over the basic Kalman filter. Surprisingly, all Schmidt gain estimation errors are opposite for this state in sign from the $K_s = 0$ case. Again, $K_s = .01$ appears to give slightly superior results. Figs. 9 and 10 demonstrate the ability of the Schmidt gain system to determine the cone angle and correct value of eddy current torque coefficient, respectively. Again, in Fig. 9 the estimation error is opposite in sign and the gain value $K_s = .01$ provides slightly superior results. The results of Fig. 10 indicate the ability to adequately determine the eddy current torque coefficient. This fact could have been assumed from Fig. 6 since the accuracy of the angular momemtum estimate was about .00002 #-ft-sec whereas the torque affects, Fig. 4, were about .0001 #-ft-sec. One can assume from Fig. 7 that a fair measure of magnetic dipole torque, M_z, could be obtained since the estimation accuracy is .0005° whereas .002° is needed in order to separate the affect of M_z.

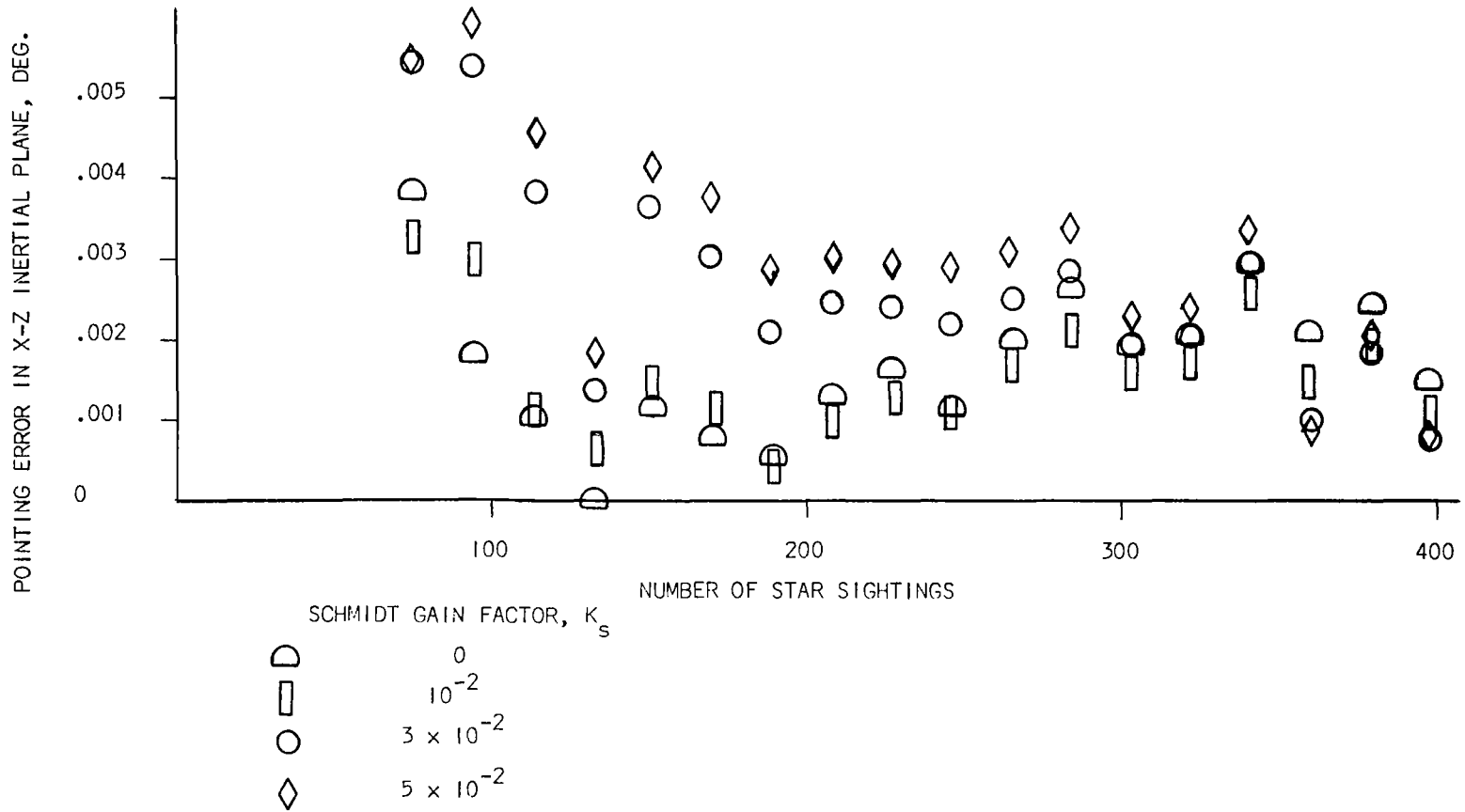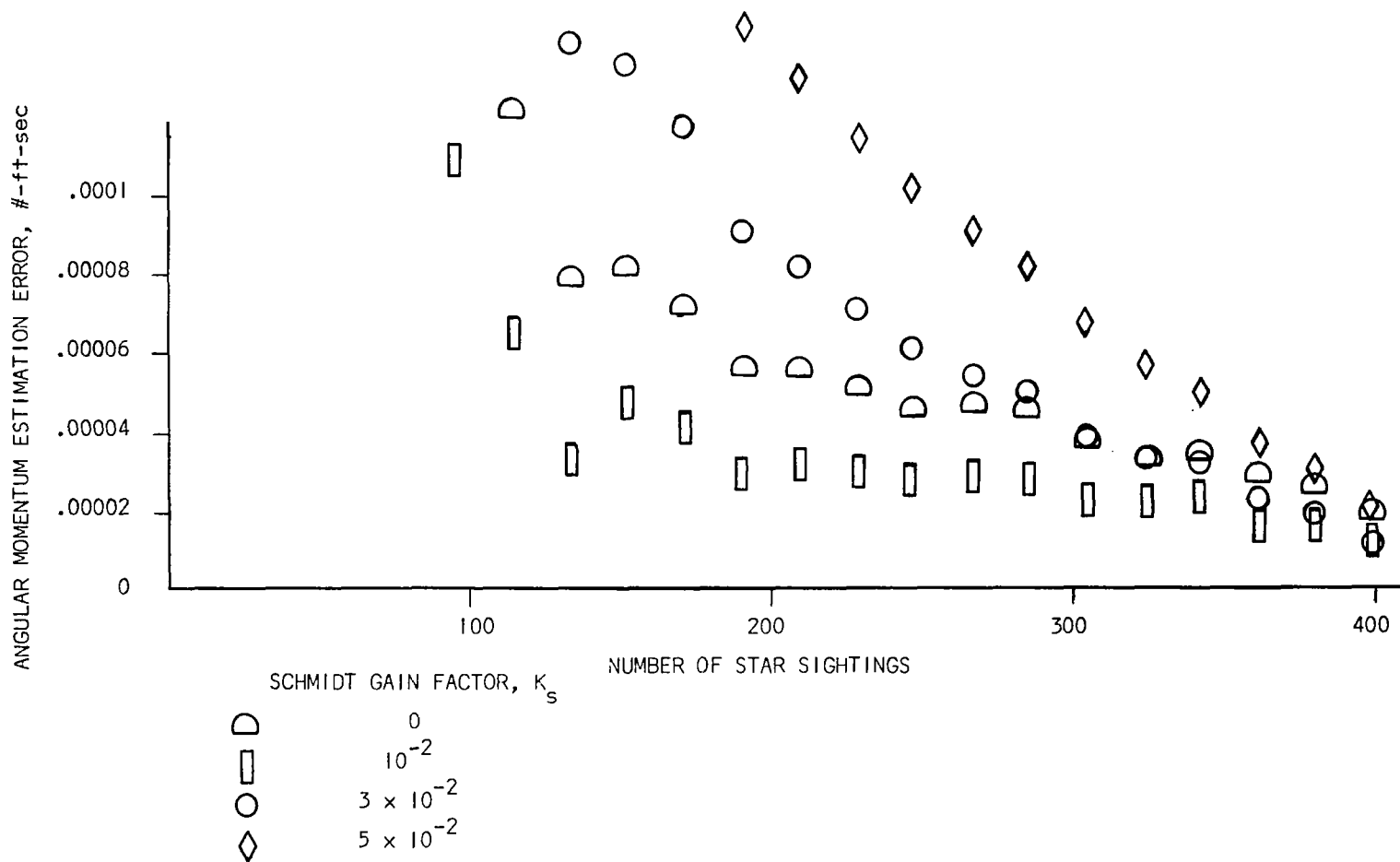FIGURE 6   SPIN ANGLE ESTIMATION ERROR FOR SCHMIDT GAIN LIMITED MEMORY.

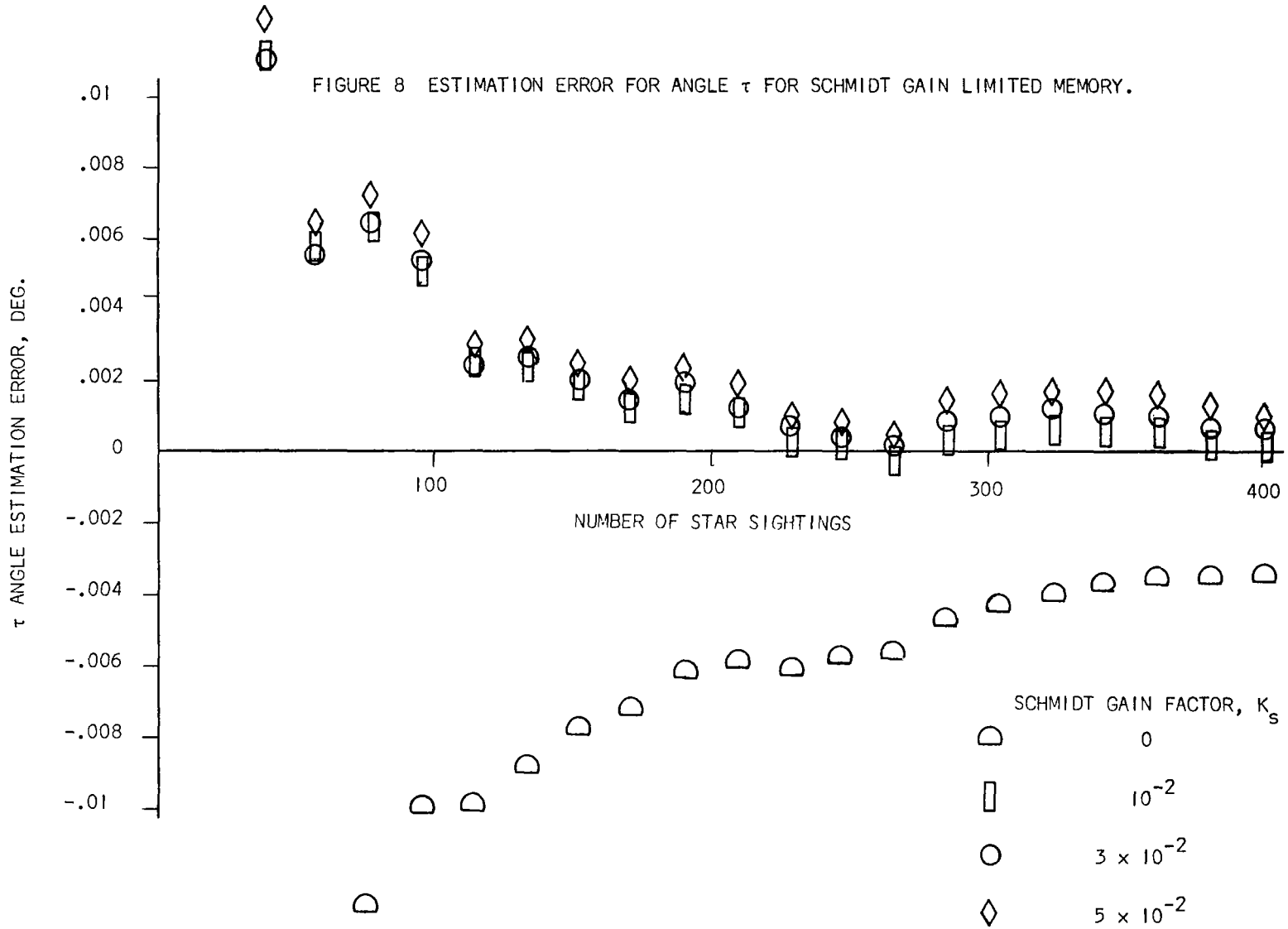FIGURE 7 ANGULAR MOMENTUM ESTIMATION ERROR FOR SCHMIDT GAIN LIMITED MEMORY.

FIGURE 8   ESTIMATION ERROR FOR ANGLE τ FOR SCHMIDT GAIN LIMITED MEMORY.

32

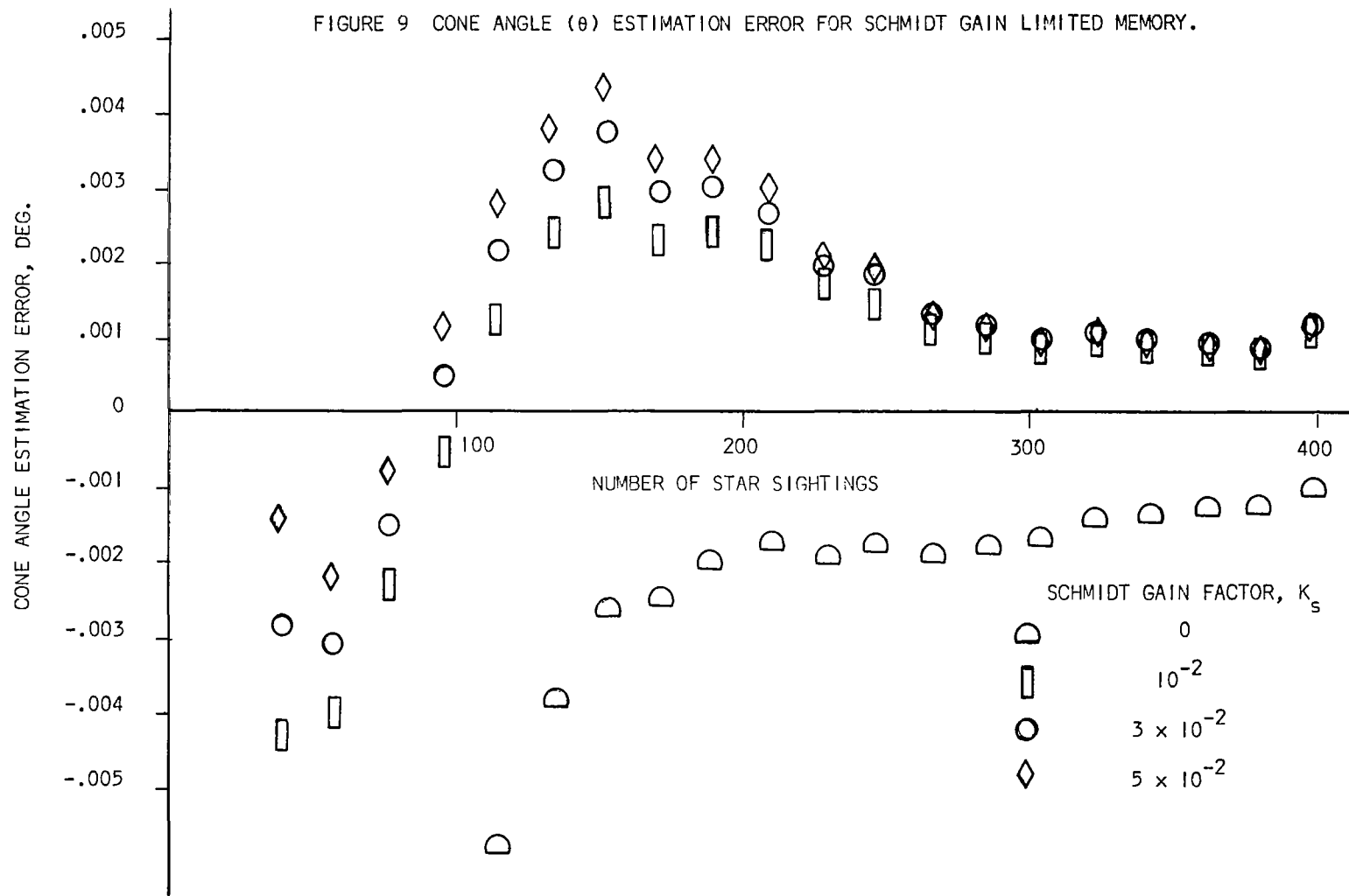FIGURE 9   CONE ANGLE (θ) ESTIMATION ERROR FOR SCHMIDT GAIN LIMITED MEMORY.
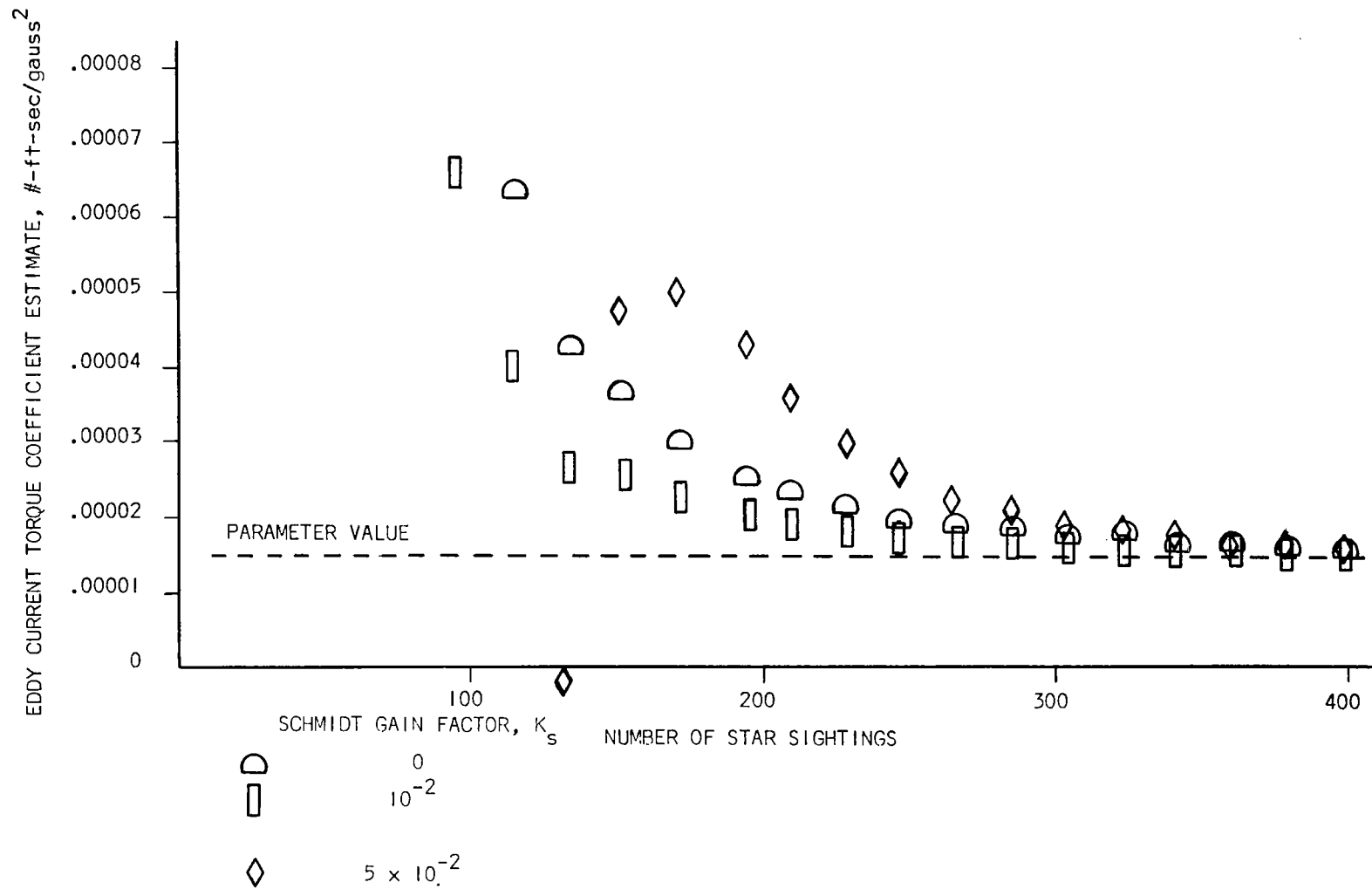
33

FIGURE 10 TORQUE COEFFICIENT ESTIMATE FOR SCHMIDT GAIN LIMITED MEMORY.

The·study results indicate that a nominal Schmidt gain of $K_s = .01$ will give adequate performance. However, it might be desirable to alter the gain factor, that is, maintain a low value during convergence and increase its value after a fairly accurate estimate has been obtained.

It should be noted that the results of Schmidt gain study indicate the estimation system tends to be nonoscillatory, that is, it does not reach a nominal zero mean error and then distribute the residual errors about the zero mean. This is probably due to insensitivity of the system to small errors. It can be concluded, however, that the Schmidt gain limited memory should be adequate for the attitude determination problem.

Limited Memory Using Noise Added to the Covariance Matrix

The second form of limited memory was to add noise to the covariance matrix to maintain a nominal value even when there is no comparable noise added to the system. The estimation accuracies for the same typical case as above for the noise aided limited memory filter cases are shown in Figs. 11-15.

The noise amplitude additions were selected by trial and error. High values for noise were selected, that is, those which gave an accurate but oscillatory converged estimates. These values were then reduced by an order of magnitude. The two estimates were then compared for various state errors in order to select a nominal (in between) value.

The noise was added to the $P^{1/2}$ equation in the form

$$P^{1/2} = P^{1/2} + W \tag{33}$$

with W a diagonal matrix; for the high noise W has elements

$$w_{11} = .4 \times 10^{-5}$$
$$w_{22} = .2 \times 10^{-5}$$
$$w_{33} = .1 \times 10^{-6}$$
$$w_{44} = .3 \times 10^{-6}$$
$$w_{55} = .1 \times 10^{-6}$$
$$w_{66} = .2 \times 10^{-7}$$
$$w_{77} = .7 \times 10^{-7}$$

35

Examination of Figs. 11-15 show that for the high noise the spin plane error, the angle τ, and the torque coefficient K, oscillate about the zero error with a rather large amplitude. The accuracy for cone angle seems adequate and could possibly be improved for angular momentum. As a result a nominal noise matrix

$$w_{11} = .7 \times 10^{-6}$$
$$w_{22} = .7 \times 10^{-6}$$
$$w_{33} = .1 \times 10^{-6}$$
$$w_{44} = .3 \times 10^{-7}$$
$$w_{55} = .3 \times 10^{-7}$$
$$w_{66} = .7 \times 10^{-7}$$
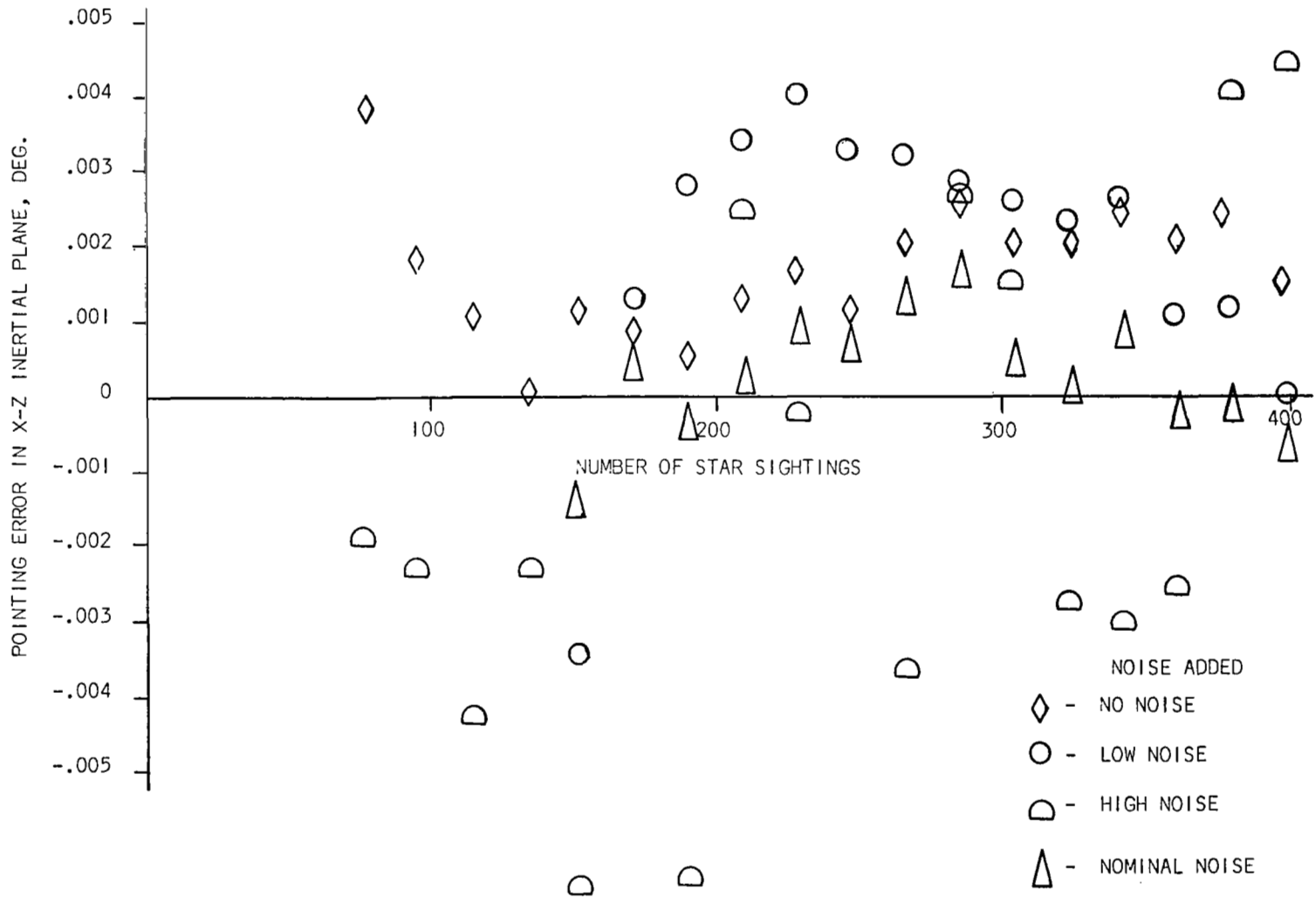$$w_{77} = .2 \times 10^{-7}$$

was selected.

The use of this latter set of noise values indicate excellent estimation of all the states illustrated in Figs. 11-15. Note that with the exception of τ, the estimations do not appear biased (as does the Kalman filter) in that errors tend to be distributed about zero. Note that the spin plane accuracy for the system is excellent, with errors after 300 stars remaining within 3 sec.

The undesirable feature of the noise addition appears to be its reduced rate of convergence. This is readily evident when comparing noise added results with the no noise Kalman filter typically as shown for spin plane accuracy in Fig. 11. Here, the high noise case did not attain an error less than .005 deg until after the 200th star.

Again the most useful situation might be to eliminate noise additions until after convergence has taken place. For example, noise might be added only after the 200th star. Also the initial data may be rerun after a good estimate has been obtained.

It can be concluded that with the proper value of noise added to the covariance matrix that an acceptable attitude determination system can be obtained. At present there does not seem to be a method, other than trial and error, for the selection of acceptable noise magnitude quantities.

FIGURE II   SPIN ANGLE ESTIMATION ERROR FOR NOISE AIDED LIMITED MEMORY.

FIGURE 12 ANGULAR MOMENTUM ESTIMATION ERROR FOR NOISE AIDED LIMITED MEMORY.

NOISE ADDED

◇ - NO NOISE

○ - LOW NOISE

⌓ - HIGH NOISE

△ - NOMINAL NOISE

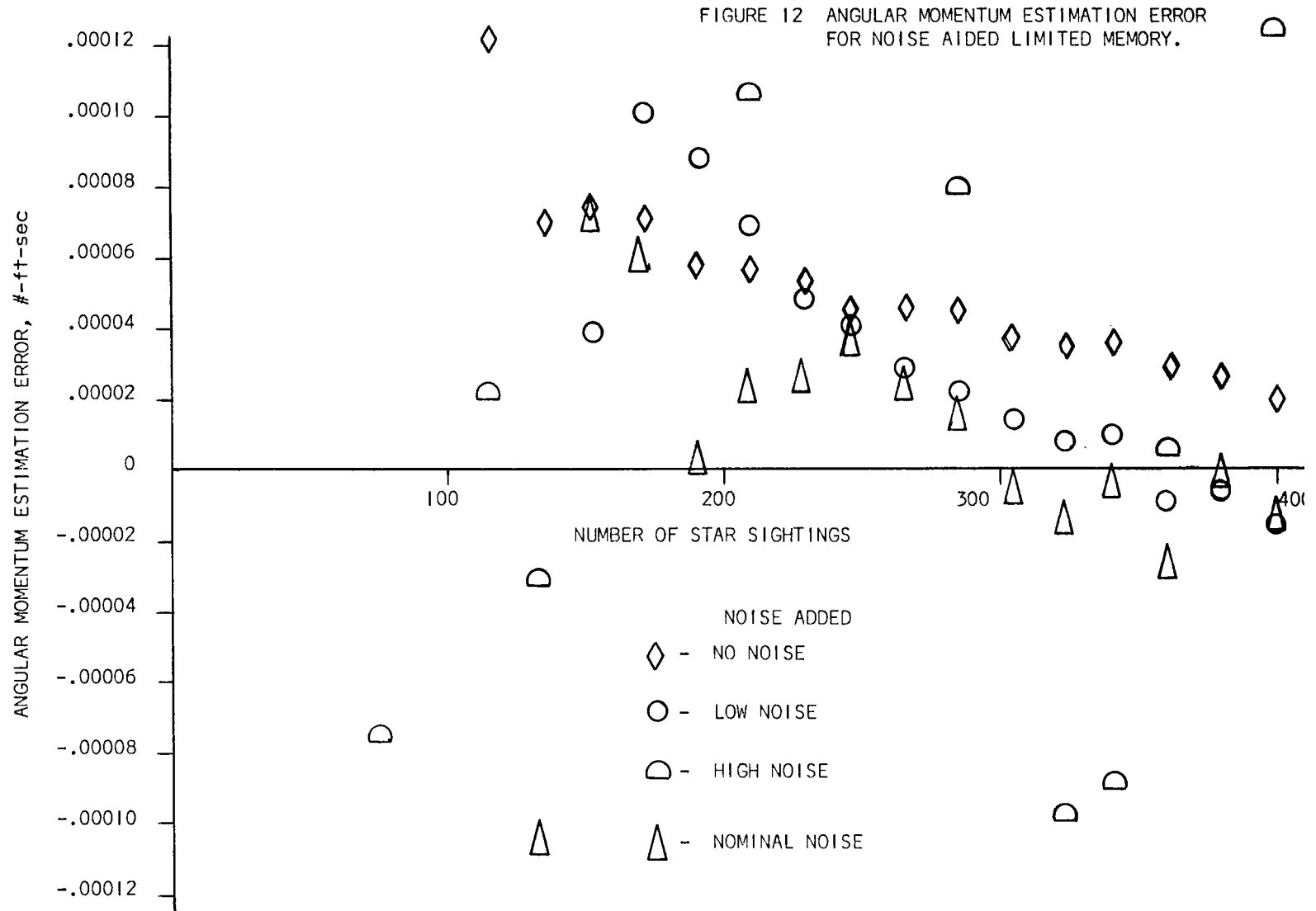NUMBER OF STAR SIGHTINGS

ANGULAR MOMENTUM ESTIMATION ERROR, #-ft-sec

38

FIGURE 13   ESTIMATION ERROR FOR ANGLE τ FOR NOISE AIDED LIMITED MEMORY.

NOISE ADDED

◇ - NO NOISE

○ - LOW NOISE

⌒ - HIGH NOISE

△ - NOMINAL NOISE

39

FIGURE 14 CONE ANGLE (θ) ESTIMATION ERROR FOR NOISE AIDED LIMITED MEMORY.

NOISE ADDED

◇ − NO NOISE
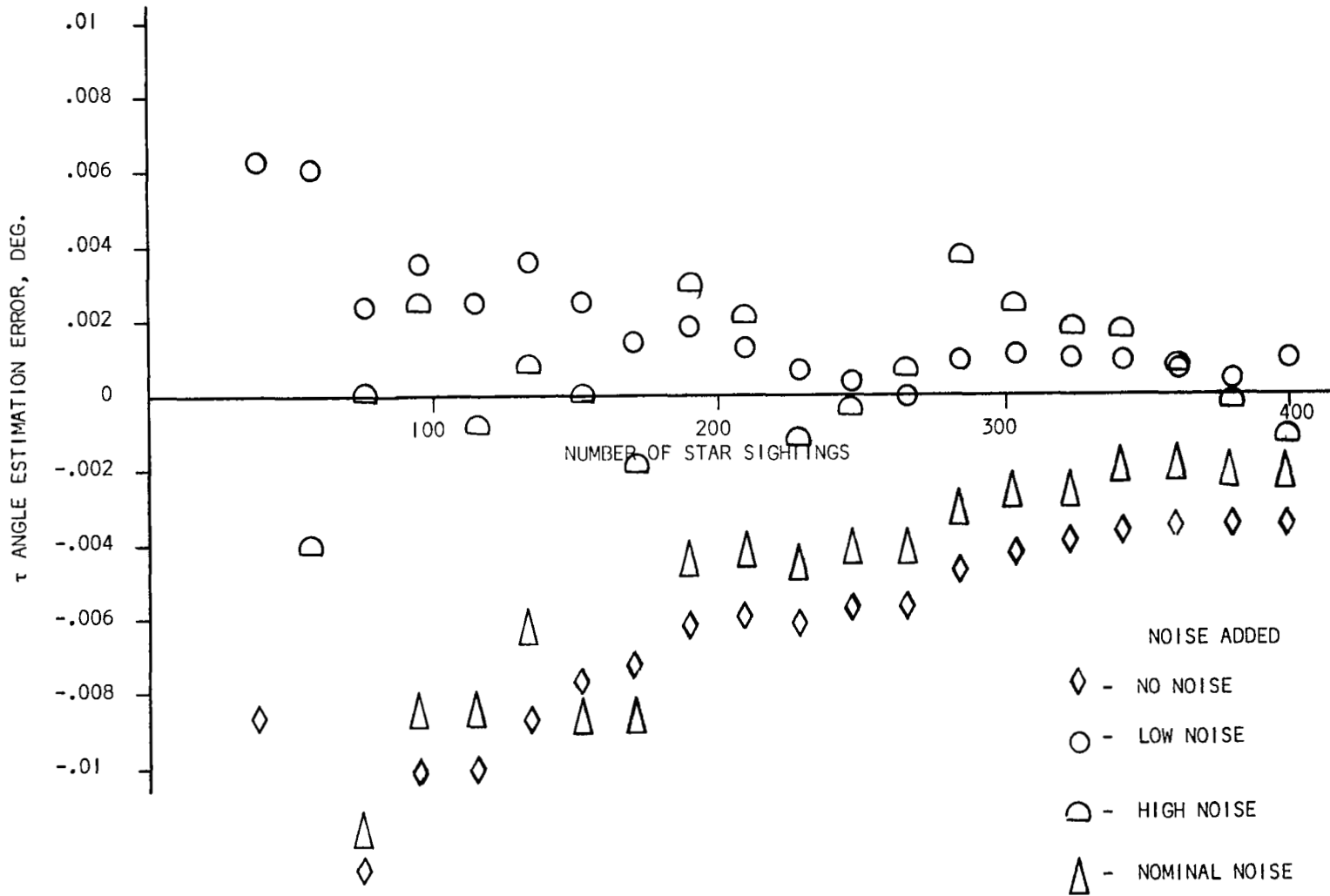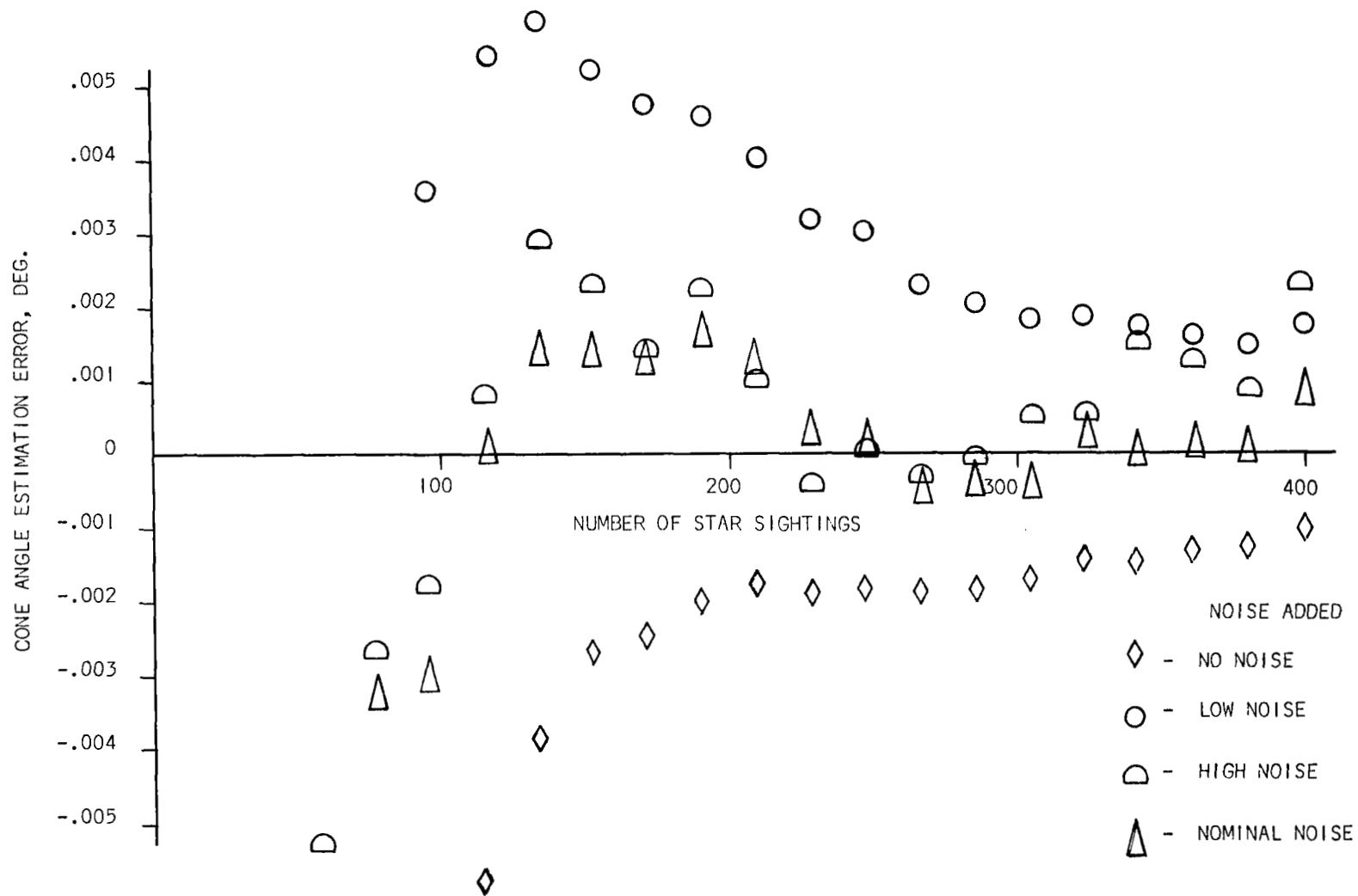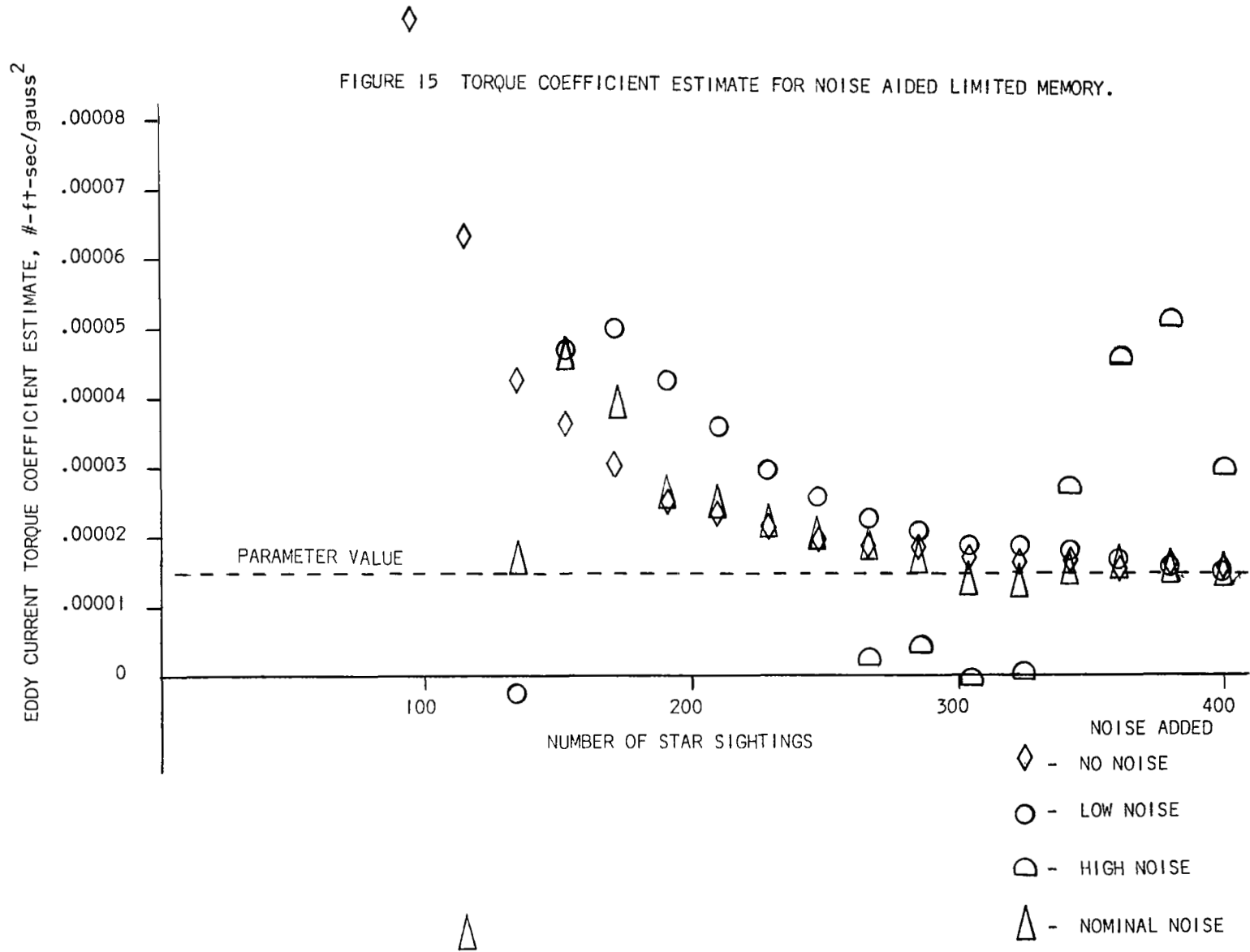
○ − LOW NOISE

⌂ − HIGH NOISE

△ − NOMINAL NOISE

40

FIGURE 15 TORQUE COEFFICIENT ESTIMATE FOR NOISE AIDED LIMITED MEMORY.

Least-Squares Limited Memory

In this section a comparison of the capability of least-squares limited memory filter considering various factors such as memory length and number of stars processed at each estimation point is made.

As noted in equation (17), it is feasible to incorporate within the filter additional measurements since the augmentation of the covariance matrix requires the integration of the state backwards in time to the qth measurement. Equation (17) considers the summation from time q+1 to p+1 but the incorporation of the qth measurement error (instead of q+1) does not seem to be a serious alteration of the least-squares procedure. (It should be noted that the qth measurement error is always available but the q+1 may not.) The positive weighting of the qth measurement is different from the limited memory scheme of Lee[4] who showed that this measurement error should be subtracted. While the computer program allows for any weighting to be used with the qth measurement all the results here treat it according to equation (17) like it were the q+1 measurement.

Actually this study did not investigate what method should be used to weight the terms of equation (17). When the model and system are identical equation (17) is correct but when model and true system differ, the method of weighting and for that matter the method for error comparison (i.e. what is meant by best fit) are no longer straightforward.

With these considerations in mind, Figs. 16-20 present the estimation errors for the same states as considered in previous sections. Considered are estimates with a memory length of 100 and 200 stars and an inclusion of either 5 or 8 stars within the memory band. These results are compared to the nonlinear Kalman filter. In most cases the results are substantially improved both in terms of more rapid convergence as well as better accuracy. For example, the system with memory length of 200 and 5 stars maintains a spin angle accuracy of 10 $\widehat{sec}$ after 75 star sightings and an angular momentum accuracy of .00002 #-ft-sec after 114 star sightings. This latter is a considerable improvement over the Kalman filter which barely reaches .00002 #-ft-sec at 399 sighting. These results are displayed con- sistently over the whole group of limited memory conditions with the

possible exception of cone angle. In addition the estimation errors tend to be more randomly distributed and not subject to the bias error displayed by the Kalman and Schmidt systems.

The system with 5 stars and 200 star memory length appears to give the best results. However, since a random selection of stars within the memory window was used, the results may differ depending upon the exact stars used. To show the true effects of differences in the number of stars and memory length a Monte Carlo simulation and a statistical analysis of the results would be required.

However, it is possible to show separately the effectiveness of processing additional stars at each measurement point in reducing the state error. This may be shown by considering the condition with measurement noise but with no data elimination from the covariance matrix. Essentially the system uses the Kalman weighting gain but uses more than the last measurements at each point.

Fig. 21 shows the effect of using one and 5 measurement stars across a 100 star length band on the precession angle error, ψ. Fig. 22 shows estimation errors for the angular momentum direction angle ξ for the same conditions as above except that two different measurement noise sequences have been included. Finally, Fig. 23 shows results similar to Fig. 22 but for cone angle, θ, and includes a condition where 12 stars are taken across a 100 star length band. In all these cases it can be seen that the convergence and accuracy considering additional measurements is greatly improved over the single measurement condition. The difference between 5 and 12 stars is not great, however. This result would indicate that the considerable improvement of the least-squares limited memory filter over the Kalman filter is due to the inclusion of additional measurements.

CONCLUDING REMARKS

A comparison study of the three limited memory filter systems, Schmidt gain, noise added, and least-squares, has been made. The study indicates

1. The three limited memory systems show a distinct improvement in accuracy after 400 stars over the Kalman filter when no system noise is present. Any of the filters would be feasible for use

43

FIGURE 16   SPIN ANGLE ESTIMATION ERROR FOR LEAST SQUARES LIMITED MEMORY.
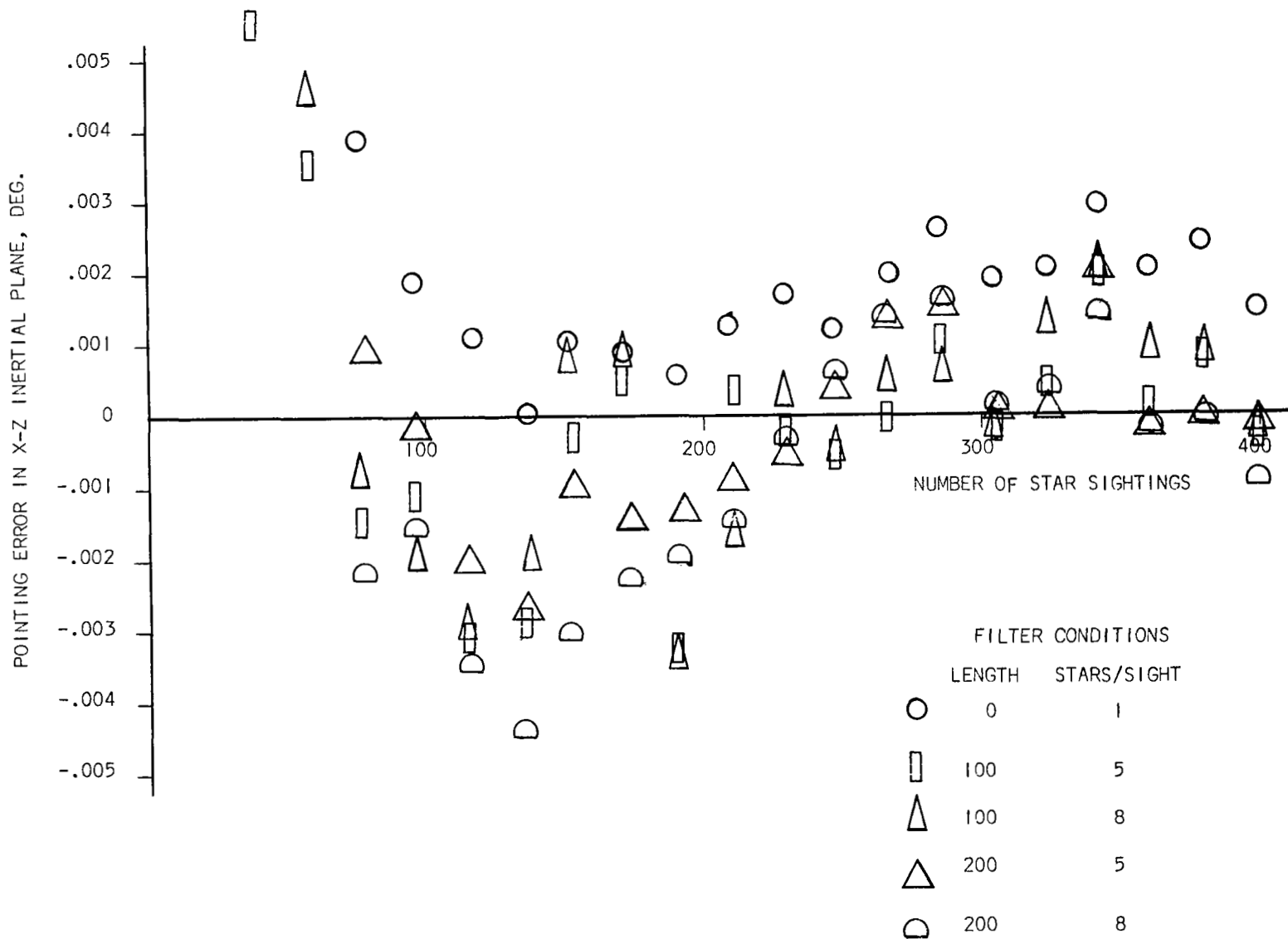
FIGURE 17 ANGULAR MOMENTUM ESTIMATION ERROR

FOR LEAST SQUARES LIMITED MEMORY.
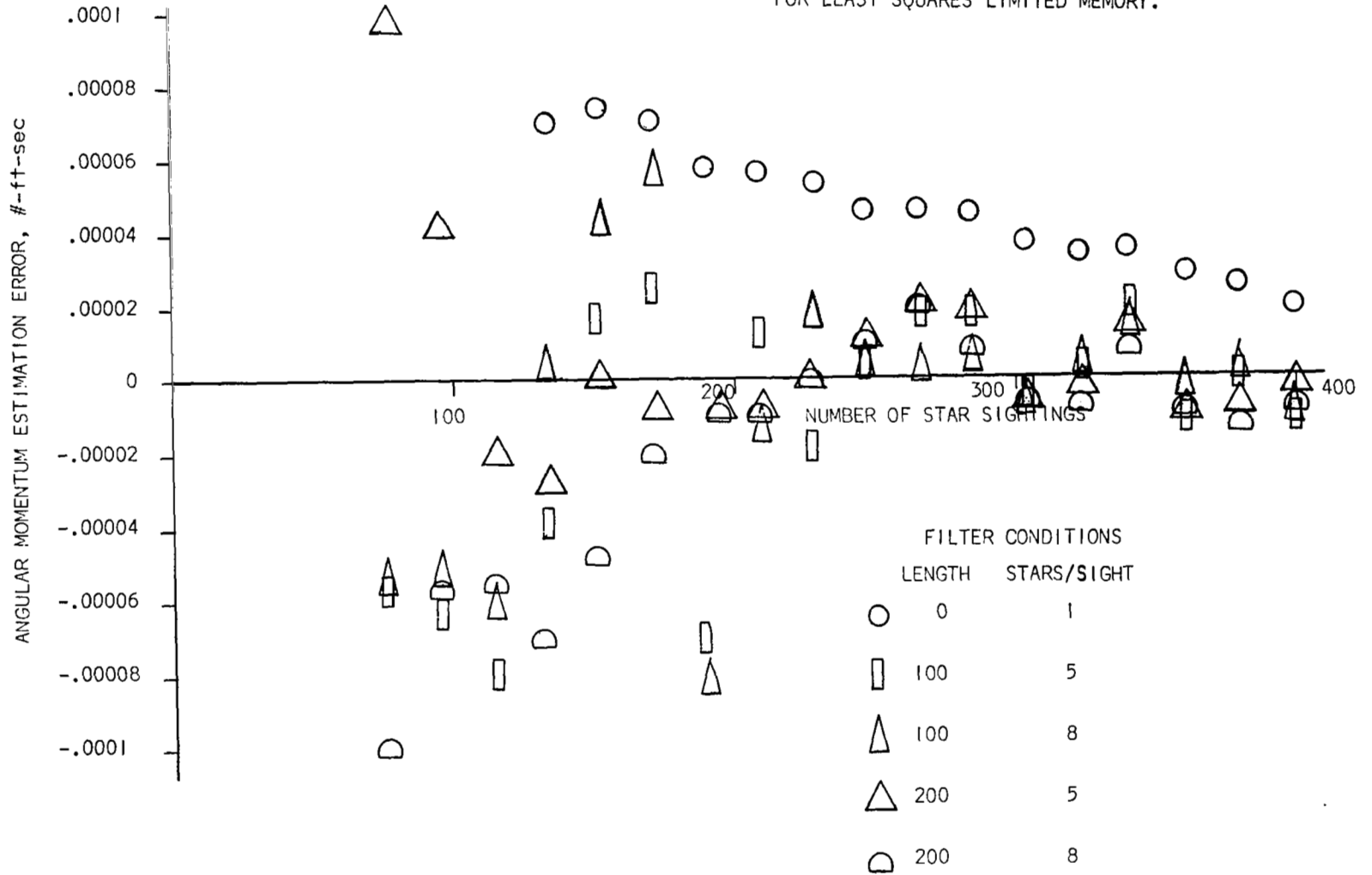
FIGURE 18   ESTIMATION ERROR FOR ANGLE τ FOR LEAST SQUARES LIMITED MEMORY.

46

FIGURE 19   CONE ANGLE (θ) ESTIMATION ERROR FOR LEAST SQUARES LIMITED MEMORY.

FILTER CONDITIONS

LENGTH	STARS/SIGHT
0	1
100	5
100	8
200	5
200	8

47
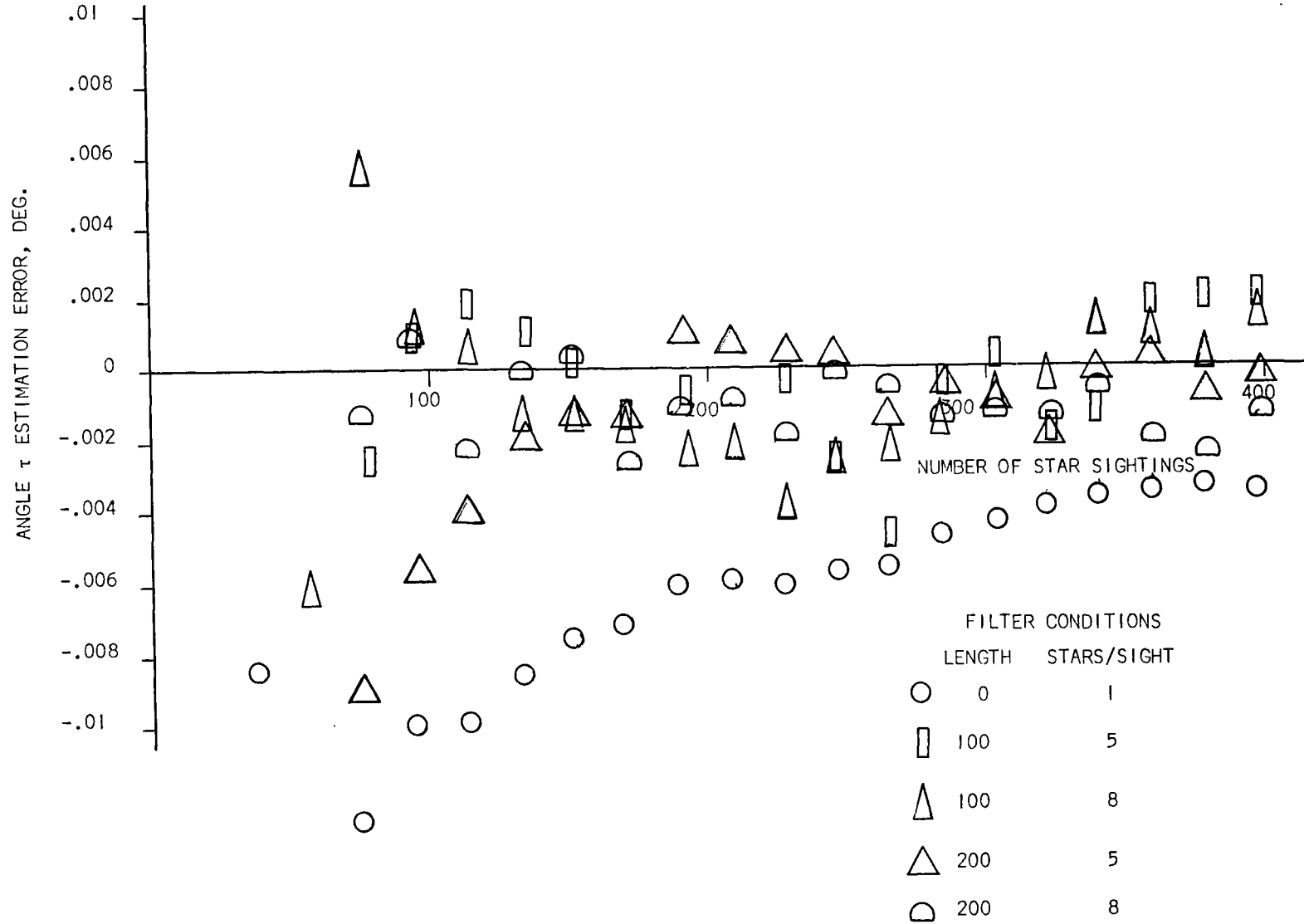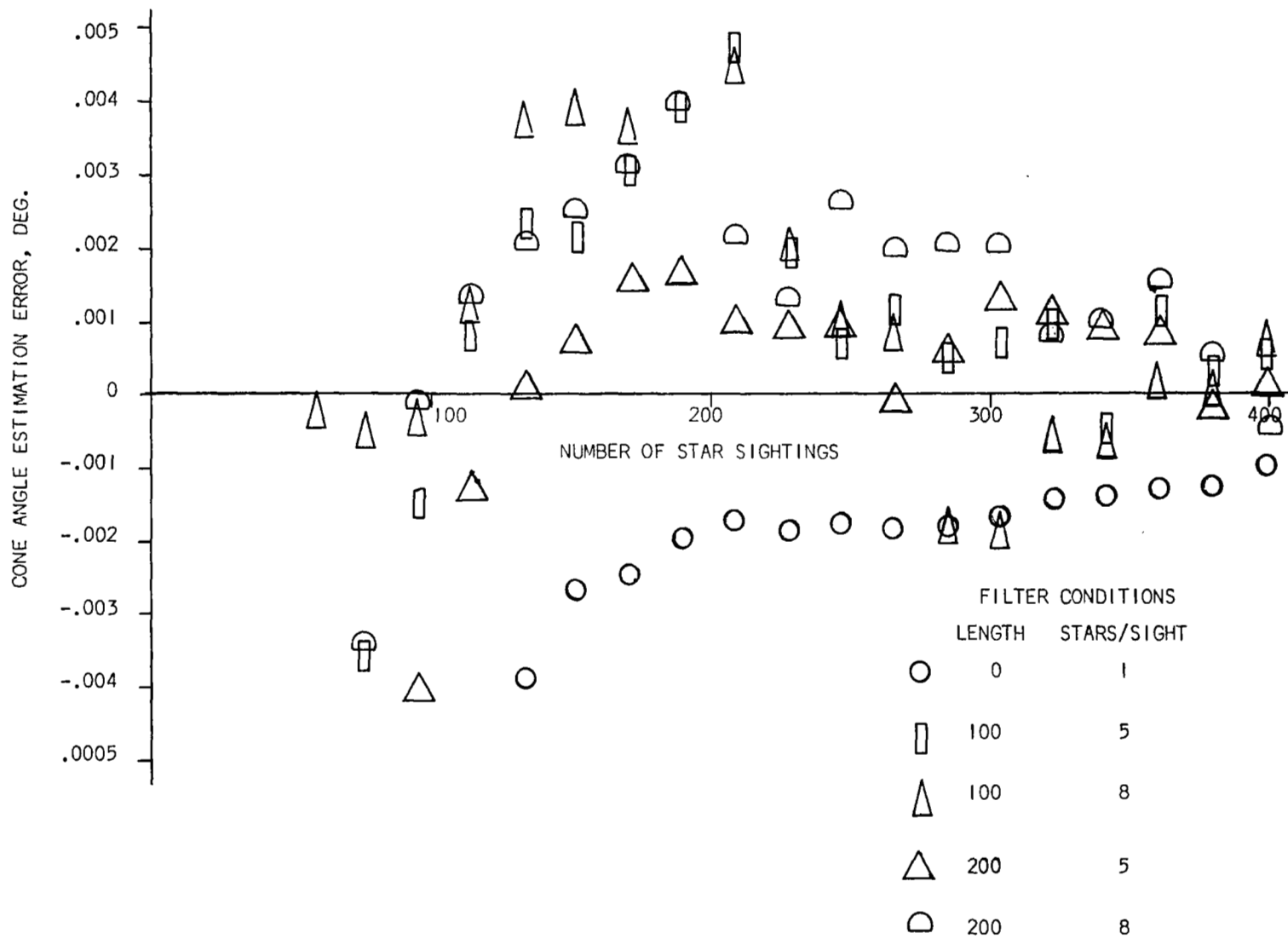
FIGURE 20 TORQUE COEFFICIENT ESTIMATE FOR LEAST SQUARES LIMITED MEMORY.
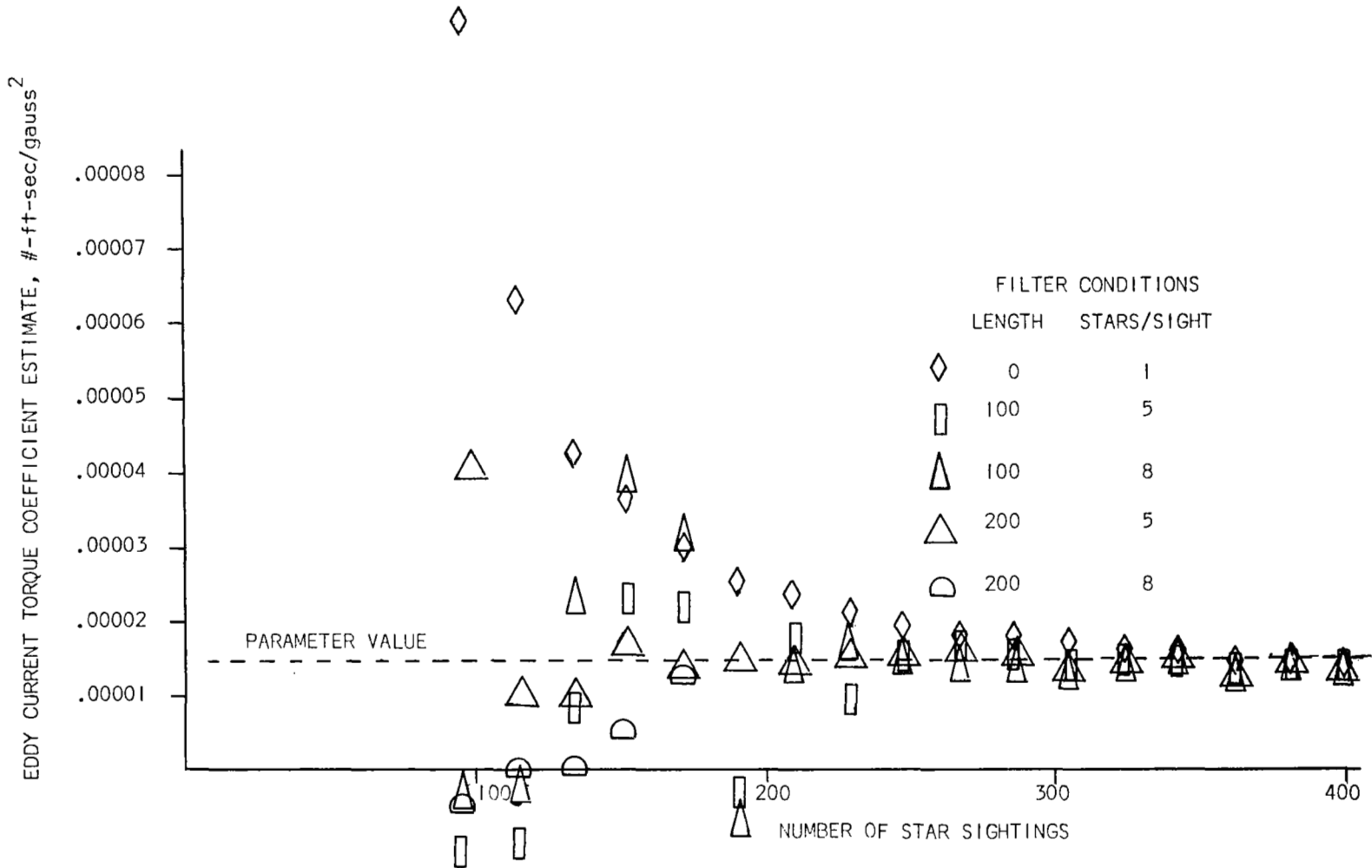
FIGURE 21 EFFECT OF ADDITIONAL MEASUREMENTS ON THE ESTIMATION ACCURACY OF

PRECESSION ANGLE (ψ).

FIGURE 22 EFFECT OF ADDITIONAL MEASUREMENTS ON THE ESTIMATION ACCURACY OF THE ANGLE ξ.
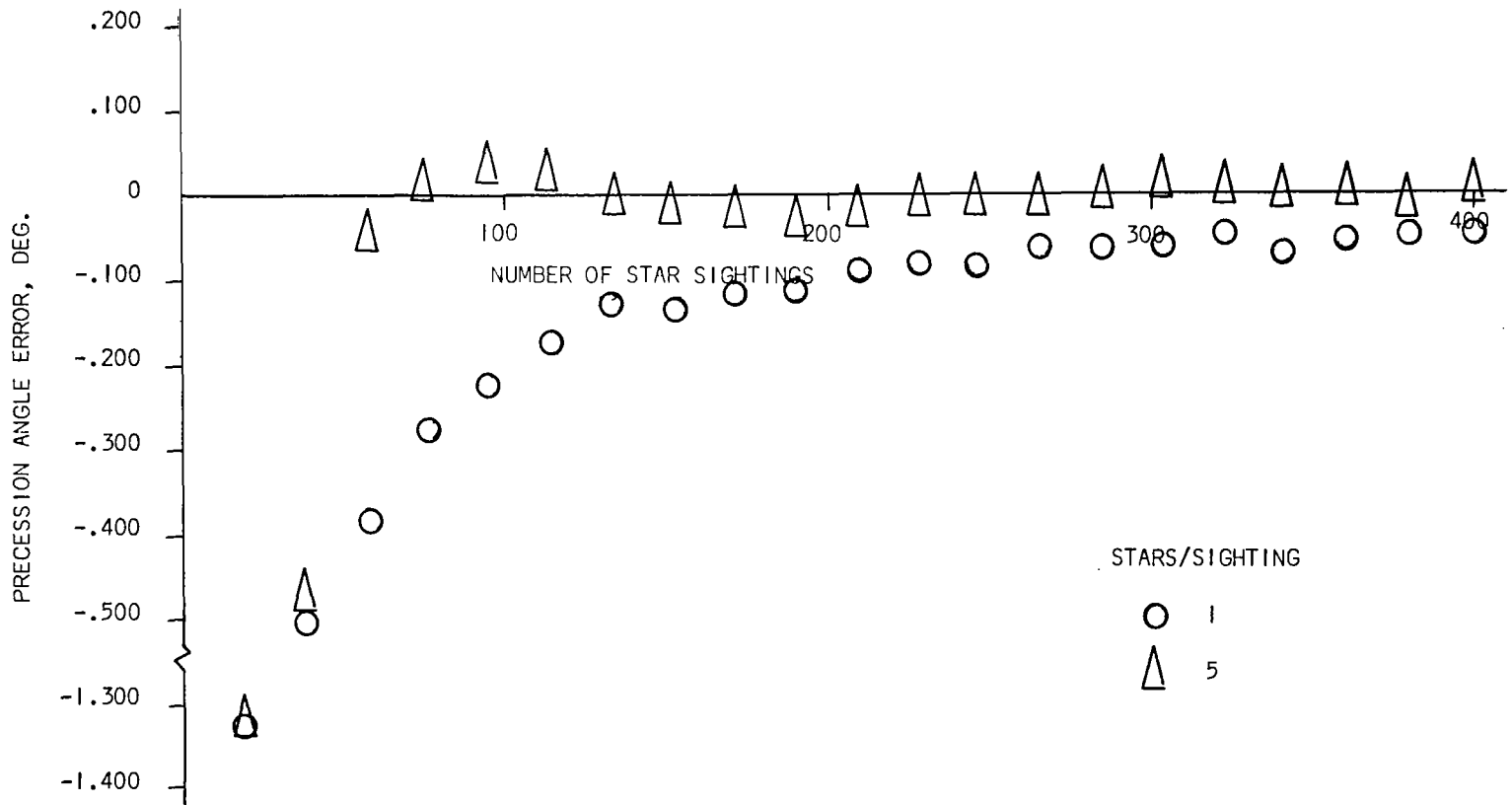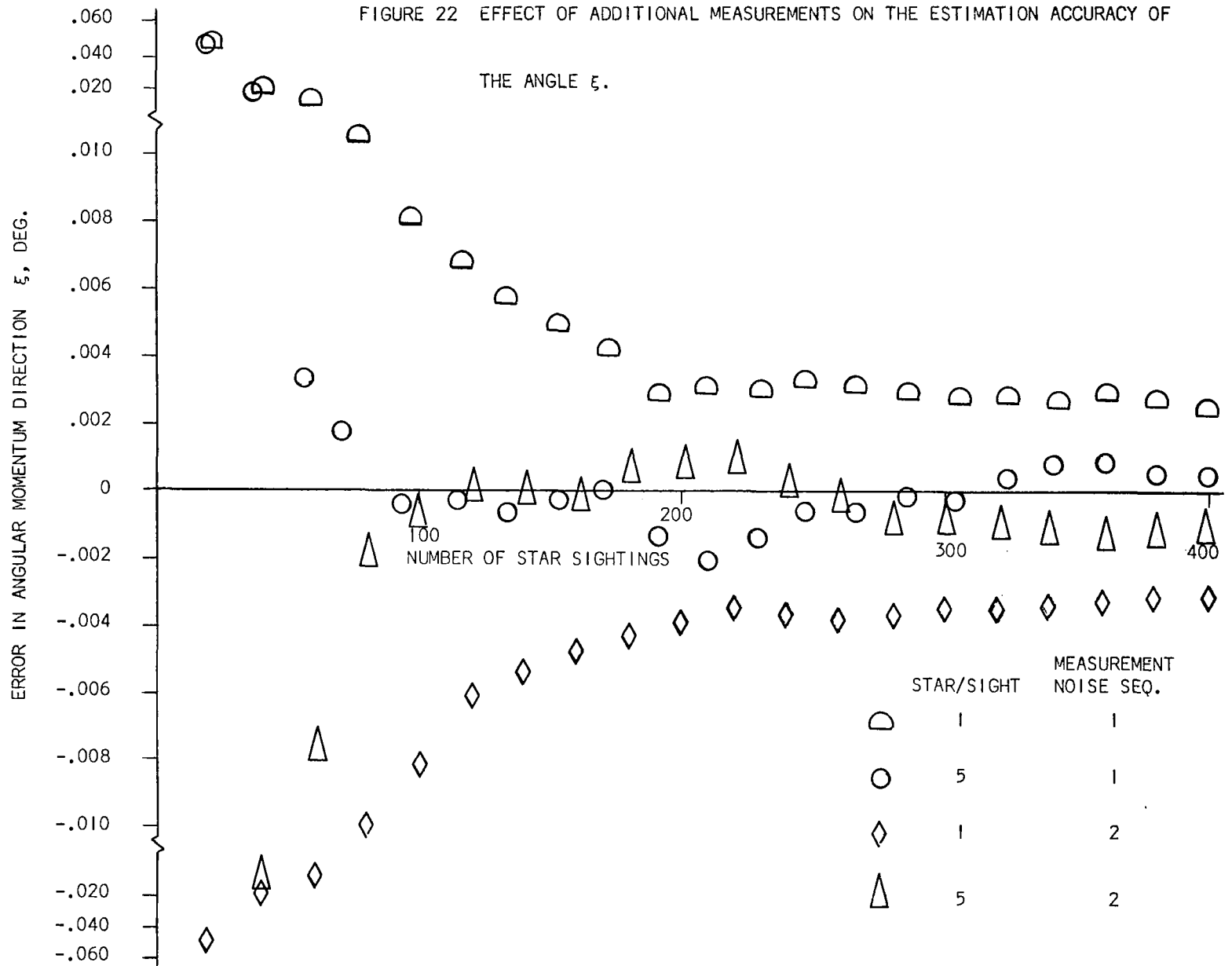
FIGURE 23   EFFECT OF ADDITIONAL MEASUREMENTS ON ESTIMATION ACCURACY OF CONE ANGLE (θ).
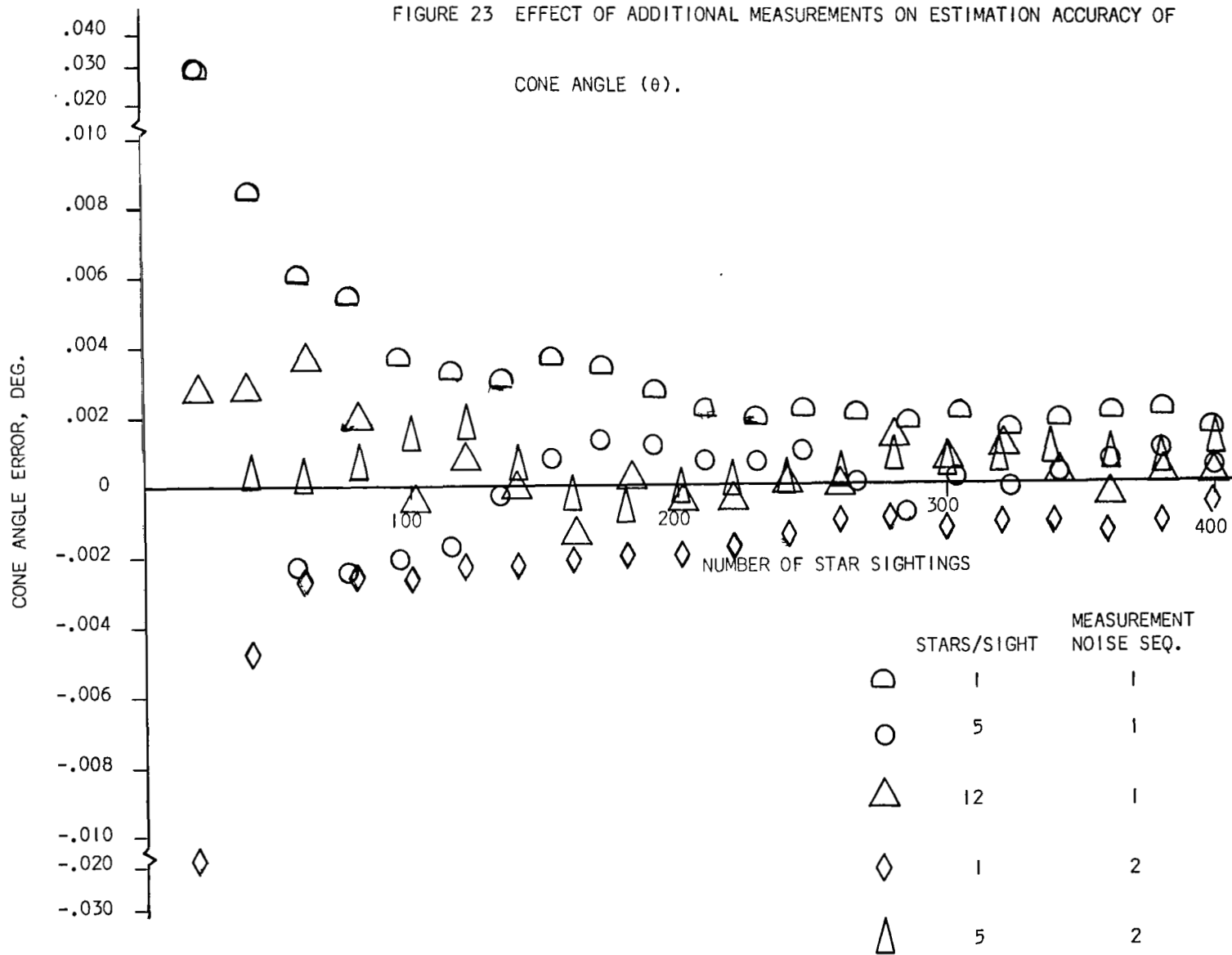
with a precision attitude determination system (10 $\widehat{sec}$ accuracy) assuming a basic star mapper precision of 10 - 30 $\widehat{sec}$.

2. The Schmidt gain limited memory exhibits a tendency to insensitivity in a manner similar to the nonlinear Kalman filter in that the state estimates appear biased and do not oscillate randomly about a zero mean error. The convergence of the Schmidt gain system is somewhat slower than that of the Kalman filter for some state variables. The gain value chosen appears to be a compromise between convergence and ultimate system accuracy.

3. The noise added limited memory filter exhibits an excellent sensitivity for the correctly selected values of noise effect added to the covariance matrix. However, trial and error must be used for the selection of the noise value and an incorrect selection can lead to either highly oscillatory or biased, low sensitivity state errors. The convergence of the noise added system is considerably worse than that of the Kalman filter for some state variables and again is dependent upon the noise values selected. An alternative to improve convergences might be to add noise in only after a sufficient number of stars have been processed.

4. The least-squares limited memory filter exhibits both excellent convergence and sensitivity for the various values of the memory length and the number of star measurements included. The converged error tends to oscillate about the true estimate with an accuracy of 3 - 6 $\widehat{sec}$ for spin error and is consistently within tolerance for the other states. The convergence is consistently better for most states than the Kalman filter. Results of studies on the convergence and accuracy indicate that the use of additional data points within the memory band contribute significant improvements over just processing the last data point.

52

APPENDIX I   MEASUREMENT AND GRADIENT EQUATIONS

The measurement system equations for the attitude determination
problem becomes quite complex and hence have been listed in this appendix
as opposed to the main body of the text.   There are also some simplifica-
tions which can be employed in some of the equations to reduce the compu-
tational difficulties.

MEASUREMENT AND GRADIENT EQUATIONS

The measurement equation (24) is a straightforward scalar product.
However, when the matrix multiplications are performed, some of the gradient
equations can be written in terms of the vector and matrix components of
the scalar product.   Defining

$$\underline{b} \;=\; \begin{pmatrix} b_1 \\ b_2 \\ b_3 \end{pmatrix} \;=\; T \, \underline{s}_i$$

and   (AI)

$$\underline{c} \;=\; \begin{pmatrix} c_1 \\ c_2 \\ c_3 \end{pmatrix} \;=\; E \, \underline{b}$$

as the transformed star vector in each frame, the gradient equations (25)
can be written as

$$\underline{a}_r^T \, \frac{\partial E}{\partial \phi} \, T \, \underline{s}_i \;=\; \underline{a}_r^T \begin{pmatrix} -E_{12}b_1 + E_{11}b_2 \\ -E_{22}b_1 + E_{21}b_2 \\ -E_{32}b_1 + E_{31}b_2 \end{pmatrix}$$

53

$$\underline{a}_r^T \frac{\partial E}{\partial \psi} T \underline{s}_i = \underline{a}_r^T \begin{pmatrix} c_2 \\ -c_1 \\ 0 \end{pmatrix}$$

$$\underline{a}_r^T \frac{\partial E}{\partial \theta} T \underline{s}_i = \underline{a}_r^T \begin{pmatrix} c_3 s\psi \\ c_3 c\psi \\ c\theta(b_1 s\phi - b_2 c\phi) - b_3 s\theta \end{pmatrix}$$

$$\underline{a}_r^T E \frac{\partial T}{\partial \tau} \underline{s}_i = \underline{a}_r^T \begin{pmatrix} -E_{11}b_3 + E_{13}b_1 \\ -E_{21}b_3 + E_{23}b_1 \\ -E_{31}b_3 + E_{33}b_1 \end{pmatrix} \qquad \text{(A2)}$$

$$\underline{a}_r^T E \frac{\partial T}{\partial \xi} \underline{s}_i = \underline{a}_r^T E \begin{pmatrix} -s_1 T_{12} + s_2 T_{11} \\ -s_1 T_{22} + s_2 T_{21} \\ -s_1 T_{32} + s_2 T_{31} \end{pmatrix}$$

where E_{ij} = ij the component of E

$\qquad\qquad T_{ij}$ = ij the component of T

$\qquad\qquad s_j$ = jth component of $\underline{s}_i$

The differential equation for calculating the state transition matrix uses the Jacobian matrix given by

$$G_{ij} = \frac{\partial g_i}{\partial x_j}$$

Using equation (26), the terms are

$$G_{i1} = G_{i2} = 0 \qquad i = 1, \ldots, n$$

$$G_{13} = \frac{-K b B_y B_z \sin 2\theta \cot \tau}{L}$$

$$G_{23} = \frac{-b \, r \sin \theta}{L}$$

54

$$G_{33} = \frac{-b\ K \cos 2\theta\ (B^2 + B_z^2)}{2L} = \frac{2\dot{\theta}\ \cos 2\theta}{\sin 2\theta}$$

$$G_{43} = \frac{-b\ K\ B_x\ B_z \sin 2\theta}{L} + \frac{B_y\ M_z \sin \theta}{r}$$

$$G_{53} = \frac{-b\ K\ B_y\ B_z \sin 2\theta}{L \sin \tau} - \frac{B_x\ M_z \sin \theta}{r \sin \tau} \tag{A3}$$

$$G_{63} = \frac{-r\ K\ b\ (B_z^2 - B^2) \sin 2\theta}{L}$$

$$G_{i3} = 0 \qquad i = 7, \ldots, n$$

In taking the $\frac{\partial g_i}{\partial x_j}$ for τ and ξ it is necessary to take into consideration that B_x, B_y, and B_z are functions of these variables. Hence

$$G_{14} = \frac{K(1 + b \cos^2\theta)(B_x B_y \cot \tau - B_y B_z \csc^2 \tau)}{L}$$
$$- \frac{M_z \cos \theta}{r} (B_z \cot \tau + B_x \csc^2 \tau + B_x)$$

$$G_{24} = \frac{B_x\ M_z}{r}$$

$$G_{34} = \frac{b\ K \sin 2\theta\ B_x\ B_z}{2L} \tag{A4}$$

$$G_{44} = \frac{K(1 + b \cos^2\theta)(B_x^2 - B_z^2)}{L}$$

$$G_{54} = \frac{K(1 + b \cos^2\theta)}{L} \left(\frac{B_x B_y}{\sin \tau} - B_y B_z \csc \tau \cot \tau \right)$$
$$- \frac{M_z \cos \theta}{r} \{B_z \csc \tau + B_x \csc \tau \cot \tau\}$$

55

$$G_{64} = \frac{2\, r\, b(1 + k \cos^2 \theta)}{L} (B_x B_z)$$

$$G_{i4} = 0 \qquad i = 7, \ldots, n$$

$$G_{15} = \frac{K(1 + b \cos^2 \theta)}{L} \cot \tau \{(B_y^2 - B_z^2) \sin \tau$$

$$- B_x B_z \cos \tau\}$$

$$+ \frac{B_y M_z \cos \theta}{r} (\cos \tau \cot \tau - \sin \tau)$$

$$G_{25} = \frac{B_y M_z \sin \tau}{r}$$

$$G_{35} = \frac{K\, b \sin 2\theta\, B_y B_z \sin \tau}{2L}$$

$$G_{45} = \frac{K(1 + b \cos^2 \theta)}{L} \{B_y(B_x \sin \tau + B_z \cos \tau)\}$$

$$+ \frac{(B_z \sin \tau + B_x \cos \tau) M_z \cos \theta}{r}$$

$$G_{55} = \frac{K(1 + b \cos^2 \theta)}{L \sin \tau} \{(B_y^2 - B_z^2) \sin \tau - B_x B_z \cos \tau\}$$

$$+ \frac{B_y M_z \cos \theta \cos \tau}{r}$$

$$G_{65} = \frac{2\, r\, K(1 + b \cos^2 \theta)}{L} B_y B_z \sin \tau$$

$$G_{i5} = 0 \qquad i = 7, \ldots, n$$

56

$$G_{16} = \frac{1}{L} - \frac{B_x M_z \cos\theta \cot\tau}{r^2} + \frac{B_z M_z \cos\theta}{r^2}$$

$$G_{26} = \frac{k \cos\theta}{L} - \frac{B_z M_z}{r^2}$$

$$G_{36} = 0$$

$$G_{46} = + \frac{B_y M_z \cos\theta}{r^2}$$

$$G_{56} = - \frac{B_x M_z \cos\theta}{r^2 \sin\tau}$$

$$G_{66} = \frac{K(B_z^2 - B^2)(1 + b\cos^2\theta)}{L}$$

$$G_{i6} = 0 \qquad i = 7, \ldots, n$$

Adding the parameters K, M_z, and b give

$$G_{17} = \frac{B_y B_z (1 + b\cos^2\theta)\cot\tau}{L}$$

$$G_{27} = 0$$

$$G_{37} = - \frac{b \sin 2\theta (B^2 + B_z^2)}{4L}$$

(A5)

$$G_{47} = \frac{B_x B_z (1 + b\cos^2\theta)}{L}$$

$$G_{57} = \frac{B_y B_z (1 + b\cos^2\theta)}{L \sin\tau}$$

$$G_{67} = \frac{r(B_z^2 - B^2)(1 + b \cos^2\theta)}{L}$$

$$G_{17} = 0 \qquad i = 7, \ldots n$$

$$G_{18} = \frac{B_x \cos\theta \cot\tau}{r} - \frac{B_z \cos\theta}{r}$$

$$G_{28} = \frac{B_z}{r}$$

$$G_{38} = 0$$

$$G_{48} = \frac{B_y \cos\theta}{r}$$

$$G_{58} = \frac{B_x \cos\theta}{r \sin\tau}$$

$$G_{i8} = 0 \qquad i = 6, \ldots, n$$

$$G_{19} = \frac{K B_y B_z \cos^2\theta \cot\tau}{L}$$

$$G_{29} = \frac{r \cos\theta}{L}$$

$$G_{39} = - \frac{K \sin 2\theta (B^2 + B_z^2)}{4L}$$

$$G_{49} = \frac{K B_x B_z \cos^2\theta}{L}$$

$$G_{59} = \frac{K B_y B_z \cos^2\theta}{L \sin\tau}$$

58

$$G_{69} = \frac{r \ K(B_z^2 - B^2) \cos^2\theta}{L}$$

$$G_{i9} = 0 \qquad i = 7, \ldots, n$$

For the instrument simulation runs, both K and M_z could be set equal to a nonzero value. For the estimation program, $M_z = 0$ was assumed. Hence the above equations were used with n = 7 and where applicable the M_z term eliminated.

PROGRAM FOR DETEMINATION OF STAR SIGHTINGS AND PREPARATION OF
SIGHTING TAPES FOR UNIVERSAL FILTER PROGRAM

```
      DIMENSION BMG(3),TMAT(3,3),BMAG(3),SVEC(3,6),AEVEC(3),AETVE(3),
     1VAL(6),SIGMA(6),GAMA(6),DER1(4),DER2(2),ANS(6),ERTIA(3),CVAL(6),
     2SVAL(6),IR(2),SMAG(6),ET(3,3),SLIT(3,3),ALAMD(3),ALPH1(3),CALAM(3)
     3,SALAM(3)
      DOUBLE PRECISION VAL,TO,T,DER2,ANS
      COMMON VAL,DER1,DER2,DELTA,T,ANS

      DATA INPUT,CONSTANTS,INITIAL STATES(VAL),CONTROL NUMBERS,ETC.,
      STAR ANGLES(SIGMA,GAMA),MAGNITUDES(SMAG),SLIT ANGLES(ALAMD,ALPH1),
      ETC.

      DATA C2,WO,VIE,VNO,ERTIA,T,M,INTER,C3/ .375,.001106,97.38,0.0,
     156.68,56.68,65.62,0.0,1,1,.333/,NAT,LGAUS,ISLIT/1,2,1/
      READ(5,100)VAL,SIGMA,GAMA,SMAG
      READ(5,102)ALAMD,ALPH1
      READ(5,101)C1,AMAG,VAR,KSTOP,ITST,IR,ITSL,K
      WRITE(6,203)

      TAPE INITIALIZATION

      REWIND 4
      READ(5,103) KAPPA,A1,A2,A3,A4,A5,A6
      WRITE(4,201)KAPPA,A1,A2,A3,A4,A5,A6,VAL

      PARAMETER INITIALIZATION

      IT = 0
      CVIE = COS(VIE/57.29578)
      SVIE = SIN(VIE/57.29578)
      DO 26 I = 2,6
   26 VAL(I) =  VAL(I)/57.29578
      TMAT(2,3) = 0.0

      SLIT VECTOR COMPONENTS

      DO 3 I=1,ITSL
      CALAM(I)=COS(ALAMD(I)/57.29578)
      SALAM(I)=SIN(ALAMD(I)/57.29578)
      SLIT(I,1)=CALAM(I)*COS(ALPH1(I)/57.29578)
      SLIT(I,2)=CALAM(I)*SIN(ALPH1(I)/57.29578)
    3 SLIT(I,3)=SALAM(I)

      STAR VECTOR COMPONENTS

      DO 4 I = 1,ITST
      SVEC(1,I)=COS(GAMA(I)/57.29578)*COS(SIGMA(I)/57.29578)
      SVEC(2,I)=COS(GAMA(I)/57.29578)*SIN(SIGMA(I)/57.29578)
    4 SVEC(3,I)=SIN(GAMA(I)/57.29578)
      DO 14 I=1,4
```

60

```
   14 SMAG(I) = 10.0**((SMAG(I)-1.0)/5.02)
C
C      INERTIAL-ANGULAR MOMENTUM AXES TRANSFORM
C
   25 DO 27 I = 2,4
      CVAL(I) = COS(VAL(I))
   27 SVAL(I) = SIN(VAL(I))
      TMAT(1,1) = CVAL(4)*CVAL(3)
      TMAT(1,2) = CVAL(4)*SVAL(3)
      TMAT(1,3) = -SVAL(4)
      TMAT(2,1) = -SVAL(3)
      TMAT(2,2) = CVAL(3)
      TMAT(3,1) = SVAL(4)*CVAL(3)
      TMAT(3,2) = SVAL(4)*SVAL(3)
      TMAT(3,3) = CVAL(4)
C
C      TEST FOR INTEGRATION TIME INTERVAL COMPLETED
C
      IF(INTER.NE.1)GO TO 6
C
C      EULER TRANSFORM(ANGULAR MOMENTUM-BODY AXES)
C
      IF(VAL(6).GT.6.2831835) VAL(6) =VAL(6) -6.2831853
      IF(VAL(5).LT. -6.2831853) VAL(5) =VAL(5) + 6.2831853
      DO 28 I = 5,6
      CVAL(I) = COS(VAL(I))
   28 SVAL(I) = SIN(VAL(I))
      ET(1,1) = CVAL(5)*CVAL(6) - CVAL(2)*SVAL(6)*SVAL(5)
      ET(1,2) = CVAL(5)*SVAL(6) + CVAL(2)*CVAL(6)*SVAL(5)
      ET(1,3) = SVAL(5)*SVAL(2)
      ET(2,1) = -SVAL(5)*CVAL(6) -CVAL(2)*SVAL(6)*CVAL(5)
      ET(2,2) = -SVAL(5)*SVAL(6) + CVAL(2)*CVAL(6)*CVAL(5)
      ET(2,3) = CVAL(5)*SVAL(2)
      ET(3,1) = SVAL(2)*SVAL(6)
      ET(3,2) = -SVAL(2)*CVAL(6)
      ET(3,3) = CVAL(2)
C
C      STAR VECTOR TRANSFORMED TO BODY AXES
C
   69 DO 9 I = 1,3
      AEVEC(I)=0.0
      DO 9 J = 1,3
    9 AEVEC(I) = AEVEC(I)+SVEC(J,M)*TMAT(I,J)
      DO 8 I = 1,3
      AETVE(I) = 0.0
      DO 8 J = 1,3
    8 AETVE(I) = AETVE(I)+ ET(I,J)*AEVEC(J)
C
C      TEST FOR ORTHOGNAL SLIT AND STAR VECTORS
C
      EVAL=SLIT(ISLIT,1)*AETVE(1)+SLIT(ISLIT,2)*AETVE(2)+SLIT(ISLIT,3)*
     1AETVE(3)
      IF(ABS(EVAL).LT.0.5E-6) GO TO 40
```

```
C
C       MAGNETIC FIELD INTENSITY COMPONENTS
C
      6 VNU =VNO +WO*T
        BETA = ATAN2(SIN(VNU)*CVIE,COS(VNU))
        ALPHA = ARSIN(SIN(VNU)*SVIE)
C       BMG(1)=BX     BMG(2)=BY     BMG(3)=BZ
        BMG(1) = C2*SIN(2.*ALPHA)*COS(BETA)
        BMG(2)= C2*SIN(2.*ALPHA)*SIN(BETA)
        BMG(3)= C2*(C3 - COS(2.*ALPHA))
        DO 30 I = 1,3
     30 BMAG(I) = TMAT(I,1)*BMG(1) + TMAT(I,2)*BMG(2) + TMAT(I,3)*BMG(3)
C
C       DIFFERENTIAL EQUATIONS
C
        ARTIA =(ERTIA(1)-ERTIA(3))/(ERTIA(3)*ERTIA(1))
        DRTIA = C1*(1.0/ERTIA(1) + ARTIA*(CVAL(2)**2))
        DER1(1)= -DRTIA*VAL(1)*(BMAG(1)**2 + BMAG(2)**2)
        DER1(2)= -C1*ARTIA*CVAL(2)*SVAL(2)*(BMAG(1)**2 + BMAG(2)**2   +
       12.*(BMAG(3)**2))/2.
        DER1(3) = DRTIA*BMAG(2)*BMAG(3)/SVAL(4) + BMAG(1)*AMAG*CVAL(2)/
       1(VAL(1)*SVAL(4))
        DER1(4) = DRTIA*BMAG(1)*BMAG(3) - BMAG(2)*AMAG*CVAL(2)/VAL(1)
        DER2(1)= ARTIA*CVAL(2)*VAL(1) + BMAG(3)*AMAG/VAL(1)
        DER2(2)= VAL(1)/ERTIA(1) + DRTIA*BMAG(2)*BMAG(3)*CVAL(4)/SVAL(4)
       1+ BMAG(1)*AMAG*CVAL(2)*CVAL(4)/(VAL(1)*SVAL(4)) - BMAG(3)*AMAG*
       2CVAL(2)/VAL(1)
C
C       TEST FOR INTEGRATION TIME INTERVAL COMPLETED
C
        IF(INTER.NE.1) GO TO 10
C
C       ANGULAR ERROR IN SPIN AND PRECESSION PLANE
C
        A1 = AETVE(1)*CALAM(ISLIT)
        A2 = AETVE(2)*CALAM(ISLIT)
        A3=AETVE(3)*SALAM(ISLIT)
        ARG1= (-A2*A3 + A1* SQRT(A1**2 + A2**2 - A3**2))/(A1**2 + A2**2)
        ASARG = ARSIN(ARG1)
        IF(AETVE(1).GT.0.0.AND.AETVE(2).GT.0.0)ASARG = 3.14159 - ASARG
        IF(AETVE(1).LT.0.0.AND.AETVE(2).GT.0.0)ASARG = 3.14159 + ASARG
        ASARG=ASARG-ALPH1(ISLIT)
C
C       INTEGRATION TIME INETERVAL TO NULL ERROR
C
        DELTA = ASARG/(DER2(1)+DER2(2))
        WRITE(6,200)ARG1,ASARG,EVAL,DELTA
C
C       CALL INTEGRATION SUBROUTINE
C
     10 CALL RKUTTA(INTER)
        GO TO 25
C
```

```
C       RANDOM NOISE GENERATOR SUBROUTINE CALL
C
   40 CALL GETRAN(IR,NAT,LGAUS,RN,Y1,Y2)
      NAT = 2
C
C       COMPUTE MEASUREMENT TIME
C
      TO =T +RN*VAR*SMAG(M)
      CHAT = FLOAT(10*ISLIT + M)
C
C       OPTICAL AXES COMPONENTS IN INERTIAL SPACE
C
      DO 110 I =1,3
      AEVEC(I) = 0.0
      DO 110 J =1,3
  110 AEVEC (I) = AEVEC(I) + TMAT(J,I)*ET(2,J)
C
C       WRITE DATA ON TAPE...STAR COUNT,STAR AND SLIT,MEASUREMENT TIME,
C       TRUE TIME,OPTICAL AXES COMPONENTS,AND STATES
C
      WRITE(4,201) K,CHAT,TO,T,AEVEC,ANS(6),ANS(5),ANS(2),ANS(4),
     1ANS(3),ANS(1)
      WRITE(6,202)ISLIT,M,TO,T,ANS,AEVEC
      K=K+1
C
C       INDEX TO NEXT STAR AND/OR SLIT
C
      IF(ISLIT.EQ.ITSL) GO TO 50
      ISLIT=ISLIT+1
      GO TO 69
   50 ISLIT=1
      M = M+1
      IF(M.GT.ITST) M=1
      IF(K.GT.KSTOP) STOP
      GO TO 69
  100 FORMAT(6E10.5,/6E10.5,/6E10.5,/6E10.5)
  101 FORMAT(3E10.5,6I5)
  102 FORMAT(6E10.5)
  103 FORMAT(I5,F3.0,2D20.12/3E16.8)
  200 FORMAT(4E16.8)
  201 FORMAT(I5,F3.0,2D20.12,3E16.8/6D20.12)
  202 FORMAT(2I5,2D20.12,/6D19.11/3E16.8//)
  203 FORMAT(2X4HDATA/,5HISLIT2X5H STAR2X10HMEAS. TIME6X9HTRUE TIME/
     12X8HMOMENTUM9X5HTHETA11X4HZETA12X3HTAU13X3HPSI13X3HPHI/
     22X38HOPTICAL (Y-AXIS) COMPONENTS OX, OY, OZ/)
      END
      SUBROUTINE GETRAN(IR,N,L,RN,Y1,Y2)
C
C       RANDOM NOISE GENERATION SUBROUTINE
C
      DIMENSION IA2(35,2),IA3(35,2),IA5 (35,2),IA1(35,2),IR(2),IA(35,2),
     1BUNNY(35)
      CON = 6.28318530717959
```

```
      Y1 = 0.0
      Y2 = 0.0
      IF(N.GT.1)GO TO 22
      DO 75 I = 1,35
   75 BUNNY(I) = .5**I
      DO 2 M = 1,L
      DO 1 I = 2,34,2
    1 IA2(I,M) = 1
      DO 9 I = 1,33,2
    9 IA2(I,M) = 0
    2 IA2(35,M) = 1
      N1 = 35
      DO 21 MI= 1,L
   18 DO 17 I = 1,35
      IF(I.EQ.1)ITEMP= IA2(34,MI)
      IF(I.EQ.2)ITEMP= IA2(35,MI)
      IF(I.GT.2)ITEMP= IA3(I-2,MI)
      IA3(I,MI) = ITEMP + IA2(I,MI)
      IF(IA3(I,MI).EQ.2) IA3(I,MI) = 0
      IA1(I,MI) = IA2(I,MI)
   17 IA2(I,MI) = IA3(I,MI)
      MAX= 35 + IR(MI)-1
      N1 = N1 + 35
      IF(N1.GE.MAX)GO TO 3
      GO TO 18
    3 N2 = N1 - MAX
      N3 = 35 - N2
      IZ = 0
      N4 = N3 + 1
      IF(N2.EQ.0)GO TO 33
      DO 19 I = N4,35
      IZ = IZ + 1
   19 IA(IZ,MI) = IA1(I,MI)
   33 CONTINUE
      IF(N3.EQ.0)GO TO 21
      DO 49 I = 1,N3
      IZ = IZ + 1
   49 IA(IZ,MI) = IA3(I,MI)
   21 N1 = 35
      GO TO 25
   22 DO 26 M = 1,L
      DO 26 I = 1,35
      IF(I.EQ.1)ITEMP = IA(34,M)
      IF(I.EQ.2)ITEMP = IA(35,M)
      IF(I.GT.2)ITEMP = IA5(I-2,M)
      IA5(I,M) = ITEMP + IA(I,M)
      IF(IA5(I,M).EQ.2)IA5(I,M) =0
   26 IA(I,M) = IA5(I,M)
   25 IF(L.EQ.1)GO TO 60
      DO 28 J = 1,35
      J1 = 36 - J
      IF(IA(J1,1).EQ.1)Y1=Y1+BUNNY(J)
      IF(IA(J1,2).EQ.1) Y2=Y2+BUNNY(J)
```

```
   28 CONTINUE
      RN = SQRT(-2.*ALOG(Y1))*SIN(CON*Y2)
      GO TO 62
   60 DO 61 J = 1,35
      J1 = 36 - J
      IF(IA(J1,1).EQ.1)Y1=Y1+BUNNY(J)
   61 CONTINUE
   62 RETURN
      END
      SUBROUTINE RKUTTA(INTER)
C
C     INTEGRATION SUBROUTINE
C
      DIMENSION DER1(4),DER2(2),VAL(6),P1(4),P2(2),R1(4),R2(2),Q1(4),
     1Q2(2),S1(4),S2(2),ANS(6)
      DOUBLE PRECISION DER2,VAL,P2,R2,Q2,S2,T,ANS
      COMMON VAL,DER1,DER2,DELTA,T,ANS
      GO TO (20,30,40,50),INTER
   20 DO 1 I=1,4
      ANS(I)= VAL(I)
      P1(I) = DELTA*DER1(I)
    1 VAL(I) = .5*P1(I)+ ANS(I)
      DO 11 I=1,2
      ANS(I+4)=VAL(I+4)
      P2(I) = DELTA*DER2(I)
   11 VAL(I+4)=.5*P2(I)+ANS(I+4)
      T=T+.5*DELTA
      INTER=2
      RETURN
   30 DO 2 I=1,4
      Q1(I) = DELTA*DER1(I)
    2 VAL(I)= .5*Q1(I)+ ANS(I)
      DO 22 I=1,2
      Q2(I)=DELTA*DER2(I)
   22 VAL(I+4)=.5*Q2(I)+ANS(I+4)
      INTER=3
      RETURN
   40 DO 3 I=1,4
      R1(I)= DELTA*DER1(I)
    3 VAL(I)= R1(I)+ ANS(I)
      DO 33 I=1,2
      R2(I)= DELTA*DER2(I)
   33 VAL(I+4) = R2(I)+ANS(I+4)
      T=T+.5*DELTA
      INTER=4
      RETURN
   50 DO 4 I=1,4
      S1(I)=DELTA*DER1(I)
    4 VAL(I)= ANS(I)+(P1(I)+2.*(Q1(I)+R1(I))+S1(I))/6.
      DO 44 I=1,2
      S2(I)= DELTA*DER2(I)
   44 VAL(I+4)=ANS(I+4)+(P2(I)+2.*(Q2(I)+R2(I))+S2(I))/6.
      DO 5 I= 2,6
```

```
5 ANS(I)= VAL(I)*57.29578
  INTER=1
  RETURN
  END
```

```
C      LIMITED MEMORY FILTER PROGRAM
C
C      DIRECTIONS FOR OPTIONS
C
C      OPTION 1. NONLINEAR KALMAN FILTER
C          SET K = -1,W(I) = 0.0,I=1,N,AND SGAIN.LT.0.0
C
C      OPTION 2. LIMITED MEMORY USING SCHMIDT GAIN
C          SET K = -1,W(I) = 0.0,I=1,N,AND SELECT VALUE FOR SGAIN.GT.0.0
C          AND .LT.+1.0 (NOMINAL SGAIN  VALUE = +.01)
C
C      OPTION 3. LIMITED MEMORY USING NOISE ADDED TO COVARIANCE MATRIX
C          SET K = -1 AND SGAIN.LT.0.0. SET NOISE TO BE ADDED TO STATES
C          BY W(I) AND IF NOISE COUPLING EXISTS BY Q(I,J).MATRIX EQUATION
C          FOR NOISE ADDITION IS P**1/2 = P**1/2 + Q*W
C
C      OPTION 4. LIMITED MEMORY USING INTERMEDIATE STAR SIGHTINGS AND
C      ADDING AND SUBTRACTING DATA FROM COVARIANCE MATRIX
C          SET W(I) = 0.0,I=1,N AND SGAIN.LT.0.0. SET K.GT.-1. IF K=0 ONLY
C          END POINT MEASUREMENTS ARE USED.K.GT.0 WILL INCLUDE K INTER-
C          MEDIATED STARS. MEMORY LENGTH IS SET BY NP - NQ.EXAMPLE,NP=1,
C          NQ=-99, LENGTH WILL BE 100.(NOTE-- NEGATIVE STARS ARE NOT
C          CONSIDERED).SET SH BETWEEN -1.0 AND +1.0 FOR WEIGHTING LAST
C          POINT IN MEASUREMENT EQUATION. SET WT BETWEEN 0.0 AND +1.0
C          FOR COVARIANCE WEIGHTING SUBTRACTION. (ELIMINATES LARGE SUB-
C          TRACTIONS FROM COVARIANCE MATRIX WHICH COULD CAUSE NONPOSITIVE
C          DEFINITE EFFECTS). IF WT.LT.0.0 THE ESTIMATION BECOMES A KALMAN
C          FILTER WITH PAST MEASUREMENT POINTS INCLUDED.
C
       DIMENSION RAN(10),NTIM(12),SSL(500),STRCH(500),SSTCH(12),STAR(3,6)
      1,MAG(6),SLIT(3,3),AMT(3,3),B(3,2),HERR(7),X(7),AMTEM(7),ET(3,3)  ,
      1P(7,7),PHI(7,7),TPHI(7,7),AMAT(2,7),TERR(7),GRAD(7),Q(7,7), W(7),
      3ISCHT(12),CX(5),SX(5),Y(7),Z(6),ERR(3)
       REAL INRTZ,INERT,MAG
       COMMON/BLK1/X/BLK2/PHI,ITIMX,T/BLK3/SSTCH,CVIE,SVIE/BLK4/CX,SX,
      1AMT,ET
       DOUBLE PRECISION X,Y,T,STRCH,SSTCH,TEM,Z
       DATA AMT(2,3)/0.0/
C
C      DATA INITIALIZATION FOR EACH RUN. STATE,STAR POSITIONS AND
C      MAGNITUDES,SLIT ORIENTATION,NOMINAL MEASUREMENT VARIANCE,SCHMIDT
C      GAIN,INITIAL TIME,NOISE EFFECTS,INERTIAS,ORBIT CONDITIONS,AND
C      CONTROL NUMBERS.INITIAL COVARIANCE MATRIX(P**1/2)
C      WRITE INTIAL DATA TO RECORD INPUT CONDITIONS
C
   43  READ(5,100) X, (HERR(I),I=1,6),(RAN(I),I=2,7)
      1,MAG,(TERR(I),I=1,3),(GRAD(I),I=1,3),VAR,SGAIN,T,W
  100  FORMAT(7E10.6/12F6.0/6F6.2,6F6.0/2E10.5
      1,E16.8/7E10.5)
       READ(5,103) INERT,INRTZ,SPAN,VIE,VNO,WO,C2,RAN(1),IMAXT,
      1IALTR,NOISE,NPRNT,NQ,K,NP,NTOTL,NSUM,NT,IAT,NSPNT,SH,WT
  103  FORMAT(5F7.2,2E10.5,E16.8/12I5,F5.0,F5.3)
       WRITE(6,100) X,(HERR(I),I=1,6),(RAN(I),I=2,7
```

```
      1),MAG,(TERR(I),I=1,3),(GRAD(I),I=1,3),VAR,SGAIN,T,W
       WRITE(6,103) INERT,INRTZ,SPAN,VIE,VNO,WO,C2,RAN(1),IMAXT,
      1IALTR,NOISE,NPRNT,NQ,K,NP,NTOTL,NSUM,NT,IAT,NSPNT,SH,WT
       READ(5,105) ((P(I,J),J=1,7),I=1,7)
  105 FORMAT(5E14.8/5E14.8/5E14.8/5E14.8/5E14.8/5E14.8/5E14.8/5E14.8
      1/5E14.8/5E14.8)
       WRITE(6,390) ((P(I,J),J=1,7),I=1,7)
       IAT = 0
        EVAL = 0.0
C
C      READ STAR CHART FROM TAPE
C
       REWIND 4
       IF(NOISE.EQ.1) GO TO 220
       READ(4,101)(SSL(I),STRCH(I),Z(1),I=1,NTOTL)
  101 FORMAT(5XF3.0,D20.12/D20.12)
       GO TO 221
  220 READ(4,102)(SSL(I),STRCH(I),Z(1),I=1,NTOTL)
  102 FORMAT(5XF3.0,20XD20.12/D20.12)
  221 CONTINUE
       REWIND 4
       DO 313 I=1,NT
  313 READ(4,361) KAT,Z(1)
       NT = NT +1
       NSUM = NSUM + NTOTL
C
C      SLIT,STAR,AND CONSTANT PARAMETER INITIALIZATION
C
       DO 300 I=1,3
       SLIT(1,I)=COS(TERR(I)/57.29578)*COS(GRAD(I)/57.29578)
       SLIT(2,I)=COS(TERR(I)/57.29578)*SIN(GRAD(I)/57.29578)
  300 SLIT(3,I)=SIN(TERR(I)/57.29578)
       DO 301 I=1,6
       MAG(I) = 10.0**((MAG(I) -1.0)/5.02)
       STAR(1,I)=COS(HERR(I)/57.29578)*COS(RAN(I+1)/57.29578)
       STAR(2,I)=COS(HERR(I)/57.29578)*SIN(RAN(I+1)/57.29578)
  301 STAR(3,I)=SIN(HERR(I)/57.29578)
       DO 303 I=1,7
       DO 303 J=1,7
       TPHI(I,J)=0.0
       DO 303 L=1,7
  303 TPHI(I,J)=TPHI(I,J) + P(I,L)*P(J,L)
       WRITE(6,201) T,X,TPHI
  201 FORMAT(3H T=,E16.8/3H X=,7E16.8/3H P=,7E16.8/3X7E16.8/3X7E16.8/
      13X7E16.8/3X7E16.8/3X7E16.8)
       DO 370 I=1,7
       DO 371 J=1,7
  371 Q(I,J) =0.0
  370 Q(I,I) =1.0
       CVIE = COS(VIE/57.29578)
       SVIE =SIN(VIE/57.29578)
       KAM = NP - NQ
       DO 302 I=1,5
```

68

```
      302 X(I)= X(I)/57.29578
        1 NP=NP+1
          NQ=NQ+1
C
C         TEST FOR EXCEEDING STAR CHART IN MEMORY
C
          IF(NP.LT.NTOTL.OR.NP.EQ.NTOTL) GO TO 93
C
C         UPDATE STAR CHART TO ELIMINATE OLD DATA AND READ IN NEW
C
          DO 41 I=1,KAM
          KAT = NQ - 1 + I
          SSL(I)=SSL(KAT)
       41 STRCH(I)=STRCH(KAT)
          KAT = KAM +1
          IF((IMAXT-NSUM).LT.(NTOTL-KAM)) NTOTL =IMAXT-NSUM+KAM
          IF (NOISE.EQ.1) GO TO 222
          READ(4,101)(SSL(I),STRCH(I),Z(1),I=KAT,NTOTL)
          GO TO 223
      222 READ(4,102)(SSL(I),STRCH(I),Z(1),I=KAT,NTOTL)
      223 CONTINUE
          REWIND 4
          DO 122 I=1,NSUM
      122 READ(4,361) KAT,Z(1)
      361 FORMAT(I5/D20.12)
          NSUM = NSUM+NTOTL-KAM
          NP= KAM+1
          NQ= 1
C
C         SELECT BY RANDOM NUMBERS THE INTERMEDIATE STARS IN OPTION 4.
C         ALTERNATES**ORDERED STATISTICS OR STAR WITHIN A SPAN
C
       93 IF(K.LT.1)GO TO 2
          IF(IALTR.EQ.1) GO TO 30
          DO 40 I=1,K
       40 CALL QRAND1(RAN(I))
C         GENERATE ORDER STATISTIC
          IF(K.EQ.1) GO TO 95
          LIM1 = K-1
       96 LINT = 1
          DO 97 I= 1,LIM1
          IF(RAN(I+1).GT.RAN(I)) GO TO 97
          TEP = RAN(I+1)
          RAN(I+1) = RAN(I)
          RAN(I) = TEP
          LINT = 1
       97 CONTINUE
          IF(LINT.EQ.1) GO TO 95
          LIM1 = I-1
          GO TO 96
C         GENERATE TIME POINTS
       95 DO 94 I = 1,K
          NTIM(I+1) = NP - IFIX(RAN(I)*FLOAT(NP-NQ))
```

```
      IF(NTIM(I+1).LT.1) NTIM(I+1) = 1
   94 CONTINUE
      GO TO 2
   30 DO 39 I=1,K
      CALL QRAND1(RAN(I))
      NTIM(I+1) = NP-(I*(NP-NQ)/(K+1)+IFIX(SPAN*(RAN(I)-0.5)))
      IF(NTIM(I+1).LT.1) NTIM(I+1)=1
   39 CONTINUE
C
C     ASSEMBLE WORKING STAR CHART(STARS TO BE USED TO OBTAIN NEW
C     ESTIMATE)
C
    2 LIM1 = K + 2
      NTIM(LIM1) = NQ
      IF(NTIM(LIM1).LT.1) NTIM(LIM1) =1
      NTIM(1) = NP
      IF(NT.LT.NSPNT+1.OR.MOD(NT,NPRNT).EQ.0) WRITE(6,401)(NTIM(I),I=1,
     112)
  401 FORMAT(11H STAR NOS.=,12I5)
      DO 92 I= 1,LIM1
      M = NTIM(I)
      SSTCH(I)=STRCH(M)
   92 ISCHT(I)=(SSL(M)+.5)
C
C     UPDATE STATE AND CALULATE STATE TRANSITION MATRIX TO NEW STAR TIME
C
      ITIMX = 1
      CALL DIFF(INERT,INRTZ,C2,WO,VNO,IAT,NT)
      TEM = T
      DO 67 I=1,7
   67 Y(I) =X(I)
C
C     CALCULATE COVARIANCE MATRIX(P**1/2) AND ADD NOISE IF ANY
C
      DO 54J=1,7
      DO 51 I=1,6
      TPHI(I,J)=Q(I,J)*W(J)
      DO 51 L=1,7
   51 TPHI(I,J) = TPHI(I,J) + PHI(I,L)*P(L,J)
   54 TPHI(7,J) = PHI(7,7)*P(7,J) +Q(7,J)*W(J)
      DO 20 I=1,7
      DO 21 J=1,7
      P(I,J) = TPHI(I,J)
   21 PHI(I,J) =0.0
      TERR(I) =0.0
   20 PHI(I,I)=1.0
      GO TO 22
   91 ITIMX = ITIMX + 1
C
C     TEST FOR VALID STAR (SINCE NEGATIVE STARS ARE NOT ALLOWED)
C
      IF(SSTCH(ITIMX).LT.(-5.0)) GO TO 80
C
```

70

```
C       INTEGRATE STATE AND STATE TRANSITION MATRIX TO OLD STAR SIGHTING
C       TIMES
C
        CALL DIFF(INERT,INRTZ,C2,WO,VNO,IAT,NT)
C
C       MEASUREMENT ANALYSIS
C       1. DETERMINE SLIT AND STAR
C
     22 ISLIT = 2
        IF(ISCHT(ITIMX).LT.20) ISLIT = 1
        ISTAR = ISCHT(ITIMX)- 10*ISLIT
C
C       2. TRANSFORM STAR VECTOR FROM INERTIAL TO BODY AXES
C
        CALL TRANS(2)
        DO 89 I=1,3
        B(I,1)= 0.0
        DO 89 J=1,3
     89 B(I,1)= B(I,1)+AMT(I,J)*STAR(J,ISTAR)
        DO 88 I=1,3
        B(I,2) = 0.0
        DO 88 J= 1,3
     88 B(I,2) =  B(I,2) + ET(I,J)*B(J,1)
C
C       3. CALCULATE MEASUREMENT EQUATION
C
        H = 0.0
        DO 87 I=1,3
     87 H = H-SLIT(I,ISLIT)*B(I,2)
        IF(ITIMX.EQ.K+2) H=SH*H
C
C       4. CALULATE GRADIENT OF H(X(K)) WITH RESPECT TO X(K)
C
        HERR(1) = 0.0
        DO 86 I=1,3
     86 HERR(1) = HERR(1) + SLIT(I,ISLIT)*(ET(I,1)*B(2,1)- ET(I,2)*B(1,1))
        HERR(2) = B(2,2)*SLIT(1,ISLIT) - B(1,2)*SLIT(2,ISLIT)
        HERR(3) = B(3,2)*(SLIT(1,ISLIT) *SX(2) + SLIT(2,ISLIT)*CX(2))+
       1(CX(3)*(SX(1)*B(1,1)-CX(1)*B(2,1))-SX(3)*B(3,1))*SLIT(3,ISLIT)
        HERR(4) = 0.0
        DO 85 I=1,3
     85 HERR(4) = HERR(4)+SLIT(I,ISLIT)*(ET(I,3)*B(1,1)-ET(I,1)*B(3,1))
        DO 84 I = 1,3
     84 B(I,1) = AMT(I,1)* STAR(2,ISTAR) - AMT(I,2)*STAR(1,ISTAR)
        DO 83 I = 1,3
        B(I,2) = 0.0
        DO 83 J = 1,3
     83 B(I,2) =  B(I,2) + ET(I,J)*B(J,1)
        HERR(5) = 0.0
        DO 82 I = 1,3
     82 HERR(5) = HERR(5) + SLIT(I,ISLIT)*B(I,2)
C
C       5. UPDATE GRADIENT AND MEASUREMENT USING STATE TRANSITION MATRIX
```

```
C        AND MEASUREMENT COVARIANCE INVERSE
C
      RAN(10)= VAR*MAG(ISTAR)*(HERR(1)+HERR(2)*(INERT/INRTZ-1.)*
     1CX(3))*X(6)/INERT
      GRAD(1) = PHI(1,1)*HERR(1)/RAN(10)
      GRAD(2) = PHI(2,2)*HERR(2)/RAN(10)
      DO 53 J=3,7
      GRAD(J) = 0.0
      DO 53 I = 1,5
   53 GRAD(J) = GRAD(J)+PHI(I,J)*HERR(I)/RAN(10)
C
C        6. ACCUMULATE WEIGHTED ERROR
C
      DO 81 I = 1,7
   81 TERR(I) = TERR(I)+GRAD(I)* H/RAN(10)
      IF(NT.LT.NSPNT+1.OR.MOD(NT,NPRNT).EQ.0) WRITE(6,404) H
  404 FORMAT(7H ERROR=,E16.8)
   80 CONTINUE
C
C      TEST FOR FIRST(NEW) OR LAST(OLDEST) STAR IN WORKING CHART(MEMORY
C      WINDOW)
C
      IF(ITIMX.LT.K+2.AND.ITIMX.GT.1) GO TO 91
C
C      TEST FOR LAST STAR
C
      IF(ITIMX.NE.1) GO TO 78
C
C      STORE GRADIENT DATA FOR FIRST AND LAST STAR TO BE USED IN P**1/2
C      UPDATE
C
      GA=0.0
      DO 77 I=1,7
      GA=GA +GRAD(I)**2
   77 AMAT(1,I)=GRAD(I)
      GAIN=SGAIN/GA
      IF(K.EQ.(-1)) GO TO 75
      GO TO 91
   78 CONTINUE
      DO 76 I=1,7
   76 AMAT(2,I)=GRAD(I)
C
C      CALCULATE NEW P**1/2 MATRIX
C
   75 LI=2
      IF(SSTCH(ITIMX).LT.(-5.0).OR.K.EQ.(-1)) LI=1
      DO 12 M=1,LI
      SI=1.0
      IF(M.EQ.2) SI=-1.0
      B(M,M) =0.0
      DO 10 I=1,7
      AMTEM(I)=0.0
      DO 11 J=1,7
```

```
   11 AMTEM(I)=AMTEM(I)+ AMAT(M,J)* P(J,I)
   10 B(M,M) = B(M,M) + AMTEM(I)**2
      IF(M.EQ.2.AND.B(2,2).GT.WT) GO TO 12
      DE = (1.0-1.0/SQRT(1.0+SI*B(M,M)))/B(M,M)
      DO 212 I=1,7
      DO 212 J=1,7
      TPHI(I,J)=P(I,J)
      DO 212 L=1,7
  212 TPHI(I,J)=TPHI(I,J) - DE*P(I,L)*AMTEM(L)*AMTEM(J)
      DO 12 I=1,7
      DO 12 J=1,7
      P(I,J)=TPHI(I,J)
   12 CONTINUE
  390 FORMAT(7E16.8/7E16.8/7E16.8/7E16.8/7E16.8/7E16.8/7E16.8//)
C
C     CALCULATE NEW ESTIMATE
C
      DO 14 I=1,7
      DO 13 J=1,7
      TPHI(I,J) =0.0
      DO 213 L=1,7
  213 TPHI(I,J)=TPHI(I,J)+P(I,L)*P(J,L)
   13 Y(I)=Y(I) + TPHI(I,J)*TERR(J)
      IF(SGAIN.GT.0.0) Y(I)=Y(I)+TERR(I)*GAIN/(1.0 + B(1,1))
   14 X(I)=Y(I)
      T = TEM
C
C     TEST FOR STAR POINTS AT WHICH TO PERFORM ACCURACY ANALYSIS AND
C     DATA PRINT
C
      IF(NT.GT.NSPNT.AND.MOD(NT,NPRNT).NE.0) GO TO 45
C
C     READ TRUE STATES AND DIRECTION OF INSTRUMENT AXIS VECTOR
C
      READ(4,360) SSTCH(1),(GRAD(I),I=1,3),Z
  360 FORMAT(28XD20.12,3E16.8/6D20.12)
      X(1)=X(1)+X(6)*(SSTCH(1)-T)/INERT
      X(2)=X(2)+X(6)*COS(X(3))*(SSTCH(1)-T)*(INERT-INRTZ)/(INERT*INRTZ)
      IF(X(2).LT.(-6.2831853)) X(2) = X(2) + 6.2831853
      IF(X(1).GT.6.2831853) X(1)=X(1)-6.2831853
C
C     COMPARE STATE AND INSTRUMENT AXIS POINTING TO OBTAIN ERRORS
C
      CALL TRANS(2)
      DO 48 I=1,3
      B(I,1) = 0.0
      DO 121 J=1,3
  121 B(I,1) = B(I,1) + AMT(J,I)*ET(2,J)
   48 ERR(I)=B(I,1)-GRAD(I)
      EPLAN=(ATAN2(GRAD(3),GRAD(1))-ATAN2(B(3,1),B(1,1)))*57.29578
      ENORM=(ATAN2(GRAD(2),GRAD(1))-ATAN2(B(2,1),B(1,1)))*57.29578
       EVAL=SQRT(ERR(1)**2+ERR(2)**2+ERR(3)**2)*57.29578
      DO 16 I=1,5
```

```
   16 X(I)=57.29578*X(I)
      DO 161 I=1,6
  161 Z(I)=Z(I)-X(I)
      E=Z(1)+Z(2)
      EK=X(7)-.1427E-4
      WRITE(7,343) NT, SSTCH(1), EPLAN, ENORM, EVAL, Z, E, EK
  343 FORMAT(I5,7(PE10.3)/5(PE10.3))
C
C     WRITE EVALUATION OF ESTIMATION ACCURACY
C
      WRITE(6,200) SSTCH(1),X,EPLAN,ENORM,EVAL,Z,E
  200 FORMAT(/,3H T=,E16.8//3H X=,7E16.8//20H SIGHTING ERRORS EP=,E16.8,
     14H EN=,E16.8,12H ANG. ERROR=,E16.8//14H STATE ERRORS=/3X7E16.8//)
      DO 17 I=1,7
   17 X(I)=Y(I)
      GO TO 46
   45 CALL TRANS(0)
      READ(4,361)KAT,Z(1)
   46 CONTINUE
C
C     WRITE DIAGONSTICS
C
      IF(MOD(NT,95).EQ.0) WRITE(6,390) ((P(I,J),J=1,7),I=1,7)
      IF(MOD(NT,95).EQ.0) WRITE(6,390) TPHI
      IF(X(1).GT.6.2831853)X(1) = X(1) - 6.2831853
      IF(X(2).LT.(-6.2831853)) X(2) = X(2) + 6.2831853
C
C     INDEX FOR NEW DATA POINT AND TEST FOR RUN COMPLETION OR
C     NONCONVERGENCE
C
      NT = NT + 1
      IF(EVAL.GT.20.0) GO TO 43
      IF(NT.GT.IMAXT) GO TO 43
      GO TO 1
      END
      SUBROUTINE DIFF(INERT,INRTZ,C2,WO,VNO,IAT,NT)
C
C     SUBROUTINE TO CALCULATE DIFFERENTIAL EQUATIONS
C
      DIMENSION BM(3),BMAG(3),DX1(2),DX2(4),DPHI(7,7),G(7,7),
     1X(7),PHI(7,7),SSTCH(12),CX(5),SX(5),AMT(3,3),ET(3,3)
      REAL INRTZ,INERT,MAG
      DOUBLE PRECISION X,T,DX1,SSTCH
      COMMON/BLK1/X/BLK2/PHI,ITIMX,T/BLK3/SSTCH,CVIE,SVIE/BLK4/CX,SX,
     1AMT,ET/BLK5/DX1,DX2,DPHI,INTER,DELTA
      DATA G(4,2),G(4,7),G(5,2),G(5,7),G(6,3),G(6,4),G(6,5),G(6,7),
     1G(7,2),G(7,7),G(3,7)/11*0.0/
   47 IF(ITIMX.GT.1) GO TO 81
      DO 83 I = 1,7
      DO 82 J = 1,7
   82 PHI(I,J) = 0.0
   83 PHI(I,I) = 1.0
   81 DELTA = SSTCH(ITIMX)-T
```

```
      INTER = 1
      IF(IAT.EQ.1)GO TO 50
      IAT = 1
   99 CALL TRANS(0)
C     COMPUTE MAGNETIC FIELD
   50 VNU=VNO+WO*T
      TMP1=SIN(VNU)*CVIE
      TMP2=COS(VNU)
      TMP3=SQRT(TMP1**2+TMP2**2)
      ALPHA=ARSIN(SIN(VNU)*SVIE)
      BM(1) = C2*SIN(2.*ALPHA)*TMP2/TMP3
      BM(2)= BM(1)*TMP1/TMP2
      BM(3)=C2*(.333-COS(2.*ALPHA))
C     COMPUTE  MAGNETIC FIELD TRANSFORM
      BMAG(1)= AMT(1,1)*BM(1)+AMT(1,2)*BM(2)+AMT(1,3)*BM(3)
      BMAG(2)=AMT(2,1)*BM(1)+AMT(2,2)*BM(2)+AMT(2,3)*BM(3)
      BMAG(3)=AMT(3,1)*BM(1)+AMT(3,2)*BM(2)+AMT(3,3)*BM(3)
      BM(1)=BMAG(1)*BMAG(2)
      BM(2)=BMAG(2)*BMAG(3)
      BM(3)=BMAG(3)*BMAG(1)
      BMAG(1)=BMAG(1)*BMAG(1)
      BMAG(2)=BMAG(2)*BMAG(2)
      BMAG(3)=BMAG(3)*BMAG(3)
C     COMPUTE DIFFERENTIAL EQUATIONS
      S2X3=SIN(2.*X(3))
      TMP1=(INRTZ-INERT)/(INRTZ*INERT)
      TMP2=X(7)*(1./INERT-TMP1*CX(3)*CX(3))
      DX1(1)= X(6)/INERT + BM(2)*TMP2*CX(4)/SX(4)
      DX1(2)= TMP1*X(6)*(-CX(3)   )
      DX2(1)= X(7)*TMP1*S2X3*(BMAG(1)+BMAG(2)+2.*BMAG(3))/4.
      DX2(2)= TMP2*BM(3)
      DX2(3)= TMP2*BM(2)/SX(4)
      DX2(4)=-X(6)*TMP2*(BMAG(1)+BMAG(2))
C     COMPUTE PARTIAL DERIVATIVES
      G(3,1) = X(7)*TMP1*BM(2)*S2X3*CX(4)/SX(4)
      G(3,2) = X(6)*TMP1*SX(3)
      G(3,3)=2.*DX2(1)*COS(2.*X(3))/S2X3
      G(3,4) = X(7)*TMP1*BM(3)*S2X3
      G(3,5)=G(3,1)/CX(4)
      G(3,6)  =-X(6)*X(7)*S2X3*TMP1*(BMAG(1)+BMAG(2))
      G(4,1) = TMP2*BM(1)*CX(4)/SX(4) - DX2(3)/SX(4)
      G(4,3) = G(3,4)/2.0
      G(4,4)=TMP2*(BMAG(1)-BMAG(3))
      G(4,5) = TMP2*BM(1)/SX(4) - DX2(3)*CX(4)/SX(4)
      G(4,6)= 2.0*TMP2*X(6)*BM(3)
      G(5,1)=TMP2*((BMAG(2)-BMAG(3))*CX(4)-BM(3)*CX(4)*CX(4)/SX(4))
      G(5,3)= X(7)*TMP1*BM(2)*S2X3*SX(4)/2.0
      G(5,4)=TMP2*(BM(1)*SX(4)+BM(2)*CX(4))
      G(5,5) = TMP2*(BMAG(2)-BMAG(3)-BM(3)*CX(4)/SX(4))
      G(5,6) = 2.0*X(6)*TMP2*BM(2)*SX(4)
      G(6,1)=1./INERT
      G(6,2) = DX1(2)/X(6)
      G(6,6) =DX2(4)/X(6)
```

```
      G(7,1) = DX2(3)*CX(4)/X(7)
      G(7,3)= DX2(1)/X(7)
      G(7,4) = DX2(2)/X(7)
      G(7,5) =DX2(3)/X(7)
      G(7,6) = DX2(4)/X(7)
      MO=0
      IF(INTER.EQ.1.AND.MOD(NT,100).EQ.0) MO=1
C     COMPUTE STATE TRANSITION MATRIX DIFFERENTIAL EQUATIONS
      DO 52 I = 1,7
      DO 52 J = 1,7
      DPHI(I,J) = 0.0
      DO 52 K=3,7
   52 DPHI(I,J) = DPHI(I,J) + G(K,I)* PHI(K,J)
   51 CALL   RKUTTA
      IF(INTER.EQ.5) RETURN
      GO TO   99
      END
      SUBROUTINE    RKUTTA
C
C     RUNGE KUTTA INTEGRATION SUBROUTINE
C
      DIMENSION ANS(6),A1(2),A11(4),A2(2),A22(4),A3(2),A33(4),A4(2),
     1A44(4),PANS(7,7),AP1(7,7),AP2(7,7),AP3(7,7),AP4(7,7),DX1(2),DX2(4)
     2,DPHI(7,7),X(7),PHI(7,7)
      COMMON/BLK1/X/BLK2/PHI,ITIMX,T/BLK5/DX1,DX2,DPHI,INTER,DELTA
      DOUBLE PRECISION ANS,A1,A2,A3,A4,DX1,T,X
      GO TO (10,20,30,40),INTER
   10 DO 1 I= 1,2
      ANS(I)=X(I)
      A1(I)= DELTA*DX1(I)
    1 X(I)=.5*A1(I)+ANS(I)
      DO 11 I =3,6
      ANS(I)=X(I)
      A11(I-2)= DELTA*DX2(I-2)
   11 X(I)= .5*A11(I-2)+ ANS(I)
      DO 61 I=1,7
      DO 61 J=1,7
      PANS(I,J)= PHI(I,J)
      AP1(I,J) = DELTA*DPHI(I,J)
   61 PHI(I,J) = .5*AP1(I,J) + PANS(I,J)
      T=T+.5*DELTA
      INTER=2
      RETURN
   20 DO 2 I=1,2
      A2(I) = DELTA*DX1(I)
    2 X(I) = .5*A2(I)+ANS(I)
      DO 22 I=3,6
      A22(I-2)= DELTA*DX2(I-2)
   22 X(I) = .5*A22(I-2)+ANS(I)
      DO 62 I =1,7
      DO 62 J=1,7
      AP2(I,J) = DELTA* DPHI(I,J)
   62 PHI(I,J) =.5*AP2(I,J)+PANS(I,J)
```

```
      INTER=3
      RETURN
   30 DO 3 I=1,2
      A3(I) = DELTA*DX1(I)
    3 X(I) = A3(I)+ANS(I)
      DO 33 I=3,6
      A33(I-2) = DELTA*DX2(I-2)
   33 X(I) =   A33(I-2)+ANS(I)
      DO 63 I=1,7
      DO 63 J=1,7
      AP3(I,J) = DELTA*DPHI(I,J)
   63 PHI(I,J)= AP3(I,J)+PANS(I,J)
      T=T+.5*DELTA
      INTER=4
      RETURN
   40 DO 4 I=1,2
      A4(I)=DELTA*DX1(I)
    4 X(I)= ANS(I)+(A1(I)+2.*(A2(I)+A3(I))+A4(I))/6.
      DO 44 I=3,6
      A44(I-2)= DELTA*DX2(I-2)
   44 X(I)=ANS(I)+(A11(I-2)+2.*(A22(I-2)+A33(I-2))+A44(I-2))/6.
      DO   64 I=1,7
      DO 64 J=1,7
      AP4(I,J) =   DELTA*DPHI(I,J)
   64 PHI(I,J) =   PANS(I,J)+(AP1(I,J)+2.*(AP2(I,J)+AP3(I,J))+AP4(I,J))/
     16.0
      INTER   =5
      RETURN
      END
      SUBROUTINE TRANS(M)
C
C     SUBROUTINE TO CALCULATE THE ANGULAR MOMENUTM AND EULER ANGLE
C     TRANSFORMATIONS
C
      DIMENSION X(7), CX(5),SX(5),AMT(3,3),ET(3,3)
      COMMON/BLK1/X/BLK4/CX,SX,AMT,ET
      DOUBLE PRECISION X
      IF(M.EQ.1) GO TO 5
C     COMPUTE ANGULAR MOMENTUM TERMS
      DO 99 I=3,5
      CX(I)= COS(X(I))
   99 SX(I)= SIN(X(I))
      AMT(1,1)=CX(4)*CX(5)
      AMT(1,2)=CX(4)*SX(5)
      AMT(1,3)=-SX(4)
      AMT(2,1)=-SX(5)
      AMT(2,2)=CX(5)
      AMT(3,1)=SX(4)*CX(5)
      AMT(3,2)=SX(4)*SX(5)
      AMT(3,3)=CX(4)
      IF(M.EQ.0)RETURN
C     COMPUTE EULER ANGLE TERMS
    5 DO 90 I=1,2
```

```
      CX(I)=COS(X(I))
   90 SX(I)=SIN(X(I))
      ET(1,1) = CX(1)*CX(2)- CX(3)*SX(1)*SX(2)
      ET(1,2) = CX(2)*SX(1)+ CX(3)*CX(1)*SX(2)
      ET(1,3) = SX(3)*SX(2)
      ET(2,1) = -SX(2)*CX(1)-CX(3)*SX(1)*CX(2)
      ET(2,2) = -SX(1)*SX(2)+ CX(3)*CX(1)*CX(2)
      ET(2,3) = CX(2)*SX(3)
      ET(3,1) = SX(3)*SX(1)
      ET(3,2) = -SX(3)*CX(1)
      ET(3,3) =   CX(3)
      RETURN
      END
C
C     RANDOM NUMBER SUBROUTINE (QRAND1) TO BE SUBSTITUTED BY USER
```

APPENDIX IV - WEIGHTED LEAST SQUARES FORMULATION FOR FILTER SYSTEM

A least-squares approach to the filter problem is suggested since the probability density function for the system need not be known or assumed as would be required for a maximum likelihood estimation. Assuming P measurements have been taken, one seeks an estimate of the states such that

$$J = \sum_{k=1}^{P} \underline{\varepsilon}^T(k) R^{-1}(k) \underline{\varepsilon}(k)$$

D1)

is a minimum where

$$\underline{\varepsilon}(k) = y(k) - \underline{H}(x(k), k)$$

D2)

The weighting factor need not be selected as $R^{-1}(k)$ but it has been shown that such a weighting function leads to the Kalman filter for the linear problem, and it seems intuitively logical to weigh measurements based upon the inverse of their covariance since it effectively states the quality of each measurement.

An extremal for a well behaved function satisfies the condition

$$\nabla_{\underline{x}} J =$$

$$\Sigma(-\underline{y} + \underline{H})^T \underline{R}^{-1} \frac{\partial \underline{H}}{\partial \underline{x}} = 0$$

D3)

The solution for equation D3) is $\hat{\underline{x}}$. Since an explicit solution in terms of $\hat{\underline{x}}$ is generally impossible an iterative technique is usually employed. Assume $\hat{\underline{x}}_o$ is a good estimate; then the first approximate correction to this term is given by

$$\hat{\underline{x}} = \hat{\underline{x}}_o + \delta \underline{x}$$

D4)

Substituting in equation D3), assuming Taylor series expansion of

79

the nonlinear terms about $\hat{\underline{x}}_o$ gives

$$o = \Sigma \ \frac{\partial \underline{H}}{\partial \underline{x}} \ ^{T}R^{-1} \ (-y = \underline{H}) \ + \ \left(\frac{\partial \underline{H}}{\partial \underline{x}}^{T} R^{-1} \ \frac{\partial \underline{H}}{\partial \underline{x}} \right) \delta \underline{x}$$

D5)

and all applicable terms have been evaluated at $\hat{\underline{x}}_o$. In evaluating equation D5) it is necessary to assume the procedure equation 8) is used in obtaining $\hat{\underline{x}}_o(k)$ for the k measurement times considering $\hat{\underline{x}}_o(p)$ is known.

The correction $\delta \underline{x}$ is obtained by solution of equation D5) as

$$\delta \underline{x} = \Sigma \ \left[\frac{\partial \underline{H}}{\partial \underline{x}} \ ^{T}R^{-1} \ \frac{\partial \underline{H}}{\partial \underline{x}} \right]^{-1} \ \Sigma \ \frac{\partial \underline{H}}{\partial \underline{x}} \ R^{-1} \ (y - \underline{H})$$

D6)

Equation D6) is the standard iterative weighted least-square solution to the estimation problem. The new estimate can be formed by equation D4) as shown in equation 3). The summation terms in equation D6) can be identified as

$$V(k) = \frac{\partial H(x(k),k)}{\partial x(p)} \ R^{-1/2}(k) \Big|_{\hat{\underline{x}}_o(k)} = \nabla \underline{x}(p) H(\underline{x}(k),k) R^{-1/2}(k) \Big|_{\hat{\underline{x}}_o(k)}$$

as noted in equation 4.

REFERENCES

1. Schmidt, S. F., "Estimation of State with Acceptable Accuracy Constraint," Interim Report 67-4, contract NAS-5-9085 Analytical Mechanics Assoc., Inc., Jan., 1967.

2. Schmidt, S. F., "Compensation for Modeling Errors in Orbit Determination Problems," Analytical Mechanics Assoc., Inc. Interim Rpt. No. 67-16, Contract NAS5-11048, November 1967.

3. Jazwinski, A. H., "Limited Memory Optimal Filtering," 1968 JACC Proceedings pp. 383-389.

4. Lee, R. C. K., "The 'Moving Window' Approach to the Problems of Estimation and Identification," Aerospace Rpt. No. TR-1001(2307)-23, June 1967.

5. Tidwell, N. W.; Nelson, G. D.; and Lewis, W. J.: Attitude-Referenced Radiometer Study. Vol. I – Attitude Determination System Design. NASA CR-66852, 1969.

6. Gainer, P. A., "A Method for Computing the Effect of An Additional Observation on A Previous Least-Squares Estimate," NASA TN D-1599, January 1963.

7. DeRusso, P., Roy, R., and Close, C., State Variables for Engineers, John Wiley & Sons, New York, 1965, pp. 451-461.

8. Cox, H., "On the Estimation of State Variables and Parameters for Noisy Dynamic Systems," IEEE Trans. on A.C., Vol. AC-9, No. 1, January 1964, pp. 5-12.

9. Lee, R. C. K., Optimal Estimation, Identification and Control, Res. Mono. #28, M.I.T. Press, Cambridge, Mass. 1964.

10. Vogelzang, N. F., Baltes, J. J. and Scharmack, D. K., "Orbital Operation and Analysis for a 15-Micron Horizon Radiance Measurement Experiment," NASA CR-66376, NASA Contract NAS 1-6010, Honeywell, Inc., May 1967.

11. Tidwell, N. W. et al, "Conceptual Mechanization Studies for A Horizon Definition Spacecraft-Attitude Control Subsystem," NASA CR-66382, NASA Contract NAS 1-6010, Honeywell, Inc., May 1967.

12. Walsh, T. M., Keating, J. C. and Hinton, D. E., "Attitude Determination of the Spin-Stabilized Project Scanner Spacecraft," NASA TN D-4740, August 1968.

13. Ostroff, A. J. and Romanczyk, K. C., "Design of An Electronically Scanned Star Sensor with Digital Output," NASA TN D-5281, June 1969.

14. Bellantoni, J. F. and Dodge, K. N., "A Square Root Formulation of the Kalman-Schmidt Filter," AIAA Journal, Vol. 5, No. 7, July 1967, pp. 1309-1314.