8976-6012-KU-000

[redacted]

FACILITY FORM 602

N65-32029
(ACCESSION NUMBER)

(THRU)

33
(PAGES)

1
(CODE)

CR 64574
(NASA CR OR TMX OR AD NUMBER)

30
(CATEGORY)

GPO PRICE $ ______

CFSTI PRICE(S) $ ______

Hard copy (HC) 2.00

Microfiche (MF) .50

ff 653 July 65

8976-6012-RU-000

MONTE CARLO SIMULATION OF MIDCOURSE GUIDANCE (TAPP MOD II)

W. H. Pace, Jr.

Prepared for

JET PROPULSION LABORATORY
California Institute of Technology
Contract No. 950045
Under NAS 7-100

Approved T A Magness
T. A. Magness,
Project Engineer
Space Systems Analysis Study

Approved R M Page
R. M. Page
Manager,
Systems Analysis Department

SPACE TECHNOLOGY LABORATORIES, INC.
A Subsidiary of Thompson Ramo Wooldridge, Inc.
One Space Park
Redondo Beach, California

CONTENTS

INTRODUCTION TO THE MONTE CARLO TECHNIQUE

The Monte Carlo simulation of midcourse guidance known as TAPP-II is basically an error-analysis tool. Three of its applications are (1) the estimation of accuracies after multiple midcourse correction, (2) the estimation of velocity requirements for midcourse corrections, and (3) the comparison of various midcourse guidance schemes.

A Monte Carlo simulation has been chosen as the most convenient means to cope with the difficulties that arise in the error analysis of multiple-correction midcourse guidance. The necessity of handling non-gaussian distributions was the motivation for previous simple Monte Carlo simulations, but the additional problems involved in analyzing complicated guidance schemes and the effects of errors in physical constants require a more sophisticated program.

The Monte Carlo error analysis technique is essentially an empirical method for determining the statistical properties of the variables of interest in a particular system. A mathematical model is used to determine the sample values of these variables by selecting values for the random inputs that are used. For each set of random inputs chosen, the corresponding outputs of the model are calculated. After a large number of samples have been calculated, the desired statistics of the outputs are computed.

The model used must be of sufficient complexity to simulate the actual system accurately, but it must also allow rapid calculation of many samples, since the confidence in the results increases with the number of samples. An analytic model (rather than one involving integration) is often used, since it is sufficiently accurate for error analysis and allows rapid computation. The model may even be linearized in stages, as it is in TAPP-II.

Samples of random variables with any desired distribution can be generated with the Random Number Generator (a computer program). The independent random variables are usually assumed to be Gaussian. Since any linear transformation of a set of Gaussian variables is also Gaussian, it is possible to start with correlated Gaussian variables instead of independent Gaussian variables.

This often substantially reduces the number of variables considered. A program known as the Random Vector Generator produces sample sets of correlated variables.

A convenient method of analyzing the results of a Monte Carlo simulation is to form cumulative distributions of the variables of interest, but sample means, variances, and correlation coefficients can also be calculated if they are desired.

SUMMARY OF CURRENT CAPABILITIES

The Monte Carlo midcourse simulation is programmed in several subroutines so that changes or additions can be made easily. The current capabilities of the program are summarized here to form a basis for the detailed explanations to follow.

A. CHARACTERISTICS OF THE CORRECTION LOGIC

1. A maximum of ten midcourse corrections at fixed times can be simulated.

2. Any type of miss for which TAPP can calculate the partial derivatives can be used. The only restrictions are that the miss can have no more than six components and no fewer than three components. Six components are sufficient to fully define the trajectory, and at least three components of miss must be used to allow calculation of three components of correction velocity. Critical plane corrections are not used.

3. The correction velocity is calculated to minimize the weighted sum of squares of the miss components remaining after the correction. This allows calculation of corrections for more than three miss components and gives the usual unique value when only three components are used.

4. The execution error model consists of a bias (shutdown or control system) error and a proportional (accelerometer or angle) error in each of three mutually perpendicular directions. Errors can be correlated between corrections.

B. TRACKING

1. Tracking knowledge in terms of the full state vector (position, velocity, and physical constants) is used in order to simulate the real-time process accurately.

2. A priori knowledge is used in each estimate of the state vector in the same manner as it would be used in real time.

C. EFFECTS OF ERRORS IN PHYSICAL CONSTANTS

1. The effects of inaccurate knowledge of physical constants are included in the trajectory calculations and, therefore, in tracking estimation.

2. Physical constants that are to be measured by tracking are included in the state vector used in the guidance logic.

D. STATISTICS OF RANDOM VARIABLES

1. The covariance matrices used in the logic are always estimates of the true values, and may be erroneous.

2. The covariance matrices used to generate the random variables are the "true" values.

E. OUTPUTS

The outputs of the program are sample cumulative distributions of the following variables:

1. Components of actual weighted miss before each correction.
2. Magnitude of actual weighted miss before each correction.
3. Magnitude of correction velocity actually fired at each time.
4. Components of actual weighted final miss.
5. Magnitude of actual weighted final miss.
6. Total midcourse velocity actually fired.

DETAILS OF THE MIDCOURSE GUIDANCE PROBLEM

In order to understand the model and equations used in the simulation, it is first necessary to understand the logic of the midcourse guidance problem. A brief description of a generalized midcourse guidance scheme is presented here to form a basis for the explanation of the simulation.

In the description of any midcourse guidance scheme it is necessary to make a distinction between variables that occur in the physical or "real" world and those that occur in the guidance logic. In all cases, the logic variables are only estimates of the physical variables. Even the statistics used in the logic are only estimates of the physical situation. The equations used for the physical world simulation must be correct to the best knowledge available, but the equations used for the guidance logic need only to be identical with the ones used in the real-time logic being simulated. That is, if inaccurate guidance equations are used the results will still be correct for that inaccurate guidance system; but if inaccurate physical equations are used the results will not be correct for any guidance system.

The following list describes the flow of information in the generalized midcourse guidance scheme diagrammed in Figure 1.

1) The injection conditions and physical constants determine the trajectory before any corrections are made.

2) The trajectory parameters at the first correction time are determined by the injection conditions and physical constants.

3) The tracking observations can be considered to be functions of position, velocity, and physical constants at any time along the free-flight trajectory. It is particularly convenient, however, to use the correction time under consideration as epoch (reference time).

4) The tracking observations have random errors which are assumed to add to the observations that correspond to the trajectory.

5) The tracking estimate of trajectory parameters and some physical constants is formed by operating on the observations.

6) The a priori estimate for the parameters at time 1 is that they are all nominal.

7) The a priori estimate is combined with the tracking estimate to produce the final values on which the correction to be commanded is based.

8) The midcourse guidance logic operates on the combined estimate in order to determine the correction velocity to be commanded. In a fixed-time logic, a correction is always fired, but in a variable-time logic the decision may be to wait for more tracking data before firing.

9) When a correction is commanded, random errors in magnitude and direction of the firing occur.

10) The correction fired is the commanded correction plus the random execution errors.

11) The trajectory parameters before the second correction time are determined by the parameters before the first correction, physical constants, and the first correction.

12) Tracking observations are obtained in the second interval just as they were in the first.

13) Tracking noise is again present.

14) The second tracking estimate is formed in the same manner as the first.

15) The a priori estimate of the parameters at t_2 is formed by using the estimate of the parameters at t_1 plus the commanded correction as initial conditions for the period from t_1 to t_2.

16) The combined estimate of the parameters at t_2 is formed in the same manner as at t_1, and the sequence continues until the last possible correction time is used.

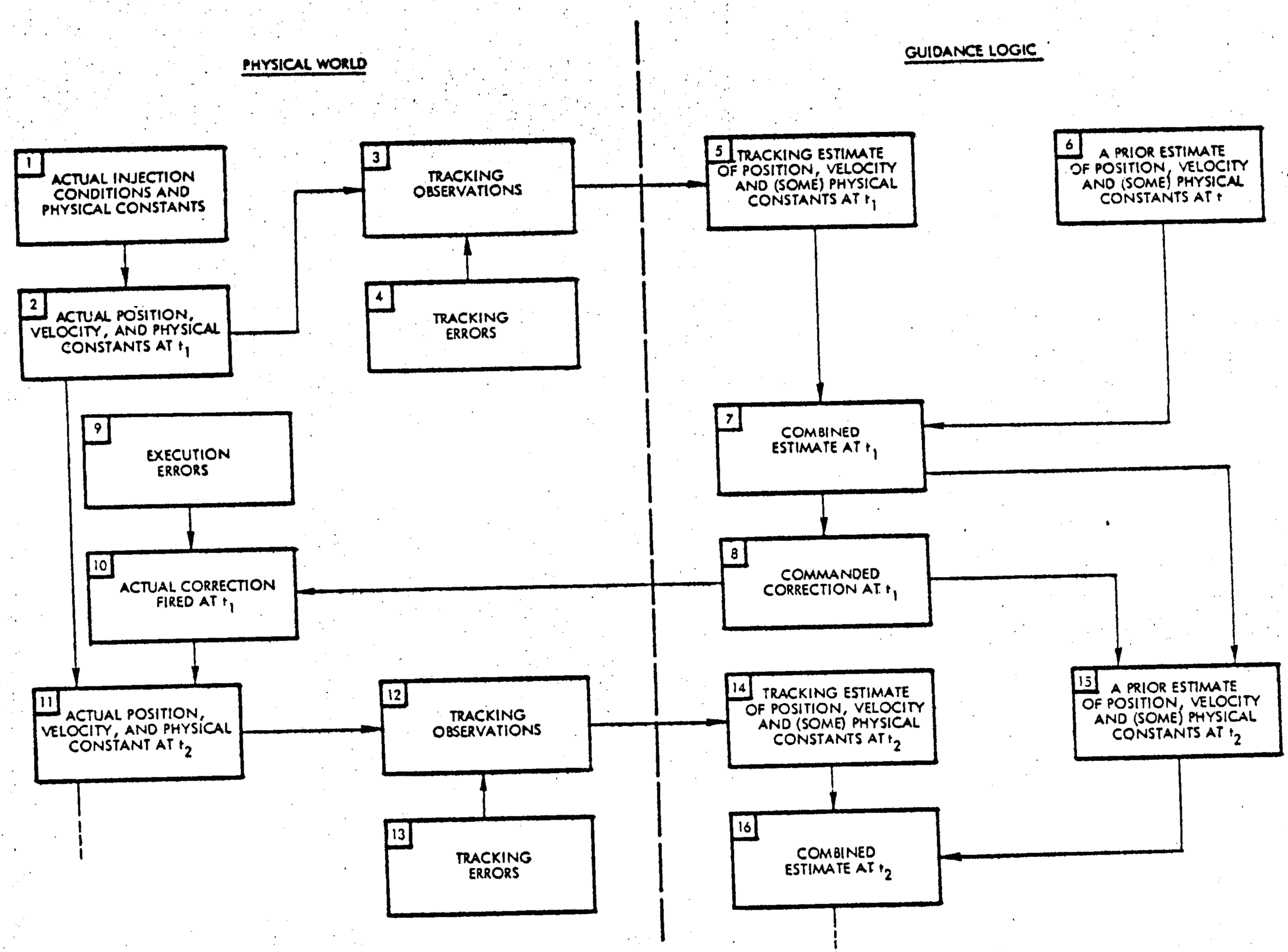

Figure 1. Information Flow in a Generalized Midcourse Guidance Scheme

ANALYTIC MODEL USED

A linear perturbation model is used in the simulation. The trajectory at each time is described by the variation of a state vector from its reference value for that time.

The state vector used consists of position, velocity, and physical constants. Station locations or any other parameters of interest can be included in the category of physical constants (provided that they are constant at all times along each sample trajectory).

The variation of the state vector at any time is assumed to be linearly related to the variation of the state vector at any other time. The partial derivatives necessary to specify this relation are calculated from a reference trajectory, which is the result of calculating a trajectory from the reference injection conditions with the reference physical constants.

The word "nominal" is not used to describe the reference trajectory, since it tends to confuse the notation. The reference trajectory is only the basis for linearization, and is not necessarily the "optimum" or "correct" trajectory.

Table 1 summarizes the symbols used in the Monte Carlo simulation.

Table 1. List of Symbols

X	=	vector of position and velocity
Y	=	vector of physical constants that are to be estimated from tracking data
Z	=	vector of physical constants that are not estimated from tracking
δX, δY, δZ	=	variations of X, Y, Z from the reference values
δX_{Ai}, δY_{Ai}, δZ_{Ai}	=	actual values of δX, δY, and δZ before the i-th correction
δX_{Pi}, δY_{Pi}	=	a priori values of δX and δY for the i-th correction
δX_{Ei}, δY_{Ei}	=	estimated values of δX, δY before the i-th correction
V_i	=	velocity commanded for i-th correction
δV_i	=	error in executing commanded velocity
Σ_0	=	actual covariance matrix of δX_{A0}, δY_{A0}, δZ_{A0}
Σ_0'	=	estimate of covariance matrix of δX_{A0}, δY_{A0}
Σ_{Pi}	=	estimate of covariance matrix of δX_{Pi}, δY_{Pi}
Σ_{Ei}	=	estimate of covariance matrix of δX_{Ei}, δY_{Ei}
$\Sigma_{\delta V_i}$	=	estimate of covariance matrix of δV_i
Λ_{Ai}	=	actual covariance matrix of execution errors that are independent between corrections
Λ_i	=	estimated covariance matrix of execution errors that are independent between corrections
Λ_0	=	actual covariance matrix of execution errors that are constant for all corrections
M	=	miss in usual units
t_j	=	tolerance of j-th component of M
M'	=	miss in units of tolerances
N	=	number of midcourse corrections
R	=	number of random cycles
i	=	correction number
k	=	cycle number

EQUATIONS

The equations used in the simulation can conveniently be considered in relation to Figure 1. In the following explanation, the equations will be identified with the numbers of the corresponding blocks in the diagram.

1. Injection Conditions

The deviations of the actual injection conditions and physical constants from the reference values are selected randomly with the covariance matrix $\sum_0$.

2. Deviations at Time 1

The deviation of the state vector from the reference value at each time is assumed to be linearly related to the deviation at the previous time. For the first midcourse correction time

$$\delta X_{A1} = \frac{\partial X_1}{\partial X_0} \delta X_{A0} + \frac{\partial X_1}{\partial Y_0} \delta Y_{A0} + \frac{\partial X_1}{\partial Z_0} \delta Z_{A0}$$

$$\delta Y_{A1} = \delta Y_{A0}$$

$$\delta Z_{A1} = \delta Z_{A0}$$

where

δX = deviations of position and velocity from the reference value

δY = deviations of the measured physical constants from the reference values

δZ = deviations of the unmeasured physical constants from the reference values

The subscript A means actual (rather than estimated) value, and the subscript i means at the i-th time (here 0 or 1).

Since the physical constants always have the same value regardless of the time along the trajectory it may seem unnecessary to use a subscript to indicate the time which is being considered. The subscript is desirable, however, for two reasons. First, it is consistent with the necessary subscript on the position and velocity vector. Second, the subscript emphasizes that the partials with respect to physical constants are evaluated with the deviation acting over a specific interval. For example $\frac{\partial X_3}{\delta Y_2}$ is used to calculate the deviation in X at time 3 caused by a deviation in Y acting from time 2 to time 3. If the time subscript were not used on Y, then the partial would probably be written as $\frac{\partial X_3}{\delta Y}$, which is clearly ambiguous, since the interval over which δY is assumed to act is not specified.

3. 4. Tracking Observations

The deviations of the tracking observations from the reference values are assumed to be linearly related to the deviation of the state vector and to the tracking noise by the equation

$$\delta R_{0,1} = A_1^{0,1} \begin{bmatrix} \delta X_{A1} \\ \delta Y_{A1} \end{bmatrix} + B_1^{0,1}\, \delta Z_{A1} + N_{0,1}$$

where

$$A_1^{0,1} = \begin{bmatrix} \frac{\partial R_{0,1}}{\delta X_1} & \frac{\partial R_{0,1}}{\delta Y_1} \end{bmatrix}$$

$$B_1^{0,1} = \begin{bmatrix} \frac{\partial R_{0,1}}{\partial Z_1} \end{bmatrix}$$

$\delta R_{0,1}$ = deviations of observation during the interval between time 0 and time 1

$N_{0,1}$ = tracking noise during the interval between time 0 and time 1

The tracking noise is random with covariance matrix $M_{0,1}$. (In the program the observation deviations are not actually calculated.)

5. Tracking Estimate

The tracking estimate (indicated by the T subscript) used in the normal weighted-least-squares estimate and is given by

$$\begin{bmatrix} \delta X_{T1} \\ \delta Y_{T1} \end{bmatrix} = (A^T WA)_1^{-1} \; (A^T W)_1 \; \delta R_{0,1}$$

$$= \begin{bmatrix} \delta X_{A1} \\ \delta Y_{A1} \end{bmatrix} + (A^T WA)_1^{-1} \; (A^T WB)_1 \; \delta Z_{A1}$$

$$+ (A^T WA)_1^{-1} \; (A^T W)_1 \; N_{0,1}$$

where

$$(A^T WA)_1 = (A_1^{0,1})^T \; W_{0,1} \; A_1^{0,1}$$

$W_{0,1}$ = the weighting matrix for the observations from time 0 to time 1

$(A^T W)_1$ and $(A^T WB)_1$ follow the same simplified notation used in $(A^T WA)_1$.

The usual least-squares estimate of orbit parameters uses the beginning of the tracking interval or some earlier time as epoch. For the midcourse simulation, however, it is convenient to have epoch at the end of the tracking interval, that is, at the correction time being considered. One way to accomplish the epoch shift is to update the normal matrix before calculating the estimate. The equation used for updating the normal matrix is (Reference 4).

$$(A^T WA)_i^{i-1,i} = \begin{bmatrix} \left[\dfrac{\partial X_i}{\partial X_{i-1}}\right]^{-1} & -\left[\dfrac{\partial X_i}{\partial X_{i-1}}\right]^{-1} \dfrac{\partial X_i}{\partial Y_{i-1}} \\ 0 & I \end{bmatrix}^T (A^T WA)_{i-1}^{i-1,i} \begin{bmatrix} \left[\dfrac{\partial X_i}{\partial X_{i-1}}\right]^{-1} & -\left[\dfrac{\partial X_i}{\partial X_{i-1}}\right]^{-1} \dfrac{\partial X_i}{\partial Y_{i-1}} \\ 0 & I \end{bmatrix}$$

The equations for updating the noise effect covariance matrix $(A^T WMWA)_{i-1}^{i-1,i}$ to $(A^T WMWA)_i^{i-1,i}$ and for updating the physical constant effect matrix $(A^T WB)^{i-1,i}$ to $(A^T WB)^{i-1,i}$ have the same form.

6. The a priori estimate $\begin{bmatrix} \delta X_{P1} \\ \delta Y_{P1} \end{bmatrix}$ for the deviations at the end of the tracking period is zero, since the mean of the distribution is the best estimate in the absence of tracking data. The covariance matrix of this estimate is simply the injection covariance matrix updated to time 1, and is given by the equation

$$\Sigma_{P1} = Q_{0,1} \Sigma_0' Q_{0,1}^T$$

where

$$Q_{0,1} = \begin{bmatrix} \dfrac{\partial X_1}{\partial X_0} & \dfrac{\partial X_1}{\partial X_0} \\ 0 & I \end{bmatrix}$$

and Σ_0' is the estimate of the injection covariance matrix (including the physical constants to be measured with tracking).

7. 16. <u>Combined Estimate of Deviations</u>

For the combined estimate, the tracking estimate and the a priori estimate are combined as if they were statistically independent. The estimated covariance matrix of the combined estimate for the i-th time is

$$\Sigma_{Ei} = \left[\Sigma_{P_i}^{-1} + (A^T WA)_i \right]^{-1}$$

where $(A^T WA)_i^{-1}$ is used as the covariance matrix of the tracking estimate.

This is correct if the weighting matrix is the inverse of the noise covariance matrix and $\delta Z = 0$. Otherwise it is not the best estimate, but it is used here in order to agree with the usual practice. It must be remembered that the logic used in the simulation should agree with the real-time logic, whether it is optimum or not.

The combined estimate is given by

$$\begin{bmatrix} \delta X_{Ei} \\ \delta Y_{Ei} \end{bmatrix} = \Sigma_{Ei} \left(\Sigma_{Pi}^{-1} \begin{bmatrix} \delta X_{Pi} \\ \delta Y_{Pi} \end{bmatrix} + (A^T WA)_i \begin{bmatrix} \delta X_{Ti} \\ \delta Y_{Ti} \end{bmatrix} \right)$$

The calculation of $\begin{bmatrix} \delta X_{Ti} \\ \delta Y_{Ti} \end{bmatrix}$ involves inverting $(A^T WA)_i$, which may be impossible. However, $(A^T WA)_i \begin{bmatrix} \delta X_{Ti} \\ \delta Y_{Ti} \end{bmatrix}$ is actually required, and it can be calculated even if $(A^T WA)_i$ cannot be inverted. The equation is

$$(A^T WA)_i \begin{bmatrix} \delta X_{Ti} \\ \delta Y_{Ti} \end{bmatrix} = (A^T WA)_i \begin{bmatrix} \delta X_{Ai} \\ \delta Y_{Ai} \end{bmatrix} + (A^T WB)\ Z_{Ai} + (A^T WN)_i$$

$(A^T WN)_i$ is the tracking noise effect and is random with covariance matrix $(A^T WMWA)_i$ where M is the covariance matrix of N.

8. Commanded Correction Velocity

The command correction velocity is calculated to minimize the weighted sum of the squares of the components of miss. Since there are only three degrees of freedom at each correction time, only three components of miss can possibly be corrected at one time. However, the velocity components can be chosen to minimize the sum of squares of more than three miss components. At most, six components of miss will be required, since six components fully define the trajectory. For example, position and velocity at the nominal time of arrival can be used as the six components of miss.

If only three components of miss are used, the least-squares correction will nominally reduce the miss to zero. Fewer than three components cannot be used with this system.

The estimated miss before the i-th correction is

$$M_{Ei} = \frac{\partial M}{\partial X_i} \delta X_{Ei} + \frac{\partial M}{\partial Y_i} \delta Y_{Ei}$$

The quantity to be minimized is

$$\left(M_{Ei} + \frac{\partial M}{\partial V_i} V_i\right)^T T \left(M_{E_i} + \frac{\partial M}{\partial V_i} V_i\right)$$

where

$$T = \left[\frac{1}{t_i^2}\right] = \text{diagonal weighting matrix for miss}$$

$$t_i = \text{tolerance on the j-th component of miss}$$

The resulting commanded velocity is

$$V_i = -\left[\left(\frac{\partial M}{\partial V_i}\right)^T T \frac{\partial M}{\partial V_i}\right]^{-1} \left(\frac{\partial M}{\partial V_i}\right)^T T M_{Ei}$$

Since the weighting matrix (T) for miss is diagonal, its effect can be taken into account by dividing each miss partial by the tolerance for the corresponding component of miss. This procedure reduces the number of matrix multiplications required and therefore saves computing time.

9. Execution Errors

The errors associated with executing the commanded velocity depend on the commanded correction and on the randomly selected values of the basic errors.

The equation is

$$\delta V_i = C_i \begin{bmatrix} K_{bi} \\ V_{\theta i} \\ V_{\phi i} \\ K_{pi} \\ \delta\theta_i \\ \delta\phi_i \end{bmatrix}$$

where

$$C_i = \begin{bmatrix} \frac{V_{xi}}{V_i} & \frac{-V_{yi}}{V_i} & \frac{-V_{zi}\, V_{xi}}{V_i\, W_i} & V_{xi} & -V_{yi} & \frac{-V_{zi}\, V_{xi}}{W_i} \\ \frac{V_{yi}}{V_i} & \frac{V_{xi}}{V_i} & \frac{-V_{zi}\, V_{yi}}{V_i\, W_i} & V_{yi} & V_{xi} & \frac{-V_{zi}\, V_{yi}}{W_i} \\ \frac{V_{zi}}{V_i} & 0 & \frac{W_i}{V_i} & V_{zi} & 0 & W_i \end{bmatrix}$$

$$W_i = \sqrt{V_{xi}^2 + V_{yi}^2} \quad ,$$

$$\begin{bmatrix} K_{bi} \\ V_{\theta i} \\ V_{\phi i} \\ K_{pi} \\ \delta\theta_i \\ \delta\phi_i \end{bmatrix} \text{ is random.}$$

x, y, and z as subscripts indicate components along the coordinate axes

K_{bi} = bias error coefficient (velocity units)

$V_{\theta i}$ = bias velocity (autopilot) error in the θ direction (velocity units)

$V_{\phi i}$ = bias velocity (autopilot) error in the ϕ direction (velocity units(

K_{pi} = proportional error coefficient (dimensionless)

$\delta\theta_i$ = error in orientation in x-y plane (radians)

$\delta\phi_i$ = error in orientation out of x-y plane (radians)

The random errors can be correlated between corrections in several ways. A simple type of correlation is included by selecting a component of the basic errors that is constant for all corrections of a particular flight. The other component is selected for each time. The covariance matrix for the fixed component is Λ_0, and the covariance matrix for the variable component is Λ_{Ai}.

The estimated covariance matrix of execution errors is used in forming the a priori estimate for the next correction time.

The equation is

$$\Sigma_{\delta Vi} = C_i \Lambda_i C_i^T$$

where Λ_i = estimate of Λ_{Ai}

This estimate neglects any correlation between corrections.

10. Actual Correction

The actual correction velocity is simply the sum of the commanded correction and the execution errors.

11. Actual Deviations at Time i

At any time after the first correction the actual deviations are given by

$$\delta X_{Ai} = \frac{\partial X_i}{\partial X_{i-1}} \left(\delta X_{Ai-1} + V_{i-1} + \delta V_{i-1}\right) + \frac{\partial X_i}{\partial Y_{i-1}} \delta Y_{Ai-1} + \frac{\partial X_i}{\partial Z_{i-1}} \delta Z_{Ai-1}$$

$$\delta Y_{Ai} = \delta Y_{Ai-1}$$

$$\delta Z_{Ai} = \delta Z_{Ai-1}$$

where $(V_{i-1} + \delta V_{i-1})$ is added to the velocity part of δX_{Ai-1}.

If the definitions are made that $V_0 = 0$ and $\delta V_0 = 0$, then the following general equation can be used for all cases.

$$\begin{bmatrix} \delta X_{Ai} \\ \delta Y_{Ai} \\ \delta Z_{Ai} \end{bmatrix} = P_{i,i-1} \left(\begin{bmatrix} \delta X_{Ai-1} \\ \delta Y_{Ai-1} \\ \delta Z_{Ai-1} \end{bmatrix} + V_{i-1} + \delta V_{i-1} \right), \quad i \geq 1.$$

$$\text{where } P_{i,i-1} = \begin{bmatrix} \frac{\partial X_i}{\partial X_{i-1}} & \frac{\partial X_i}{\partial Y_{i-1}} & \frac{\partial X_i}{\partial Z_{i-1}} \\ 0 & I & 0 \\ 0 & 0 & I \end{bmatrix}$$

12. 13. 14. General Tracking Estimate

The general tracking estimate is given by

$$\begin{bmatrix} \delta X_{Ti} \\ \delta Y_{Ti} \end{bmatrix} = \begin{bmatrix} \delta X_{Ai} \\ \delta Y_{Ai} \end{bmatrix} + (A^T WA)_i^{-1} (A^T WB)_i \, \delta Z_{Ai} + (A^T WA)_i^{-1} (A^T W)_i \, N_{i-1,i}$$

This estimate is not calculated separately in the simulation because the matrix $A^T WA$ may be singular. Instead, the combined estimate is calculated since it does not require $(A^T WA)^{-1}$.

15. General A Priori Estimate

For all correction times after the first the a priori estimate is the result of updating the sum of the previous combined estimate and the previous commanded correction. The equation for the i-th time is

$$\begin{bmatrix} \delta X_{Pi} \\ \delta Y_{Pi} \end{bmatrix} = Q_{i,i-1} \left(\begin{bmatrix} \delta X_{Ei-1} \\ \delta Y_{Ei-1} \end{bmatrix} + V_{i-1} \right)$$

where

$$Q_{i,i-1} = \begin{bmatrix} \dfrac{\partial X_i}{\partial X_{i-1}} & \dfrac{\partial X_i}{\partial Y_{i-1}} \\ 0 & I \end{bmatrix}$$

and it is understood that V_{i-1} adds to the velocity part of the state vector. IF $\begin{bmatrix} \delta X_{E0} \\ \delta Y_{E0} \end{bmatrix}$ and V_0 are defined to be zero, then this equation is also correct for $i = 1$.

The estimated covariance matrix of the a priori estimate for the i-th time is

$$\Sigma_{Pi} = Q_{i,i-1} \left(\Sigma_{Ei-1} + \Sigma_{\delta V_{i-1}} \right) Q^T_{i,i-1}$$

Since $\Sigma_{E0} = \Sigma'_0$ and $\Sigma_{\delta V_0}$ can be defined to be zero, the equation is valid even for $i = 1$.

SEQUENCE OF CALCULATIONS IN SIMULATION

Figure 2 shows a logical sequence for performing the calculations involved in the simulation (exclusive of processing the data). It should be noticed that there are two loops involved. The first loop calculates the deviations and corrections for each correction time on a randomly selected trajectory, and the second loop selects the trajectories.

The input tape used is normally generated by TAPP-I and contains the updating partials and tracking matrices. The output tape is used to store the sample values that are used as input for the processing routine.

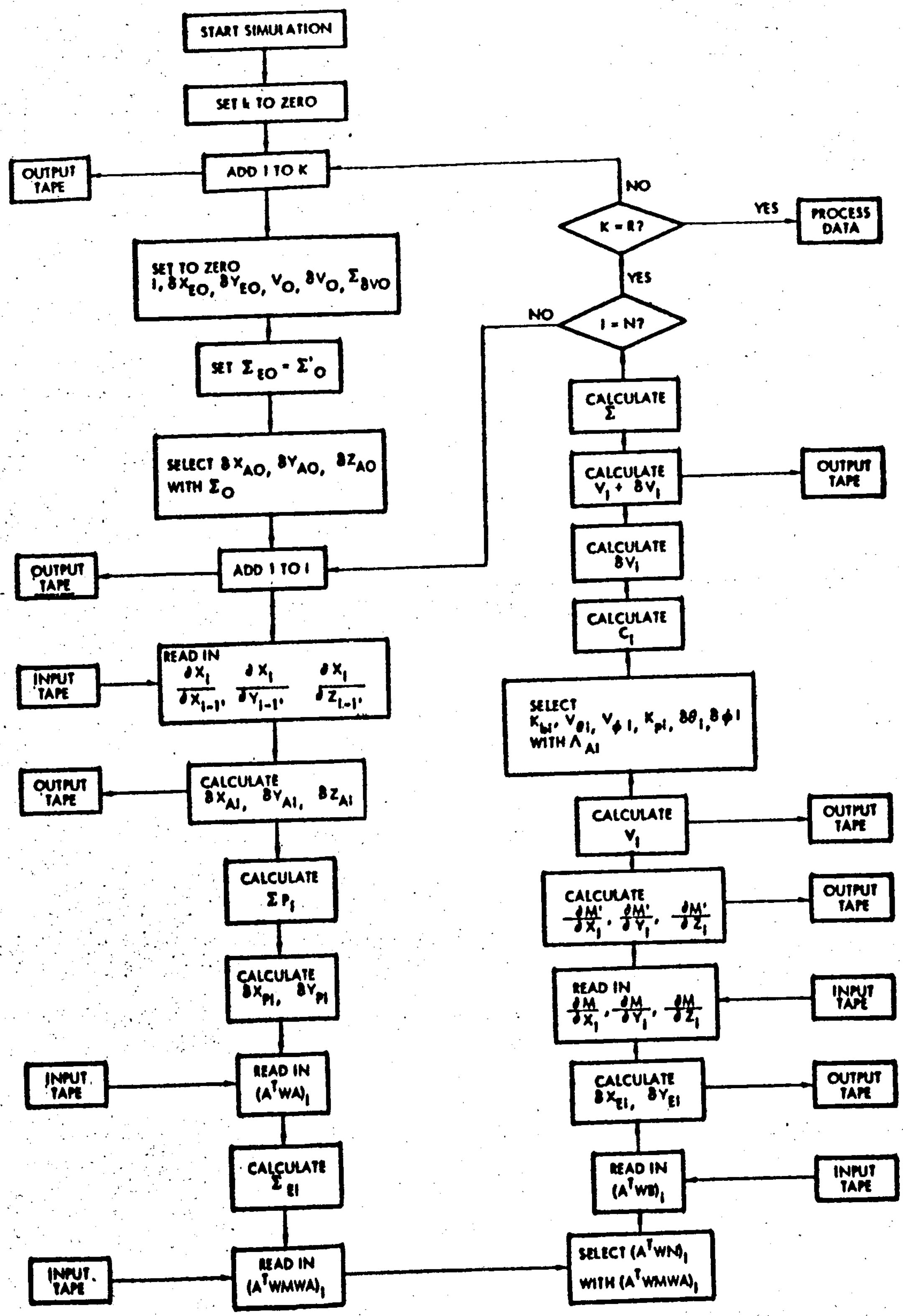

Figure 2. Simulation Logic Diagram

FUNCTION OF THE PROCESSOR

The processor is the routine used to calculate the desired statistics from the sample values resulting from the simulation. Many different kinds of statistics may prove useful from time to time. For this reason all of the sample values from the simulation that are likely to be used are currently stored on tape, even though some of them are not now used. Thus, it probably will not be necessary to modify the simulation output tape whenever a new processor is programmed.

The inputs to the processor that are available from the tape are the following:

$$\delta X_{Ai}, \quad \delta Y_{Ai}, \quad \delta Z_{Ai}$$

$$\delta X_{Ei}, \quad \delta Y_{Ei}$$

$$\frac{\partial M'}{\partial X_i}, \quad \frac{\partial M'}{\partial Y_i}, \quad \frac{\partial M'}{\partial Z_i}$$

$$V_i$$

$$V_i + \delta V_i$$

The current processor compiles cumulative distributions of several variables of interest from the random samples generated by the simulation. Specifically, distributions of the following variables are obtained:

1. Components of actual weighted miss before each correction.
2. Magnitude of actual weighted miss before each correction.
3. Magnitude of correction velocity actually fired at each time.
4. Components of actual weighted final miss.
5. Magnitude of actual weighted final miss.
6. Total midcourse velocity actually fired.

EQUATIONS FOR PROCESSOR

Since miss is never calculated explicitly in the simulation, it must be calculated in the processor. The weighted miss before the i-th correction is calculated from

$$M'_{Ai} = \frac{\partial M'}{\delta X_i} \delta X_{Ai} + \frac{\partial M'}{\partial Y_i} \delta Y_{Ai} + \frac{\partial M'}{\delta Z_i} \delta Z_{Ai}$$

The weighted miss after the last (N-th) correction is calculated from

$$M'_{AF} = M'_{AN} + \frac{\partial M'}{\delta V_N} (V_N + \delta V_N)$$

The velocity actually fired at each time is calculated by the simulation. The magnitudes of each of the above quantities are calculated from

$$\left| M'_{Ai} \right| = \left[(M'_{Ai})^T \; M_{Ai} \right]^{1/2}$$

$$\left| M'_{AN} \right| = \left[(M'_{AN})^T \; M'_{AN} \right]^{1/2}$$

$$\left| V_i + \delta V_i \right| = \left[(V_i + \delta V_i)^T \; (V_i + \delta V_i) \right]^{1/2}$$

The total velocity fired for any one mission is

$$\sum_{i=1}^{N} \left| V_i + \delta V_i \right| .$$

SEQUENCE OF CALCULATIONS IN PROCESSOR

Figure 3 shows a sequence for making the calculations in the processor. This is not necessarily the exact way that the program operates, but it is equivalent.

The processor operation may be considered in two parts. First, the variables of interest are calculated from the data supplied by the simulation. These variables are misses and midcourse velocities in the current processor. Second, the desired statistics are calculated. Cumulative distributions are currently used.

The results of the first phase of processing are stored on a tape which is then read back as the input to the second phase. With the current method of processing this tape is read only once with no rewinding (an obvious advantage over rewinding). This is made possible by keeping track of the maximum and minimum values of each variable as it is calculated instead of searching the tape at the end.

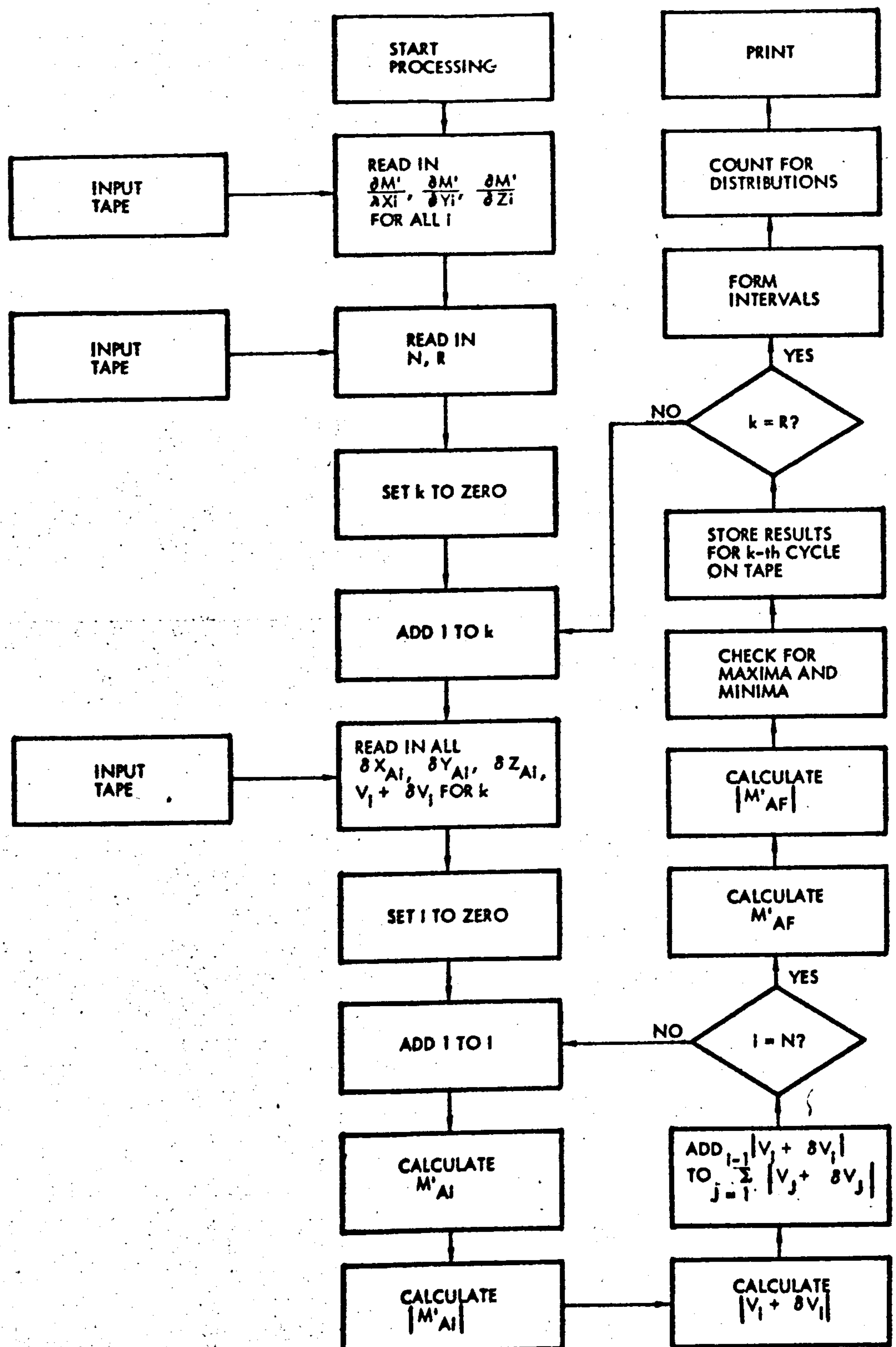

Figure 3. Processor Calculation Sequence

INPUTS REQUIRED

Inputs Required by TAPP-II

Part of the inputs to TAPP-II are made by hand, and the remaining inputs are normally obtained from TAPP-I. Additional manual inputs to TAPP-I are also required in order to generate the data required by TAPP-II.

Manual Inputs to TAPP-II

The injection and execution covariance matrices are manual inputs. These are:

Σ_0 = the actual injection covariance matrix

Σ_0' = the estimated injection covariance matrix

Λ_{Ai} = the actual covariance matrix of the independent portion of the execution errors

Λ_i = the estimated covariance matrix of the independent portion of the execution errors

Λ_0 = the actual covariance matrix of the correlated (bias) portion of the execution errors

Several numbers necessary for bookkeeping purposes are also required. These are:

1. The number of measured physical constants
2. The number of unmeasured physical constants
3. The number of correction times (N)
4. The number of random cycles (R)
5. The number of levels to be used in the cumulative distributions

The tolerances (t_j) on the miss are also required.

Additional Inputs to TAPP-I

The following inputs to TAPP-I are required in addition to the inputs required for a normal TAPP-I run:

1. The midcourse correction times

2. The physical constants to be considered, with the measured constants first and the unmeasured constants second.

3. The components of miss to be considered.

Inputs to TAPP-II from TAPP-I

The updating and tracking accuracy data are supplied by TAPP-I. These are the following:

1. Updating partials

$$\frac{\partial X_i}{\partial X_{i-1}}, \frac{\partial X_i}{\partial Y_{i-1}}, \frac{\partial X_i}{\partial Z_{i-1}}$$

2. Normal matrix for times t_i

$$(A^T WA)_i$$

including the effects of orbit parameters and the estimated physical constants.

3. Matrix to account for effects of physical constants not estimated.

$$(A^T WB)_i$$

4. Partials of miss with respect to orbit parameters and all physical constants considered at t_i.

$$\frac{\partial M}{\partial X_i} \quad \frac{\partial M}{\partial Y_i} \quad \frac{\partial M}{\partial Z_i}$$

5. Actual tracking noise effect covariance matrix

$$(A^T WMWA)_i$$

These values are supplied to TAPP-I on tape.

MODES OF OPERATION

Because of the large amount of storage needed for the TAPP-II simulation it will be necessary to write much of the information on tape. While the use of tapes may appear to be a disadvantage, it allows at least three distinct modes of operation, which can be used to reduce computing time when thoroughly investigating a particular trajectory.

MODE 1:

The normal operating sequence for the Monte Carlo simulation is the following:

1. The TAPP-I program is read into core. It calculates all of the partial derivatives needed to run the simulation and to process the data and writes them on tape.

2. The simulation program and the partial tape are read into core. This destroys the TAPP-I program. The simulation calculates for the desired number of cycles and writes the raw data on tape.

3. The processor program is read into core. This destroys the simulation program, but all of the partials are saved. The processor then calculates the desired outputs from the raw data tape.

All of these steps are done automatically in the normal (first run) case, but the tapes containing the partials and the raw data can be saved for re-runs.

MODE 2:

Re-runs can be made with the same partials by simply starting with Step 2. This saves the time required to calculate the partials when things other than the trajectory, correction times, or variables considered are changed.

MODE 3:

If only the method of processing the data is changed, the calculations can begin at Step 3 with the addition of reading in the partial tape, since the partials will not already be in core for this case.

A third type of re-run is the calculation of a new type of miss from previously obtained raw data. For this re-run it is necessary to calculate the required partials with TAPP-I. Since two different kinds of miss can normally be specified, this third type of re-run would be required only if a third type of miss were needed. Since the probability of needing distributions of three types of miss is low, this type of calculation is not included as a separate mode. If such a result is actually necessary, it may be obtained by running twice in Mode 1.

REFERENCES

1. Pace, W. H., Jr. "Monte Carlo Simulation of Midcourse Guidance for TAPP-II", Space Technology Laboratories Memo 9861. 5-55, 1 March 1962.

2. Pace, W. H., Jr. "Modifications and Additions to the Monte Carlo Simulation of Midcourse Guidance for TAPP-II", Space Technology Laboratories Memo 9861. 5-71, 27 April 1962.

3. Pace, W. H., Jr. "Modes of Operation of the Monte Carlo Simulation of Midcourse Guidance for TAPP-II", Space Technology Laboratories Memo 9861. 5-72, 27 April 1962.

4. Pace, W. H., Jr. "Equations for Updating Tracking Matrices in TAPP-II", Space Technology Laboratories Memo 9861. 5-113, 21 September 1962.

5. Cutting, E. "Preliminary Specification for TAPP (Mod II)", Jet Propulsion Laboratory Memo, 5 March 1962.