X-553-69-544
PREPRINT

NASA TM X- 63979

GEOSTAR-I
A
GEOPOTENTIAL
AND STATION POSITION RECOVERY SYSTEM

C. E. VELEZ
G. P. BRODSKY

NOVEMBER 1969

GSFC

GODDARD SPACE FLIGHT CENTER
GREENBELT, MARYLAND

X-553-69-544
PREPRINT

GEOSTAR-I
A GEOPOTENTIAL AND STATION POSITION RECOVERY SYSTEM

C. E. Velez
G. P. Brodsky
Program Systems Branch
Mission Trajectory and Analysis Division

November 1969

GODDARD SPACE FLIGHT CENTER
Greenbelt, Maryland

ABSTRACT

The GEOSTAR-I multiple arc geopotential and station position estimation system is described. A detailed presentation of the mathematical model and formal documentation of the principal program components are included.

ACKNOWLEDGMENTS

The authors wish to gratefully acknowledge the guidance and technical advice given in the satellite geodesy and orbit determination areas by Mssrs. Frank Lerch, Carl Wagner and James Murphy from the Mission Trajectory Determination Branch, Mission and Trajectory Analysis Division. The programming, testing and documentation assistance given by Mrs. Gail Finley, Mr. Roscoe Pullen, Mr. K. Ludwig, and co-workers, IBM Corporation, under NASA Contract No. NAS5-10022, and Mssrs. Jerry Morris and Richard Hamblen, Computer Usage Development Corporation, under NASA Contract No. NAS5-11726 is also acknowledged.

CONTENTS

CONTENTS (continued)

GEOSTAR-I
A GEOPOTENTIAL AND STATION POSITION RECOVERY SYSTEM

by
C. E. Velez and G. P. Brodsky
Goddard Space Flight Center

I. INTRODUCTION

The GEOSTAR-I system is a multiple arc, multiple satellite geopotential coefficient and station position recovery system. Its principal feature is the capability to process the full spectrum of available tracking data for the determination of a number of geodetic parameters, simultaneously utilizing the accuracy of the measurements and maintaining computational efficiency.

The GEOSTAR-I system is basically comprised of:

(1) A modified version of NONAME which is a single arc, definitive orbit and station recovery system consisting of ODP and ancillary data handling programs (Reference 1).

(2) The matrix algebra programs MERGE, SOLVE and EIGENVALUE from the multiple arc lunar potential estimation program LUNGFISH (Reference 2).

GEOSTAR-I is designed to operate in either a single or multiple arc mode. In the single arc mode, the parameter set is solved for by using a differential correction process with a least squares estimator which uses the a priori estimates of the parameters and their covariance matrices.

In the multiple arc mode, the single arc program generates a system of normal equations which corresponds to each individual arc and parameter set. The matrix algebra programs combine and process these systems of equations for a multiple arc, linear least squares solution, or essentially one differential correction iteration.

Under the current capabilities of the system, each arc can have a parameter set consisting of a maximum of 50 parameters of the following types:

- Geopotential coefficients
- Tracking station coordinates
- Arc dependent parameters including state, drag and solar constants, and, in the single arc mode, tracking instrument errors.

The maximum number of parameters which can be processed in a multiple arc solution is 500. Moreover, by using the same system of normal equations, the determination of any subset of the original parameter set is permitted by the "suppress" feature of the system.

Several optional methods, available in the system, enable the detection and analysis of an ill-conditioned normal matrix. These methods include matrix pseudoinversion with or without rank reduction, gradient, or steepest descent methods, and parameter transformations in which, by a canonical decomposition of the normal equations, the linear combination of parameters well-determined by the data can be detected.

Some of the principal capabilities of the single arc program include:

- Cowell type numerical integration of the equations of motion and linear variational equations in which the integrator used for position and velocity is independent of the integrator used to obtain the partials of position with respect to the parameters.
- Simulated data (without noise) generation, and processing capabilities, including the processing of rectangular coordinate data.
- Complete flexibility in the choice of parameters to be estimated, so that in the single or multiple arc operations, arc dependent parameters may be excluded.
- Position partials generation capability in the orbit generator mode of the program, permitting independent investigations on the sensitivity of the satellite position with respect to variations in the geopotential harmonics.
- All the current data reduction and analysis capabilities of the single arc operational NONAME system.
- An optional variable stepsize integrator, providing efficient integration of the equations of motion and the variational equations associated with satellites having high eccentricities.

The GEOSTAR-I system, written in 360 FORTRAN, is currently operational on the IBM 360, Models 95 and 91. The operating instructions, system testing procedures and test results can be found in Reference 3.

II. SYSTEM DESCRIPTION

The primary module of the GEOSTAR-I system is a modified version of the single arc orbit determination and geodetic parameter estimation program from the NONAME system. All the current data reduction capabilities, operating instructions, and overall design of this program have been maintained. The principal modifications made to this program, as discussed in the following sections of this report, are the incorporation of improved numerical integration techniques, the computation of partial derivatives for geopotential coefficient estimation, and interfaces with LUNGFISH multiple arc matrix algebra programs. The details of the present single arc NONAME ODP not directly related to these modifications will not be discussed in this report but may be found in Reference 1.

In the single arc mode, the GEOSTAR-I ODP can be used to estimate orbital and geodetic parameters from satellite tracking data by differential corrections using a Bayesian least squares method of estimation. The maximum single arc parameter set permitted is 50 parameters, optionally selected from the following:

- Geopoential coefficients through $C_{30,30}$ and $S_{30,30}$ (maximum 50)
- Tracking station coordinates (maximum 30)
- State, or epoch position and velocity vectors
- Physical constants related to atmospheric drag and solar radiation forces (maximum 2)
- Tracking errors, including measurement biases and station timing biases (maximum 44).

In the multiple arc mode, the single arc ODP is used to construct a system of normal equations for a specified parameter set from the given parameter estimates and observations for each arc. These normal equations are in turn converted to the LUNGFISH matrix processors' required format (B matrix), and written on tape for use in either the MERGE or SOLVE programs. Each B matrix tape contains arc identification, normal equations, parameter estimates, and parameter identification labels defining the parameter set associated with the corresponding normal equations.

The MERGE program is used to copy the various single arc B matrix tapes onto one logical merged tape which may consist of several physical reels of tape. The resulting "merged" B matrix can then be input to the SOLVE program for a multiple arc solution.

More precisely, the SOLVE program performs the following functions:

(i) On option, selected parameters are "suppressed" or deleted from each individual input B matrix, thus providing selectivity in the parameters to be used in a particular multiple arc solution;

(ii) The arc parameters, if present, are then eliminated from each B matrix, producing 1) a set of arc independent or "reduced" matrices which are used to solve for the arc independent parameters; and 2) a set of "backsubstitution matrices" which are used to solve for the arc dependent parameters;

(iii) The reduced matrices are then combined (essentially added), producing the "combined" matrix which is then inverted to produce a solution set for the gravity and station parameters;

(iv) Finally, the arc dependent parameters are computed for each arc using the saved "backsubstitution" matrices and the arc independent solution set.

The SOLVE program essentially completes a single iteration of a multiple arc differential correction process. This program outputs the updated estimates of the gravity station position and arc parameters in both printed and punched card form. The card output is in the GEOSTAR-I ODP format in order that successive iterations may be readily performed. Optional output includes the variance-covariance and correlation matrices for the parameters as well as any of the interim matrices used in the solution, including tapes containing the combined and backsubstitution matrices. These tapes can then be used directly in SOLVE with an additional B matrix tape to obtain a new arc dependent and independent parameter set solution reflecting the arcs present on both the combined and B matrix tapes.

An example of the above operations is displayed in Figure 1, in which arcs 1 and 2 are processed through the MERGE-SOLVE loop, followed by a SOLVE operation using combined and backsubstitution matrices together with an additional arc 3.

Because of the large numbers of geodetic parameters which can be considered in a given multiple arc solution, it may frequently happen that the resulting combined matrix is ill-conditioned due to high correlations or poor observability of the particular parameter sets selected.

Various investigators have suggested different approaches to either avoid or cope with this problem. One of these is to determine, on the basis of an analytic perturbation analysis on the mean orbital elements associated with a particular arc, the magnitude of the perturbation resulting from a particular harmonic. A GSFC program called HAP (Reference 4) is currently available for this purpose. This method, however, does not currently determine the cumulative effect of linear combinations of harmonics, which may lead to problems resulting from unmodeled parameters. Other methods involving either the examination of the correlation matrices or the use of constraining a priori information can also be useful tools for this problem.

In addition to these methods, algorithms have been developed (References 5 and 6) which, in certain cases, enable one to either precisely determine which parameters can be estimated with physical significance by direct examination of the normal system, or avoid numerical difficulties by obtaining least square solutions which do not involve the inversion of a poorly conditioned matrix. Several well-known methods of this type have been made available in the GEOSTAR-I system.

The first method involves the diagonalization of the normal matrices and uses the eigenvalues and eigenvectors to determine the parameters or linear combinations of parameters which can be estimated with physical significance. The canonical decomposition of the normal matrix into eigenvectors and eigenvalues is performed by the program EIGENVALUE which accepts the combined matrix from SOLVE as input. The EIGENVALUE program will eliminate any station or arc

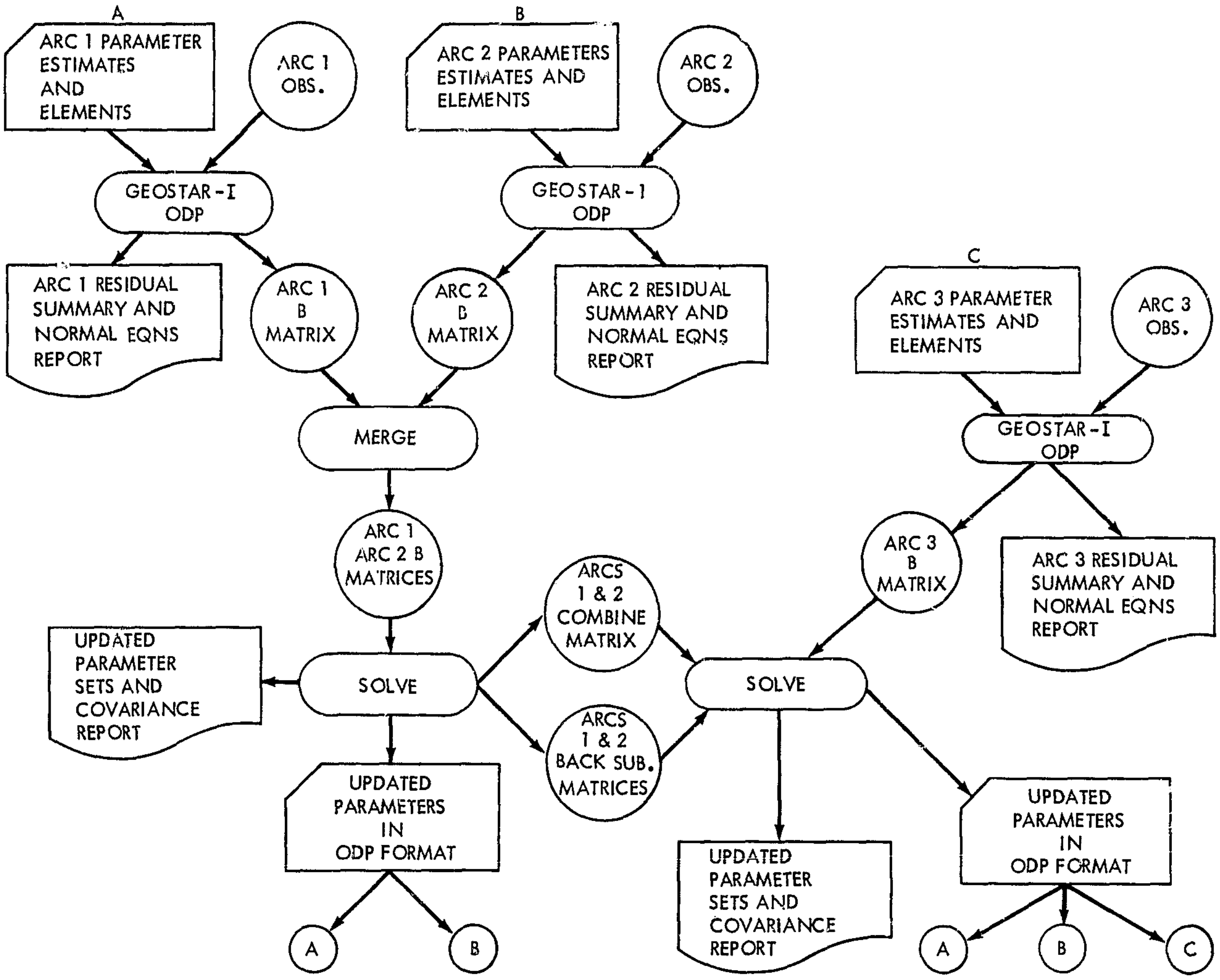

A
ARC 1 PARAMETER ESTIMATES AND ELEMENTS
ARC 1 OBS.
B
ARC 2 PARAMETERS ESTIMATES AND ELEMENTS
ARC 2 OBS.
GEOSTAR-I ODP
GEOSTAR-1 ODP
ARC 1 RESIDUAL SUMMARY AND NORMAL EQNS REPORT
ARC 1 B MATRIX
ARC 2 B MATRIX
ARC 2 RESIDUAL SUMMARY AND NORMAL EQNS REPORT
C
ARC 3 PARAMETER ESTIMATES AND ELEMENTS
ARC 3 OBS.
MERGE
GEOSTAR-I ODP
ARC 1 ARC 2 B MATRICES
ARC 3 B MATRIX
ARC 3 RESIDUAL SUMMARY AND NORMAL EQNS REPORT
ARCS 1 & 2 COMBINE MATRIX
UPDATED PARAMETER SETS AND COVARIANCE REPORT
SOLVE
SOLVE
ARCS 1 & 2 BACK SUB. MATRICES
UPDATED PARAMETERS IN ODP FORMAT
UPDATED PARAMETER SETS AND COVARIANCE REPORT
UPDATED PARAMETERS IN ODP FORMAT
A
B
A
B
C

Figure 1.

parameters before forming the decomposition, allowing the examination of the geopotential coefficient parameters exclusively, while retaining the effects of the eliminated parameters. Details concerning this method can be found in Reference 5 and is briefly outlined in Section 3.3.3 of this report.

The second method involves the computation of a unique minimal norm least squares solution of the normal equations by using a generalized inverse in lieu of a Gauss-Jordan inverse of an ill-conditioned matrix. The generalized inverse used is the Penrose pseudoinverse and is computed using the Andree Algorithm. Details concerning this method can be found in Reference 6. This option has been incorporated into the SOLVE program so that one can elect to use either the Gauss-Jordan elimination subroutine for a direct inverse, or the Andree subroutine for the pseudoinverse of the combined matrix.

Finally, in the single arc ODP, an option is available to use a gradient or "steepest descent" method whenever an ill-conditioned matrix is encountered during the iterative process. This method is given in detail in Reference 7 and outlined in Section 3.1 of this report.

Before proceeding with the description of the mathematical methods used in the GEOSTAR-I system, it is noted that future GEOSTAR systems which are currently under development or planning will include the following capabilities:

- A maximum of 250 parameters for a single arc
- The use of a priori information in the multiple arc differential correction process
- The incorporation of analytic partial derivatives of the elements with respect to the geopotential coefficients
- Techniques to handle pass dependent instrumentation biases efficiently in a multiple arc environment
- The use of advanced numerical integration techniques, such as generalized multistep methods and multistep starters, to improve the efficiency of the numerical integration process
- The use of semi-analytic techniques to solve the linear variational equations in which the state transition matrix is utilized in the solution for the partials.
- The use of advanced data management methods to handle the large amounts of tracking data required for a full geopotential solution.

III. METHOD

3.1 Geopotential Coefficient Estimation

Essentially, the geopotential coefficients are estimated by augmenting the existing NONAME ODP normal equations by including these parameters. This augmented system of normal equations is solved within either the framework of the current single arc differential correction process in NONAME, or the multiple arc processing within the LUNGFISH system.

The normal equations in the GEOSTAR-I ODP are of the form:

$$\left[A^T WA + P_0^{-1}\right] \Delta x = A^T W \Delta O - P_0^{-1} (\Sigma \Delta x) \tag{1}$$

where if m is the number of observations, and n the number of parameters, then

A = an $m \times n$ matrix of measurement partials with respect to the parameters to be estimated, i.e., $A = \{a_{i,j}\}$, where, if M_i is the i^{th} observation and x_j is the j^{th} parameter to be estimated, then

$$a_{ij} = \frac{\partial M_i}{\partial x_j} \qquad \begin{array}{l} i = 1, 2, \cdots m \\ j = 1, 2, \cdots n \end{array}$$

W = matrix of measurement weights

P_0 = a priori covariance matrix on the a priori estimates of the parameters

Δx = $n \times 1$ parameter correction vector to be determined

ΔO = $m \times 1$ residual vector

$\Sigma \Delta x$ = the accumulated correction vectors over previous iterations.

These normal equations are precisely those which exist in the NONAME ODP with the exception of the quantities needed to estimate the geopotential coefficient parameters $C_{n,m}$ and $S_{n,m}$. Assuming that the a priori estimates and covariances of these parameters are available, the additional quantities required are of the form

$$\frac{\partial M(t)}{\partial x_j} ,$$

where $M(t)$ is some measurement at observation time t, and x_j is one of the harmonics.

By the chain rule,

$$\frac{\partial M(t)}{\partial x_j} = \frac{\partial M(t)}{\partial \overline{x}} \frac{\partial \overline{x}(t)}{\partial x_j} + \frac{\partial M(t)}{\partial \dot{\overline{x}}} \frac{\partial \dot{\overline{x}}(t)}{\partial x_j} , \tag{2}$$

where $\bar{x} = (x, y, z)$ and $\dot{\bar{x}} = (\dot{x}, \dot{y}, \dot{z})$ are the satellite position and velocity vectors at observation time t. Note that here and throughout this report, the following notation is used for derivatives with respect to vectors:

If $\bar{f}$ and $\bar{g}$ are vectors, $\bar{f} = (f_1, f_2, f_3)$, $\bar{g} = (g_1, g_2, g_3)$ and h is a scalar function, then

$$\frac{\partial h}{\partial \bar{g}} = \left(\frac{\partial h}{\partial g_1}, \frac{\partial h}{\partial g_2}, \frac{\partial h}{\partial g_3}\right),$$

$$\frac{\partial \bar{g}}{\partial h} = \text{col}\left(\frac{\partial g_1}{\partial h}, \frac{\partial g_2}{\partial h}, \frac{\partial g_3}{\partial h}\right),$$

and

$$\frac{\partial \bar{f}}{\partial \bar{g}} = \left\{\frac{\partial f_i}{\partial g_j}\right\} = \begin{bmatrix} \frac{\partial f_1}{\partial g_1} & \frac{\partial f_1}{\partial g_2} & \frac{\partial f_1}{\partial g_3} \\ \frac{\partial f_2}{\partial g_1} & \frac{\partial f_2}{\partial g_2} & \frac{\partial f_2}{\partial g_3} \\ \frac{\partial f_3}{\partial g_1} & \frac{\partial f_3}{\partial g_2} & \frac{\partial f_3}{\partial g_3} \end{bmatrix}$$

Now the instantaneous observation partials with respect to satellite position and velocity

$$\frac{\partial M(t)}{\partial \bar{x}} \qquad \frac{\partial M(t)}{\partial \dot{\bar{x}}}$$

are computed precisely as in the current NONAME ODP for all the observation types considered in the system. Hence by Equation 2, all that remains to be computed to form the full observation partials are the position and velocity partials

$$\frac{\partial \bar{x}(t)}{\partial x_j}, \quad \frac{\partial \dot{\bar{x}}(t)}{\partial x_j}.$$

The equations used to compute these position partials can be easily derived by examination of the equations for the two-body and perturbative accelerations acting on the satellite. These accelerations can be e- pressed as

$$\ddot{\bar{x}} = \frac{\partial U(\bar{x})}{\partial \bar{x}} + \bar{F}_{drag}(\bar{x}, \dot{\bar{x}}) + \bar{F}_{sr}(\bar{x}) + \bar{F}_{sun}(\bar{x}) + \bar{F}_{moon}(\bar{x}), \tag{3}$$

where $U(\bar{x})$ is the geopotential given by

$$U(\bar{x}) = \frac{GM}{r} \sum_{n=0}^{30} \sum_{m=0}^{n} \left(\frac{a_e}{r}\right)^n \left[C_{nm} \cos m\lambda + S_{nm} \sin m\lambda\right] P_n^m(\sin\psi) , \tag{4}$$

where

r - radius from center of earth to satellite

a_e - earth's semi-major axis

GM - gravitational constant times mass of earth

λ - geocentric longitude (positive east)

ψ - geocentric latitude

$P_n^m(\sin\psi)$ - associated Legendre polynomials

and where $\bar{F}_{drag}$, $\bar{F}_{sr}$, $\bar{F}_{sun}$, $\bar{F}_{moon}$ are the acceleration vectors due to atmospheric drag, solar radiation pressure, solar and lunar gravity respectively. The equations for these forces are precisely those used in NONAME and can be found in Reference 1.

Now if x_j represents any physical parameter occurring in the right hand side of Equation 3, or a state vector element, then the variation of the acceleration with respect to this parameter can be expressed as

$$\frac{\partial \ddot{\bar{x}}}{\partial x_j} = \frac{\partial \ddot{\bar{x}}}{\partial \bar{x}} \frac{\partial \bar{x}}{\partial x_j} + \frac{\partial \ddot{\bar{x}}}{\partial \dot{\bar{x}}} \frac{\partial \dot{\bar{x}}}{\partial x_j} + \frac{\partial \ddot{\bar{x}}}{\partial x_j}$$

obtained by differentiation, noting that by Equation 3, $\ddot{\bar{x}}$ is a function of position and velocity. By interchanging the order of differentiation, this equation can be rewritten as

$$\frac{d^2}{dt^2}\left[\frac{\partial \bar{x}(t)}{\partial x_j}\right] = \frac{\partial \ddot{\bar{x}}}{\partial \bar{x}} \frac{\partial \bar{x}}{\partial x_j} + \frac{\partial \ddot{\bar{x}}}{\partial \dot{\bar{x}}}\left[\frac{d}{dt} \frac{\partial \bar{x}}{\partial x_j}\right] + \frac{\partial \ddot{\bar{x}}}{\partial x_j} \tag{5}$$

which is a 2nd order linear differential equation in the position partial $\partial\bar{x}(t)/\partial x_j$ i.e., if vector functions $\bar{y}^{(j)}$, $\dot{\bar{y}}^{(j)}$, $\bar{f}^{(j)}$ are defined by

$$\bar{y}^{(j)}(t) = \frac{\partial \bar{x}(t)}{\partial x_j}$$

$$\dot{\bar{y}}^{(j)}(t) = \frac{\partial \dot{\bar{x}}(t)}{\partial x_j}$$

$$\bar{f}^{(j)}(t) = \frac{\partial \ddot{\bar{x}}}{\partial x_j}$$

and the matrix functions $A(t)$, $B(t)$ by

$$A(t) = \frac{\partial \ddot{\bar{x}}(t)}{\partial \bar{x}},$$

$$B(t) = \frac{\partial \ddot{\bar{x}}(t)}{\partial \dot{\bar{x}}},$$

then by Equation 5, $\bar{y}^{(j)}(t)$ satisfies the equation

$$\ddot{\bar{y}}^{(j)}(t) = A(t)\bar{y}^{(j)}(t) + B(t)\dot{\bar{y}}^{(j)}(t) + \bar{f}^{(j)}(t), \tag{6}$$

which is called the linear variational equation with respect to the parameter x_j. Note that the Jacobian matrices $A(t)$ and $B(t)$ are independent of the particular parameter under consideration, so that they are the same for all the parameters. These matrices, as currently computed in the NONAME program, are obtained from the differentiation of Equation 3 with respect to position and velocity, where only the more significant contributions are retained to improve efficiency. For example, the term

$$\frac{\partial}{\partial \bar{x}}\left[\frac{\partial U}{\partial \bar{x}}\right]$$

only includes tesserals to third order and zonals to fourth order; see Reference 1 for details concerning these approximations.

The terms in Equation 6 which are not currently defined in the NONAME ODP are the forcing terms $\bar{f}^{(j)}(t) = \partial\ddot{\bar{x}}/\partial x_j$, where x_j is a geopotential coefficient $C_{n,m}$ or $S_{n,m}$. These are computed as follows:

Letting $\bar{\phi}$ denote the vector of satellite spherical coordinates, $\bar{\phi} = (r, \lambda, \psi)$, from Equation 3 it follows that

$$\frac{\partial \ddot{\bar{x}}}{\partial C_{nm}} = \frac{\partial}{\partial C_{nm}}\left[\frac{\partial U}{\partial \bar{x}}\right] = \frac{\partial}{\partial C_{nm}}\left[\frac{\partial U}{\partial \bar{\phi}}\frac{\partial \bar{\phi}}{\partial \bar{x}}\right], \tag{7}$$

or

$$\frac{\partial \ddot{\bar{x}}}{\partial C_{nm}} = \frac{\partial}{\partial C_{nm}} \left(\frac{\partial U}{\partial \bar{\Phi}}\right) \left(\frac{\partial \bar{\Phi}}{\partial \bar{x}}\right) ;$$

likewise,

$$\frac{\partial \ddot{\bar{x}}}{\partial S_{nm}} = \frac{\partial}{\partial S_{nm}} \left(\frac{\partial U}{\partial \bar{\Phi}}\right) \left(\frac{\partial \bar{\Phi}}{\partial \bar{x}}\right) , \tag{8}$$

where $\partial U/\partial \bar{\Phi}$ is the vector

$$\frac{\partial U}{\partial \bar{\Phi}} = \left(\frac{\partial U}{\partial r} , \frac{\partial U}{\partial \lambda} , \frac{\partial U}{\partial \psi}\right)$$

and $(\partial \bar{\Phi}/\partial \bar{x})$ is the transformation matrix given by

$$\frac{\partial \bar{\Phi}}{\partial \bar{x}} = \begin{bmatrix} \frac{\partial r}{\partial x} & \frac{\partial r}{\partial y} & \frac{\partial r}{\partial z} \\ \frac{\partial \lambda}{\partial x} & \frac{\partial \lambda}{\partial y} & \frac{\partial \lambda}{\partial z} \\ \frac{\partial \psi}{\partial x} & \frac{\partial \psi}{\partial y} & \frac{\partial \psi}{\partial z} \end{bmatrix} = \begin{bmatrix} x/r & y/r & z/r \\ -y/s^2 & x/s^2 & 0 \\ -xz/r^2 s & -yz/r^2 s & s/r^2 \end{bmatrix} \tag{9}$$

where

$$s = \left(x^2 + y^2\right)^{1/2} ,$$

and where, by differentiating Equation 4, the partials

$$\frac{\partial}{\partial C_{nm}} \left(\frac{\partial U}{\partial \bar{\Phi}}\right) = \left(\frac{\partial}{\partial C_{nm}} \frac{\partial U}{\partial r} , \frac{\partial}{\partial C_{nm}} \frac{\partial U}{\partial \lambda} , \frac{\partial}{\partial C_{nm}} \frac{\partial U}{\partial \psi}\right)$$

$$\frac{\partial}{\partial S_{nm}} \left(\frac{\partial U}{\partial \bar{\Phi}}\right) = \left(\frac{\partial}{\partial S_{nm}} \frac{\partial U}{\partial r} , \frac{\partial}{\partial S_{nm}} \frac{\partial U}{\partial \lambda} , \frac{\partial}{\partial S_{nm}} \frac{\partial U}{\partial \psi}\right)$$

are given by:

$$
\begin{aligned}
\frac{\partial}{\partial C_{nm}} \frac{\partial U}{\partial r} &= -\left(\frac{GM}{r^2}\right)\left(\frac{a_e}{r}\right)^n (n+1) \cos m\lambda \, P_n^m (\sin \psi) \\
\frac{\partial}{\partial C_{nm}} \frac{\partial U}{\partial \lambda} &= -\frac{GM}{r}\left(\frac{a_e}{r}\right)^n (m \sin m\lambda) P_n^m (\sin \psi) \\
\frac{\partial}{\partial C_{nm}} \frac{\partial U}{\partial \psi} &= \frac{GM}{r}\left(\frac{a_e}{r}\right)^n \cos m\lambda \left[P_n^{m+1} (\sin \psi) - m \tan \psi P_n^m (\sin \psi)\right] \\
\frac{\partial}{\partial S_{nm}} \frac{\partial U}{\partial r} &= -\frac{GM}{r^2}\left(\frac{a_e}{r}\right)^n (n+1) \sin m\lambda \, P_n^m (\sin \psi) \\
\frac{\partial}{\partial S_{nm}} \frac{\partial U}{\partial \lambda} &= \frac{GM}{r}\left(\frac{a_e}{r}\right)^n (m \cos m\lambda) P_n^m (\sin \psi) \\
\frac{\partial}{\partial S_{nm}} \frac{\partial U}{\partial \psi} &= \frac{GM}{r}\left(\frac{a_e}{r}\right)^n \sin m\lambda \left[P_n^{m+1} (\sin \psi) - m \tan \psi P_n^m (\sin \psi)\right] ,
\end{aligned}
\tag{10}
$$

and the forcing functions are then computed using Equations 7, 8 and 9.

Using these forcing terms in Equation 6, the required partials with respect to geopotential coefficients as well as other parameters are obtained by numerically integrating this equation for each j, using the initial conditions

$$\bar{y}^{(j)} (t_0) = \dot{\bar{y}}^{(j)} (t_0) = 0$$

for all j corresponding to geopotential coefficients.

The measurement partials are then formed using Equation 2 and used in the normal system Equation 1. Upon completing the formation of these equations, the ODP will either output them on a tape in the SOLVE format for a multiple arc solution, or perform an iterative single arc solution for the parameter correction vector Δx. In the single arc process, the normal matrix

$$S = \left[A^T W A + P_0^{-1} \right]$$

is first inverted using the Gauss-Jordan method with partial pivoting. The resulting matrix is tested to determine the condition of the matrix S. If it is found to be ill-conditioned, the solution is determined by the gradient technique. This is essentially a step in the method of steepest descent for obtaining least square solutions, which finds the scale-factor multiple of the right hand side of Equation 1, denoted by b, which minimizes the quadratic form

$$R(\Delta x) = (S\,\Delta x - b)^T (S\,\Delta x - b)$$

i.e., a scalar λ is found such that if $\Delta x = \lambda b$, then $R(\lambda b)$ is minimized, or

$$\frac{\partial R}{\partial \lambda} = 0 \cdot$$

This solution is found to be

$$\lambda = \frac{b^T y}{y^T y} ,$$

where

$$y = Sb ,$$

and the correction vector is then given by $\Delta x = \lambda b$.

In either case, the parameter set is then updated and the process is repeated until a convergence criterion is met, or a maximum iteration count is exceeded.

The multiple arc process is described in Section 3.3.

3.2 Numerical Integration and Interpolation

The numerical integration of the equations of motion (3) for position and the variational equations (6) for the position partials, is performed by using a summed Cowell type integration method. These formulas are used in Lagrangian or ordinate form, and the coefficients are available in the system for orders 4 through 15. The starting procedure used is an 8th order Runge-Kutta, and the Hermite interpolation formula, utilizing functional derivatives, is used to interpolate position and velocity vectors as required by the process.

3.2.1 The Integration of the Equations of Motion

The basic integration formulas for the integration of Equation 3 are given by

$$\bar{x}_{n+1} = h_e^2 \left[{}^{II}\bar{S}_n + \sum_{i=0}^{k_e} \alpha_i \ddot{\bar{x}}_{n-i} \right] \quad \text{(Störmer Predictor)} \qquad (11a)$$

$$\overline{x}_{n+1} = h_e^2 \left[{}^{II}\overline{S}_n + \sum_{i=0}^{k_e} \alpha_i^* \ddot{\overline{x}}_{n+1-i} \right] \text{ (Cowell Corrector)} \qquad (11b)$$

$$\dot{\overline{x}}_{n+1} = h_e \left[{}^{I}\overline{S}_n + \sum_{i=0}^{k_e} \beta_i \ddot{\overline{x}}_{n-i} \right] \text{ (Adams Predictor)} \qquad (11c)$$

$$\dot{\overline{x}}_{n+1} = h_e \left[{}^{I}\overline{S}_n + \sum_{i=0}^{k_e} \beta_i^* \ddot{\overline{x}}_{n+1-i} \right] \text{ (Moulton Corrector)}, \qquad (11d)$$

where

h_e = the stepsize used for the integration of the equations of motion,

k_e = $p - 2$, where p is the order of the formulas (11),

${}^{I}\overline{S}_n$, ${}^{II}\overline{S}_n$ = the first and second sums of the accelerations, which can be defined as

$${}^{I}\overline{S}_n = \nabla^{-1} \ddot{\overline{x}}_n$$

$${}^{II}\overline{S}_n = \nabla^{-1}\, {}^{I}\overline{S}_n = \nabla^{-2} \ddot{\overline{x}}_n ,$$

α_i, α_i^*, β_i, β_i^* = the ordinate integration coefficients which depend on the order p.

The ordinate form coefficients can be formulated by defining the difference form coefficients recursively and converting to ordinate form. In the GEOSTAR-I system, these coefficients have been precomputed in rational form (Reference 8) and entered into the system as a permanent data file for orders 4 through 15.

The algorithm used to integrate the equations of motion is the following:

(i) Compute a set of "starting" values for the accelerations

$$\ddot{\overline{x}}_{k_e - i} , \qquad i = 1, 2, \cdots k_e ,$$

and sums ${}^{I}\overline{S}_{k_e}$, ${}^{II}\overline{S}_{k_e}$ using an independent procedure;

(ii) Using Equations 11a and 11c with $n = k_e$, obtain predicted values

$$\overline{x}_{n+1}^{(P)} , \qquad \dot{\overline{x}}_{n+1}^{(P)} ;$$

(iii) Evaluate the force model using the last computed position and velocity vector to obtain $\ddot{\overline{x}}_{n+1}$;

(iv) Using Equations 11b and 11d, obtain corrected values

$$\bar{x}^{(C)}_{n+1}, \qquad \dot{\bar{x}}^{(C)}_{n+1};$$

(v) Compare the magnitude of the vector

$$\left| \bar{x}^{(C)}_{n+1} - \bar{x}^{(P)}_{n+1} \right|$$

with a predictor-corrector tolerance. If this difference vector is sufficiently small, the predictor-corrector cycle is complete and step (vi) is then executed. If it is not sufficiently small, the predicted values are replaced by the corrected values and steps (iii) to (v) are repeated. The maximum number of iterations allowed is three. Note that it is possible to complete a predictor-corrector cycle with only one force model evaluation.

(vi) Compute updated sums by

$$\begin{aligned} {}^{I}\bar{S}_{n+1} &= \ddot{\bar{x}}_{n+1} + {}^{I}\bar{S}_{n} \\ {}^{II}\bar{S}_{n+1} &= {}^{I}\bar{S}_{n+1} + {}^{II}\bar{S}_{n}, \end{aligned}$$

completing the integration step. Steps (ii) through (vi) are then repeated with $n = n + 1$.

3.2.2 Integration of the Linear Variational Equations

The integration formulas used to integrate the linear variational Equations 6 are given by

$$\bar{y}^{(j)}_{n+1} = h_v^2 \left[{}^{II}\bar{P}^{(j)}_{n} + \sum_{i=0}^{k_v} \alpha_i^* \bar{y}^{(j)}_{n+1-i} \right] \qquad (12a)$$

$$\dot{\bar{y}}^{(j)}_{n+1} = h_v \left[{}^{I}\bar{P}^{(j)}_{n} + \sum_{i=0}^{k_v} \beta_i^* \ddot{\bar{y}}^{(j)}_{n+1-i} \right], \qquad j = 1, 2, \cdots \qquad (12b)$$

where $y^{(j)}_{n+1}$ indicates the j^{th} position partial at t_{n+1}, and

h_v = the stepsize used for the integration of the variational Equations 6,

k_v = $p - 2$, where p is the order of the formulas (12),

${}^{I}\bar{P}^{(j)}_{n}$, ${}^{II}\bar{P}^{(j)}_{n}$ = the first and second sums of the acceleration partials with respect to the j^{th} parameter,

α_i^*, β_i^* = the ordinate corrector integration coefficients as in formulas (11b) and (11d), depending on the order of formulas (12).

The GEOSTAR-I system was designed so that the stepsize and order used to integrate the variational equations need not be the same as those used to integrate the variational equations. This allows efficient utilization of the varying accuracy requirements between the position and position partials. For example, in certain cases, it may be possible to obtain sufficiently accurate position partials using two or three times the stepsize used for the satellite position, or half the order, thereby improving overall efficiency.

The method used to integrate the variational equations employs a closed form solution of Equations 12, or the "corrector-only" technique. The required equations are derived as follows:

From Equation 6, a typical variational equation at time t_{n+1} can be expressed as

$$\ddot{\bar{y}}_{n+1} = A_{n+1}\,\bar{y}_{n+1} + B_{n+1}\,\dot{\bar{y}}_{n+1} + \bar{f}_{n+1}, \tag{13}$$

where for the moment, the j superscript is dropped. From Equations 12 it follows that

$$\begin{aligned} \bar{y}_{n+1} &= h_v^2\left[{}^{II}\bar{P}_n + \alpha_0^*\,\ddot{\bar{y}}_{n+1} + \sum_{i=1}^{k_v} \alpha_i^*\,\ddot{\bar{y}}_{n+1-i}\right] \\ \dot{\bar{y}}_{n+1} &= h_v\left[{}^{I}\bar{P}_n + \beta_0^*\,\ddot{\bar{y}}_{n+1} + \sum_{i=1}^{k_v} \beta_i^*\,\ddot{\bar{y}}_{n+1-i}\right]. \end{aligned} \tag{14}$$

Letting

$$\begin{aligned} \bar{X}_n &= h_v^2\left[{}^{II}\bar{P}_n + \sum_{i=1}^{k_v} \alpha_i^*\,\ddot{\bar{y}}_{n+1-i} + \alpha_0^*\,\bar{f}_{n+1}\right] \\ \bar{V}_n &= h_v\left[{}^{I}\bar{P}_n + \sum_{i=1}^{k_v} \beta_i^*\,\ddot{\bar{y}}_{n+1-i} + \beta_0^*\,\bar{f}_{n+1}\right] \end{aligned} \tag{15}$$

and substituting Equation 13 into the right hand side, it is possible to rewrite Equation 14 as

$$\begin{aligned} \bar{y}_{n+1} &= h_v^2\,\alpha_0^*\,A_{n+1}\,\bar{y}_{n+1} + h_v^2\,\alpha_0^*\,B_{n+1}\,\dot{\bar{y}}_{n+1} + \bar{X}_n \\ \dot{\bar{y}}_{n+1} &= h_v\,\beta_0^*\,A_{n+1}\,\bar{y}_{n+1} + h_v\,\beta_0^*\,B_{n+1}\,\dot{\bar{y}}_{n+1} + \bar{V}_n\ , \end{aligned} \tag{16}$$

which can be expressed as the matrix equation

$$\begin{bmatrix} \bar{y}_{n+1} \\ \dot{\bar{y}}_{n+1} \end{bmatrix} = \begin{bmatrix} h_v^{\,2} \alpha_0^{\,*} A_{n+1} & h_v^{\,2} \alpha_0^{\,*} B_{n+1} \\ h_v \beta_0^{\,*} A_{n+1} & h_v \beta_0^{\,*} B_{n+1} \end{bmatrix} \begin{bmatrix} \bar{y}_{n+1} \\ \dot{\bar{y}}_{n+1} \end{bmatrix} + \begin{bmatrix} \bar{X}_n \\ \bar{V}_n \end{bmatrix} .$$

or letting H be the 6 × 6 matrix

$$H = \begin{bmatrix} h_v^{\,2} \alpha_0^{\,*} A_{n+1} & h_v^{\,2} \alpha_0^{\,*} B_{n+1} \\ h_v \beta_0^{\,*} A_{n+1} & h_v \beta_0^{\,*} B_{n+1} \end{bmatrix} , \tag{17}$$

and I the identity 6 × 6 matrix, then the 6 × 1 vector

$$\begin{bmatrix} \bar{y}_{n+1} \\ \dot{\bar{y}}_{n+1} \end{bmatrix}$$

is the solution of the equation

$$[I - H] \begin{bmatrix} \bar{y}_{n+1} \\ \dot{\bar{y}}_{n+1} \end{bmatrix} = \begin{bmatrix} \bar{X}_n \\ \bar{V}_n \end{bmatrix} , \tag{18}$$

and hence is given by

$$\begin{bmatrix} \bar{y}_{n+1} \\ \dot{\bar{y}}_{n+1} \end{bmatrix} = [I - H]^{-1} \begin{bmatrix} \bar{X}_n \\ \bar{V}_n \end{bmatrix} . \tag{19}$$

We note that the matrix H is independent of the particular parameter being considered so that the inversion in Equation 19 is performed only once per step regardless of the number of parameters being estimated. Using Equation 19, the solution of Equations 12 for all the position and velocity partials can be expressed as

$$\begin{bmatrix} \bar{y}_{n+1}^{(1)} & \bar{y}_{n+1}^{(2)} \cdots\cdot \bar{y}_{n+1}^{(N)} \\ \dot{\bar{y}}_{n+1}^{(1)} & \dot{\bar{y}}_{n+1}^{(2)} \cdots\cdot \dot{\bar{y}}_{n+1}^{(N)} \end{bmatrix}_{6\times N} = [I - H]^{-1}_{6\times 6} \begin{bmatrix} \bar{X}_n^{(1)} & \bar{X}_n^{(2)} \cdots\cdot \bar{X}_n^{(N)} \\ \bar{V}_n^{(1)} & \bar{V}_n^{(2)} \cdots\cdot \bar{V}_n^{(N)} \end{bmatrix}_{6\times N} \tag{20}$$

where N is the total number of partials being computed, and where the $\overline{X}_n^{(i)}$ and $\overline{V}_n^{(i)}$ are defined in the natural way from Equations 15 as

$$\overline{X}_n^{(i)} = h_v^2 \left[{}^{II}\overline{P}_n^{(i)} + \sum_{j=1}^{k_v} \alpha_j^* \ddot{\overline{y}}_{n+1-j}^{(i)} + \alpha_0^* \overline{f}_{n+1}^{(i)} \right]$$

$$\overline{V}_n^{(i)} = h_v \left[{}^{I}\overline{P}_n + \sum_{j=1}^{k_v} \beta_j^* \ddot{\overline{y}}_{n+1-j} + \beta_0^* \overline{f}_{n+1}^{(i)} \right] . \tag{21}$$

The algorithm used to integrate the system of N linear variational equations is the following:

(i) Compute a set of "starting" values for the acceleration partials

$$\ddot{\overline{y}}_{k_v-i}^{(j)} , \qquad i = 1, 2, \cdots k_v$$

and sums ${}^{I}\overline{P}_{k_v}^{(j)}$, ${}^{II}\overline{P}_{k_v}^{(j)}$, using an independent procedure, for each $j = 1, 2, \cdots N$.

(ii) Letting $n = k_v$, compute the values of satellite position and velocity at time $t_{n+1} = t_{k_v} + h_v$, denoted by $\overline{x}_{n+1}'$ and $\dot{\overline{x}}_{n+1}'$. Normally, if $h_v = h_e$, these values can be obtained by integrating the equations of motion as described in Section 3.2.1 to time $t = t_{n+1}$, i.e., $\overline{x}_{n+1}' = \overline{x}_{n+1}$ and $\dot{\overline{x}}_{n+1}' = \dot{\overline{x}}_{n+1}$. If $h_v \neq h_e$, these values are obtained by interpolation, where the integration of the equations of motion continues until enough position and velocity data is available.

(iii) Using the vectors $\overline{x}_{n+1}'$, $\dot{\overline{x}}_{n+1}'$ compute the Jacobian matrices A_{n+1}, B_{n+1} and the forcing terms $\overline{f}_{n+1}^{(j)}$;

(iv) Using the matrices A_{n+1}, B_{n+1}, form $[I - H]$ by Equation 17 and using the vectors

$$\overline{f}_{n+1}^{(j)} , \ddot{\overline{y}}_{n-i}^{(j)} , {}^{I}\overline{P}_n^{(j)} , {}^{II}\overline{P}_n^{(j)}$$

compute the vector quantities

$$\overline{X}_n^{(j)} , \overline{V}_n^{(j)} \qquad j = 1, 2, \cdots N$$

using Equations 21.

(v) Invert $[I - H]$ and perform the multiplication in Equation 20 to obtain the values for the N position and velocity partials, completing the integration step to time t_{n+1}. Steps (ii) through (v) are then repeated with $n = n + 1$.

3.2.3 The Integration for the Case of Velocity-Free Accelerations

For the case in which the accelerations acting on the satellite do not depend on the velocity vector, i.e., $\overline{F}_{drag}$ in Equation 3 is zero, the number of computations performed for the integration, as described in Sections 3.2.1 and 3.2.2 can be considerably reduced.

In this case, for the integration of the equations of motion, Equation 11c which predicts the velocity vector is not used, and Equation 11d, the velocity corrector, is used only once, after convergence of the position corrector formula Equation 11c.

For the integration of the variational equations, note that the Jacobian matrix B(t) in Equation 6 is zero, so that Equation 13 is simply of the form

$$y_{n+1} = A_{n+1} \bar{y}_{n+1} + \bar{f}_{n+1} .$$

By using the notation of Section 3.2.2, the corrector formula to be solved can be expressed as

$$\bar{y}_{n+1} = h_v^2 \alpha_0^* A_{n+1} \bar{y}_{n+1} + \bar{X}_n$$

with solution

$$\bar{y}_{n+1} = [I - H]^{-1} \bar{X}_n \; ,$$

where H is now the 3 × 3 matrix $h_v^2 \alpha_0^* A_{n+1}$, and I the 3 × 3 identity. The velocity components are obtained from

$$\dot{\bar{y}}_{n+1} = h_v \beta_0^* A_{n+1} y_{n+1} + \bar{V}_n \; .$$

Hence, for this case, Equation 20 is replaced by the equations

$$\begin{aligned}\left[\bar{y}_{n+1}^{(1)}, \bar{y}_{n+1}^{(2)}, \cdots \bar{y}_{n+1}^{(N)}\right]_{3\times N} &= [I - H]^{-1}\left[\bar{X}_n^{(1)}, \bar{X}_n^{(2)}, \cdots, \bar{X}_n^{(N)}\right] \\ \left[\dot{\bar{y}}_{n+1}^{(1)}, \dot{\bar{y}}_{n+1}^{(2)}, \cdots \dot{\bar{y}}_{n+1}^{(N)}\right]_{3\times N} &= \left[h_v \beta_0^* A_{n+1}\right]_{3\times 3}\left[\bar{y}_{n+1}^{(1)} \cdots \bar{y}_{n+1}^{(N)}\right]_{3\times N} + \left[\bar{V}_n^{(1)}, \bar{V}_n^{(2)} \cdots, \bar{V}_n^{(N)}\right]\end{aligned} \tag{22}$$

which can be seen to require significantly fewer computations.

3.2.4 The Starting Procedure

The method used to compute the required starting arrays of accelerations and acceleration partials is an eight order, single-step Runge-Kutta type numerical integration.

This method is defined as follows:

Given an arbitrary system of first order differential equations

$$\dot{\bar{x}} = \bar{f}(t, \bar{x}) \tag{23}$$

with initial vector $\bar{x}(t_0) = \bar{x}_0$, the solution vector at time $t_1 = t_0 + h_1$, h_1 an arbitrary stepsize, is approximated by

$$\bar{x}\left(t_1\right) = \bar{x}\left(t_0\right) + h_1 \sum_{i=0}^{9} c_i \bar{k}_i \quad , \tag{24}$$

where, for each i,

$$\bar{k}_i = \bar{f}\left(t_0 + a_i h_1, \bar{x}_0 + \sum_{j=0}^{i-1} b_{ij} \bar{k}_j\right) \qquad i = 1, 2, \cdots 9 \quad , \tag{25}$$

and where a_i, b_{ij} and c_i are constants defining the method. These can be found in Reference 9 and are also given in the subroutine description RK in Section 4.2.

Note that to compute $\bar{x}(t_2) = \bar{x}(t_1 + h_2)$ where $h_1 \neq h_2$, it is possible to repeat the above process starting with $\bar{x}(t_1)$ and using h_2 in the definition of the $\bar{k}_i$ (Equation 25, so that the method advances from step to step independently of the stepsizes taken, as is usually the case for single-step methods.

Now, by the usual procedure, the three-dimensional system of 2nd order equations given by Equation 3 can be expressed as a six-dimensional system of 1st order equations; likewise, the variational Equations 6 can be reduced to a system of N 1st order equations of dimension 6, so that both systems (Equations 6 and 3) can be reduced to a system of the form given by Equation 23.

The algorithm used to form the starting values for the multistep integrations is the following:

Case I: $h_v = h_e$

In this case, the number of time points produced by the single step method is $\ell = \max(k_v, k_e)$, i.e., using the given initial state for the equations of motion and the initial values

$$\bar{y}^{(j)}\left(t_0\right) = \dot{\bar{y}}^{(j)}\left(t_0\right) = 0 ,$$

for the variational equations, the single step integrator is used to compute the values:

$$\left.\begin{matrix} \bar{x}_i, \dot{\bar{x}}_i, \ddot{\bar{x}}_i \\ \\ \bar{y}_i^{(j)}, \dot{\bar{y}}_i^{(j)}, \ddot{\bar{y}}_i^{(j)} \end{matrix}\right\} \text{at } t_i = t_0 + ih_e, \qquad \begin{matrix} i = 1, 2, \cdots \ell \\ j = 1, 2, \cdots N \end{matrix} \quad .$$

Case II: $h_v \neq h_e$

In this case, the number of time points produced by the single step method is $\ell = k_v + k_e$, i.e., the single step integrator is used to compute the values:

$$\bar{x}_i, \dot{\bar{x}}_i, \ddot{\bar{x}}_i \text{ at } \quad \text{at } t_i = t_0 + ih_e, \qquad i = 1, 2, \cdots k_e$$

$$\bar{y}_i^{(j)}, \dot{\bar{y}}_i^{(j)}, \ddot{\bar{y}}_i^{(j)} \text{ at } t_i = t_0 + ih'_v, \qquad \begin{array}{l} i = 1, 2, \cdots k_v \\ j = 1, 2, \cdots N . \end{array}$$

In either case, the stepsize used by the single step method is independent of the stepsizes h_v or h_e, and is generally chosen to be a fraction of the stepsize, h_e, used to integrate the equations of motion.

Once the above starting values have been obtained, the starting first and second sums are computed as follows:

First, the sums

$${}^{I}\bar{S}_{k_e-1},\ {}^{II}\bar{S}_{k_e-1},\ {}^{I}\bar{P}^{(j)}_{k_v-1},\ {}^{II}\bar{P}^{(j)}_{k_v-1}$$

are computed by inverting the respective corrector formulas (11d), (11b), (12b), and (12a), with $n = k_e - 1$ or $n = k_v - 1$.

For example, from Equation 11d,

$${}^{I}\bar{S}_{k_e-1} = \frac{\dot{\bar{x}}_{k_e}}{h_e} - \sum_{i=0}^{k_e} \beta_i^* \ddot{\bar{x}}_{k_e-i}$$

where we see that all the quantities on the right side are given by the starting values. Once these have been computed, the required starting sums are given by

$$\begin{aligned} {}^{I}\bar{S}_{k_e} &= {}^{I}\bar{S}_{k_e-1} + \ddot{\bar{x}}_{k_e} \\ {}^{II}\bar{S}_{k_e} &= {}^{I}\bar{S}_{k_e} + {}^{II}\bar{S}_{k_e-1} \end{aligned} \qquad (26)$$

and likewise for ${}^{I}\bar{P}^{(j)}_{k_v}$, ${}^{II}\bar{P}^{(j)}_{k_v}$. These equations complete the starting procedure.

3.2.5 Variable Stepsize Integration

A version of the GEOSTAR-I ODP program is available which uses an integration method allowing dynamic stepsize modification during the integration. This version is designed specifically for the integration of the equations of motion and the concomitant variational equations associated with high eccentricity satellites, where normally fixed stepsize integration methods are extremely inefficient.

In the variable step integration program, the equations of motion and variational equations are integrated with the same integration methods outlined in Sections 3.1.1 and 3.1.2, with the exception that $h_e = h_v = h$, i.e., a common stepsize is used.

The stepsize variations are based on the concept of local error control, where the stepsize is selected so that the local error, denoted by Rn, at each step of the integration satisfies the constraint equation

$$T_2 \leq Rn \leq T_1 \tag{27}$$

where T_1, T_2 are specified upper and lower bounds.

In the GEOSTAR-I ODP, the local error Rn is estimated by the quantity

$$Rn = C|\bar{x}_n^{(P)} - \bar{x}_n^{C}|$$

where C is an error constant depending on the order of the formulas (Equations 11) being used, and $\bar{x}_n^{(P)}$, $\bar{x}_n^{(C)}$ are the predicted and finally accepted satellite position vectors respectively, computed at time t_n.

The variable step integration algorithm is then the following:

At each step n, the test (Equation 27) is performed. We have three cases:

(i) $Rn > T_1$; the stepsize is decreased, the n^{th} computed point is rejected and recomputed with the new stepsize, where the required back values in Equations 11 or 12 are obtained by interpolation.

(ii) $Rn < T_2$; the stepsize is increased, the n^{th} computed point is accepted and the integration proceeds with the new stepsize, where the required back values are obtained by interpolation.

(iii) Rn satisfies Equation 27; the integrator proceeds to the next step using the same stepsize.

In either case (i) or (ii), the new stepsize is computed using the formula

$$h_{new} = h\left[\frac{T_3}{Rn}\right]^{1/k_e},$$

where T_3 is a specified value for the "allowable" local error: $T_2 \leq T_3 \leq T_1$.

The starting procedure used for this integration method is the same as that described in Section 3.2.4 with the exception that the initial stepsize is modified if necessary so that Equation 27 is satisfied at the first Cowell step.

Because of core storage limitations, the following restrictions were imposed on this version of the ODP:

(i) The maximum number of position partials allowed is 20;

(ii) As indicated above, the stepsizes used by the equations of motion and the variational equations are equal, although different orders may be used.

3.2.6 Interpolation

The position and position partials are computed at non-step points as required by the observations, or in the integration process, using a 5 point Hermite interpolation scheme.

Each component of the interpolated vectors is obtained by the formula

$$y(t) = \sum_{i=0}^{k} h_j (t) y_i + \dot{h}_j (t) \dot{y}_j ,$$

where

$$h_j (t) = 1 - 2(t - t_j) \dot{\ell}_j (t_j) \ell_j^2 (t)$$

$$\dot{h}_j (t) = (t - t_j) \ell_j^2 (t)$$

$$\ell_j (t) = \prod_{\substack{i=0 \\ i \neq j}}^{k} \frac{(t - t_i)}{(t_j - t_i)}$$

$$\dot{\ell}_j (t) = \ell_j (t) \sum_{\substack{i=0 \\ i \neq j}}^{k} \frac{1}{(t - t_i)}$$

and where $k = 4$.

3.3 Multiple Arc Processing

GEOSTAR-I multiple arc solutions are obtained by using the MERGE and SOLVE programs with essentially the same algebraic and data manipulation techniques as used in the current LUNGFISH system. The ODP interfaces with these programs, their basic methods and capabilities, as well as the methods of parameter transformation and pseudoinversion, are reviewed in the following sections.

3.3.1 ODP Interfaces; Matrix and Parameter Set Handling Procedures

The GEOSTAR-I ODP—MERGE—SOLVE interfaces can be summarized as follows:

(i) B Matrix—The principal data link between the ODP program and the multiple arc processors MERGE and SOLVE. A B matrix is defined as a matrix containing the known elements of the normal equations (Equation 1, Section 3.1), i.e., the B matrix contains the elements of the matrix

$$S = \left[A^T WA + P_0^{-1}\right] ,$$

and the right hand side vector

$$b = A^T W\Delta O - P_0^{-1} (\Sigma \Delta x) .$$

The elements of the B matrix are arranged so that the geopotential and station position parameters precede the arc parameters. Figure 2 displays the parameter sequences as they occur in the ODP normal matrix and as they are required in the B matrix. In addition to the normal equations, the B matrix tape contains arc and parameter identification labels, as well as the parameter values. This labeling scheme is also indicated in Figure 2. The precise format specifications for this tape are given in Appendix B, with further description given in the subroutine WTBMAT in Section 4.2.

(ii) SOLVE Punched Card Output—After a multiple arc solution is obtained, the SOLVE program will output a deck of punched cards containing the updated arc dependent and independent parameters. These cards are in the input format required by the ODP program for the parameter estimates, facilitating an iterative process. It is noted that although SOLVE performs the station location calculations in rectangular coordinates (Figure 2), the updated station location parameters are converted to spherical coordinates for output.

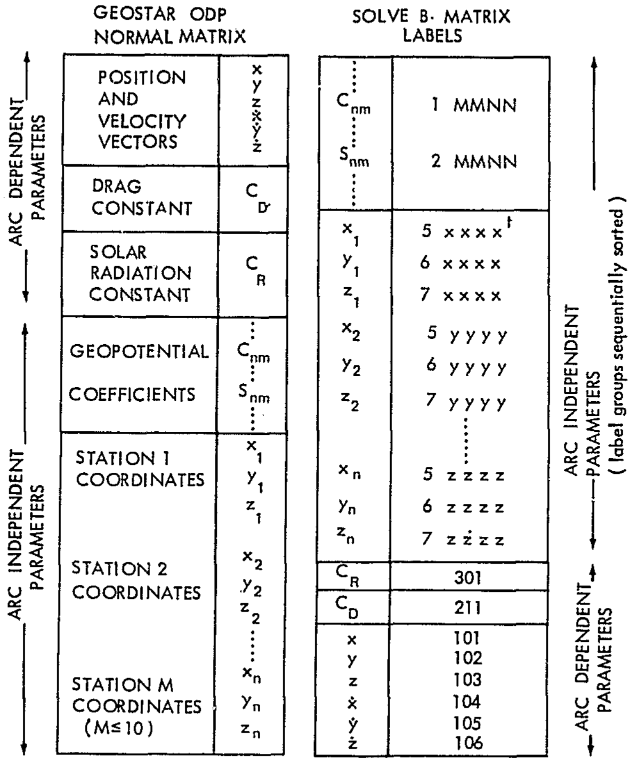
The essential matrix and parameter set handling procedures available in the GEOSTAR-I system are:

(i) Merging—The MERGE program is used to copy B matrices and parameter set matrices from up to four input tapes onto one tape. The output tape used may already contain

B matrices, in which case the input matrices are copied after them. Also, the input tapes may contain several B matrices each. The "merged" tape can then be input into the SOLVE program, where the arcs used in the solution can be any subset of the total set of arcs contained on the tape.

(ii) Combining—The SOLVE program can optionally perform the following matrix handling procedures:

(a) A combined matrix, representing the normal equations and parameter sets for the arc independent parameters over all the arcs used in the solution can be output in the B matrix format. This tape can then be used, together with an additional B matrix tape containing an arbitrary number of additional B matrices in a subsequent SOLVE run.



† "xxxx," "yyyy," "zzzz" represent 4 digit station identification numbers used in the ODP.

Figure 2—GEOSTAR ODP and B Matrix Parameter Sequences.

(b) A combined matrix, as in (a), parameter set matrices for the arc dependent parameters, and the "backsubstitution" matrices, over all the arcs used in the solution can be output on three tapes. As in (a), these tapes can then be used, together with another B matrix tape, in a subsequent SOLVE run, allowing the arc dependent parameters of a previous SOLVE run to reflect the solution of the arc independent parameters over all the arcs used.

(c) Matrices containing data over the same period for the same satellite can be combined in such a way as to include the arc dependent parameters. This matrix can then be output on tape, as well as used to form a single arc solution. This is called the "COMBINE ARCS" feature.

(iii) Multiple Reel Processing—The IBM 360 system multiple reel volume capability can be used with the MERGE, SOLVE programs to effectively handle large numbers of large matrices. The multireel capability essentially allows that tapes generated as single reels may be treated as a multiple reel volume for subsequent program inputs. To process a large number of matrices, the procedure recommended is to first use the MERGE program to collect as many matrices as possible onto a two or three reel volume, then use several of these multireel volumes as one large multireel volume in SOLVE.

3.3.2 Algebraic Matrix Operations

The matrix operations used to determine the solution of the parameters in a multiple arc environment are:

(i) Elimination of Arc Parameters

(ii) Combining of Normal Equations

(iii) Backsubstitution

(iv) Suppression of Parameters

To illustrate these operations, let

$$S_n \Delta x_n = b_n \qquad n = 1, 2, \cdots k \tag{1}$$

be k sets of normal equations, where the parameter sets x_n are comprised of arc independent (I) and arc dependent (D) parameters, i.e.,

$$x_n = \left(x_n^I, x_n^D\right) \qquad n = 1, 2, \cdots k \;,$$

where the $\{x_n^I\}$ may or may not have common elements. For each n, we can express these systems in partitioned form:

$$S_n \Delta x_n = \begin{bmatrix} B_n & C_n^T \\ C_n & D_n \end{bmatrix} \begin{bmatrix} \Delta x_n^I \\ \Delta x_n^D \end{bmatrix} = \begin{bmatrix} b_n^I \\ b_n^D \end{bmatrix} . \tag{2}$$

To solve Equation 1 for the parameter corrections Δx_n simultaneously, without forming a considerably larger matrix than any of the S_n, the following procedure is used:

First, the <u>Elimination of the Arc Parameters</u> operation is used to form a reduced system of equations. From Equation 2, we have the component equations

$$B_n \Delta x_n^I + C_n^T \Delta x_n^D = b_n^I \tag{3a}$$

$$n = 1, 2, \cdots k .$$

$$C_n \Delta x_n^I + D_n \Delta x_n^D = b_n^D \tag{3b}$$

Solving Equation 3b for Δx_n^D, we get

$$\Delta x_n^D = D_n^{-1} \left[b_n^D - C_n \Delta x_n^I \right] . \tag{4}$$

Substituting Equation 4 into Equation 3a and rearranging, we obtain the equation in Δx_n^I :

$$\left[B_n - C_n^T D_n^{-1} C_n\right] \Delta x_n^I = b_n^I - C_n^T D_n^{-1} b_n^D . \qquad (5)$$

Defining the matrices and vectors

$$S_n' = B_n - C_n^T D_n^{-1} C_n$$

$$b_n' = b_n^I - C_n^T D_n^{-1} b_n^D ,$$

we obtain the system of reduced normal equations

$$S_n' \Delta x_n^I = b_n' \qquad n = 1, 2, \cdots k ,$$

where each equation contains only arc independent parameters, i.e., only geopotential or station position parameters.

Next, the Combining of Normal Equations operation is used to "combine" or form a matrix and vector union of the S_n', b_n' denoted by

$$S' = \bigcup_{i=1}^{k} S_i' \qquad \text{(Combined Matrix)}$$

$$b' = \bigcup_{i=1}^{n} b_i' \qquad \text{(Combined right hand sides)} ,$$

which can be regarded as matrix addition, where the rows and columns of S' and elements of b' correspond to all the distinct arc independent parameters occurring over all the S_i', say x^I. Hence the elements of the S_i' and b_i' corresponding to the same parameter, as i varies, are simply added to form single elements in S' and b'. Note that the dimension of S' is then equal to the total number of distinct arc independent parameters. We then have the "combined" normal equation

$$S' \Delta x^I = b' , \qquad (6)$$

where Δx^I is the correction vector, to be determined, containing the corrections to all the distinct geopotential and station position parameters contained in the k equations (Equation 1). This correction vector is obtained by matrix inversion

$$\Delta x^I = S'^{-1} b'$$

or on option, by pseudoinversion (Section 3.3.4).

The solution for the arc dependent parameters is then obtained by the Backsubstitution operation:

$$\Delta x_n^D = D_n^{-1} \left[b_n^D - C_n \Delta x_n^I \right] ,$$

where each x_n^I is a subset of the total arc independent correction vector Δx^I.

The Suppression of Parameters operation is used whenever it is desirable to examine the effects of suppressing the corrections to certain parameters occurring in either the S_i matrices or in the combined matrix S'. This is performed by simply striking out all the rows and columns of the S_i, b_i or S', b' corresponding to these parameters specified for suppression.

3.3.3 EIGENVALUE and Ill-Conditioned Systems

When the system of normal equations

$$S \Delta x = b \tag{1}$$

is ill-conditioned due to poor observability of the parameter set x selected, numerical difficulties may prevent a meaningful solution. A method is available which can, in certain cases, allow one to determine a subset of the original parameter set which is well-determined by the data contributing to S. This method is described in References 5 and 10, and is now outlined.

The basic idea involves a coordinate transformation into a coordinate system with basis elements consisting of eigenvectors of the coefficient matrix S in Equation 1. Let $\{\lambda_i\}_{i=1}^n$ be the eigenvalues of S, where S is $n \times n$, and let $\{\bar{e}_i\}_{i=1}^n$ be the associated eigenvectors, i.e.,

$$S \bar{e}_i = \lambda_i \bar{e}_i \qquad \text{for all } i = 1, 2, \cdots n . \tag{2}$$

Since S is a positive definite or semi-definite symmetric matrix, its eigenvalues are real nonnegative numbers, and the eigenvectors form an orthogonal basis. Let E be the matrix

$$E = \left[\bar{e}_1, \bar{e}_2, \cdots, \bar{e}_n \right] .$$

Then E is an orthogonal matrix, so that $E^{-1} = E^T$, or

$$E^T E = EE^T = I . \tag{3}$$

Moreover, from Equation 2, E can be seen to satisfy the matrix equation

$$SE = DE , \tag{4}$$

where $D = \mathrm{diag}(\lambda_1, \lambda_2, \cdots \lambda_n)$. We can consider D as the matrix S expressed in the coordinate system with basis elements $\{\bar{e}_i\}_{i=1}^{n}$; in fact, we have from Equation 4

$$E^T SE = E^T DE = E^T ED = D . \tag{5}$$

We transform coordinates of the parameter correction and right hand side vectors by

$$\Delta y = E^T \Delta x$$

$$c = E^T b$$

i.e., Δy and are the correction vectors Δx and b in the new coordinate system. We then have (since $E^T = E^{-1}$)

$$E \Delta y = \Delta x$$

and

$$Ec = b .$$

Therefore,

$$S \Delta x = SE \Delta y = b$$

by Equation 1, and

$$E^T SE \Delta y = D \Delta y = E^T b = c$$

and hence in the new coordinate system, the transformed parameter correction vector Δy satisfies the equation

$$D \Delta y = c . \tag{6}$$

Note however, that D is a diagonal matrix, so that the elements of the transformed parameter correction vector Δy which are not well-determined can be discerned by inspection. For example, if $\Delta y = (\Delta y_1, \Delta y_2, \cdots \Delta y_n)$, and if the i^{th} diagonal element of D, or the i^{th} eigenvalue of S, were smaller than some minimum value, the i^{th} component of the correction vector Δy , namely Δy_i , would not be well-determined. But note that each element of Δy is simply a linear combination of elements of the original correction vector Δx, i.e.,

$$\begin{aligned} \Delta y_i &= \bar{e}_i^T \Delta x \\ &= e_{i2} \Delta x_1 + e_{i2} \Delta x_2 + \cdots + e_{in} \Delta x_n \end{aligned}$$

so that the above scheme can be used to establish which linear combinations of original parameter corrections are not well-determined. This in turn can be used to discuss which particular parameter corrections are not well-determined. For example, let $(\lambda_1, \lambda_2, \cdots \lambda_k)$ $(k \leq n)$ be the set of relatively large or "significant" eigenvalues of S, with corresponding eigenvectors $(\bar{e}_1, \bar{e}_2, \cdots \bar{e}_k)$. If it is found that a particular parameter correction Δx_j has a small coefficient in each linear combination

$$\bar{e}_i^T \Delta x = \sum_{m=0}^{n} e_{im} \Delta x_m \qquad i = 1, 2, \cdots k$$

or equivalently, that the j^{th} component of each of the eigenvectors $\{\bar{e}_i\}_{i=1}^{k}$ is small, then this parameter correction cannot be determined from the given matrix S, so that one would then suppress the corresponding parameter from the parameter set, and try to solve the resulting smaller system. Examples of this method, as well as some possible extensions, can be found in Reference 5.

In the GEOSTAR-I system, the calculation of the eigenvalues and eigenvectors of the real symmetric matrix is performed using a SHARE subroutine package called HOW. The matrix is first reduced to tridiagonal form using Householder's method. The eigenvalues are then computed using a method based on Sturm sequences, and the eigenvectors are computed using Wilkinson's method.

The program EIGENVALUE accepts as input the combined matrix resulting from a multiple arc SOLVE run containing geopotential and station location parameters. EIGENVALUE then eliminates the station position parameters, obtaining a reduced normal matrix containing only geopotential coefficient parameters. Note that this reduced matrix still contains the "effects" of the eliminated parameters. This matrix is then nomalized and decomposed into eigenvalues and eigenvectors using the HOW subroutine package. Normalization is performed using the usual harmonic coefficient normalization factors

$$N_{nm} = \left[(n-m)!\,(2n+1)\,K/(n+m)!\right]^{-1/2} ,$$

where

$$K = \begin{cases} 1 & \text{if} \quad m = 0 \\ 2 & \text{if} \quad m \neq 0 . \end{cases}$$

The eigenvalues and eigenvectors are then printed for user analysis of the type described in the first part of this section.

3.3.4 Pseudoinversion and Ill-Conditioned Systems

A numerical approach to the problem of finding a least squares solution of an arbitrary overdetermined linear system

$$Ax = b , \tag{1}$$

where A is an $m \times n$ matrix, $m \geq n$, is the use of the Penrose pseudoinverse or a generalized inverse of the matrix A. Methods of this type were developed to cope with the numerical difficulties arising either from the formation of the normal matrix $A^T A$, or, once formed, with the ill-conditioning or possible singularity of $A^T A$.

The Penrose pseudoinverse of A is defined as a matrix $A^{\#}$ which satisfies the postulates:

$$\begin{aligned} &(i) \quad A A^{\#} A = A \\ &(ii) \quad A^{\#} A A^{\#} = A^{\#} \\ &(iii) \quad (A A^{\#})^T = A^{\#T} A^T = A A^{\#} \\ &(iv) \quad (A^{\#} A)^T = A^T A^{\#T} = A^{\#} A . \end{aligned} \tag{2}$$

It can be shown (Reference 6) that the pseudoinverse of any matrix A always exists, is unique and satisfies the following:

(i) if A^{-1} exists, then $A^{\#} = A^{-1}$;

(ii) if

$$x_0 = A^{\#} b , \tag{3}$$

then x_0 is a least squares solution of Equation 1, that is, the quadratic form

$$R(x) = (Ax - b)^T (Ax - b)$$

is minimized by the vector x_0;

(iii) if more than one least squares solution of Equation 1 exists, then x_0 is the smallest in magnitude of all such solutions, i.e.,

$$x_0^T x_0 \leq x_1^T x_1$$

for all vectors x_1 which are least squares solutions of Equation 1.

The methods which have been developed to compute generalized inverses are either of the type where the matrix $A^{\#}$ is formed directly from the matrix A, generally requiring more core storage or sophisticated algorithms (Reference 11), or, using the already formed normal matrix $A^T A$, the pseudoinverse is found by the equation

$$A^{\#} = (A^T A)^{\#} A^T .$$

In the latter case, given the normal equations

$$[A^T A] x = A^T b = c ,$$

the pseudo solution would be given simply by

$$x = [A^T A]^{\#} c . \qquad (4)$$

In GEOSTAR-I, the pseudoinverse solution for the combined normal system of equations is obtained as in Equation 4. The algorithm used to compute the pseudoinverse of the normal matrix is called the Andree algorithm, which is described in detail in Reference 5 and is outlined in the subroutine description ANDREE in Section 4.4. This subroutine is called on option by the SOLVE program instead of the Gauss-Jordan subroutine. One of the advantages of using this subroutine is its rank determination capability. This feature computes the computational rank of the matrix based on a control parameter indicating the "noise" level of the matrix. The advantage of this feature is that ill-conditioning can be detected by direct examination of the matrix and the rank will be reduced only if necessary. If the matrix is of maximum rank, then by property (3) (i) of the pseudoinverse, the pseudoinverse obtained would be equal to the inverse computed by the Gauss-Jordan method, i.e., in this case

$$(A^T A)^{\#} = (A^T A)^{-1} .$$

If it is not of maximum rank, then numerical difficulties which would result from a direct inverse computation are avoided. It is remarked, however, that the applicability of the pseudoinverse to the problems of physical parameter estimation appears to be in question. The problem seems to be the determination of the physical significance of the pseudo solution in practical applications where noise in the data and physical correlations, as well as numerical noise, contribute to the ill-conditioning of the normal matrix. Studies using simulated data have been encouraging in that the pseudoinverse methods were found to offer distinct advantages over conventional techniques (Reference 6). Further studies are currently underway to determine the applicability of these methods on problems corrupted by noise.

IV. NEW AND UPDATED MODULES IN NONAME ODP AND LUNGFISH MATRIX PROGRAMS

This section describes modifications and additions made to the subroutine structures of the NONAME ODP and LUNGFISH matrix programs for the GEOSTAR-I system.

To obtain the GEOSTAR-I ODP, the NONAME ODP program was modified to allow for geopotential coefficient estimation and new integration methods. To implement these capabilities, the following existing NONAME ODP modules were modified:

MAIN
COEFL
EGRAV
ESTIM
DNVRT1 (DNVERT)
ORB1
PREDCT
VEVAL
INPUT-BLOCK DATA
DRAG
SUNGRV
F(FRCS)

and the following new modules were developed:

For Geopotential Estimation:
READGP
SOLVGP
STORGP
SUMTOB
WTBMAT

For Numerical Integration:
ORBIT
CKDIFF*
CSTEP
EPHQAN
HEMINT
INV2,3
MMATRX
RK
SUMS
SWTEST
TABLE
TABLEB*
TEST*

To achieve the multiple arc capability of the GEOSTAR-I system, the LUNGFISH MERGE and SOLVE programs are used. Also, for multiple arc correlation analysis, the LUNGFISH EIGENVALUE program is used.

The MERGE program is used without any modifications. The SOLVE program was modified to allow for handling of a larger number of station location parameters in rectangular and geodetic coordinates, to extend the "combine" matrix capabilities, to include pseudoinversion, and to satisfy I/O interface requirements with the GEOSTAR-I ODP. To implement these capabilities, the following existing SOLVE modules were modified:

MAIN
BEDIT
CALTYP
INVERT
LBLSUP
OPARC
OPGRAV
OPSTAT
SUPRSS
UPCOMB
ELIM

*Subroutines for the variable stepsize version only; see Appendix A.

and the following new modules were developed:

ANDREE
PHLINN
OUTRAD

The LUNGFISH EIGENVALUE program, which computes the eigenvalues and eigenvectors of the combined normal equations resulting from a SOLVE operation, was modified to allow for the elimination of the station parameters from the matrix before decomposition, so that the geopotential parameters contributing to the solution can be examined separately, while including the effects of the eliminated parameters. This was accomplished by adding the module ELIM.

In the following sections, those modules listed above which are either new or significantly modified existing subroutines will be documented in detail. The remaining modules which received minor modifications are only briefly outlined, with changes indicated. Further details of these, as well as the unmodified NONAME ODP or LUNGFISH SOLVE subroutines can be found in References 1 and 2.

4.1 Modifications to Existing NONAME ODP Modules

The modifications made to the NONAME executive program MAIN, and some of the other existing modules, are designed to:

- Call the new subroutines.
- Extend data tables and variables to accommodate the geopotential coefficients through order 30 and the new geopotential partials.
- Extend the position partial computation algorithms to include partials with respect to geopotential coefficients.
- Modify the computational algorithm to allow for independent integrations of the equations of motion and the variational equations.
- Extend the program options applicable to geopotential coefficients and integration features.

A summary of these modifications follows:

- MAIN — the ODP control program. Modified to control the new subroutines READGP, SOLVGP, STORGP, WTBMAT, INV3 and to include the I/O required by the geopotential estimation process. To allow for effective compilation, this program was divided into four subprograms: MAIN, OPTCRD, STATRD, OUTPUT.

- ESTIM — sums all observation partials into the normal equations matrix and right hand sides for each observation data point forming the normal equations.

The subroutine was modified at entry three to exit before the solution is formed (subroutine SOLVGP replaces that portion of the routine), and also, to provide weights using a full variance-covariance matrix (using the VARCOV option card).

- ELEMK — the subroutine ELEM which converts inertial vectors to Keplerian elements was modified and called ELEMK to provide the I/O in the calling sequence rather than COMMON.

- COEEL, ORB1 — modified to extend the geopotential coefficient data tables to order 30.

- INPUT — modified to extend the geopotential coefficient data tables to order 30, extend integration coefficients (orders 4-15), extend the available program options, and extend data base constants required for the new program options.

- PREDCT — computes the observation partials using position partials. The subroutine was modified to accommodate the additional measurement partials with respect to the geopotential coefficients, and six more observation types (rectangular coordinate measurement types-x, y, z, $\dot{x}$, $\dot{y}$, $\dot{z}$).

- DRAG — computes the acceleration of a satellite due to drag forces. Modified to not compute density above 1000 kilometers.

- SUNGRV — computes the acceleration on a satellite due to the gravitational attraction of a disturbing body. Modified to save variables for the computation of the partials of the acceleration due to the sun with respect to instantaneous position and velocity.

- FRCS — an executive routine calling various subprograms which evaluate the accelerations of a satellite due to the various forces acting on it. Modifications were made to allow for the case of a two-body gravity model, and for the recomputation of the ephemeris quantities when necessary.

- DNVRT1 — the subroutine DNVERT, which is a double precision matrix inversion routine using the Gauss-Jordan method of condensation with partial (column) pivoting, was modified to indicate an error condition due to a negative or zero pivot element.

4.2 New GEOSTAR-I ODP Modules and Significantly Modified NONAME ODP Modules

The following GEOSTAR-I modules were written to allow for geopotential estimation, satisfy I/O interface requirements with the LUNGFISH MERGE and SOLVE programs, and numerically integrate the equations of motion and the variational equations.

The orbit generator portion of the NONAME ODP program was replaced by a new set of subroutines which improves efficiency and provides additional capabilities to the system. Those NONAME subroutines which evaluate the force model associated with the satellite accelerations and acceleration partials have been left essentially unchanged, except for some additions which may improve the accuracy of the acceleration partials.

Those subroutines effected by the new GEOSTAR-I orbit generator are summarized in Table 1.

Table 1

Replaced NONAME Orbit Generator Routines	New GEOSTAR-I ODP Routines (approximate equivalents)
COWELL	INV2, CSTEP, MMATRX, SUMS, TEST,* CKDIFF*
HERMIT	HEMINT
REARG, HHEMIT	TABLEB*
F, EGRAV, VEVAL	FRCS, EGRAV, VEVAL
INTGST	TABLE, RK
ORBIT	ORBIT, EPHQAN, SWTEST

*Variable stepsize version.

CSTEP

Purpose:

- To integrate the satellite equations of motion using a summed ordinate form of the Stőrmer/Cowell predictor-corrector formulas for position, and Adams-Bashforth/Moulton formulas for velocity.

- To integrate the variational equations using a summed "corrector-only" form of the Cowell formula for position partials and Moulton formula for velocity partials.

Called By:

ORBIT

Calls:

FRCS
HEMINT
VEVAL
INV2
MMATRX

Method:

- Integration of Equations of Motion

Let t_{n+1} be the time point at which the satellite position and velocity is to be computed. Letting $\ddot{\bar{x}}_i$ be the satellite acceleration vector at time t_i, scaled by the factor h^2, the predicted position and velocity vectors are computed from:

$$\bar{x}_{n+1} = {}^{II}\bar{S}_n + \sum_{i=0}^{k_e} \alpha_i \ddot{\bar{x}}_{n-i} \qquad \text{(Stőrmer)}$$

$$\dot{\bar{x}}_{n+1} = \frac{1}{h}\left[{}^{I}\bar{S}_n + \sum_{i=0}^{k_e} \beta_i \ddot{\bar{x}}_{n-i}\right], \qquad \text{(Bashforth)}$$

and successive corrections by:

$$\bar{x}_{n+1} = {}^{II}\bar{S}_n + \sum_{i=0}^{k_e} \alpha_i^* \ddot{\bar{x}}_{n+1-i} \qquad \text{(Cowell)}$$

$$\dot{\bar{x}}_{n+1} = \frac{1}{h}\left[{}^{I}\bar{S}_{n} + \sum_{i=0}^{k_e}{}' \beta_i^{*}\ddot{\bar{x}}_{n+1-i}\right], \qquad \text{(Moulton)}$$

where $\ddot{\bar{x}}_{n+1}$ is computed using the subroutine FRCS whenever a successive correction is necessary. The maximum number of corrections allowed is three. After convergence, the sums ${}^{I}\bar{S}_{n+1}$, ${}^{II}\bar{S}_{n+1}$ are computed using:

$${}^{I}\bar{S}_{n+1} = {}^{I}\bar{S}_{n} + \ddot{\bar{x}}_{n+1}$$

$${}^{II}\bar{S}_{n+1} = {}^{II}\bar{S}_{n} + {}^{I}\bar{S}_{n+1} .$$

In the case when no drag is used, the velocity predictor is not used and the corrector is only applied once.

The nominal values for the stepsize h and order used for the integration of the equations of motion are:

$$h = 100 \text{ sec}$$

$$\text{Order} = 11 \qquad (k_e = 9)$$

which can be changed on option.

- Integration of Variational Equations

Let t_{n+1} be the time point at which the position and velocity partials are to be computed. Let $\bar{y}_i$, $\dot{\bar{y}}_i$ denote position and velocity partials with respect to a particular parameter at time, t_i and $\ddot{\bar{y}}_i$ the corresponding acceleration partial, scaled by h^2. Note that this h^2 need not be the same as the one used in the satellite acceleration vector.

The satellite position and velocity vectors at time t_{n+1} are first computed. If the stepsizes being used are not equal, these vectors are obtained by interpolation. Next, using the subroutine VEVAL, the acceleration partials with respect to position and velocity,

$$A_{n+1} = \frac{\partial\ddot{\bar{x}}_{n+1}}{\partial\bar{x}}$$

$$B_{n+1} = \frac{\partial\ddot{\bar{x}}_{n+1}}{\partial\dot{\bar{x}}} ,$$

and with respect to the parameter x_j,

$$\bar{f}_{n+1} = \frac{\partial\ddot{\bar{x}}_{n+1}}{\partial x_j} ,$$

are computed.

The vectors $\bar{X}_n$, $\bar{V}_n$ are then formed by:

$$\bar{X}_n = h^2 \alpha_0^* \bar{f}_{n+1} + {}^{II}\bar{S}_n + \sum_{i=1}^{k_v} \alpha_i^* \ddot{\bar{y}}_{n+1-i}$$

$$\bar{V}_n = \frac{1}{h} \left[h^2 \beta_0^* f_{n+1} + {}^{I}\bar{S}_n + \sum_{i=1}^{k_v} \beta_i^* \ddot{\bar{y}}_{n+1-i} \right] ,$$

where ${}^{I}\bar{S}_n$, ${}^{II}\bar{S}_n$ are the associated sums for the acceleration partials. The solution

$$\begin{bmatrix} y_{n+1} \\ \bar{y}_{n+1} \end{bmatrix}$$

is then computed by

$$\begin{bmatrix} \bar{y}_{n+1} \\ \dot{\bar{y}}_{n+1} \end{bmatrix} = [I - H]^{-1} \begin{bmatrix} \bar{X}_n \\ \bar{V}_n \end{bmatrix} ,$$

where I is the 6 × 6 identity matrix and

$$H = \begin{bmatrix} h^2 \alpha_0^* A_{n+1} & h^2 \alpha_0^* B_{n+1} \\ h \beta_0^* A_{n+1} & h \beta_0^* B_{n+1} \end{bmatrix}_{(6\times 6)}$$

In the case when there is no drag, $B_n \equiv 0$ for all n, so that we simply have

$$\bar{y}_n = \left[I - h^2 \alpha_1^* A_{n+1} \right] \bar{X}_n ,$$

and $\ddot{\bar{y}}_{n+1}$ computed directly from the corrector formula

$$\dot{\bar{y}}_{n+1} = \bar{V}_n + h \beta_0^* A_{n+1} \bar{y}_{n+1} .$$

The above computations are performed for each position and velocity partial required. Note however, that the matrix $(I - H)^{-1}$ is independent of the particular partials being computed, so that it is only formed once.

The nominal values for the stepsize and order used for the integration of the variational equation are:

$$h = 100 \text{ sec}$$

$$\text{Order} = 7 \quad (k_v = 5) ,$$

which can be changed on option, independently of those used for the integration of the equations of motion.

- Integration Coefficients

The integration coefficients are brought in through the COMMON/ABCOEF/from the BLOCK DATA subprogram. The Adams-Bashforth Predictor Coefficients (ALPHA), Adams-Moulton Corrector Coefficients (ALPHAS), Störmer Predictor Coefficients (BETA) and Cowell Corrector Coefficients (BETAS) for the summed ordinate forms of these.equations were computed to 18 digit accuracy (Reference 8). Two indices, IB(1) and IB(2), computed in the ORBIT subprogram indicate which set of coefficients to use for the equations of motion and which set of coefficients to use for the variational equations based on the requested or nominal order formula to be used in these integrations.

Calling Sequence:

CALL CSTEP (IEQ)

COMMON Blocks Used:

ABCOEF
WORKER
LIMITS
XYZ
ANPART
GRBLOK
DC

Variables Not In COMMON:

FORTRAN Name	Format	Description
IEQ	I	Indicates which equations are to be integrated
SCO (3)	D	Nonsummed corrected value of position vector
SPCO (3)	D	Nonsummed predicted value of position vector
SN (6, 50)	D	The storage array for the vectors $\overline{X}_n$ and $\overline{V}_n$ described in the method
C (6, 50)	D	Instantaneous values of the position partials

EGRAV

Purpose:

- Calculates the satellite acceleration vector in rectangular coordinates due to the geopotential.
- Calculates acceleration partials in rectangular coordinates with respect to geopotential coefficients.

Called By:

FRCS
VEVAL

Method

The accelerations due to the geopotential are given by:

$$\ddot{\overline{x}}_E = \frac{\partial U}{\partial \overline{x}} = \frac{\partial U}{\partial \overline{\phi}} \frac{\partial \overline{\phi}}{\partial \overline{x}} ,$$

where $\overline{\phi} = (r, \lambda, \psi)$ and U is the earth's potential. The components of the vector

$$\frac{\partial U}{\partial \overline{\phi}} = \left(\frac{\partial U}{\partial r}, \frac{\partial U}{\partial \lambda}, \frac{\partial U}{\partial \psi} \right)$$

are given by:

$$\frac{\partial U}{\partial r} = \frac{GM}{r^2} \left[1 + \sum_{n=2}^{30} \left(\frac{a_e}{r} \right)^n \sum_{m=0}^{n} \left(C_{nm} \cos m\lambda + S_{nm} \sin m\lambda \; (n + 1) P_n^m (\sin \psi) \right. \right]$$

$$\frac{\partial U}{\partial \lambda} = \frac{GM}{r} \sum_{n=2}^{30} \left(\frac{a_e}{r} \right)^n \sum_{m=0}^{n} \left(S_{nm} \cos m\lambda - C_{nm} \sin m\lambda \right) m P_n^m (\sin \psi)$$

$$\frac{\partial U}{\partial \psi} = \frac{GM}{r} \sum_{n=2}^{30} \left(\frac{a_e}{r} \right)^n \sum_{m=0}^{n} \left(C_{nm} \cos m\lambda + S_{nm} \sin m\lambda \right) \left[P_n^{m+1} (\sin \psi) - m \tan \psi \, P_n^m (\sin \psi) \right]$$

where

GM = gravitational constant times mass of earth

r = satellite radius vector

a_e = earth's semi-major axis

λ = geocentric longitude of satellite

ψ = geocentric latitude of satellite

P_n^m = associated Legendre polynomials

C_{nm}, S_{nm} = harmonic coefficients

The matrix $\partial\bar{\phi}/\partial\bar{x}$ is given by:

$$\frac{\partial\bar{\phi}}{\partial\bar{x}} = \begin{bmatrix} x/r & y/r & z/r \\ -y/s^2 & x/s^2 & 0 \\ -xz/r^2 s & -yz/r^2 s & s/r^2 \end{bmatrix}, \qquad s = \left(x^2+y^2\right)^{1/2}.$$

To compute the Legendre polynomials P_n^m and the trigonometric terms, the following recursive relations are used:

$$P_n^0(x) = \left[(2n-1)\,xP_{n-1}^0(x) - (n-1)P_{n-2}^0\right]\Big/n$$

$$P_n^m(x) = P_{n-2}^m + (2n-1)\left(1-x^2\right)^{1/2} P_{n-1}^{m-1}(x),$$

and for sectorials ($m = n$), since the first term is zero,

$$P_n^m(x) = (2n-1)\left(1-x^2\right)^{1/2} P_{n-1}^{m-1}(x),$$

$$\sin m\lambda = 2\cos\lambda\,\sin(m-1)\lambda - \sin(m-2)\lambda$$

$$\cos m\lambda = 2\cos\lambda\,\cos(m-1)\lambda - \cos(m-2)\lambda$$

$$m\tan\psi = (m-1)\tan\psi + \tan\psi\,.$$

The gravitational accelerations in rectangular coordinates are then computed by:

$$\ddot{x}_E = \frac{\partial U}{\partial \bar{\phi}} \frac{\partial \bar{\phi}}{\partial \bar{x}} = \left(\frac{\partial U}{\partial r}, \frac{\partial U}{\partial \lambda}, \frac{\partial U}{\partial \psi}\right) \left[\frac{\partial \bar{\phi}}{\partial \bar{x}}\right] .$$

The acceleration partials with respect to the harmonics C_{nm} and S_{nm} are obtained from

$$\frac{\partial \ddot{\bar{x}}_E}{\partial C_{nm}} = \frac{\partial}{\partial C_{nm}} \left(\frac{\partial U}{\partial r}, \frac{\partial U}{\partial \lambda}, \frac{\partial U}{\partial \psi}\right) \left[\frac{\partial \bar{\phi}}{\partial \bar{x}}\right]$$

$$= \left(\frac{\partial}{\partial C_{nm}} \frac{\partial U}{\partial r}, \frac{\partial}{\partial C_{nm}} \frac{\partial U}{\partial \lambda}, \frac{\partial}{\partial C_{nm}} \frac{\partial U}{\partial \psi}\right) \left[\frac{\partial \bar{\phi}}{\partial \bar{x}}\right]$$

and

$$\frac{\partial \ddot{\bar{x}}_E}{\partial S_{nm}} = \left(\frac{\partial}{\partial S_{nm}} \frac{\partial U}{\partial r}, \frac{\partial}{\partial S_{nm}} \frac{\partial U}{\partial \lambda}, \frac{\partial}{\partial S_{nm}} \frac{\partial U}{\partial \psi}\right) \left[\frac{\partial \bar{\phi}}{\partial \bar{x}}\right] ,$$

where

$$\frac{\partial}{\partial C_{nm}} \frac{\partial U}{\partial r} = -\frac{GM}{r^2} \left(\frac{a_e}{r}\right)^n (n+1) \cos m\lambda P_n^m (\sin \psi)$$

$$\frac{\partial}{\partial C_{nm}} \frac{\partial U}{\partial \lambda} = -\frac{GM}{r} \left(\frac{a_e}{r}\right)^n m \sin m\lambda P_n^m (\sin \psi)$$

$$\frac{\partial}{\partial C_{nm}} \frac{\partial U}{\partial \psi} = \frac{GM}{r} \left(\frac{a_e}{r}\right)^n \cos m\lambda \left[P_n^{m+1} (\sin \psi) - m \tan \psi P_n^m (\sin \psi)\right]$$

$$\frac{\partial}{\partial S_{nm}} \frac{\partial U}{\partial r} = -\frac{GM}{r^2} \left(\frac{a_e}{r}\right)^n (n+1) \sin m\lambda P_n^m (\sin \psi)$$

$$\frac{\partial}{\partial S_{nm}} \frac{\partial U}{\partial \lambda} = \frac{GM}{r} \left(\frac{a_e}{r}\right)^n m \cos m\lambda P_n^m (\sin \psi)$$

$$\frac{\partial}{\partial S_{nm}} \frac{\partial U}{\partial \psi} = \frac{GM}{r} \left(\frac{a_e}{r}\right)^n \sin m\lambda \left[P_n^{m+1} (\sin \psi) - m \tan \psi P_n^m (\sin \psi)\right]$$

The C and S coefficients are described by indices n and m. For the zonal harmonics, m = 0. For the sectorial harmonics, m = n. For the tesseral harmonics, m < n. Therefore, the C coefficients only fill one triangle of a matrix, as do the S. To conserve space, both sets of coefficients were combined into one matrix. A diagram of that matrix and the computations for the subscripts corresponding to the n and m indices appear in Table 2 and a list of the SAO denormalized coefficients used by the program appears in the FMODEL COMMON block description in Section 5.1.

Table 2

Structure of 30 × 33
Harmonic Coefficients Array

CS (30, 33)

$$\begin{array}{ccccccc}
C_1^0 & C_1^1 & S_{30}^{30} & S_{30}^{29} & \cdots & S_{30}^{1} & S_{30}^{0} \\
C_2^0 & C_2^1 & C_2^2 & S_{29}^{29} & & & S_{29}^{0} \\
\vdots & & \ddots & \ddots & & & \\
C_{29}^{0} & & & & & S_2^1 & S_2^0 \\
C_{30}^{0} & C_{30}^{1} & \cdots & C_{29}^{29} & C_{30}^{30} & S_1^1 & S_1^0
\end{array}$$

	Index	Matrix Subscript Computation
C coefficients	n	N
	m	M + 1
S coefficients	n	31 - N
	m	33 - M

Calling Sequence:

CALL EGRAV (THETG, AE, GM, RASAT, DX, SWITCH)

COMMON Blocks Used:

FMODEL	ESTGP
XYZ	SETSW
GRBLOK	IORBIT
LIMITS	

Variables Not in COMMON:

FORTRAN Name	Format	Description
THETG	D	Right ascension of the Greenwich meridian
AE	D	Equatorial radius of earth
GM	D	Gravitational constant times mass of earth
RASAT	D	Right ascension of the satellite
DX (3)	D	Satellite acceleration due to earth's gravity in rectangular coordinates
SWITCH	L	A switch which if set to TRUE indicates that the acceleration partials with respect to the geopotential coefficients will be computed
LAMBDA	D	λ-geocentric longitude of the satellite
P (32, 30)	R	P_n^m -associated Legendre polynomials of degree m and order n
C (30, 33)	R	C_{nm}-coefficients of the cosine spherical harmonic
S (30, 33)	R	S_{nm}-coefficients of the sine spherical harmonic
PCNM (3)	D	Intermediate quantities used in the computation of acceleration partials with respect to harmonic coefficients
T0	D	
T1	D	

EPHQAN

Purpose:

To control and determine when to calculate the three components of the moon's inertial unit vector and geocentric distance, the sun's inertial unit vector and geocentric distance, and the equation of the equinoxes.

Called By:

ORBIT

Calls:

COEFF
DJUL
EQN
MOONAD
SUN
TDIF

Method:

In order to calculate the ephemeris quantities, the coefficients of nine 4th order degree polynomials are estimated using 11 calculated values of each quantity spaced over a 2.5 day arc as observations. The degree of the polynomial, the number of observations, and the length of arc can be changed by altering these quantities in the BLOCK DATA subprogram.

Calling Sequence:

CALL EPHQAN

COMMON Blocks Used:

CONST 3
COFIT
COSAVE
PCOFIT

Variables Not in COMMON:

FORTRAN Name	Format	Description
DJ	D	Julian date of ephemeris times used to compute the ephemeris polynomials
AO (9, 11)	D	Array containing all ephemeris quantities to be used for the polynomials fit
TIME (11)	D	Times of the ephemeris quantities
VAR (11)	D	Ephemeris quantities to be used for the polynomial fit
S1	D	S3 /2
E	D	True obliquity of date
CE	D	Cosine of true obliquity of date
SE	D	Sine of true obliquity of date

GTIMIN/GTIMOT

Purpose:

Measures the elapsed time of each iteration and computes the total elapsed time of a run in hundredths of seconds.

Called By:

MAIN

Method:

- The subroutine GTIMIN saves the time at which it is called but returns nothing to the program. However this subroutine must be called prior to the calling of GTIMOT.
- The subroutine GTIMOT returns the elapsed time as measured by subtracting the time of day GTIMIN was called from the time of day GTIMOT was called.

Calling Sequence:

```
CALL GTIMOT (TX)
```

TX—Name of the single precision variable containing the value of the elapsed time in seconds.

HEMINT

Purpose:

To interpolate for position, velocity and the associated partials.

Called By:

ORBIT
CSTEP

Method:

The position and velocity components of the equations of motion and the associated partials are interpolated using the Hermite formula:

$$f(x) = \sum_{j=0}^{k} h_j(x)\, f_j + \dot{h}_j(x)\, \dot{f}_j \;,$$

where

$$h_j(x) = 1 - 2\left(x - x_j\right) \ell_j'\left(x_j\right) \ell_j^2(x)$$

$$\dot{h}_j(x) = \left(x - x_j\right) \ell_j^2(x)$$

$$\ell_j(x) = \prod_{\substack{i=0 \\ i \neq j}}^{k} \frac{\left(x - x_i\right)}{\left(x_j - x_i\right)}$$

$$\ell_j'(x) = \ell_j(x) \sum_{\substack{i=0 \\ i \neq j}}^{k} \frac{1}{\left(x - x_i\right)} \;.$$

The order k is set to max (5, P/2).

Calling Sequence:

CALL HEMINT (TI, STEPZ, K1, IEQ)

COMMON Blocks Used:

WORKER
LIMITS

Variables Not in COMMON:

FORTRAN Name	Format	Description
TI	D	Interpolation time
STEPZ	D	Distance between data points
K1	I	Starting index of points to be interpolated
IEQ	I	Indicates which equations are being interpolated
M	I	Interpolation order (≥5)
SMI	D	Legendre interpolating polynomial
SJMI	D	Derivative of Legendre interpolating polynomial
HX	D	Hermite coefficient
HXB	D	Derivative of Hermite coefficient

INV2/INV3

Purpose/Method:

To invert a matrix by using the Gauss-Jordan method of condensation with partial (column) pivoting.

Note:

The INV2 and INV3 subprograms are the same except that one passes variables through COMMON whereas the other passes variables using a calling sequence. The INV2 subprogram is used specifically to invert the matrix containing the Jacobian of accelerations with respect to position and velocity: the INV3 subprogram is used to invert matrices as required by other subprograms.

INV2 Called By:

CSTEP

INV3 Called By:

MAIN
COEFF

Calling Sequence:

CALL INV3 (N, A, M, DETERM)

Variables:

FORTRAN Name	Format	Description
A (50, 50)	D	Matrix to be inverted and subsequent inverted matrix
N	I	Dimension of portion of matrix to be inverted
M	I	Maximum dimension of matrix to be inverted
DETERM	D	Determinant of matrix inverted

MMATRX

Purpose:

To multiply an $n \times n$ matrix with an $n \times m$ matrix where n need not equal m.

Called By:

CSTEP

Method:

Standard matrix multiplication,

$$C = A \times B ,$$

where

$$c_{ij} = \sum_{k=0}^{n} a_{ik} b_{kj} \qquad \begin{array}{l} i = 1, 2, \cdots n \\ j = 1, 2, \cdots m . \end{array}$$

Calling Sequence:

CALL MMATRX (A, B, N, M, C)

Variables:

FORTRAN Name	Format	Description
A (6, 6)	D	Dimensions of input and output matrices.
B (6, 50)	D	
N, M	I	Input and output matrices
C (6, 50)	D	Output matrix

ORBIT

Purpose:

To control the integration of the orbit and the variational equations.

Called By:

MAIN
ORB1

Calls:

SWTEST
EPHQAN
FRCS
VEVAL
TABLE
SUMS
CSTEP
HEMINT
REFCOR

Method:

Orbit has five functions: 1) to initialize the required variables and constants, 2) to compute the starting table for the integration, 3) to compute the initial first and second sums for the integration, 4) to control the calculation of the ephemeris quantities, and 5) to control the orbit and variational equations' integration and interpolation processes.

Calling Sequences:

CALL ORBIT (PXPX0)

COMMON Blocks Used:

CONST1
ESTGP
NON2
XYZOUT
COFIT
CONST3
WORKER
LIMITS
GRBLOK
ANPART
IORBIT
CONVRG
ABCOEF

Variables Not in COMMON:

FORTRAN Name	Format	Description
PXPX0 (50, 6)	D	An array containing partials of position and velocity with respect to parameters to be estimated
B0	R	1/2 area of satellite/mass of satellite
TI	D	Observation time in days
IEND	I	Total number of array points for position, velocity, and position partials
NO	I	Number of array points to reset
IEQ	I	Equation indicator
NEO	I	
ETIME	D	Time to recompute the ephemeris quantities
TOUT	D	Time from epoch to current integration time in minutes

READGP

Purpose:

- To read in data to modify the stored values of the geopotential coefficients.
- To read data determining which geopotential coefficients are to be estimated and to set up a table of the requested parameters.
- To read other data options defining the geopotential estimation problem.

Called By:

OPTCARD

Method:

The call to READGP is initiated by the COEFGP option card. The remaining geopotential coefficient option cards are in two categories: EST and CHG cards.

CHG cards contain the coefficient name and value to replace the prestored data base in COMMON/FMODEL/CS. No other action takes place. There is no limit on the number of change cards.

EST cards request the gravity coefficient indicated to be estimated. The initial estimate is taken from COMMON/FMODEL/CS unless the EST1 card contains a value for this estimate. The limit of EST cards is 50. The arrays of coefficient subscript indicators, estimated values and sigmas are sorted to place all C's first, then S's. Note that the a priori values and estimate values are initially the same.

Calling Sequence:

CALL READGP (INTP, IARRAY, DT1, T2, T3, T4, T5)

COMMON Blocks Used:

FMODEL
ESTGP
NON2

Variables Not in COMMON:

FORTRAN Name	Format	Description
INTP	I	Data set to read from, usually FT05
IARRAY (1)	I	Resets logical indicator of ESTGP/ISTATE

IARRAY (2), (3)	I	Resets value of /ESTGP/ITERGP
IARRAY (4)	I	Resets value of/NON1/JTEMP
DT1	D	Not used
T2	D	Sets logical indicator of/LIMITS/ ISWT (13)
T3-T5	D	Not used

RK

Purpose:

To integrate the equations of motion and the variational equations to obtain the starting values of position, velocity, and partials for the multistep Cowell integrator.

Called By:

TABLE

Calls:

FRCS
VEVAL

Method:

The method used to obtain the starting values for the multistep integration of the equations of motion and variational equations is an eight order Runge-Kutta integrator. This method is defined as follows: Given a system of first order differential equations

$$\dot{\bar{x}} = \bar{f}(t, \bar{x})$$

with initial vector $\bar{x}(t_0) = \bar{x}_0$, the solution vector at time $t_1 = t_0 + h$, h a given stepsize, is computed by:

$$x(t_1) = x(t_0) + \frac{h}{840}\left[41(\bar{K}_0 + \bar{K}_9) + 27(\bar{K}_3 + \bar{K}_5) + 272\bar{K}_4 + 216(\bar{K}_6 + \bar{K}_8)\right],$$

where

$$\bar{K}_0 = \bar{f}(t_0, \bar{x}_0)$$

$$\bar{K}_1 = \bar{f}\left(t_0 + \frac{4h}{27}, \bar{x}_0 + \frac{4}{27}\bar{K}_0\right)$$

$$\bar{K}_2 = \bar{f}\left(t_0 + \frac{2h}{9}, \bar{x}_0 + \frac{1}{18}\left[\bar{K}_0 + 3\bar{K}_1\right]\right)$$

$$\bar{K}_3 = \bar{f}\left(t_0 + \frac{h}{3}, \bar{x}_0 + \frac{1}{12}\left[\bar{K}_0 + 3\bar{K}_2\right]\right)$$

$$\bar{K}_4 = \bar{f}\left(t_0 + \frac{h}{2},\ \bar{x}_0 + \frac{1}{8}\left[\bar{K}_0 + 3\bar{K}_3\right]\right)$$

$$\bar{K}_5 = \bar{f}\left(t_0 + \frac{2h}{3},\ \bar{x}_0 + \frac{1}{54}\left[13\bar{K}_0 - 27\bar{K}_2 + 42\bar{K}_3 + 8\bar{K}_4\right]\right)$$

$$\bar{K}_6 = \bar{f}\left(t_0 + \frac{h}{6},\ \bar{x}_0 + \frac{1}{4320}\left[389\bar{K}_0 - 54\bar{K}_2 + 966\bar{K}_3 - 824\bar{K}_4 + 243\bar{K}_5\right]\right)$$

$$\bar{K}_7 = \bar{f}\left(t_0 + h,\ \bar{x}_0 + \frac{1}{20}\left[-231\bar{K}_0 + 81\bar{K}_2 - 1164\bar{K}_3 + 656\bar{K}_4 - 122\bar{K}_5 + 800\bar{K}_6\right]\right)$$

$$\bar{K}_8 = \bar{f}\left(t_0 + \frac{5h}{6},\ \bar{x}_0 + \frac{1}{288}\left[-127\bar{K}_0 + 18\bar{K}_2 - 678\bar{K}_3 + 456\bar{K}_4 - 9\bar{K}_5 + 576\bar{K}_6 + 4\bar{K}_7\right]\right)$$

$$\bar{K}_9 = \bar{f}\left(t_0 + h,\ \bar{x}_0 + \frac{1}{820}\left[1481\bar{K}_0 - 81\bar{K}_2 + 7104K_3 - 3376\bar{K}_4 + 72\bar{K}_5 - 5040\bar{K}_6 - 60\bar{K}_7 + 720\bar{K}_8\right]\right)$$

The stepsize h used in this integrator is nominally set at 24 seconds and can be modified on option. The subroutine accepts as input the values of the equations being integrated at time t_i, the request time $t_{i+1} = t_i + h_{Cowell}$. It then performs N integration steps with the Runge-Kutta stepsize h, where N is the integer

$$N = \left[\frac{t_{i+1} - t_i}{h}\right]$$

followed by one final integration step with $h = t_{i+1} - (t_i + Nh)$.

Calling Sequence:

CALL RK

COMMON Blocks Used:

WORKER
RKST
RKT
ANPART
GRBLOK

Variables Not in COMMON:

FORTRAN Name	Format	Description
R* (10, 6, 50)	D	The K_i required by the method
F (6, 50)	D	The f_i required by the method

*NOTE: R is equivalenced to MYMD of COMMON/PRIORI/, thus writing over the flux tables after initial use.

SOLVGP

Purpose:

To solve the normal equations for the parameter corrections. This subroutine performs an accuracy check on the inverted normal matrix. If it is found that numerical difficulties were encountered by the normal inversion process due to poor conditioning, a gradient method is used to obtain the parameter corrections.

Called By:

MAIN

Calls:

SYMMET
DNVRT1

Method:

As input, SOLVGP is given the normal equations

$$S\Delta x = b$$

to solve for the parameter corrections Δx, where

$$S = \left[A^T WA + P_0^{-1}\right] ,$$

the normal matrix summed over all measurements and

$$b = A^T W\Delta O - P_0^{-1} \Sigma\Delta x ,$$

the right hand side of the normal equations. In these equations:

A = the matrix of measurement partials with respect to the parameters to be estimated

W = measurement weight matrix

P_0 = a priori covariance matrix

ΔO = the vector of residuals

$\Sigma\Delta x$ = the accumulated change in Δx over previous iterations.

Reference the subroutine ESTIM (Reference 1, Vol. 2) for details on the formation of the normal equations.

SOLVGP performs the following sequence of computations on the normal matrix S:

(i) The matrix S is first normalized to form the matrix H where the elements of H are defined by:

$$h_{ij} = \frac{s_{ij}}{\sqrt{s_{ii}}\,\sqrt{s_{jj}}}, \qquad S = \{s_{ij}\} .$$

If $s_{ii} < 0$ for any i, transfer to step viii.

(ii) Invert H using the subroutine DNVRT1; error exit from DNVERT resulting from zero divisor defaults to the gradient option (viii). DNVRT1 uses the Gauss-Jordon method of condensation with partial pivoting. The result of this inversion process is D where, hopefully, $H^{-1} = D$.

(iii) Test for the validity of the inverted matrix, $H^{-1} = D$ by multiplying the matrix by its supposed inverse and comparing the result to an identity matrix i.e., compute the maximum element of the product matrix HD where 1 is subtracted from the diagonal elements.

(iv) The resulting maximum, MAX, is tested against some tolerance, TOL, (initially 1×10^{-8}). If MAX > TOL, then the resulting inverted matrix D is not accurately H^{-1} and is considered ill-conditioned. The gradient option (viii) is the default for this condition.

(v) For MAX < TOL, the solution is valid and the elements of S^{-1} are given by

$$s_{ij}^{-1} = \sqrt{s_{ii}}\, d_{ij}\, \sqrt{s_{jj}} .$$

(vi) Then the correction vector Δx is computed by $\Delta x = S^{-1} b$ and S^{-1} is the variance-covariance matrix.

(vii) Program exit.

(viii) Gradient option: This technique is an attempt to get a solution if errors resulted at (i), (ii) or (iv) above.

For approximating the correction vector Δx, evaluate $y = Sb$, compute scale factor

$$\sigma = \frac{b^T y}{y^T y} ,$$

and calculate Δx as x [illegible] b.

The error signals of 1, 2, and 3 are used for output to indicate to the user that the Δx corrections obtained at that iteration are approximations resulting from the use of the gradient option. For error 3 the D matrix is available for display.

Calling Sequence:

CALL SOLVGP (DELTA, NPARAM)

COMMON Blocks Used:

PRIORI
BEQ

Variables Not in COMMON:

FORTRAN Name	Format	Description
DELTA (50)	D	Parameter corrections
NERR	I	Error indicator; zero = no error

STORGP

Purpose/Method:

The purpose of STORGP is to initialize the internal arrays of geopotential coefficients (CNM and SNM) used by the ORBIT subroutine VEVAL with the values from the data base reference. /FMODEL/CS.STORGP is called at the beginning of each DC iteration. Thus, in a DC geopotential estimation problem or when altering the geopotential model by the CHG card option, consistent values of the model are used. The arrays CNM and SNM contain values for the tesserals through order 3 and zonals through order 4.

Called By:

MAIN

Calling Sequence:

CALL STORGP

COMMON Blocks Used:

FMODEL
CSVEVL

SUMS

Purpose:

To compute the first and second sums, ${}^{I}\bar{S}_n$, ${}^{II}\bar{S}_n$ where n = order - 1, necessary for the summed form of the predictor-corrector formulas used in the multistep integration process. This is done only once for each iteration—during the initialization of the orbit generator.

Called By

ORBIT

Method:

Let $\ddot{\bar{x}}_n$ denote the acceleration vector, or acceleration partial with respect to a parameter, where it has been scaled by the factor h^2. The first and second sums are defined by:

$$
\begin{aligned}
{}^{I}\bar{S}_n &= \nabla^{-1}\ddot{\bar{x}}_n \\
{}^{II}\bar{S}_n &= \nabla^{-1}\,{}^{I}\bar{S}_n = \nabla^{-2}\ddot{\bar{x}}_n \ ,
\end{aligned}
$$

and are computed as follows. First, the sums ${}^{II}\bar{S}_{n-1}$ and ${}^{II}\bar{S}_{n-1}$ are computed by inverting the corrector formulas for $\bar{x}_n$ and $\dot{\bar{x}}_n$:

$$
\begin{aligned}
{}^{I}\bar{S}_{n-1} &= h\dot{\bar{x}}_n - \sum_{i=0}^{k} \beta_i^*\,\ddot{\bar{x}}_{n-i} \\
{}^{II}\bar{S}_{n-1} &= \bar{x}_n - \sum_{i=0}^{k} \alpha_i^*\,\ddot{\bar{x}}_{n-i} \ ,
\end{aligned}
$$

where α_i^*, β_i^* are the Cowell and Moulton coefficients respectively. The required sums are then computed by

$$
\begin{aligned}
{}^{I}\bar{S}_n &= {}^{I}\bar{S}_{n-1} + \ddot{\bar{x}}_n \\
{}^{II}\bar{S}_n &= {}^{II}\bar{S}_{n-1} + {}^{I}\bar{S}_n
\end{aligned}
$$

Calling Sequence:

CALL SUMS (IEQ, J3, J4)

COMMON Blocks Used:

ABCOEF
WORKER
LIMITS

Variables Not in COMMON:

FORTRAN Name	Format	Description
IEQ	I	Indicates which equation is being used
J3	I	Indicates first equation to be used
J4	I	Indicates last equation to be used

SUMTOB

Purpose/Method:

Moves the (K + I)-th parameter and its associated row of the normal equations:

(SUM1) × (PARBUF) = (SUM2), with (LBLBUF) to the (L + J)-th position of the normal equations in the B matrix form

(BMATRX) × (PARAM) = (BRHS), with (LABEL).

Called By:

WTBMAT

Calling Sequence:

```
CALL SUMTOB (N, K, I, L, J)
```

COMMON Blocks Used:

PRIORI
BEQ

Variables Not in COMMON:

FORTRAN Name	Dim.	Format	Description
N	1	I	Total number of parameters (=matrix size)
K	1	I	Number of parameter of all preceding parameter groups in the input block SUM2 $(0 \leq K \leq N)$
I	1	I	Position of the parameters of the current parameter group to be moved $(1 \leq K + I \leq N)$
L	1	I	Number of parameters in output parameter block PARAM $(0 \leq L \leq N)$
J	1	I	Position of the parameter in the current parameter group in B-matrix order $(1 \leq L + J \leq N)$

TABLE

Purpose:

To form the initial table of starting values for the equations of motion and the variational equations to be used in the multistep Cowell integrator.

Called By:

ORBIT

Calls:

RK

Method:

The computation of the starting values required by the multistep integrator is done by using a single step 8th order Runge-Kutta integration method. The algorithm used to form the starting tables is the following:

Case I: Equations of motion and variational equations being integrated with the same stepsize.

In this case, the total number of time points produced for the starting tables is

$$K = \max(N_1, N_2) ,$$

where

$$N_i = P_i - 2 ;$$

P_1 and P_2 the orders of the integrators to be used for the equations of motion and variational equations respectively. The required times $t_i = t_0 + ih$, $i = 1, 2, \cdots k$ are computed and used as input to the subroutine RK together with the initial values to compute the necessary starting table.

Case II: Equations of motion and variational equations being integrated with different stepsizes.

In this case, the total number of time points produced for the starting tables is

$$K = N_1 + N_2 ,$$

where N_i is defined as above. If h_1 and h_2 are the stepsizes to be used to integrate the equations of motion and the variational equations respectively, then the required times are given by

$$t_i = t_0 + ih_1 \qquad i = 1, \cdots N_1$$

$$t_j = t_0 + jh_2 \qquad j = 1, \cdots N_2$$

and are used as input to the subroutine RK together with the initial values to obtain arrays containing the required starting tables. The starting tables required at the two stepsizes are then obtained by sorting the values contained in the RK produced arrays.

Because of core storage limitations, the restriction

$$P_1 + P_2 \leq 22$$

is required whenever two different stepsizes are used.

Calling Sequence:

CALL TABLE

COMMON Blocks Used:

WORKER
RKT

Variables Not in COMMON:

FORTRAN Name	Format	Description
IND (30)	I	Array of indicators to designate which starting points belong to the equations of motion and which starting points belong to the variational equations
K	I	Total number of starting points needed
RT1	I	Time at which starting points are to be computed for the equations of motion
RT2	I	Time at which starting points are to be computed for the variational equations

IRT1	I	Number of starting points to be computed for the equations of motion
IRT2	I	Number of starting points to be computed for the variational equations
ITR	I	Indicates which equations are being re-organized

SWTEST

Purpose:

To set internal switches which:

- identify which perturbative effects are being applied to the satellite accelerations and which perturbation effects are being applied to the computation of the position partials;
- indicate what perturbations have to be recomputed if the variational equations and the equations of motion are being integrated with two different stepsizes;
- indicate which variational equations are being integrated.

Called By:

ORBIT

Method:

The routine defines an array of indicators which are used to define computed GO TO statements in the force computations of the acceleration partials. The values for the indicators are determined by:

(i) whether the equations of motion and variational equations are being integrated together or separately;

(ii) what forces are to be applied to the variational equations;

(iii) what forces are to be applied to the equations of motion.

Calling Sequence:

CALL SWTEST (SWITCH)

COMMON Blocks Used:

WORKER
ESTGP
LIMITS
SETSW

Variables Not in COMMON:

FORTRAN Name	Format	Description
SWITCH	L	A switch which if set to TRUE indicates that the equations of motion and the variational equations are being integrated with the same stepsize; if the switch is set to FALSE, it indicates that the equations of motion and the variational equations are being integrated with two different stepsizes.

VEVAL

Purpose:

To compute the partials of the satellite acceleration vectors with respect to the position and velocity and, if present, emissivity, drag, and harmonic coefficients.

Called By:

ORBIT
RK
CSTEP

Calls:

REFCOR
EGRAV
DENSTY
DOTPRD

Method:

The acceleration partials are computed from accelerations of the form

$$\ddot{\bar{x}} = \frac{\partial U}{\partial \bar{x}} + \bar{F}_{drag} + \bar{F}_{sr} + \bar{F}_{sun} + \bar{F}_{moon} ,$$

where

U = the earth's geopotential

$\bar{F}_{drag}$ = acceleration vector due to atmospheric drag

$\bar{F}_{sr}$ = acceleration vector due to solar radiation pressure

$\bar{F}_{sun}$ = acceleration vector due to solar gravity

$\bar{F}_{moon}$ = acceleration vector due to lunar gravity

In the current NONAME ODP version of this subroutine, the acceleration partials include effects of the earths potential to fourth order terms, drag and lunar gravity. The details of these computations can be found in Reference 1, Vol. 2. The GEOSTAR-1 modification of this subroutine can be summarized as follows:

(i) effects due to solar gravity and radiation pressure have been included in the computation of acceleration partials. These solar gravity effects are given by:

$$\frac{\partial \bar{F}_{sun}}{\partial \bar{x}} = \begin{bmatrix} \frac{\partial \ddot{x}_s}{\partial x} & \frac{\partial \ddot{x}_s}{\partial y} & \frac{\partial \ddot{x}_s}{\partial z} \\ \frac{\partial \ddot{y}_s}{\partial x} & \frac{\partial \ddot{y}_s}{\partial y} & \frac{\partial \ddot{y}_s}{\partial z} \\ \frac{\partial \ddot{z}_s}{\partial x} & \frac{\partial \ddot{z}_s}{\partial y} & \frac{\partial \ddot{z}_s}{\partial z} \end{bmatrix}$$

$$\frac{\partial \ddot{x}_s}{\partial x} = -\frac{GM_s}{|\bar{x}_s - \bar{x}|^3}\left[1 - 3\left(x_s - x\right)^2 \Big/ |\bar{x}_s - \bar{x}|^2\right]$$

$$\frac{\partial \ddot{y}_s}{\partial y} = -\frac{GM_s}{|\bar{x}_s - \bar{x}|^3}\left[1 - 3\left(y_s - y\right)^2 \Big/ |\bar{x}_s - \bar{x}|^2\right]$$

$$\frac{\partial \ddot{z}_s}{\partial z} = -\frac{GM_s}{|\bar{x}_s - \bar{x}|^3}\left[1 - 3\left(z_s - z\right)^2 \Big/ |\bar{x}_s - \bar{x}|^2\right]$$

$$\frac{\partial \ddot{x}_s}{\partial y} = \frac{\partial \ddot{y}_s}{\partial x} = -3\,GM_s\left[\left(x_s - x\right)\left(y_s - y\right)\right] \Big/ |\bar{x}_s - \bar{x}|^5$$

$$\frac{\partial \ddot{x}_s}{\partial z} = \frac{\partial \ddot{z}_s}{\partial x} = -3\,GM_s\left[\left(x_s - x\right)\left(z_s - z\right)\right] \Big/ |\bar{x}_s - \bar{x}|^5$$

$$\frac{\partial \ddot{y}_s}{\partial z} = \frac{\partial \ddot{z}_s}{\partial y} = -3\,GM_s\left[\left(y_s - y\right)\left(z_s - z\right)\right] \Big/ |\bar{x}_s - \bar{x}|^5 ,$$

where

GM_s = gravitational constant times mass of sun in earth masses,

$\bar{x}_s = \left(x_s, y_s, z_s\right)$ = solar position vector,

and the solar radiation effects are given by:

$$\frac{\partial \bar{F}_{sr}}{\partial \bar{x}} = \begin{bmatrix} \frac{\partial \ddot{x}_{sr}}{\partial x} & \frac{\partial \ddot{x}_{sr}}{\partial y} & \frac{\partial \ddot{x}_{sr}}{\partial z} \\ \frac{\partial \ddot{y}_{sr}}{\partial x} & \frac{\partial \ddot{y}_{sr}}{\partial y} & \frac{\partial \ddot{y}_{sr}}{\partial z} \\ \frac{\partial \ddot{z}_{sr}}{\partial x} & \frac{\partial \ddot{z}_{sr}}{\partial y} & \frac{\partial \ddot{z}_{sr}}{\partial z} \end{bmatrix}$$

$$\frac{\partial \ddot{x}_{sr}}{\partial x} = - C_R \frac{A}{M} \left[1 - \left(\frac{x_s - x}{|\bar{x}_s - \bar{x}|} \right)^2 \right] \quad |\bar{x}_s - \bar{x}|$$

$$\frac{\partial \ddot{y}_{sr}}{\partial y} = - C_R \frac{A}{M} \left[1 - \left(\frac{y_s - y}{|\bar{x}_s - \bar{x}|} \right)^2 \right] \quad |\bar{x}_s - \bar{x}|$$

$$\frac{\partial \ddot{z}_{sr}}{\partial z} = - C_R \frac{A}{M} \left[1 - \left(\frac{z_s - z}{|\bar{x}_s - \bar{x}|} \right)^2 \right] \quad |\bar{x}_s - \bar{x}|$$

$$\frac{\partial \ddot{x}_{sr}}{\partial y} = \frac{\partial \ddot{y}_{sr}}{\partial x} = - C_R \frac{A}{M} \left[(x_s - x) \; (y_s - y) \right] \Big/ |\bar{x}_s - \bar{x}|^3$$

$$\frac{\partial \ddot{x}_{sr}}{\partial z} = \frac{\partial \ddot{z}_{sr}}{\partial x} = - C_R \frac{A}{M} \left[(x_s - x) \; (z_s - z) \right] \Big/ |\bar{x}_s - \bar{x}|^3$$

$$\frac{\partial \ddot{y}_{sr}}{\partial z} = \frac{\partial \ddot{z}_{sr}}{\partial y} = - C_R \frac{A}{M} \left[(y_s - y) \; (z_s - z) \right] \Big/ |\bar{x}_s - \bar{x}|^3 ,$$

where

C_R = satellite reflectivity constant

A = satellite cross-sectional area

M = satellite mass

(ii) Computations have been rearranged so that any of the particular effects acting on the acceleration partials can be excluded on cption. Only an option to compute two-body partials has been included.

(iii) The subroutine computes partials of acceleration with respect to harmonic coefficients by calling the EGRAV subroutine. The formulation is documented in that subroutine.

(iv) Computations have been arranged so that if the variational equations are being integrated with a stepsize different from that used for the equations of motion, the intermediate quantities which are used in the computation of the acceleration partials (normally brought in through COMMON) are recomputed in VEVAL.

(v) To allow for possible variations in the low order harmonics currently used in the acceleration partials computation, these coefficients are now initialized by the STORGP subroutine and brought into VEVAL through COMMON (CSVEVL).

Calling Sequence:

CALL VEVAL

COMMON Blocks Used:

CONST1
CONST3
DRGBLK
CSVEVL
SETSW
VRBLOK
FLXBLK
ESTGP
GRBLOK
MOONGR
LIMITS
COFIT
XYZ
ANPART

Variables Not in COMMON:

FORTRAN Name	Format	Description
RASAT	D	Right ascension of the satellite
THETG	D	Right ascension of Greenwich at integration time
LAMBDA	D	Satellite longitude
ELEM(6)	D	Satellite position and velocity vectors referenced to the true equator and equinox of date
VARD(9)	D	An array containing lunar and solar position vectors and vector magnitudes
EQ	D	Equation of equinox
CONS	R	Constant used in the computation of the acceleration partials with respect to the drag parameter
DPXUVM	D	Dot product of satellite and lunar position vector
RRSUN	D	Square of lunar or solar position vector
DPXUV	D	Dot product of satellite and solar position vector

P(3), PMAG(3)	D	Vector and vector magnitude used in the sunlight determination computation
SUNMAT (3, 3)	D	Matrix of the satellite acceleration partials due to solar gravity with with respect to instantaneous position
SOLMAT (3, 3)	D	Matrix of the satellite accelera-partials due to solar radiation with respect to instantaneous position
SLUMAT (3, 3)	D	Matrix of the satellite acceleration partials due to lunar gravity with respect to instantaneous position

WTBMAT

Purpose:

To output on the specified device the normal equations, together with the latest values and the labels of the parameters, in the B matrix format.

Called By:

MAIN

Calls:

SUMTOB
SYMMET

Method:

The parameter sequence in the normal equations in GEOSTAR-I ODP is different from that required by the B matrix format. Figure 2, Section 3.3.1 shows this parameter order. WTBMAT takes each parameter type group and assigns labels and sorts the parameters into the B matrix format within that particular group.

The subprogram SUMTOB is used to shift rows of a particular parameter type group into the designated B matrix areas. After all parameter groups are sorted and labeled, the normal equations matrix is written by rows in the B matrix format.

The total variance, as required by the SOLVE program, is computed as:

$$V1 = (RMS)^2 * [NOB - (NPARAM - 1)] ,$$

where

RMS = total weighted standard deviation

NOB = total number of observations used

NPARAM = total number of parameters being estimated.

Calling Sequence:

```
CALL WTBMAT (RMSTOT, NOWTOB, NSTEST, NBIAS,
        NPARAM, BNAME, IDMAT, BMT)
```

COMMON Blocks Used:

CONST1	STANUM
ESTGP	PRIORI
CELEM	BEQ

Variables:

FORTRAN Name	Format	Description
NOWTOB	I	NOB, number of weighted observations used in computing RMSTOT
NSTEST	I	Number of stations whose locations are to be estimated
NBIAS	I	Number of instrumental bias parameters
NPARAM	I	Total number of parameters to be estimated
BNAME (3)	R	B matrix name (12 EBCDIC characters)
IDMAT	I	B matrix identification number ($\leq$99999)
BMT	I	Output data set reference number
V1	D	The total variance
RMSTOT	R	Total weighted standard deviation

4.3 Modifications to Existing SOLVE, EIGENVALUE Modules

The modifications made to the SOLVE subroutines are designed to:

- Call the new subroutines.
- Allow for 4-digit station code labeling.
- Change the punched card formats to conform to the GEOSTAR-I ODP input requirements.
- Accept station location parameters in rectangular coordinates and convert them to spherical coordinates for printing and punching.

A summary of these modifications for the GEOSTAR-I SOLVE follows:

- CALTYP—Modified to recognize 4-digit station numbers, and to adjust its label count accordingly.
- INVERT—Modified to call for the pseudoinverse solution (ANDREE) on option, with corresponding changes in matrix handling, etc.
- LBLSUP—Modified to recognize 4-digit station numbers and their special suppression features, and to zero out the proper labels accordingly.
- OPARC—Modified to punch out updated state parameters in a format suitable for input back into ODP for a GEOSTAR-I iteration.
- OPGRAV—Modified to punch out updated harmonics in a format suitable for input back into NONAME for a GEOSTAR-I iteration.
- OPSTAT—Modified to recognize 4-digit station parameters; accept B matrix input given in rectangular coordinates; output rectangular or spherical coordinates as required; punch out STAPOS cards in a format suitable for input back into the ODP for a GEOSTAR-I iteration.
- SUPRSS—Modified to recognize an input CB (combined) matrix.
- UPCOMB—Modified to recognize 4-digit station numbers.

In addition the SETEIG subroutine in the EIGENVALUE program has been modified to call the subroutine ELIM which eliminates the station position parameters from the input matrix.

4.4 New GEOSTAR-I SOLVE Modules and Significantly Modified LUNGFISH SOLVE Modules

The following GEOSTAR-I modules were written to allow for pseudoinversion, to satisfy the I/O interface requirements with the GEOSTAR-I ODP program and to increase the "combined" matrix handling capabilities of the original SOLVE program. The subroutine MAIN, the SOLVE control program, was modified considerably to achieve the above requirements and hence these modifications are detailed in the following pages, as well as the new modules.

The modifications to MAIN are now detailed.

The first section of MAIN reads data cards one through five. In this section, modifications have been made as follows:

(i) Three additional options, IOPT5, IOPT6, and IOPT7 have been included in the read-in of the option data card, data card #2. These options are: to allow the input of a combined matrix with associated backsubstitution and parameter set matrices (IOPT5); to compute the Penrose pseudoinverse instead of a normal inverse (IOPT6; the ANDREE algorithm is used); and to accept incoming B matrices as representing the same arc (although different data types), allowing for the combining of the entire matrices, including state parameters (IOPT7).

(ii) In accordance with IOPT5, input tapes 19 or 29 (containing input parameter sets and input backsubstitution matrices to go with an input combined matrix) are repositioned so that they may be used in subsequent parts of the program.

The second section of MAIN reads data cards six and seven, B matrix information, and then finds and transfers the B matrix from input tape 18 to work tape 28 (suppressing the B matrix as required).

The following changes have been made in this section:

(i) Provision has been made for reading in optional values for AED and FLAT (parameters used in the conversion of rectangular to geodetic coordinates). These values are read in on B matrix data card 5; if the appropriate fields are left blank, nominal values for AED and FLAT are used.

(ii) For the combine arcs option (IOPT7), provision was made for identifying such matrices by setting their ITYPE value in the matrix header to 1, thus identifying them as B matrices. The combined arcs matrix, when finally output, will be for all functional purposes equivalent to a B matrix.

(iii) Provision was made to recognize an input combined matrix (IOPT5).

(iv) Coding was added to write a negative dummy record at the end of unit 19 once all the parameter sets were read in. This is for use in OPSTAT, OPGRAV and OPARC.

The third section of MAIN deals with the elimination of arc parameters from the input matrices.

The following change has been made:

> If this is a combine arcs run (IOPT7), then switches are set so as not to eliminate any arc parameters. Thus the arc parameters will be combined along with the rest of the matrix.

The fourth section of MAIN deals with the combining of the various arc-eliminated B matrices.

The following changes have been made:

(i) Coding from ELIM has been placed in MAIN to write out a negative dummy record only once on unit 29, at the end of all the backsubstitution.

(ii) Coding has been added to complete implementation of the combined arcs option (IOPT7). Once the combining of matrices has been completed, then a parameter set is read in from tape 19 and added immediately after the combined matrix on unit 28. The resulting matrix on tape 28 is then a B matrix, suitable for reintroduction into SOLVE, input to MERGE, etc.

(iii) Depending on the exact value of IOPT7, the option has been introduced either to stop at this point and output the combined arcs matrix on tape 28, or to continue on to invert and print out final solutions.

The fifth section of MAIN inverts the matrix by calling INVERT and its associated subroutines.

The following change has been made in this part of MAIN:

> The call statement for INVERT now includes IOPT6 for use in INVERT (INVERT calls ANDREE for the pseudoinverse solution if IOPT6 is on).

The sixth section of MAIN deals with the output.

The following change has been made in this section:

A mechanism was introduced for handling standard and optional values of AE and FLAT.

The seventh and last major section of MAIN deals with backsubstitution and the printout of related output.

The following change has been made in this section:

If IOPT7 is non-zero, i.e. if this is a combined arcs run where no arcs were eliminated, all subsequent backsubstitution procedures are skipped.

The very last sequence edits whatever matrices have been designated. One change has been added:

A data card is read in; if it is not 77777 in value, then the program loops back to start, beginning with the first data card.

ANDREE

Purpose:

To compute the pseudoinverse of a given matrix.

Called By:

INVERT

Method:

The Andree algorithm is used to obtain the pseudoinverse from an input matrix. By pivoting about each largest diagonal element and discarding those diagonal elements which are unacceptably small (in the ill-conditioned case), an S-matrix is obtained so that $SBS^T = E$, where B is the input matrix and E is a diagonal matrix of unitary or zero elements. If no elements were discarded in the pivot search, then E is the identity matrix and B^{-1} is obtained by a normal inversion, $S^T S = B^{-1}$.

If some elements were discarded however, then the algorithm computes a U matrix from the S matrix. Non-diagonal elements are set to the negative of their corresponding values in the S matrix if the diagonal for that row was eliminated, i.e., is zero in the E matrix. Otherwise, non-diagonal elements are zeroed, and the diagonal elements of the U matrix are equal to the diagonal elements of the E matrix.

$U^T BU$ is formed and all rows and columns that are zero in the result are deleted. The remainder becomes the upper left portion of a matrix whose other portions are zero matrices: this is our new matrix $C^{\#}$, and $B^{\#}$, the pseudoinverse equals $UC^{\#} U^T$.

Calling Sequence:

CALL ANDREE (V, N, NR, EPS, B, U, R, ISV)

Variables:

FORTRAN Name	Format	Description
V(140, 140)	D	The matrix to be pseudoinverted
N	I	The dimension of V
EPS	R	Number of inaccurate digits, used in determining whether to zero a given pivot element

ISV	I	Work tape unit number (set at 25)
NR	I	The computed rank of the matrix
B(140)	D	Work array
U(140, 140)	D	Work array
R(140)	D	Work array

The pseudoinverse of the input matrix is placed in the input array V for output.

OUTRAD

Purpose:

Converts radius to angular or time equivalents.

Called By:

OBSTAT

Method:

Proper combination of integer and real number arithmetic for the separation of higher integer units from remainders.

Calling Sequence:

```
CALL OUTRAD (RAD, IH, IM, S, K)
```

Variables:

FORTRAN Name	Format	Description
RAD	D	Radian input
IH	I	Equivalent number in hours or degrees
IM	I	Minutes
S	D	Seconds
K	I	Input indicator specifying time or angular conversion

PHLINN

Purpose:

To convert tracking station geodetic rectangular coordinates to geodetic spherical coordinates.

Called By:

OPSTAT

Calls:

DARCTN

Method:

An iterative technique is used to compute geodetic latitude and elevation above spheroid from geocentric X, Y, Z of the input station. See Figure 3.

Initially set

$$t = e^2 Z .$$

Then for each iteration solve for t as follows:

$$Z_t = Z + t$$

$$N = \left(X^2 + Y^2 + Z_t^2\right)^{1/2}$$

$$\sin\phi = Z_t/N$$

$$\nu = a/\left(1 - e^2 \sin^2 \phi\right)$$

$$t = \nu e^2 \sin\phi .$$

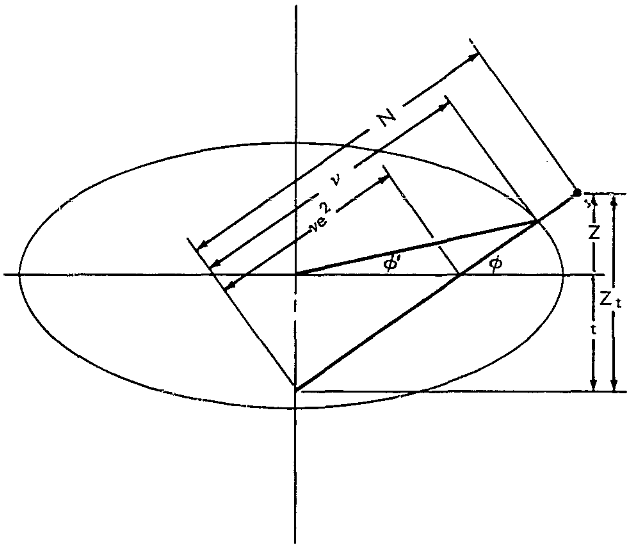


Figure 3—Diagram of geodetic latitude.

When t converges, compute

$$\phi = \sin^{-1}\phi \qquad \text{(latitude)}$$

$$H = N - \nu \qquad \text{(height)}$$

and

$$\lambda = \tan^{-1}(Y/X) \quad \text{(longitude)} .$$

Calling Sequence:

CALL PHLINN (M, X, Y, Z, AE, FLAT, PHI, XAMBDA, H)

Variables

FORTRAN Name	Format	Description
M	I	Station identification indicator
X Y Z	D	Rectangular coordinates of station position
AE	D	Semi-major axis of reference ellipsoid
FLAT	D	Flattening coefficient of reference ellipsoid
PHI	D	Geodetic latitude
XAMBDA	D	Geodetic east longitude
H	D	Height above spheriod, in meters

V COMMON BLOCK VARIABLE DESCRIPTION

The following sections contain the COMMON block descriptions of the COMMON areas used in the GEOSTAR-I ODP and SOLVE programs. These descriptions include the variables contained in these areas, their type and meaning, and the subroutines which either define or use these variables, for those which are not used in a major number of the routines. Also, following each section of COMMON block variable descriptions, a section containing a cross reference table of all the COMMON areas as used in each subroutine is presented.

5.1 NONAME—GEOSTAR-I ODP COMMON Blocks

This section contains a description of all the COMMON areas used in the GEOSTAR-I ODP program.

/ABCOEF/

COMMON /ABCOEF/ALPHA(102), ALPHAS(102), BETA(102), BETAS(102), IB(2)

Variable	Type	Description	Program Where Defined	Program Where Used
ALPHA(102)	D	Adams-Bashforth predictor coefficients for orders 4-15	BLOCK DATA	CSTEP SUMS
ALPHAS(102)	D	Adams-Moulton corrector coefficients for orders 4-15	BLOCK DATA	CSTEP SUMS
BETA(102)	D	Störmer predictor coefficients for orders 4-15	BLOCK DATA	CSTEP SUMS
BETAS(102)	D	Cowell corrector coefficients for orders 4-15	BLOCK DATA	CSTEP SUMS
IB(2)	I	IB(1)—starting point for the set of integration coefficients to be used for the equations of motion integration IB(2)—starting point for the set of integration coefficients to be used for the variational equations integration	ORBIT	CSTEP SUMS

/ANPART/

COMMON /ANPART/PRDD(3, 6)

Variable	Type	Description	Program Where Defined	Program Where Used
PRDD(3, 6)	D	Matrix of the satellite acceleration partials with respect to instantaneous position and velocity. The matrix is of the form: $\left[\begin{array}{ccc\|ccc} \frac{\partial \ddot{x}}{\partial x} & \frac{\partial \ddot{x}}{\partial y} & \frac{\partial \ddot{x}}{\partial z} & \frac{\partial \ddot{x}}{\partial \dot{x}} & \frac{\partial \ddot{x}}{\partial \dot{y}} & \frac{\partial \ddot{x}}{\partial \dot{z}} \\ \frac{\partial \ddot{y}}{\partial x} & \frac{\partial \ddot{y}}{\partial y} & \frac{\partial \ddot{y}}{\partial z} & \frac{\partial \ddot{y}}{\partial \dot{x}} & \frac{\partial \ddot{y}}{\partial \dot{y}} & \frac{\partial \ddot{y}}{\partial \dot{z}} \\ \frac{\partial \ddot{z}}{\partial x} & \frac{\partial \ddot{z}}{\partial y} & \frac{\partial \ddot{z}}{\partial z} & \frac{\partial \ddot{z}}{\partial \dot{x}} & \frac{\partial \ddot{z}}{\partial \dot{y}} & \frac{\partial \ddot{z}}{\partial \dot{z}} \end{array}\right]$ where those quantities within the dotted brackets are computed only if drag is applied.	VEVAL	CSTEP ORBIT RK TABLEB

/BEQ/

COMMON /BEQ/SUM2(50), PARBUF(50), LBLBUF(50), BMATRX(50, 50),
BRHS(50), PARAM(50), LABEL(50), IPARAM(50)

Variable	Type	Description	Program Where Defined	Program Where Used
SUM2(50)	D	Right hand side of normal equations in GEOSTAR-I parameter order	ESTIM	ESTIM WTBMAT SUMTOB SOLVGP
PARBUF(50)	D	Buffer for parameter values in GEOSTAR-I order, by row	WTBMAT	WTBMAT SUMTOB
LBLBUF(50)	I	Numerical identifiers for the type of parameters in PARBUF, in the same order	WTBMAT	WTBMAT SUMTOB
BMATRX(50, 50)	D	Symmetrical matrix of the normal equations in B matrix form, for multi-arc processing	WTBMAT	WTBMAT SUMTOB
BRHS(50)	D	The associated right hand side of BMATRX	WTBMAT	WTBMAT SUMTOB
PARAM(50)	D	Parameter values in B matrix order by row	WTBMAT	WTBMAT
LABEL(50)	I	Numerical identifiers for the type of parameters in PARAM, in the same order	WTBMAT	WTBMAT SUMTOB
IPARAM(50)	I	Array used by subroutine SUMTOB to order the parameter values	SUMTOB	SUMTOB

/CELEM/

COMMON /CELEM/ELEMST(6), ORBELA(6), XNV, EC

Variable	Type	Description
		Orbital Elements - Inertial Position and Velocity Vectors
ELEMST(1)	D	X - meters
ELEMST(2)		Y - meters
ELEMST(3)		Z— meters
ELEMST(4)		$\dot{X}$ - meters/second
ELEMST(5)		$\dot{Y}$ - meters/second
ELEMST(6)		$\dot{Z}$ - meters/second
		Orbital Elements - Osculating Keplerian
ORBELA(1)	D	Semi-major axis - meters
ORBELA(2)		Eccentricity
ORBELA(3)		Inclination - radians
ORBELA(4)		Longitude of ascending node - radians
ORBELA(5)		Argument of pericenter - radians
ORBELA(6)		Mean anomaly - radians
XNU	D	True anomaly - radians
EC	D	Eccentric anomaly - radians

Program Where Used			Purpose
ELEM	Input	-	values of ELEMST(6) (inertial position and velocity vectors) from MAIN program
	Output	-	return values of ORBELA(6) (osculating orbital elements), XNV and EC to MAIN program
POSVEL	Input	-	values of ORBELA(6) (osculating orbital elements) from MAIN program
	Output	-	values of ELEMST(6) (inertial position and velocity vectors) to MAIN program
ESTIM	Input	-	values of ELEMST(6) (inertial position and velocity vectors).

/CGEOS/

COMMON /CGEOS/DAYREF, DAYSTP, DAYSTA, ISATID, SIG1, SIG2,
MTYPE, NMEAS, ISTA

NOTE

Subprograms GEOSRD or DODSRD are utilized by the system to read records of observational data, select and delete records on request, and preprocess optical and time observational data as requested by various indicator variables. Part of the final observational data is returned to MAIN through COMMON/CGEOS/. The remainder of the observational data is returned to MAIN through COMMON/PREBLK/. Each record of observational data may contain one or two observations.

Variable	Type	Description	Program Where Defined	Program Where Used
DAYREF	D	Reference data in days since Jan. 0.0 in A1 time	MAIN	
DAYSTP	D	Stop date for the DC arc in days since Jan. 0.0 in A1 time	MAIN	
DAYSTA	D	Epoch of observations in days elapsed from Jan. 0.0 UTC of the year of the epoch of the current run.	GEOSRD DODSRD Converted to A1 time in MAIN	MAIN
ISATID	D	Satellite identification code	GEOSRD DODSRD	MAIN
SIG1	R	Standard deviation of 1st observation in this record	GEOSRD DODSRD	MAIN
SIG2	R	Standard deviation of 2nd observation if the record contains two observations	GESORD DODSRD	MAIN

/CGEOS/ (continued)

NOTE

Units of SIG1 and SIG2 are:

right ascension:	seconds of arc
declination:	seconds of arc
azimuth:	seconds of arc
elevation:	seconds of arc
X angle:	degrees
Y angle:	degrees
range:	meters
range rate:	meters/sec
direction cosines:	dimensionless

Variable	Type	Description	Program Where Defined	Program Where Used
MTYPE	I	Observation Type in this record = 1 - α and δ (right ascension and declination) = 2 - r (range) = 3 - $\dot{r}$ (range rate) = 4 - Doppler (converted to range rate) = 5 - ℓ, m (Minitrack direction cosines) = 6 - X, Y angles = 7 - azimuth and elevation	GEOSRD DODSRD	MAIN PREDCT
NMEAS	I	Number of observations in this record NMEAS = 1 for MTYPE = 2, 3, 4 = 2 for MTYPE = 1, 5, 6, 7		MAIN
ISN	I	GSFC code number for station recording observations in this record		MAIN

/COFIT/

COMMON /COFIT/C(5, 9), DAY1, CENTER, II4

Variable	Type	Definition	Program Where Defined	Program Where Used
C(5, 9)	D	Coefficients of a 4th degree polynomial fit to nine ephemeris quantities. These coefficients are estimated by using 11 values of each quantity calculated by subprograms MOONAD and SUN equally spaced over a 2.5 day interval starting at DAY1.	EPHQAN	FRCS VEVAL
		Specifically C represents coefficients for:		

C(1, 1)-C(5, 1)
C(1, 2)-C(5, 2)
C(1, 3)-C(5, 3)
	D	X_m	EPHQAN	FRCS
				VEVAL

C(1, 4)-C(5, 4)	D	R_m	EPHQAN	FRCS VEVAL

C(1, 5)-C(5, 5)
C(1, 6)-C(5, 6)
C(1, 7)-C(5, 7)
	D	X_s	EPHQAN	FRCS
				VEVAL

C(1, 8)-C(5, 8)	D	R_s	EPHQAN	FRCS VEVAL
C(1, 9)-C(5, 9)	D	E_q	EPHQAN	FRCS VEVAL PREDCT

/COFIT/ (continued)

Variable	Type	Description	Program Where Defined	Program Where Used
DAY1	D	Days elapsed since January 0.0 of the epoch year of the initial elements	ORBIT	FRCS VEVAL
CENTER	D	= DAY1 + 1.25 days	EPHQAN	FRCS VEVAL PREDCT
II4	I	Order of ephemeris polynomials	EPHQAN	FRCS VEVAL

where

X_m = Unit vector to moon referred to true equator and equinox of date

R_m = Distance to moon - meters

X_s = Unit vector to sun referred to true equator and equinox of date

R_s = Distance to sun - meters

E_q = Equation of the equinoxes - radians

/CONST1/

COMMON /CONST1/THETG0(10), THDOT1, GM, AE, FINV, CD, ASAT, MSAT, ADDR, SRAD, EMISS, MSUN, MMOON, RPRESS

Variable	Type	Description
THETG0(10)	D	Greenwich Mean Sidereal Time in radians at 0 hours UT1 on January 0 for the years 1960-1969
THDOT1	D	Relative increase of angle between mean Greenwich meridian and the mean equinox of date - radians per 24 hours UT1 time
GM	D	GM_E - meters ^{3}/seconds ^{2}
AE	D	R_E - meters
FINV	R	1/f
CD	R	C_D
ASAT	R	A_s - meters^2
MSAT	R	M_s - kilograms
ADDR	I	C_D estimation indicator. If ADDR >0, C_D will be estimated.
SRAD	I	C_R estimation indicator. If SRAD >0, C_R will be estimated.
EMISS	R	C_R
MSUN	R	$M_\odot / M_E$
MMOON	R	M_M / M_E
RPRESS	R	$P_\odot$

where

G = universal gravitational constant
M_E = mass of sun
M_M = mass of moon
M_s = mass of satellite
A_s = area of satellite
f = flattening of earth's reference ellipsoid
R_E = semi-major axis of earth's reference ellipsoid
C_D = satellite drag coefficient
C_R = satellite reflectivity coefficient
$P_\odot$ = solar radiation pressure in the vicinity of the earth.

/CONST1/ (continued)

		Value	For the year
THETG0(1)	=	1.722 186 300 radians	1960
(2)	=	1.735 222 656 radians	1961
(3)	=	1.731 056 239 radians	1962
(4)	=	1.726 889 824 radians	1963
(5)	=	1.722 723 442 radians	1964
(6)	=	1.735 759 816 radians	1965
(7)	=	1.731 593 399 radians	1966
(8)	=	1.727 427 000 radians	1967
(9)	=	1.723 260 602 radians	1968
(10)	=	1.736 296 992 radians	1969
THDOT1	=	.017 202 791 266 radians per 24 hours UT1 time	
RPRESS	=	$.45 \times 10^{-5}$ newtons/meter^2	

The following nominal values may be changed by optional input cards:

$GM_E = 3.986\ 032 \times 10^{14}$ meters^3/seconds^2

$R_E = 6378165.$ meters

$1/f = 298.252$

$C_D = 0$

$A_s = 0$

$M_s = 0$

$C_R = 0$

$M_\odot/M_E = 332951.3$

$M_M/M_E = .0122999$

ADDR and SRAD are used to request adjustments to C_D and C_R respectively, and are initially defined to be 0. If it is desired to adjust C_D, option card DRAG is read to redefine ADDR to be +1. If it is desired to adjust C_R, option card SOLARD is read to redefine SRAD to be +1.

The CONST1 variables are defined in the BLOCK DATA subprogram.

/CONST2/

COMMON /CONST2/DPI, DTWOPI, DRAD, DRSEC, PI, TWOPI, RAD, RSEC

Data base containing constants necessary for angular calculations in radians

Variable	Type	Description
DPI	D	π in radians, 3.1415926535897932
DTWOPI	D	2π in radians, 6.2831853071795864
DRAD	D	Conversion from degrees to radians, .017453292519943296
DRSEC	D	Conversion from seconds of arc to radians, $.484813681109536 \times 10^{-5}$
PI	R	π in radians, 3.141593
TWOPI	R	2π in radians, 6.283185
RAD	R	Conversion from degrees to radians, .01745329
RSEC	R	Conversion from seconds of arc to radians, $.4848137 \times 10^{-5}$

The CONST2 variables are defined in the BLOCK DATA subprogram.

/CONST3/

COMMON /CONST3/THDOT2, THDT2S, AESQ, GM3(2), IY1, B, B0, APGM,
APLM, FSQ32, FFSQ32

Variable	Type	Description	Program Where Defined
THDOT2	D	Total rotation of the mean Greenwich meridian with respect to the mean equinox of date - radians per 24 hours of UT1 time	BLOCK DATA
THDT2S	D	Total rotation of the mean Greenwich meridian with respect to the mean equinox of date - radians per second of UT1 time (same as THDOT2 except for units)	BLOCK DATA
AESQ	D	R_E^2 - meters^2	ORBIT
GM3 (1)	R	$3GM_M$ - meters^3/seconds^2	ORBIT
GM3 (2)	R	$3GM_\odot$ - meters^3/seconds^2	ORBIT
IY1	I	Years elapsed from 1959 to the year of the epoch of the initial elements	ORBIT
B	R	$\frac{1}{2}\frac{A_s}{M_s} C_D$ (= 0 if $M_s \leq 0$)	ORBIT
B0	R	$\frac{1}{2}\frac{A_s}{M_s}$ (= 0 if $M_s \leq 0$)	ORBIT
APGM	R	$\frac{A_s}{M_s} P_\odot\ C_R$ (= 0 if $M_s \leq 0$)	ORBIT
APLM	R	$\frac{A_s}{M_s} P_\odot$ (= 0 if $M_s \leq 0$)	ORBIT
FSQ32	R	$\frac{3}{2} R_E f^2$	APPER ORBIT
FFSQ32	R	$R_E f \left(1 + \frac{3}{2} f\right)$	APPER ORBIT

/CONST3/ (continued)

where

G = universal gravitational constant

M_M = mass of moon in earth masses

$M_{\odot}$ = mass of sun in earth masses

M_s = mass of satellite - kilograms

R_E = semi-major axis of earth's reference ellipsoid

f = flattening of earth's reference ellipsoid

A_s = satellite cross sectional area - meters^2

C_D = satellite drag coefficient

C_R = satellite reflectivity coefficient

$P_{\odot}$ = solar radiation pressure in the vicinity of the earth - newtons/meter^2

The following variables are defined in BLOCK DATA:

THDOT2 = 6.300 388 098 445 593 radians per 24 hours of UT1 time
THDT2S = .729 211 585 468 2 × 10^{-4} radians per second of UT1 time.

The remaining variables are calculated in subprogram ORBIT.

/CONVRG/

COMMON /CONVRG/TOREFT

			Program	Program
			Where	Where
Variable	Type	Description	Defined	Used
TOREFT	L	Initialized to FALSE in BLOCK DATA. The OUTPUT option card is used to set this switch to TRUE which indicates an ORB1 tape has been requested and also to reference all quantities to the reference date.	BLOCK DATA OPTCRD	MAIN ORBIT

/CORB1/

COMMON /CORB1/RANDOT, PERDOT, PERHT, APHT, PRD

Variable	Type	Description	Program Where Defined	Program Where Used
RANDOT	D	Rate of change of the right ascension of ascending node (deg/day)	MAIN	MAIN
PERDOT	D	Rate of change of the argument of perigee (deg/day)	MAIN	MAIN
PERHT	D	Perigee height, considering oblateness (km)	APPER	MAIN
APHT	D	Apogee height, considering oblateness (km)	APPER	MAIN
PRD	D	Period (seconds)	MAIN	MAIN

/COSAVE/

COMMON /COSAVE/SA(5, 9), S2, S3, SCENTE

Variable	Type	Description	Program Where Defined	Program Where Used
SA(5, 9)	D	The saved coefficients of the polynomials used to determine the ephemeris quantities	EPHQAN	FRCS
S2	D	The overlap span from one arc of ephemeris quantities to the next	EPHQAN	EPHQAN
S3	D	The length of arc used to fit the ephemeris quantities. The nominal value set in the program is 2.5 days	BLOCK DATA	EPHQAN
SCENTE	D	The saved reference time point for the ephemeris polynomials	EPHQAN	FRCS

/CPARTL/

COMMON /CPARTL/PXPX0(50, 6), PMPX0(50, 2), PMPSTA(3, 2)

Variable	Type	Description	Program Where Defined	Program Where Used
PXPX0(50, 6)	D	Array containing partials of position and velocity with respect to the parameters to be estimated. The matrix is of the form:	ORBIT	PREDCT MAIN

$$\begin{bmatrix}
\frac{\partial x}{\partial x_0} & \frac{\partial y}{\partial x_0} & \frac{\partial z}{\partial x_0} & \frac{\partial \dot{x}}{\partial x_0} & \frac{\partial \dot{y}}{\partial x_0} & \frac{\partial \dot{z}}{\partial x_0} \\
\frac{\partial x}{\partial y_0} & \frac{\partial y}{\partial y_0} & \frac{\partial z}{\partial y_0} & \frac{\partial \dot{x}}{\partial y_0} & \frac{\partial \dot{y}}{\partial y_0} & \frac{\partial \dot{z}}{\partial y_0} \\
\frac{\partial x}{\partial z_0} & \frac{\partial y}{\partial z_0} & \frac{\partial z}{\partial z_0} & \frac{\partial \dot{x}}{\partial z_0} & \frac{\partial \dot{y}}{\partial z_0} & \frac{\partial \dot{z}}{\partial z_0} \\
\frac{\partial x}{\partial \dot{x}_0} & \frac{\partial y}{\partial \dot{x}_0} & \frac{\partial z}{\partial \dot{x}_0} & \frac{\partial \dot{x}}{\partial \dot{x}_0} & \frac{\partial \dot{y}}{\partial \dot{x}_0} & \frac{\partial \dot{z}}{\partial \dot{x}_0} \\
\frac{\partial x}{\partial \dot{y}_0} & \frac{\partial y}{\partial \dot{y}_0} & \frac{\partial z}{\partial \dot{y}_0} & \frac{\partial \dot{x}}{\partial \dot{y}_0} & \frac{\partial \dot{y}}{\partial \dot{y}_0} & \frac{\partial \dot{z}}{\partial \dot{y}_0} \\
\frac{\partial x}{\partial \dot{z}_0} & \frac{\partial y}{\partial \dot{z}_0} & \frac{\partial z}{\partial \dot{z}_0} & \frac{\partial \dot{x}}{\partial \dot{z}_0} & \frac{\partial \dot{y}}{\partial \dot{z}_0} & \frac{\partial \dot{z}}{\partial \dot{z}_0} \\
\frac{\partial x}{\partial C_D} & \frac{\partial y}{\partial C_D} & \frac{\partial z}{\partial C_D} & \frac{\partial \dot{x}}{\partial C_D} & \frac{\partial \dot{y}}{\partial C_D} & \frac{\partial \dot{z}}{\partial C_D} \\
\frac{\partial x}{\partial'} & \frac{\partial y}{\partial C_R} & \frac{\partial z}{\partial C_R} & \frac{\partial \dot{x}}{\partial C_R} & \frac{\partial \dot{y}}{\partial C_R} & \frac{\partial \dot{z}}{\partial C_R} \\
\frac{\partial x}{\partial C_{nm}} & \frac{\partial y}{\partial C_{nm}} & \frac{\partial z}{\partial C_{nm}} & \frac{\partial \dot{x}}{\partial C_{nm}} & \frac{\partial \dot{y}}{\partial C_{nm}} & \frac{\partial \dot{z}}{\partial C_{nm}} \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots
\end{bmatrix}$$

/CPARTL/ (continued)

Variable	Type	Description	Program Where Defined	Program Where Used
PMPX0(50, 2)	D	The partials of the measurements at a given time with respect to the parameters being estimated. The matrix is of the form:	PREDCT (calculated only when M_1 and M_2 have non-zero sigmas)	MAIN (as input to subprogram ESTIM)

$$\begin{bmatrix} \frac{\partial M_1}{\partial x_0} & \frac{\partial M_2}{\partial x_0} \\ \frac{\partial M_1}{\partial y_0} & \frac{\partial M_2}{\partial y_0} \\ \frac{\partial M_1}{\partial z_0} & \frac{\partial M_2}{\partial z_0} \\ \frac{\partial M_1}{\partial \dot{x}_0} & \frac{\partial M_2}{\partial \dot{x}_0} \\ \frac{\partial M_1}{\partial \dot{y}_0} & \frac{\partial M_2}{\partial \dot{y}_0} \\ \frac{\partial M_1}{\partial \dot{z}_0} & \frac{\partial M_2}{\partial \dot{z}_0} \\ \frac{\partial M_1}{\partial C_D} & \frac{\partial M_2}{\partial C_D} \\ \frac{\partial M_1}{\partial C_R} & \frac{\partial M_2}{\partial C_R} \\ \frac{\partial M_1}{\partial C_{nm}} & \frac{\partial M_2}{\partial C_{nm}} \\ \vdots & \vdots \end{bmatrix}$$

NOTE: The PXPX0 and PMPX0 arrays are packed so that if a particular parameter type is not to be estimated, the lower parameter partials move up.

/CPARTL/ (continued)

Variable	Type	Description	Program Where Defined	Program Where Used
PMPSTA(3, 2)	D	The partials of the measurements at a given time with respect to tracking station coordinates. The matrix is of the form: $\begin{bmatrix} \frac{\partial M_1}{\partial u_1} & \frac{\partial M_2}{\partial u_1} \\ \frac{\partial M_1}{\partial u_2} & \frac{\partial M_2}{\partial u_2} \\ \frac{\partial M_1}{\partial u_3} & \frac{\partial M_2}{\partial u_3} \end{bmatrix}$	PREDCT (calculated only for observations with non-zero sigmas made by stations whose coordinates are to be adjusted)	MAIN (as input to subprogram ESTIM)

where

$$\left.\begin{matrix} \begin{pmatrix} x_0 \\ y_0 \\ z_0 \end{pmatrix} \\ \begin{pmatrix} \dot{z}_0 \\ \dot{y}_0 \\ \dot{z}_0 \end{pmatrix} \end{matrix}\right\} =$$

geocentric inertial rectangular coordinates of the position and velocity vectors of the satellite at the initial epoch

/CPARTL/ (continued)

C_D = satellite aerodynamic drag coefficient

C_R = satellite reflectivity coefficient

$$\left.\begin{matrix}M_1\\ M_2\end{matrix}\right\} = \begin{array}{l}\text{observation or observations of the satellite which}\\ \text{is one of the following types or pair of types:}\end{array}$$

$$\left\{\begin{matrix}M_1\\ M_2\end{matrix}\right\} = \begin{array}{l}\text{right ascension}\\ \text{declination}\end{array}$$

$$\left\{\begin{matrix}M_1\\ M_1\\ M_1\end{matrix}\right\} = \begin{array}{l}\text{range}\\ \text{range rate}\\ \text{range rate (derived from Doppler)}\end{array}$$

$$\left\{\begin{matrix}M_1\\ M_2\end{matrix}\right\} = \begin{array}{l}\text{minitrack direction cosine}\\ \text{minitrack m direction cosine}\end{array}$$

$$\left\{\begin{matrix}M_1\\ M_2\end{matrix}\right\} = \begin{array}{l}\text{X angle (30 foot antenna)}\\ \text{Y angle (30 foot antenna)}\end{array}$$

$$\left\{\begin{matrix}M_1\\ M_2\end{matrix}\right\} = \begin{array}{l}\text{azimuth angle}\\ \text{elevation angle}\end{array}$$

$\begin{pmatrix}u_1\\ u_2\\ u_3\end{pmatrix}$ = geocentric earth-fixed rectangular coordinates of the tracking station (computed from its geodetic spherical coordinates); u_3 axis is directed toward earth's positive direction of axis of rotation; u_1 axis is directed toward intersection of Greenwich meridian and earth's equator.

/CQUANT/

COMMON /CQUANT/STAX(50), STAY(50), STAZ(50), NHAT(3, 50),
ZHAT(3, 50), EHAT(3, 50), THPRIM(2, 50)

Variable	Type	Description	Program Where Defined	Program Where Used

STAX(N)
STAT(N)
STAZ(N)
	D	$\begin{pmatrix}u_1\\u_2\\u_3\end{pmatrix}_N$	SQUANT (Calculated on 1st call only)	MAIN
				ESTIM
				OBSDOT
				PLHOUT
				PREDCT

NHAT(N)
ZHAT(N)
EHAT(N)
	D	$\begin{pmatrix}N\\Z\\E\end{pmatrix}_N$	SQUANT (Calculated on 1st call only)	OBSDOT
				PREDCT

THPRIM(1, N)	D	$\sin(\phi_N - \phi_N')$	SQUANT	Not currently used by GEOSTAR-I
THPRIM(2, N)	D	$\cos(\phi_N - \phi_N')$	SQUANT	Not currently used by GEOSTAR-I

where:

ϕ_N = geodetic latitude of Nth station

ϕ_N' = geocentric latitude of Nth station

$\begin{pmatrix}u_1\\u_2\\u_3\end{pmatrix}_N$ = geocentric earth fixed rectangular coordinates of the Nth tracking station (computed from its geodetic spherical coordinates); u_3 axis is directed towards earth's positive direction of axis of rotation; u_1 axis is directed towards intersection of Greenwich meridian and earth's equator

/CQUANT/ (continued)

$\begin{pmatrix} N \\ Z \\ E \end{pmatrix}_N =$ unit vectors centered at the Nth tracking stations directed North, Vertical and East respectively. The components of these vectors are referred to the topocentric equatorial, earth fixed coordinate system (principal axis parallel to the principal axis of the geocentric system).

/CSTHET/

COMMON /CSTHET/CTHETG, STHETG

Variable	Type	Description	Program Where Defined	Program Where Used
CTHETG	D	Cosine of the apparent right ascension of the mean Greenwich meridian	PREDCT	XEFIX
STHETG	D	Sine of the apparent right ascension of the mean Greenwich meridian	PREDCT	XEFIX

/CSTINF/

```
COMMON /CSTINF/MEASNO(4), NOBS(4), RDMEAN(4), RMS0(4), RND(4),
MEASWT(4), WTMEAN(4), RMSWT0(4), WTRND(4), TYPRMS(14), NOTYPE(14)
```

NOTE

For each data reduction and parameter estimation run made, the following restrictions apply:

50 = maximum number of tracking stations allowed

4 = maximum number of different observation types allowed per station.

At the end of each iteration, subprogram STAINF is called to provide a statistical summary for that particular iteration on a station by station basis. At the end of the iteration, STAINF calculates the required statistical information for each of the observation types (maximum 4) for each station and returns this information to MAIN via COMMON/CSTINF/. MAIN then prints this statistical summary for each station. The dimension of 4 shown in the above COMMON/ CSTINF/ block for each variable represents the maximum number of observation types allowed per station. At any given time, the information in this block pertains to only one station. The statistical information corresponding to the same index in the various arrays pertains to the specific observation type implied by that particular index number.

/CSTINF/ (continued)

Variable	Type	Description	Program Where Defined	Program Where Used
		NOTE Let N (= 1, 2, 3, or 4) be the index in the array indicating a specific observation under consideration as described in the following table:		
MEASNO(N)	I	Code number (MTYPE) indicating type of observation for which statistical information was computed.	STAINF	MAIN
		MTYPE = 1 for right ascension = 2 for range = 3 for range rate = 4 for range rate (derived from Doppler) = 5 for direction cosine = 6 for X angle = 7 for azimuth angle = 8 for declination = 12 for m direction cosine = 13 for Y angle = 14 for elevation angle = 16 for geocentric coordinate x = 17 for geocentric coordinate y = 18 for geocentric coordinate z = 19 for geocentric coordinate $\dot{x}$ = 20 for geocentric coordinate $\dot{y}$ = 21 for geocentric coordinate $\dot{z}$		
NOBS(N)	I	Total number of MTYPE observations for this station	STAINF	MAIN
RDMEAN(N)	R	Mean of the residuals of all MTYPE observations for this station	STAINF	MAIN

/CSTINF/ (continued)

Variable	Type	Description	Program Where Defined	Program Where Used
RMSO(N)	R	RMS of the residuals about zero for all MTYPE observations for this station (calculated only if there are 10 or more MTYPE observations for this station)	STAINF	MAIN
RND(N)	R	Random normal deviate of residuals for all MTYPE observations for this station (calculated only if there are 10 or more MTYPE observations for this station)	STAINF	MAIN
MEASWT(N)	I	Number of weighted MTYPE observations for this station	STAINF	MAIN
WTMEAN(N)	R	Weighted mean of the residuals for all weighted MTYPE observations for this station	STAINF	
RMSWTO(N)	R	Weighted RMS of the residuals about 0 for the weighted MTYPE observations for this station (calculated only if there are 10 or more observations for this station)	STAINF	MAIN
WTRND(N)	R	Weighted random normal deviate of the residuals of the MTYPE observations for this station (calculated only if there are 10 or more observations for this station)	STAINF	MAIN
TYPRMS	R	Total RMS for each observation type in the solution	STAINF	MAIN
NOTYPE	I	Number of observations per observation type if zero, that data type is not in the solution	STAINF	MAIN

/CSVEVL/

COMMON /CSVEVL/CNM(3, 3), SNM(3, 3)

These arrays are used only by subroutine VEVAL. They are initialized by STORGP from the data base array of geopotential coefficients /FMODEL/CS.

Variable	Type	Description

CNM(3, 3)
SNM(3, 3)
	D
	D
		Contain constants for the geopotential coefficients C or S; zonals through 4th order and tesserals through 3rd order are included.

Variable Name with FORTRAN Subscript	Values Using Geophysical Notation $C_{n,m}$
CNM(1, 1)	C(2, 0)
CNM(2, 1)	C(2, 1)
CNM(3, 1)	C(3, 1)
CNM(1, 2)	C(3, 0)
CNM(2, 2)	C(2, 2)
CNM(3, 2)	C(3, 2)
CNM(1, 3)	C(4, 0)
CNM(2, 3)	–
CNM(3, 3)	C(3, 3)

Same as above for SNM

/CVRCOV/

COMMON /CVRCOV/VRCOV(50, 50), JHIGH, VRCSW

Variable	Type	Description	Program Where Defined	Program Where Used
VRCOV	R	Initial variance-covariance matrix, the inverse of which is the parameter weight matrix used in ESTIM. BLOCK DATA defines the six diagonal elements to be used for the state vector. The entire matrix may be filled by using the VARCOV option card.	BLOCK DATA or input on cards in OPTCRD	ESTIM MAIN
JHIGH	I	Initialized as 6. If VARCOV option cards are read in, JHIGH indicates the dimension of the portion of VRCOV that is filled.	BLOCK DATA OPTCRD	ESTIM MAIN
VRCSW	L	Initialized to FALSE; if VARCOV option cards are read in, VRCSW is automatically set to TRUE indicating that the initial variance-covariance matrix has been redefined.	BLOCK DATA OPTCRD	ESTIM

/DC/

COMMON /DC/H(6, 6), DETER, MM1

Variable	Type	Description	Program Where Defined	Program Where Used
H(6, 6)	D	Matrix containing the Jacobian of accelerations with respect to position and velocity	CSTEP	INV2
DETER	D	Determinant of matrix inverted by INV2	INV2	
MM1	I	Dimension of matrix to be inverted	CSTEP	INV2

/DRGBLK/

COMMON /DRGBLK/C(4), SPSISQ, C3, C1, VEL, XDOTR, YDOTR, RHO, HT

Variable	Type	Description	Program Where Defined	Program Where Used

C(1)
C(2)
C(3)
C(4)
	R	Used for transfer of intermediate calculations	DENSTY	VEVAL

SPSISQ	R	$\frac{z^2}{r^2} = \sin^2 \phi$	DRAG VEVAL	VEVAL
C3 BBRHO	R	$\frac{\rho}{v_r}$	DRAG VEVAL	VEVAL
C1 BBRHOV	R	$\frac{1}{2} \frac{\rho}{v_r} \frac{A_s}{M_s} C_D = a_D$	DRAG VEVAL	VEVAL
VEL	R	$\dot{v}_r$ - meters/second	DRAG VEVAL	VEVAL
XDOTR VELR(1)	R	$\dot{x}_r$ - meters/second	DRAG VEVAL	VEVAL
YDOTR VELR(2)	R	$\dot{y}_r$ - meters/second	DRAG VEVAL	VEVAL
RHO VELR(3)	R	ρ - kilograms/meter^3	DRAG VEVAL	VEVAL
HT	R	h - meters	DRAG VEVAL	DENSTY VEVAL

where

z = inertial z component of satellite position vector

r = geocentric distance of satellite

ρ = atmospheric density at the height of the satellite

/DRGBLK/ (continued)

A_s = satellite cross-sectional area-meters^2

M_s = satellite mass - kilograms

C_D = satellite drag coefficient

v_r = magnitude of relative velocity

h = height of satellite

$\dot{x}_r$ = inertial x component of relative velocity vector of satellite

$\dot{y}_r$ = inertial y component of relative velocity vector of satellite

ϕ = geocentric latitude of satellite

a_D = magnitude of acceleration due to aerodynamic drag

The variables C(1), C(2), C(3), and C(4) are coefficients of the equation

$$\log_{10} \rho \quad = \quad C(1) + C(2)h + C(3)h^2 + C(4)h^3 \ .$$

Subroutine VEVAL requires the quantity $\frac{\partial}{\partial h} \log_{10} \rho$, so that only the coefficients C(2), C(3) and C(4) are used by that program. The corresponding variables in VEVAL are:

DENSTY	VEVAL
C(1)	C0
C(2)	EXPT(1)
C(3)	EXPT(2)
C(4)	EXPT(3)

XDOTR = VELR(1)
YDOTR = VELR(2)
RHO = VELR(3).

After calculation using these variables, VEVAL then uses C0 and EXPT as scratch storage. The equivalence of the same variables in DRAG and VEVAL with different names are shown in the above list in brackets. These are

C3 = BBRHO   C1 = BBRHOV

/ESTGP/

COMMON /ESTGP/CSA(50), CSE(50), CSSIG(50), PERTRB, LABELG, NCSN(50),
NCSM(50), NC, NS, NPRTL, ITERGP, ISTATE

/ESTGP/ defines variables used when estimating the earth's geopotential coefficients or gravity parameters (GP).

Variable	Type	Description	Program Where Defined	Program Where Used
CSA(50)	R	Array of a priori values of geopotential coefficients for estimation	At first iteration CSE = CSA.	ESTIM
CSE(50)	R	Array of estimated values of geopotential coefficients for estimation	READGP from Data Base, /FMODEL/ CS(30, 33) or input cards.	ESTIM MAIN WTBMAT
CSSIG(50)	R	Array of a priori standard deviations of the geopotential coefficients for estimation	READGP	ESTIM
PERTRB	R	Not used		
LABELG	I	Position in the GRPAR array corresponding to the gravity coefficients. LABELG is initialized to zero in BLOCK DATA.	BLOCK DATA	OPTCRD EGRAV

/ESTGP/ (continued)

Variable	Type	Description	Program Where Defined	Program Where Used

NCSN(50)
NCSM(50)
	I	Arrays of n and m subscripts designating which C_{nm} or S_{nm} are represented in CSA and CSE. NCSN contains the n subscript; NCSM, the m subscript. Subscripts for C_{nm} are first in the arrays, with S_{nm} following. The first NC places are for C_{nm} and the following NS places are for S_{nm}.	Input cards as processed by READGP	MAIN
				EGRAV
				WTBMAT
				STATRD

NC	I	Number of C_{nm} to estimate	READGP	MAIN EGRAV STATRD WTBMAT
NS	I	Number of S_{nm} to estimate	READGP	MAIN EGRAV STATRD WTBMAT
NPRTL	I	Total number of GP to estimate, NC + NS which is initialized to 0 in BLOCK DATA	BLOCK DATA READGP	MAIN OPTCRD ORBIT STATRD ESTIM EGRAV VEVAL WTBMAT
ITERGP	I	Number of iterations to compute numerical partials; defined as 0 in BLOCK DATA and redefined on the COEFGP option card	BLOCK DATA READGP	MAIN

/ESTGP/ (continued)

Variable	Type	Description	Program Where Defined	Program Where Used
ISTATE	L	The switch is initialized to TRUE in BLOCK DATA, indicating that state parameters are to be estimated. The COEFGP option card is used to set this switch to FALSE which indicates that state parameters will not be estimated.	BLOCK DATA READGP	MAIN STATRD ORBIT SWTEST WTBMAT

/FLXBLK/

COMMON /FLXBLK/DSTART, DAY, AVFLX(15), DFLX(15), AP(15)

Variable	Type	Description	Program Where Defined	Program Where Used
DSTART	D	Epoch of initial elements - days from January 0.0 of the year of the epoch of the initial elements	MAIN	DENSTY GEOSRD PREDCT
DAY2	D	Epoch of calculation - days from January 0.0 of the year of the epoch of the initial elements	FRCS	DENSTY
AVFLX(15)	R	Solar flux values averaged over 55 days	MAIN	DENSTY
DFLX(15)	R	Daily values of the solar flux for 15 days starting with DSTART	MAIN	DENSTY
AP(15)	R	Geomagnetic activity index (dimensionless)	MAIN	DENSTY

Units of AVFLX(15) and DFLX(15) are 10^{-22} watts/meter ^{2}/cycle/sec of bandwidth.

/FMODEL/

COMMON /FMODEL/MODEL(8), CS(30, 33), INDEX1, INDEX3

Variable	Type	Description	Program Where Defined	Program Where Used
MODEL(8)	D	Alphanumeric information describing gravity coefficient set used. This array is printed by MAIN for identification purposes.	BLOCK DATA	MAIN COEFL EGRAV
CS(30, 33)	R	The C and S coefficients in the spherical harmonic expansion of the expression for the geopotential	BLOCK DATA	MAIN COEFL EGRAV READGP
INDEX1	I	N = index of highest degree of spherical harmonic coefficients contained in CS	BLOCK DATA	MAIN COEFL EGRAV
INDEX3	I	M = index of highest order of spherical harmonic coefficient contained in CS	BLOCK DATA	MAIN COEFL EGRAV

The C and S coefficients are described by indices n and m. For the zonal harmonics, m = 0. For the sectorial harmonics, m = n. For the tesseral harmonics, $m < n$. Therefore, the C coefficients only fill one triangle of a matrix, as do the S. To conserve space, both sets of coefficients were combined into one matrix. A diagram of that matrix and the computations for the subscripts corresponding to the n and m indices appear in the EGRAV subroutine description in Section 4.2. The maximum geopotential field allowed is (30×30). A(15×15) geopotential (modified SAO C - 5 gravity model) is currently defined in BLOCK DATA. The coefficients of this model are listed in Table 3.

/FMODEL/ (continued)

Table 3

SAO Denormalized Coefficients (C - 5)

C(2, 0)	=	$-1082.645 \cdot 10^{-6}$
S(2, 0)	=	0.0
C(2, 2)	=	$+1.536 \cdot 10^{-6}$
S(2, 2)	=	$-0.872 \cdot 10^{-6}$
C(3, 0)	=	$+2.546 \cdot 10^{-6}$
S(3, 0)	=	0.0
C(3, 1)	=	$+2.091 \cdot 10^{-6}$
S(3, 1)	=	$+0.287 \cdot 10^{-6}$
C(3, 2)	=	$+0.251 \cdot 10^{-6}$
S(3, 2)	=	$-0.184 \cdot 10^{-6}$
C(3, 3)	=	$+0.782 \cdot 10^{-7}$
S(3, 3)	=	$+0.226 \cdot 10^{-6}$
C(4, 0)	=	$+1.649 \cdot 10^{-6}$
S(4, 0)	=	0.0
C(4, 1)	=	$-0.543 \cdot 10^{-6}$
S(4, 1)	=	$-0.445 \cdot 10^{-6}$
C(4, 2)	=	$+0.738 \cdot 10^{-7}$
S(4, 2)	=	$+0.148 \cdot 10^{-6}$
C(4, 3)	=	$+0.509 \cdot 10^{-7}$
S(4, 3)	=	$-0.114 \cdot 10^{-7}$
C(4, 4)	=	$-0.112 \cdot 10^{-8}$
S(4, 4)	=	$+0.486 \cdot 10^{-8}$
C(5, 0)	=	$+0.210 \cdot 10^{-6}$
S(5, 0)	=	0.0
C(5, 1)	=	$-0.677 \cdot 10^{-7}$
S(5, 1)	=	$-0.882 \cdot 10^{-7}$
C(5, 2)	=	$+0.102 \cdot 10^{-6}$
S(5, 2)	=	$-0.375 \cdot 10^{-7}$
C(5, 3)	=	$-0.172 \cdot 10^{-7}$
S(5, 3)	=	$+0.231 \cdot 10^{-9}$
C(5, 4)	=	$-0.206 \cdot 10^{-8}$
S(5, 4)	=	$+0.498 \cdot 10^{-9}$
C(5, 5)	=	$+0.384 \cdot 10^{-9}$
S(5, 5)	=	$-0.146 \cdot 10^{-8}$
C(6, 0)	=	$-0.646 \cdot 10^{-6}$
S(6, 0)	=	0.0
C(6, 1)	=	$-0.370 \cdot 10^{-7}$
S(6, 1)	=	$-0.212 \cdot 10^{-7}$
C(6, 2)	=	$+0.858 \cdot 10^{-8}$
S(6, 2)	=	$-0.455 \cdot 10^{-7}$
C(6, 3)	=	$-0.112 \cdot 10^{-8}$
S(6, 3)	=	$+0.643 \cdot 10^{-9}$
C(6, 4)	=	$-0.167 \cdot 10^{-9}$
S(6, 4)	=	$-0.196 \cdot 10^{-8}$
C(6, 5)	=	$-0.253 \cdot 10^{-9}$
S(6, 5)	=	$-0.370 \cdot 10^{-9}$
C(6, 6)	=	$-0.932 \cdot 10^{-11}$
S(6, 6)	=	$-0.361 \cdot 10^{-10}$
C(7, 0)	=	$+0.333 \cdot 10^{-6}$
S(7, 0)	=	0.0
C(7, 1)	=	$+0.144 \cdot 10^{-6}$
S(7, 1)	=	$+0.114 \cdot 10^{-6}$
C(7, 2)	=	$+0.363 \cdot 10^{-7}$
S(7, 2)	=	$+0.162 \cdot 10^{-7}$
C(7, 3)	=	$+0.352 \cdot 10^{-8}$
S(7, 3)	=	$+0.254 \cdot 10^{-9}$
C(7, 4)	=	$-0.323 \cdot 10^{-9}$
S(7, 4)	=	$-0.217 \cdot 10^{-9}$
C(7, 5)	=	$+0.269 \cdot 10^{-10}$
S(7, 5)	=	$+0.191 \cdot 10^{-10}$
C(7, 6)	=	$-0.145 \cdot 10^{-10}$
S(7, 6)	=	$+0.437 \cdot 10^{-11}$
C(7, 7)	=	$+0.102 \cdot 10^{-11}$
S(7, 7)	=	$+0.180 \cdot 10^{-11}$
C(8, 0)	=	$+0.270 \cdot 10^{-6}$
S(8, 0)	=	0.0
C(8, 1)	=	$-0.520 \cdot 10^{-7}$
S(8, 1)	=	$+0.450 \cdot 10^{-7}$
C(8, 2)	=	$+0.214 \cdot 10^{-8}$
S(8, 2)	=	$+0.320 \cdot 10^{-8}$
C(8, 3)	=	$-0.374 \cdot 10^{-9}$
S(8, 3)	=	$+0.404 \cdot 10^{-10}$
C(8, 4)	=	$-0.277 \cdot 10^{-9}$
S(8, 4)	=	$-0.157 \cdot 10^{-10}$
C(8, 5)	=	$-0.959 \cdot 10^{-11}$
S(8, 5)	=	$+0.214 \cdot 10^{-10}$
C(8, 6)	=	$-0.475 \cdot 10^{-12}$
S(8, 6)	=	$+0.888 \cdot 10^{-11}$
C(8, 7)	=	$-0.444 \cdot 10^{-13}$
S(8, 7)	=	$+0.158 \cdot 10^{-12}$
C(8, 8)	=	$-0.316 \cdot 10^{-12}$
S(8, 8)	=	$+0.130 \cdot 10^{-12}$
C(9, 0)	=	$+0.530 \cdot 10^{-7}$
S(9, 0)	=	0.0
C(9, 1)	=	$+0.760 \cdot 10^{-7}$
S(9, 1)	=	$+0.784 \cdot 10^{-8}$
C(9, 2)	=	$-0.277 \cdot 10^{-9}$
S(9, 2)	=	$+0.242 \cdot 10^{-8}$
C(10, 0)	=	$+0.540 \cdot 10^{-7}$
S(10, 0)	=	0.0
C(10, 01)	=	$+0.649 \cdot 10^{-7}$
S(10, 01)	=	$-0.779 \cdot 10^{-7}$
C(10, 02)	=	$-0.624 \cdot 10^{-8}$
S(10, 02)	=	$-0.250 \cdot 10^{-8}$
C(10, 03)	=	$-0.379 \cdot 10^{-9}$
S(10, 03)	=	$+0.175 \cdot 10^{-9}$
C(10, 04)	=	$-0.436 \cdot 10^{-10}$
S(10, 04)	=	$-0.654 \cdot 10^{-10}$
C(11, 0)	=	$-0.302 \cdot 10^{-6}$
S(11, 0)	=	0.0
C(11, 01)	=	$-0.313 \cdot 10^{-7}$
S(11, 01)	=	$+0.880 \cdot 10^{-8}$
C(12, 0)	=	$+0.357 \cdot 10^{-6}$
S(12, 0)	=	0.0
C(12, 01)	=	$-0.400 \cdot 10^{-7}$
S(12, 01)	=	$-0.402 \cdot 10^{-7}$
C(12, 02)	=	$-0.470 \cdot 10^{-8}$
S(12, 02)	=	$-0.230 \cdot 10^{-9}$
C(12, 12)	=	$-0.278 \cdot 10^{-18}$
S(12, 12)	=	$+0.718 \cdot 10^{-20}$
C(13, 0)	=	$+0.114 \cdot 10^{-6}$
S(13, 0)	=	0.0
C(13, 12)	=	$-0.126 \cdot 10^{-18}$
S(13, 12)	=	$+0.117 \cdot 10^{-18}$
C(13, 13)	=	$-0.239 \cdot 10^{-19}$
S(13, 13)	=	$+0.212 \cdot 10^{-19}$
C(14, 0)	=	$-0.179 \cdot 10^{-6}$
S(14, 0)	=	0.0
C(14, 01)	=	$-0.788 \cdot 10^{-8}$
S(14, 01)	=	$+0.280 \cdot 10^{-8}$
C(14, 11)	=	$+0.947 \cdot 10^{-21}$
S(14, 11)	=	$-0.473 \cdot 10^{-21}$
C(14, 12)	=	$+0.140 \cdot 10^{-20}$
S(14, 12)	=	$-0.132 \cdot 10^{-19}$
C(14, 14)	=	$-0.193 \cdot 10^{-21}$
S(14, 14)	=	$-0.414 \cdot 10^{-22}$
C(15, 09)	=	$-0.241 \cdot 10^{-18}$
S(15, 09)	=	$-0.483 \cdot 10^{-18}$
C(15, 12)	=	$-0.138 \cdot 10^{-19}$
S(15, 12)	=	$-0.190 \cdot 10^{-20}$
C(15, 13)	=	$-0.770 \cdot 10^{-21}$
S(15, 13)	=	$-0.374 \cdot 10^{-21}$
C(15, 14)	=	$+0.114 \cdot 10^{-22}$
S(15, 14)	=	$-0.558 \cdot 10^{-22}$
C(17, 13)	=	$+0.159 \cdot 10^{-22}$
S(17, 13)	=	$+0.280 \cdot 10^{-22}$

/GRBLOK/

COMMON /GRBLOK/GRPAR(3, 50)

Variable	Type	Description	Program Where Defined	Program Where Used
GRPAR(3, 50)	D	Array of acceleration partials with respect to the parameters being estimated. The array is of the form:	DRAG FRCS EGRAV VEVAL	ORBIT RK CSTEP TABLEB

$$\begin{bmatrix} \frac{\partial \ddot{x}}{\partial C_D} & \frac{\partial \ddot{x}}{\partial C_R} & \frac{\partial \ddot{x}}{\partial C_{nm}} & \cdots \\ \frac{\partial \ddot{y}}{\partial C_D} & \frac{\partial \ddot{y}}{\partial C_R} & \frac{\partial \ddot{y}}{\partial C_{nm}} & \cdots \\ \frac{\partial \ddot{z}}{\partial C_D} & \frac{\partial \ddot{z}}{\partial C_R} & \frac{\partial \ddot{z}}{\partial C_{nm}} & \cdots \end{bmatrix}$$

where

$$\left.\begin{matrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{matrix}\right\} = \text{coordinates of the satellite acceleration}$$

and

C_D = drag coefficient,

C_R = reflectivity coefficient,

C_{nm} = geopotential coefficient.

/IORBIT/

COMMON /IORBIT/TREQ, NTER, INITSW

Variable	Type	Description	Program Where Defined	Program Where Used
TREQ	D	Observation time (or output time)	MAIN	ORBIT HEMINT
NTER	I	Iteration number which is initialized to 1 in BLOCK DATA	BLOCK DATA MAIN	ORBIT
INITSW	L	Switch to initialize the orbit generator	MAIN	ORBIT EGRAV ORB1

/LIMITS/

COMMON /LIMITS/S1(3, 50), S2(3, 50), CTOL, ITER, ISCT(2),
ISWT(30), SW(30), ORDER(2)

Variable	Type	Description	Program Where Defined	Program Where Used
S1(3, 50)	D	First sums necessary for the predictor-corrector formulas	SUMS	CSTEP TABLEB
S2(3, 50)	D	Associated second sums	SUMS	CSTEP TABLEB
CTOL	D	Predictor-corrector tolerance	BLOCK DATA OPTCRD	CSTEP
ITER	I	Number of predictor-corrector iterations required to satisfy CTOL tolerance (must be ≤ 3)	CSTEP	ORBIT
ISCT(2)	I	ISCT(1)-number of points produced at a particular stepsize for equations of motion ISCT(2)-number of points produced at a particular stepsize for the variational equations	ORBIT	TEST TABLEB
ISWT(30)*	L	Array of logical external switches which set various options in the program	BLOCK DATA OPTCRD MAIN	FRCS ORBIT SWTEST STATRD
SW(30)*	L	Array of logical internal switches which are set on condition during the execution of a run	BLOCK DATA ORBIT FRCS	EGRAV VEVAL CSTEP HEMINT FRCS SWTEST TABLE TABLEB TEST

/LIMITS/ (continued)

Variable	Type	Description	Program Where Defined	Program Where Used
ORDER(2)	I	ORDER(1)-order of the multistep predictor-corrector integration formulas for the equations of motion ORDER(2)-order of the multistep corrector integration formulas for the variational equations	BLOCK DATA ORBIT	HEMINT TABLEB TEST

The following values, as initially assigned by BLOCK DATA, may be changed by optional input cards:

CTOL = 5.0×10^{-5}

ORDER(1) = 11

ORDER(2) = 7

NOTE: Only ISWT numbers (5), (6), (7), (8), (9), (12), (13), (15), (21), (22), (23), (24), (25), (28), (29), (30), and SW numbers (1), (2), (3), (6), (8), (16), (20), (22), and (24) are being used.

Those ISWT and SW numbers which are presently being used are defined in Tables 4 and 5 respectively.

Table 4

Variable	Description
ISWT(5)	Setting this switch to TRUE means use two-body gravity model for variational equations computation.
ISWT(6)	Setting this switch to TRUE means omit drag model for variational equations computation.
ISWT(7)	Setting this switch to TRUE means omit solar radiation model for variational equations computation.
ISWT(8)	Setting this switch to TRUE means omit solar gravity model for variational equations computation.
ISWT(9)	Setting this switch to TRUE means omit lunar gravity model for variational equations computation.
ISWT(12)	Setting this switch to TRUE means use only two-body gravity model in position acceleration computation.
ISWT(13)	Setting this switch to TRUE means output position partials at each data point.
ISWT(15)	Setting this switch to TRUE means output Keplerian elements at each data point.
ISWT(21)	Setting the switch to TRUE means use two-body gravity model in position acceleration computation.
ISWT(22)	Setting the switch to TRUE means use lunar gravity model in position acceleration computation.
ISWT(23)	Setting the switch to TRUE means use solar gravity model is position acceleration computation.
ISWT(24)	Setting the switch to TRUE means use drag model in position acceleration computations.
ISWT(25)	Setting the switch to TRUE means use solar radiation model in position acceleration computations.
ISWT(28)	Setting the switch to TRUE means a B matrix (normal equations) to be created during first iteration.
ISWT(29)	Setting the switch to TRUE means read position partials from disk.
ISWT(30)	Setting the switch to TRUE means write position partials onto disk.

Table 5

Variable	Description
SW(1)	Setting this switch to TRUE means the initial table values have been completed for the equations of motion.
SW(2)	Setting this switch to TRUE means the initial table values have been completed for the variational equations.
SW(3)	Setting this switch to TRUE means the local error of the table values is being tested for the possibility of doubling or halving the initial stepsize.
SW(6)	Setting this switch to TRUE means doubling of stepsize in TABLE or a step increase in CSTEP will take place.
SW(8)	Setting this switch to TRUE means halving stepsize in TABLE or a step decrease in CSTEP will take place.
SW(16)	Setting this switch to TRUE means reset saved ephemeris quantities to original values.
SW(20)	Setting this switch to TRUE means optimize the stepsize at the first CSTEP point for the equations of motion and the variational equations.
SW(22)	Setting this switch to TRUE means integrate the equations of motion and the variational equations with the same stepsize.
SW(24)	Setting this switch to TRUE means either the equations of motion or the variational equations have been integrated up to the data point—go back and integrate the other to that point.

/MOONGR/

COMMON /MOONGR/RHOM(3, 2), RHOSQ(2), RH03(2)

Variable	Typ.e	Description	Program Where Defined	Program Where Used
RHOM(3, 2)	D	RHOM(I, 1) = $\bar{x} - \bar{x}_m$ RHOM(I, 2) = $\bar{x} - \bar{x}_s$	SUNGRV VEVAL	VEVAL
RHOSQ(2)	D	$\rho_1^2 = R_m^2 + R^2 - 2(\bar{x} \cdot \bar{x}_m)$ $\rho_2^2 = R_s^2 + R^2 - 2(\bar{x} \cdot \bar{x}_s)$	SUNGRV VEVAL	VEVAL
RHO3 (2)	D	ρ_1^3 ρ_2^3	SUNGRV VEVAL	VEVAL

where

$\bar{x}$ = satellite position vector

$\bar{x}_m$ = lunar position vector

$\bar{x}_s$ = solar position vector

R = satellite position vector magnitude

R_m = lunar position vector magnitude

R_s = solar position vector magnitude.

/NON1/

COMMON /NON1/ORBTSW, GRAVSW, XYZFSW, XYZLSW, PLTLSW, PLHSW,
TOREF0, STAPSW, DODSTP, PTAPE, ITRUOR, IBTAPE, JTEMP, DATP

Variable	Type	Description	Program Where Defined	Program Where Used
ORBTSW	L	Setting this switch to TRUE means an orbit generator run will be made.	BLOCK DATA OPTCRD	MAIN STATRD OUTPUT
GRAVSW	L	Setting this switch to TRUE means that the gravity field being used will be printed.	BLOCK DATA OPTCRD	MAIN STATRD
XYZFSW	L	Setting this switch to TRUE means that the ground track will be printed only on the first iteration of a data reduction run.	BLOCK DATA OPTCRD	MAIN STATRD
XYZLSW	L	Setting this switch to TRUE means that the ground track will be printed only on the last iteration of a data reduction run.	BLOCK DATA OPTCRD	MAIN STATRD
PLTLSW	L	Setting this switch to TRUE means that a GEORGE tape has been requested at the end of the run.	BLOCK DATA OPTCRD	MAIN STATRD
PHLSW	L	A switch which is initialized to FALSE indicating that a station's position is in rectangular coordinates and is to be stored in STASIG array. Setting this switch to TRUE means that a station's position is in geodetic coordinates and to be stored in PLHSIG array.	BLOCK DATA OPTCRD	MAIN STATRD

/NON1/ (continued)

Variable	Type	Description	Program Where Defined	Program Where Used
TOREF0	L	Setting this switch to TRUE means that an ORB1 TAPE will be created.	BLOCK DATA OPTCRD	MAIN STATRD
STAPSW	L	A switch which is initialized to TRUE. Setting this switch to FALSE means that a STAPOS option card and station position cards are to be read.	BLOCK DATA OPTCRD	MAIN STATRD
DODSTP	L	Setting this switch to TRUE means that a DOD's formatted observation tape is to be used during a data reduction run.	BLOCK DATA MAIN	MAIN STATRD
PTAPE	L	Setting this switch to TRUE means that a simulated data tape comprised of rectangular coordinate type data will be created during an orbit generator run.	BLOCK DATA OPTCRD	MAIN STATRD
ITRUOR	I	A variable which defines a device to input or output simulated observational type data. This variable is initialized to 36.	BLOCK DATA OPTCRD	
IBTAPE	I	A variable which defines a device to output a B matrix (normal equations). This variable is initialized to 18.	BLOCK DATA OPTCRD	MAIN

/NON1/ (continued)

Variable	Type	Description	Program Where Defined	Program Where Used
JTEMP	I	A variable which defines the type of data reduction run that is being made. The variable is initialized to 0 which means that a data reduction run using real data will be made. JTEMP can be redefined to have a value of 1 which indicates that a data reduction run using simulated data will be made. JTEMP can also be redefined to have a value of 2 which indicates that a data reduction run using real data will be made and that the computed observations will be stored on tape, creating a simulated data tape.	BLOCK DATA OPTCRD	MAIN STATRD
DATP	I	A variable which defines a device to store and then use real or simulated data. The variable is initialized to 35 indicating that real data will be stored and used in a subsequent iteration. On option, the variable can be redefined to 36 indicating that simulated data will be stored and used.	BLOCK DATA OPTCRD	MAIN

All logicals are initialized to FALSE in the BLOCK DATA subprogram except STAPSW.

/NON2/

COMMON /NON2/XYZTP, RVTP, IOBS, INPAR, NPARAM

Variable	Type	Description	Program Where Defined	Program Where Used
XYZTP	I	A variable which defines a device to output XYZ type data. The variable is initialized to 12 for XYZ tape output but the ORBIT option card redefines the device as the printer (6).	BLOCK DATA OPTCRD	MAIN STATRD
IOBS	I	A variable which defines the input device for either real observational type data or rectangular coordinate type data. The variable is initialized to 10.	BLOCK DATA OPTCRD	MAIN
INPAR	I	A variable reflecting whether the state, drag or solar radiation parameters are being estimated	BLOCK DATA OPTCRD	MAIN ESTIM READGP WTBMAT ORBIT PREDCT
NPARAM	I	The total number of parameters to be estimated	OPTCRD	MAIN ESTIM SOLVGP WTBMAT

/NON2/ (continued)

NOTE

INPAR is initialized to 6 which indicates that state parameters will be estimated. The COEFGP option card redefines INPAR to be 0 if state parameters are not being estimated. Therefore,

Label	Value If Present	Description
INPAR = COEFGP	6	Estimate x_0, y_0, z_0, $\dot{x}_0$, $\dot{y}_0$, $\dot{z}_0$
+ADDR	1	Estimate C_D
+SRAD	1	Estimate C_R
NPARAM = INPAR	6, 7, or 8	
+NPRTL	NP	Number of GP to be estimated
+(3* NSTEST)	NS	Number of station coordinates to be estimated
+NBIAS	NB	Number of biases to be estimated

where

$$\left.\begin{array}{c} x_0 \\ y_0 \\ z_0 \\ \dot{x}_0 \\ \dot{y}_0 \\ \dot{z}_0 \end{array}\right\} = \text{state parameters}$$

C_D = drag coefficient

C_R = solar radiation coefficient

GP = geopotential coefficients.

/NON3/

COMMON /NON3/TIMING, TTL, CDNAME, ATYPE, UNITS, BLANK, NCARDS

Variable	Type	Description	Program Where Defined	Program Where Used
TIMING	D	The title "TIMING"	BLOCK DATA	MAIN STATRD
TTL(50)	D	Title array for parameters being estimated	BLOCK DATA	MAIN STATRD
CDNAME(32)	D	Array of option card names. There are 32 option cards available.	BLOCK DATA	OPTCRD STATRD
ATYPE(21)	D	Array of data type names. There are 21 data types available.	BLOCK DATA	MAIN OPTCRD STATRD
UNITS(15)	D	An array of the names of the units used for print-out	BLOCK DATA	MAIN OPTCRD STATRD
BLANK	D	1 blank space	BLOCK DATA	MAIN OPTCRD STATRD
NCARDS	I	Total number of option cards available (presently 32 option cards are available)	BLOCK DATA	OPTCRD

/NON4/

COMMON /NON4/BSEND, BSTRT, RMSTOT, EDITN, BYTPE, BSTANO, IRSUPR, NBIAS, NSTA, ISTEST, NSTEST, NOPRPR, IDSAT

Variable	Type	Description	Program Where Defined	Program Where Used
BSEND(44)	D	End time of biases being estimated	OPTCRD	MAIN STATRD
BSTRT(44)	D	Start times of biases being estimated	OPTCRD	MAIN STATRD
RMSTOT	R	Input value for the desired total RMS. The nominal value is 200.0.	BLOCK DATA OPTCRD	MAIN
EDITN	R	Input sigma multiplier. The nominal value is 3.5.	BLOCK DATA OPTCRD	MAIN
BYTPE(44)	I	Bias types to be estimated	OPTCRD	MAIN
BSTANO(44)	I	Station numbers for which biases are requested	OPTCRD	MAIN
IRSUPR(4)	I	Indicates which iterations to suppress residual printout. This is initialized to 0.	BLOCK DATA OPTCRD	STATRD
NBIAS	I	Number of biases being estimated. This variable is initialized to 0.	BLOCK DATA OPTCRD	MAIN
NSTA	I	Number of stations to be used in a run (not allowed to exceed 50). This variable is initialized to 0.	BLOCK DATA OPTCRD	MAIN STATRD
ISTEST	I	Array of station numbers corresponding to those stations that are to be estimated	OPTCRD	STATRD

/NON4/ (continued)

Variable	Type	Description	Program Where Defined	Program Where Used
NSTEST	I	Number of stations that are being estimated. This variable is initialized to 0.	BLOCK DATA OPTCRD	MAIN STATRD PREDCT ESTIM
NOPRPR	I	Number of stations requesting data to be preprocessed. This variable is initialized to 0.	BLOCK DATA OPTCRD	STATRD
IDSAT	I	Satellite identification number	OPTCRD	MAIN

/NON5/

COMMON /NON5/DORBIT, DAYEND, DORB1, DORB1E, EPSEC, DRATE, ORBRT, RATE, SEC, IEPHM, IHM, IYMD, IYBEG, IEPYMD, IYREF

Variable	Type	Description	Program Where Defined	Program Where Used
DORBIT	D	Integration start time which is initialized to 0.0	BLOCK DATA OPTCRD	MAIN STATRD
DAYEND	D	Integration end time which is initialized to 0.0	BLOCK DATA OPTCRD	MAIN STATRD
DORB1	D	Integration start time when creating an ORB1 tape is initialized to -1.0	BLOCK DATA OPTCRD	MAIN STATRD
DORB1E	D	Integration end time for creating an ORB1 tape	OPTCRD	MAIN STATRD
EPSEC	D	Number of seconds into the epoch day	OPTCRD	MAIN STATRD
DRATE	D	Print time interval in days for a ground track request. The value is initialized to 999.0.	BLOCK DATA OPTCRD	MAIN STATRD
ORBRT	D	Output time interval in seconds for an ORB1 tape	OPTCRD	MAIN STATRD
RATE	R	Print time interval in seconds for an orbit generator run	OPTCRD	MAIN STATRD
SEC	R	Number of seconds into the epoch day	OPTCRD	MAIN STATRD
IEPHM	I	Epoch date (packed hours and minutes)	OPTCRD	MAIN STATRD
IHM	I	Selected end time (packed hours and minutes) of arc for data reduction run	OPTCRD	MAIN STATRD

/NON5/ (continued)

Variable	Type	Description	Program Where Defined	Program Where Used
IYMD	I	Selected end time (packed year, month, and day) of arc for data reduction run	OPTCRD	MAIN STATRD
IYBEG	I	Reference year of the epoch elements	OPTCRD	MAIN STATRD
IEPYMD	I	Date of the epoch elements (packed year, month, and day)	OPTCRD	MAIN STATRD
IYREF	I	Reference date of the epoch elements (packed year, month, and day)	OPTCRD	MAIN STATRD

/NON6/

COMMON /NON6/ORBEL, SATNME, XYZNOM, PLHNOM, HN,
SLATN, LATDN, LATMN, LONDN, LONMN, SLONN

Variable	Type	Description	Program Where Defined	Program Where Used
ORBEL(6)	D	Epoch orbital elements	MAIN	STATRD
SATNME	D	Satellite name for printout purposes	MAIN	STATRD
XYZNOM(10, 6)	R	Nominal variance-covariance of a station position in earth fixed rectangular coordinate system	STATRD	MAIN
PLHNOM(10, 6)	R	Variance-covariance of a station position in the geodetic coordinate system	STATRD	MAIN
HN(10)	R	Nominal station height in meters	STATRD	MAIN
SLATN(10)	R	Number of seconds in the nominal station geodetic latitude	STATRD	MAIN
LATDN(10)	I	Number of degrees in the nominal geodetic latitude	STATRD	MAIN
LATMN(10)	I	Number of minutes in the nominal station geodetic latitude	STATRD	MAIN
LONDN(10)	I	Number of degrees in the nominal station east longitude	STATRD	MAIN
LONMN(10)	I	Number of minutes in the nominal station east longitude	STATRD	MAIN
SLONN(10)	R	Number of seconds in the nominal station east longitude	STATRD	MAIN

/OUTFOR/

COMMON /OUTFOR/TITLE, CENVRG, NITER

Variable	Type	Description	Program Where Defined	Program Where Used
TITLE(36)	D	Title identification information for a run	MAIN	OUTPUT
CENVRG	R	Convergence criteria on RMS	MAIN	OUTPUT
NITER	I	Upper bound on total number of iterations for a run	MAIN	OUTPUT

/PCES/

COMMON /PCES/PCESW

Variable	Type	Description	Program Where Defined	Program Where Used
PCESW	L	The switch is initialized to FALSE in BLOCK DATA. The PCE option card is used to set the switch to TRUE, which indicates that rectangular coordinate type simulated data will be processed.	BLOCK DATA OPTCRD	MAIN STATRD PREDCT OUTPUT

NOTE

Rectangular coordinate type simulated data (PCE Data) is processed in a fashion simular to the ordinary data types. The differences occur in the input format and measurement partial computations, where changes were made so that rectangular coordinate "measurement" types could be processed.

/PCOFIT/

COMMON /PCOFIT/NO

Variable	Type	Description	Program Where Defined	Program Where Used
NO	I	Number of observations needed to calculate the coefficients of the polynomial used to determine the ephemeris quantities. The nominal value set in the program is 11.	BLOCK DATA	EPHQAN

/PREBLK/

COMMON /PREBLK/OBS1, OBS2, IPREPR(4, 50), RFINDX(2, 50), INDPRE(2, 50), VHFCHN

NOTE

Subprogram GEOSRD is the program utilized by MAIN to read records of observational data, select and delete records on request, and preprocess optical and time observational data as requested by various indicator variables. Part of the final observational data is returned through COMMON/PREBLK/. The remainder of the observational data is returned through COMMON /CGEOS/. Each record of observational data may contain one or two observations.

Variables	Type	Description	Program Where Defined	Program Where Used
OBS1	D	1st observation contained in record	GEOSRD PROCES	MAIN PROCES PREDCT
OBS2	D	2nd observation if the record contains two observations (Refer to indicator NMEAS in COMMON /GEOS/.)	GEOSRD PROCES	MAIN PROCES PREDCT

NOTE

The indicator variables MTYPE and NMEAS contained in COMMON/CGEOS/ must be used in order to interpret OBS1 and OBS2. These are used as follows:

$$\left.\begin{array}{l}\text{MTYPE} = 1\\ \text{NMEAS} = 2\end{array}\right\}\begin{array}{l}\text{OBS1} = \text{right ascension - radians}\\ \text{OBS2} = \text{declination - radians}\end{array}$$

$$\left.\begin{array}{l}\text{MTYPE} = 2\\ \text{NMEAS} = 1\end{array}\right\}\ \text{OBS1} = \text{range - meters}$$

$$\left.\begin{array}{l}\text{MTYPE} = 3\\ \text{NMEAS} = 1\end{array}\right\}\ \text{OBS1} = \text{range rate - meters/sec.}$$

/PREBLK/ (continued)

NOTE (continued)

MTYPE = 4
NMEAS = 1
	OBS1 = range rate - meters/sec
	(derived from Doppler)

MTYPE = 5
NMEAS = 2
	OBS1 = ℓ direction cosine - dimensionless
	OBS2 = m direction cosine - dimensionless

MTYPE = 6
NMEAS = 2
	OBS1 = X angle - radians
	OBS2 = Y angle - radians

MTYPE = 7
NMEAS = 2
	OBS1 = azimuth - radians
	OBS2 = elevation - radians

Variables	Type	Description	Program Where Defined	Program Where Used
IPREPR(1, N)	I	Indicator for preprocessing optical data including provisions for active or passive time correction	MAIN	PROCES GEOSRD
(2, N)	I	Indication for applying new index of refraction	MAIN	PROCES
(3, N)	I	Indicator for applying constant timing corrections	MAIN	PROCES GEOSRD
(4, N)	I	Not used		PROCES GEOSRD
RFINDX(1, N)		New values of index of refraction in ppm deviation from 1, from Nth preprocessing option card	MAIN	PROCES GEOSRD
INDPRE(1, N)	I	Station number from Nth preprocessing option card	MAIN	GEOSRD
INDPRE(2, N)	I	Measurement type from Nth preprocessing option	MAIN	GEOSRD

The observational preprocessing information contained in the arrays IPREPR(4, 50), RFINDX(2, 50) and INDPRE(2, 50) are read from the option card PREPRO. This information is then interpreted by subprograms GEOSRD and PROCES for appropriate preprocessing.

/PRIORI/

```
COMMON /PRIORI/BEGYMD, ENDYMD, SFLUX(2499), MGFLUX(2499), DUMMY(10000)
        VARCOV(6, 6), STASIG(3, 3, 10), PLHSIG(3, 3, 10), ELEMIN(6), XSTA(10) YSTA(10),
        ZSTA(10), DRAGO, EMISS0, DRAGSG, EMISSG, BBIAS(44), BIAS0(44), BIASSG(44)
```

R(10, 6, 20) is equivalenced in RK to BEGYMD, thus writing over the flux tables after initial use, and SUM1(50, 50) is equivalenced in MAIN to BEGYMD, thus writing over the initial integration starting tables after initial use.

Variable	Type	Description	Program Where Defined	Program Where Used

BEGYMD	I	Beginning date of the tables of solar flux and geomagnetic activity data in the form YYMMDD where
		YY = last 2 digits of year
		MM = month
		DD = day
			BLOCK DATA	MAIN

ENDYMD	I	End date of the tables of solar flux and geomagnetic activity data in the form YYMMDD	BLOCK DATA	MAIN
SFLUX(2499)	R	Daily values of solar flux at 2800 Mc (10.7 cm) in units of watts/meters ^{2}/cycle/second/ bandwidth × 10^{-22} measured at Ottawa ARO, adjusted to 1 astronomical unit from BEGYMD to ENDYMD	BLOCK DATA	MAIN
MGFLUX(2499)	R	Daily values of geomagnetic activity indices (dimensionless) from BEGYMD to ENDYMD	BLOCK DATA	MAIN
DUMMY*(1000)	R	Scratch storage array		

*DUMMY is used to increase the total length of the PRIORI COMMON block in order to store intermediate values needed in the computation of the starting table (see subprogram RK).

/PRIORI/ (continued)

Variable	Type	Description	Program Where Defined	Program Where Used
VARCOV(6, 6)	R	Not used, replaced by /CVRCV/VRCOV		
STASIG(3, 3, 10)	R	Variance-covariance matrix of the a priori estimate of the geocentric earth-fixed rectangular coordinates of the tracking stations to be adjusted-meters. (Maximum of 10 stations allowed to be adjusted simultaneously.)	MAIN	MAIN ESTIM
PLHSIG(3, 3, 10)	R	Variance-covariance matrix of the a priori estimate of the geodetic latitude, longitude and height above ellipsoid of the tracking stations to be adjusted—angles in radians, height in meters	MAIN SQUANT (as output from subroutine PLHOUT)	MAIN ESTIM
ELEMIN(6)	D	A priori estimate of the geocentric inertial rectangular coordinates of the position and velocity vectors of the satellite at the initial epoch to be adjusted—meters and meters/sec		

XSTA(10)
YSTA(10)
ZSTA(10)
	D	A priori estimate of the geocentric earth-fixed rectangular coordinates of the tracking stations to be adjusted—meters	MAIN	ESTIM

DRAG0	R	C_{D_0}		
EMISS0	R	C_{R_0}		
DRAGSG	R	$\sigma\left(C_{D_0}\right)$		
EMISSG	R	$\sigma\left(C_{R_0}\right)$		
BBIAS(44)	R	b	MAIN	MAIN ESTIM

Variable	Type	Description	Program Where Defined	Program Where Used
BIAS0(44)	R	b_0	MAIN	MAIN ESTIM
BIASSG(44)	R	$\sigma(b_0)$	MAIN	MAIN ESTIM

where:

C_{D_0} = a priori value of the satellite's drag coefficient

C_{R_0} = a priori value of the satellite's reflectivity coefficient

$\sigma\left(C_{D_0}\right)$ = standard deviation of C_{D_0}

$\sigma\left(C_{R_0}\right)$ = standard deviation of C_{R_0}

b_0 = array containing the a priori values of the measurement biases or station timing biases to be adjusted

b = array containing the current adjusted values of the measurement biases or station timing biases

$\sigma(b_0)$ = array containing the standard deviations of b_0.

Data for SFLUX and MGFLUX are defined in BLOCK DATA for the periods

BEGYMD = 650101 (January 1, 1965)

to

ENDYMD = 680430 (April 30, 1968).

During the initialization process of the main program, values for 15 days of solar flux and geomagnetic data, with the first value corresponding to the day of the epoch of the initial elements, are placed in another common block (COMMON/FLXBLK/). Appropriate provisions are made for estimating values if the epoch of the initial elements lies outside the range of the table. After these data have been placed in COMMON/FLXBLK/, the storage area in COMMON/PRIORI/, starting with the variable BEGYMD and ending with the last value of MGFLUX, is utilized by subroutines RK and ESTIM to store temporary integration arrays and the matrix SUM1(50, 50). As a result the stored

/PRIORI/ (continued)

values of solar flux and geomagnetic activity, data are overlaid and no longer available for use. This fact prevents "stacking" cases "back to back" since the required solar flux and geomagnetic data are available only on the 1st pass through the program.

After solution, the matrix SUM1(50, 50) contains the variance-covariance matrix of all parameters which are adjusted by the DC.

/RKST/

COMMON /RKST/RSTEP

Variable	Type	Description	Program Where Defined	Program Where Used
RSTEP	D	Starting table stepsize. It should generally be a fraction of the Cowell stepsize. The nominal value is 20 seconds but can be changed using the RKSTEP option card.	BLOCK DATA OPTCRD	RK

/RKT/

COMMON /RKT/CSTEPT

Variable	Type	Description	Program Where Defined	Program Where Used
CSTEPT	D	Required Cowell stepsize time points for the starting table	TABLE	RK

/SCRTCH/

```
COMMON /SCRTCH/DUMMY(500)
```

This common block is used as a scratch storage area by the following programs:

MAIN
COEFF
SQUANT.

It is only used for the storage of information that need not be saved. Usage of this common block in other routines is permitted under the same restrictions.

Program	Usage of /SCRTCH/
MAIN	To store station positions in output format at the beginning of a case and at the end of each iteration when station position adjustment has been requested. Used to pass station variance-covariance information to subroutine SQUANT when station position adjustment has been requested.
SQUANT	To obtain station positions at the beginning of each case for conversion to earth-fixed rectangular coordinates. To return station positions to the main program, in output format, when station position adjustment has been requested.
COEFF	To store matrices needed for calculation of coefficients for polynomial fit.

/SETSW/

COMMON /SETSW/IND(9), I10, INDX

Variable	Type	Description	Program Where Defined	Program Where Used
IND(9)	I	An array of indicators which define computed GO TO statements in the force computations for the variational equations	SWTEST	VEVAL
I10	I	An indicator which when set to: 1-indicates state parameters will not be estimated, 2-indicates that only state parameters will be estimated, 3-indicates that state and force model parameters are to be estimated.	SWTEST	VEVAL MAIN ESTIM DRAG FRCS
INDX	I	An indicator which when set to 0 is the same as I10 set to 1; a value of 6 is the same as I10 set to 2 or 3.	SWTEST	VEVAL MAIN ESTIM DRAG FRCS

NOTE

Indicators I10 and INDX are used to pack the GRPAR, PXPX0 and PMPX0 arrays so that if a particular parameter type is not to be estimated, the lower parameter partials move up.

/SIGMAC/

COMMON /SIGMAC/SIGCHG, IMTYPE, ISTNO, SIGSTD, CULL, NSIG, NCULL

This common block is used to transfer optional input card information to the observation processing modules.

Variable	Type	Description
		The arrays, SIGCHG, IMTYPE, ISTNO, store the optional inputs used to override the standard observation sigmas defined by the observation types in SIGSTD. This permits specification of an observation sigma by station and/or observation type.
SIGCHG(50)	R	SIGCHG(J) = Value of sigma in units of observation type (see SIGSTD)
IMTYPE(50)	I	Observation type for SIGCHG(J)
ISTNO(50)	I	Applicable station number for SIGCHG(J)
SIGSTD(14)	R	Pre-stored array of observation sigmas by type, which is initialized in BLOCK DATA to the following:

Type	Sigma Value	Observation
(1)	20. seconds of arc	rt. ascension
(2)	25. meters	range
(3)	10. meters/sec	range rate
(4)	1.	Doppler
(5)	0.3	direction cosine,
(6)	50. degrees	X angle
(7)	50. seconds of arc	azimuth
(8)	20. seconds of arc	declination
(12)	0.3 dimensionless	direction cosine, m
(13)	50. degrees	Y angle
(14)	50. seconds of arc	elevation
(16)	100. meters	geocentric coordinate x
(17)	100. meters	geocentric coordinate y
(18)	100. meters	geocentric coordinate z
(19)	0.1 meters/sec	geocentric coordinate $\dot{x}$
(20)	0.1 meters/sec	geocentric coordinate $\dot{y}$
(21)	0.1 meters/sec	geocentric coordinate $\dot{z}$

/SIGMAC/ (continued)

Variable	Type	Description
CULL(2, 100)	I	Defines the range of observations, by sequence number to be deleted from a DC run; where CULL(1, K) is the first number and CULL(2, K) is the last number. K is the number of sets of observations to be deleted.
NSIG	I	Number of variables in arrays, SIGCHG, IMTYPE, ISTNO which is initialized to 0 in BLOCK DATA.
NCULL	I	Number of sets of observations to be deleted, K. Initialized to 0 in BLOCK DATA.

These variables are defined in subprogram OPTCRD from optional input cards and are subsequently used by subprograms DODSRD and GEOSRD.

/STANUM/

COMMON /STANUM/NAME(50), ISTANO(50)

Variable	Type	Description	Program Where Defined	Program Where Used
NAME(N)	D	Identifying name of the Nth tracking station (assumed to have a maximum of six alphanumeric characters)	BLOCK DATA	MAIN STATRD PLHOUT
ISTANO(N)	I	Identifying number of the Nth tracking station (currently assumed to have a maximum of four digits)	BLOCK DATA	STATRD MAIN

The arrays NAME and ISTANO are also stored by STATRD as the station position cards are read in. A maximum of 50 tracking stations are allowed by GEOSTAR-I.

/VRBLOK/

COMMON /VRBLOK/RTXYSQ, COSLAM(31), SINLAM(31), RR

Variable	Type	Description	Program Where Defined	Program Where Used
RTXYSQ	D	$\sqrt{x^2 + y^2}$ - meters	EGRAV	EGRAV VEVAL
COSLAM(N)	R	cos (N - 1)λ; N = 1, 2, ··· 31	EGRAV BLOCK DATA	EGRAV VEVAL
SINLAM(N)	R	sin (N - 1)λ; N = 1, 2, ··· 31	EGRAV BLOCK DATA	EGRAV VEVAL
RR	D	r - meters	EGRAV (same value as R placed in COMMON /XYZ/)	VEVAL

where:

x, y = geocentric inertial x and y components of the position vector of the satellite

r = geocentric distance of satellite

λ = geodetic longitude of the sub-satellite point.

BLOCK DATA initially defines:

RTXYSQ = 0
COSLAM(1) = 1
COSLAM(2) – COSLAM(31) = 0
SINLAM(1) – SINLAM(31) = 0

/VRBLOK/ (continued)

Computing Note

Particular attention should be paid to the fact that subroutine VEVAL establishes the variables in this block as:

```
COMMON /VRBLOK/A1, G2, CSLM(31), SNLM(30), R1
  DOUBLE PRECISION A1, R1
```

The correspondence of the /VRBLOK/ variables in VEVAL with the /VRBLOK/ variables in EGRAV are:

VEVAL	EGRAV
A1	RTXYSQ
G2	COSLAM(1)
CSLM(1) - CSLM(30)	COSLAM(2) - COSLAM(31)
CSLM(31)	SINLAM(1)
SNLM(1) - SNLM(30)	SINLAM(2) - SINLAM(31)
R1	RR

/WORKER/

COMMON /WORKER/X(20, 3, 50), XD(20, 3, 50), XDD(20, 3, 50), XXDD(3, 50), CDEL(2), TT(2), T(2), KI(2), N(2), NEQ, IPA(2)

Variable	Type	Description	Program Where Defined	Program Where Used
X(20, 3, 50)	D	Array of position vectors for the equations of motion and the variational equations in meters	ORBIT	CSTEP SUMS HEMINT RK FRCS TABLE TABLEB CKDIFF
XD(20, 3, 50)	D	Corresponding array of velocity vectors in meters/second	ORBIT	CSTEP SUMS HEMINT RK FRCS TABLE TABLEB CKDIFF
XDD(20, 3, 50)	D	Corresponding array of acceleration vectors in meters/second^2	ORBIT	CSTEP SUMS HEMINT RK FRCS TABLE TABLEB CKDIFF
XXDD(3, 50)	D	Current acceleration vector at a given time point for the equations of motion and the variational equations	ORBIT	CSTEP SUMS HEMINT RK FRCS TABLE TABLEB CKDIFF

/WORKER/ (continued)

Variable	Type	Description	Program Where Defined	Program Where Used
CDEL(2)	D	CDEL(1) = h_1 CDEL(2) = h_2	MAIN OPTCRD BLOCK DATA	CSTEP SUMS HEMINT RK FRCS TABLE TABLEB CKDIFF
TT(2)	D	TT(1) = h_1^2 TT(2) = h_2^2	ORBIT	CSTEP SUMS HEMINT RK FRCS TABLE TABLEB CKDIFF
T(2)	D	T(1) is the time of the current position, velocity, and acceleration vectors from epoch for the equations of motion. T(2) is the time of the current position, velocity, and acceleration vectors from epoch for the variational equations.	ORBIT	CSTEP SUMS HEMINT RK FRCS TABLE TABLEB CKDIFF
KI(2)	I	KI(1) indicates the position (in the position, velocity, and acceleration arrays) of the current position, velocity and acceleration vectors for the equations of motion. KI(2) indicates the position (in the position, velocity, and acceleration arrays) of the current position, velocity, and acceleration vectors for the variational equations	ORBIT	CSTEP SUMS HEMINT RK FRCS TABLE TABLEB CKDIFF

/WORKER/ (continued)

Variable	Type	Description	Program Where Defined	Program Where Used
N(2)	I	$N(1) = P_1 - 2$ $N(2) = P_2 - 2$	ORBIT	CSTEP SUMS HEMINT RK FRCS TABLE TABLEB CKDIFF
NEQ	I	Number of equations being integrated	MAIN	CSTEP SUMS HEMINT RK FRCS TABLE TABLEB CKDIFF
IPA(2)	I	IPA(1) is the storage location of the interpolated vectors for the equations of motion; IPA(2) is the storage location of the interpolated vectors for the variational equations	ORBIT	CSTEP SUMS HEMINT RK FRCS TABLE TABLEB CKDIFF

where

h_1 = stepsize used to integrate the equations of motion

h_2 = stepsize used to integrate the variational equations

p_1 = order of formulas used to integrate the equations of motion

p_2 = order of formulas used to integrate the variational equations.

/XYZ/

COMMON /XYZ/X, Y, Z, XDOT, YDOT, ZDOT, R, RSQ, RQ, TI

Variable	Type	Description	Program Where Defined	Program Where Used
X Y Z	D	Satellite position vector in meters	FRCS (Defined from variables input through COMMON /WORKER/ from CSTEP or RK	EGRAV DRAG VEVAL
XDOT YDOT ZDOT	D	Satellite velocity vector in meters/second.	FRCS (Defined from variables input through COMMON /WORKER/ from CSTEP or RK	EGRAV DRAG VEVAL
R	D	r - meters	EGRAV FRCS VEVAL	DENSTY DRAG EGRAV VEVAL
RSQ	D	r^2 - meters^2	FRCS VEVAL	VEVAL
RQ	D	r^3 - meters^3	FRCS VEVAL	VEVAL
TI	D	Current integration time of the variational equations.	FRCS CSTEP	VEVAL

/XYZ/ (continued)

where:

$$\bar{r} = \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} \qquad \bar{v} = \begin{pmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \end{pmatrix}$$

geocentric inertial rectangular coordinates of the position and velocity vectors of the satellite at the epoch when subroutine FRCS is called by either subroutine CSTEP or RK for the calculation of accelerations during the integration process. $\bar{r}$ and $\bar{v}$ are passed by CSTEP and RK to FRCS by means of the COMMON /WORKER/. If the equations of motion and the variational equations are being integrated with the same stepsize, then FRCS places this information in COMMON /XYZ/ for use in subprograms DRAG, EGRAV, DENSTY, and VEVAL. However, if two different stepsizes are being used, then $\bar{r}$ and $\bar{v}$ are interpolated vectors at the time needed by the variational equations and placed in the COMMON /XYZ/ by CSTEP for use in VEVAL.

/XYZOUT/

COMMON /XYZOUT/XYZEND(6)

NOTE

The array XYZEND(6) contains the geocentric inertial rectangular coordinates of the position and velocity vectors of the satellite at each observation time or requested print time.

Variable	Type	Description	Program Where Defined	Program Where Used
XYZEND(1)	D	x - meters	MAIN (as output from subroutine ORBIT)	MAIN (as input to subroutine ORBIT) PREDCT
XYZEND(2)	D	y - meters		
XYZEND(3)	D	z - meters		
XYZEND(4)	D	x - meters/second		
XYZEND(5)	D	y - meters/second		
XYZEND(6)	D	z - meters/second		

where

$$\left.\begin{matrix}\bar{r} = \begin{Bmatrix} x \\ y \\ z \end{Bmatrix} \\ \bar{v} = \begin{Bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{Bmatrix}\end{matrix}\right\} \text{geocentric inertial rectangular coordinates of the position and velocity vectors of the satellite}$$

5.2 NONAME—GEOSTAR-I ODP COMMON Block Cross Reference Table

This section contains a cross reference table describing the COMMON area structure in the GEOSTAR-I ODP.

GEOSTAR-I COMMON Cross Reference Table

SUBROUTINES

COMMON BLOCKS	MAIN	APPER	BLKSTA	BLDAT1	BLDAT2	BLDAT3	CKDIFF	COEFF	COEFL	CSTEP	DENSTY	DODSRD	DRAG	EGRAV	ELEM	ELEMK	EPHQAN	ERROR	ESTIM	FRCS	GDET	GEOSRD	HEMINT	INV2	OBSDOT	OPTCRD	ORB1	ORBIT	OUTPUT	
ABCOEF					X					X																		X		
ANPART										X																		X		
BEQ	X				X														X							X				
CELEM	X	X													X				X							X	X			
CGEOS	X											X										X				X	X			
COFIT	X				X												X			X								X		
CONST1	X	X			X							X	X		X	X			X	X						X	X	X	X	
CONST2	X				X						X	X			X	X						X				X	X	X		
CONST3	X	X											X				X			X							X	X		
CONVRG	X																									X		X	X	
CORB1	X	X																									X			
COSAVE					X												X			X										
COWS	X				X													X								X		X	X	
CPARTL	X																				X									
CQUANT	X		X	X															X						X					
CSTINF	X																													
CVRCOV	X				X														X							X				
DC																								X						
DRGBLK											X		X																	
ERR22										X																				
ESTGP	X													X					X							X		X	X	
FLXBLK	X										X	X								X		X				X	X			
FMODEL	X				X				X					X												X	X		X	
GRBLOK										X			X	X						X								X		
IORBIT	X													X													X	X		
LIMITS	X				X					X				X						X			X			X	X	X	X	
NON1	X				X																					X				
NON2	X				X														X									X		
NON3	X				X																					X				
NON4	X				X																					X				
NON5	X				X																					X				
NON6	X																													
OUTFOR	X																												X	
PCES	X																									X			X	
PCOFIT					X												X													
PREBLK	X											X										X				X				
PRIORI	X					X													X							X			X	
RKST					X																					X		X	X	
SCRTCH	X		X					X																						
SETSW	X									X			X	X					X	X								X		
SIGMAC				X								X										X				X				
STANUM	X		X									X										X								
VRBLOK					X									X																
WORKER	X				X		X			X										X			X			X		X	X	
XYZ										X	X		X	X						X								X		
XYZOUT	X																								X			X		

GEOSTAR-I COMMON Block Cross Reference Table (Continued)

SUBROUTINES

COMMON BLOCKS	PLHOUT	POSVEL	PREDCT	PRNTPR	PROCES	READGP	REFIMP	REFCOR	RK	SOLVGP	SQUANT	STAINF	STATRD	STORGP	SUMST	SUMTOB	SUNGRV	SWTEST	TABLE	TABLEB	TEST	VEVAL	WTBMAT	XEFIX	XINERT	YEFIX	YINERT
ABCOEF															X												
ANPART									X											X		X					
BEQ										X						X							X				
CELEM		X																					X				
CGEOS								X					X														
COFIT			X																			X					
CONST1	X	X	X								X		X									X	X				
CONST2			X										X														
CONST3			X		X												X					X					
CONVRG													X														
COWS																			X		X						
CPARTL			X																								
CQUANT	X		X				X				X		X														
CSTHET			X																					X	X	X	X
CSTINF												X															
CSVEVL														X								X					
CVRCOV													X														
DRGBLK																						X					
ERR22																					X						
ESTGP			X			X							X					X				X	X				
FLXBLK			X																			X					
FMODEL						X								X													
GRBLOK									X											X		X					
LIMITS			X						X				X		X			X	X	X	X	X					
MOONGR																	X					X					
NON1													X														
NON2			X			X							X														
NON3													X														
NON4													X														
NON5													X														
NON6													X														
PCES			X										X														
PREBLK				X	X																						
PRIORI									X	X	X		X			X							X				
RKST									X																		
RKT									X										X								
SCRTCH											X		X														
SETSW									X									X				X					
SIGMAC													X														
SSTEP																				X	X						
STANUM	X						X						X										X				
VRBLOK																						X					
WORKER									X						X			X	X	X	X						
XYZ																				X		X					
XYZOUT			X		X																						

Note... Common Blocks COWS, ERR22, and SSTEP are used by the Variable Step Program

5.3 LUNGFISH—GEOSTAR-I SOLVE COMMON Blocks

This section contains a detailed description of the COMMON areas used in the GEOSTAR-I SOLVE program.

/EARTH/

COMMON /EARTH/AE, FLAT

Variable	Type	Description	Program Where Defined	Program Where Used
AE	D	Semi-major axis of earth	MAIN	OPSTAT
FLAT	D	Flattening coefficient	MAIN	OPSTAT

/INPUT/

COMMON /INPUT/IOPT1, IOPT2, IOPT3, MEORG, MEPAR, MESUP,
MERED, MEDIC, MECOM, MEINV, NCMSUP, JCMSUP(500),
IDIN, BNAMIN(3), NBSUP, JBSUP(500), NARG

Variable	Type	Description	Program Where Defined	Program Where Used
IOPT1	I	This variable can have the values: (0)—check input data cards and B matrix tape; (1)—check input cards only; (2)—no checks will be made, for illegal values or format	MAIN	INCK
IOPT2	I	This variable can have the values: (0)—normal solve run (1)—process only one input B matrix	MAIN	INCK
IOPT3	I	This variable can have the values: (0)—intermediary matrices generated during a solution saved on tape #30. (1)—intermediary matrix to be printed	MAIN	INCK
MEORG	I	Indicates the type of edit requested for the original input B matrix	MAIN	INCK
MEPAR	I	Indicates the type of edit requested for the original input parameter set matrices	MAIN	INCK
MESUP	I	Indicates the type of edit requested for the suppressed matrices	MAIN	INCK
MERED	I	Indicates the type of edit requested for the reduced matrices	MAIN	INCK
MEDIC	I	Indicates the type of edit requested for the backsubstitution matrices	MAIN	INCK

/INPUT/ (continued)

Variable	Type	Description	Program Where Defined	Program Where Used
MECOM	I	Indicates the type of edit requested for the final combined matrix	MAIN	INCK
MEINV	I	Indicates the type of edit requested for the inverse of the final combined matrix	MAIN	INCK
NCMSUP	I	The number of parameters to be suppressed from the combined matrix	MAIN	MAIN INCK
JCMSUP(500)	I	Array of parameter labels to be suppressed from the combined matrix	MAIN	MAIN INCK
IDIN	I	Identification number of each B matrix used	MAIN	MAIN INCK
BNAMIN(3)	R	B matrix name	MAIN	MAIN
NBSUP	I	Number of parameters to be suppressed from each B matrix	MAIN	MAIN INCK
JBSUP(500)	I	Array of parameter labels to be suppressed from the current B matrix	MAIN	MAIN INCK
NARG	I	Total number of parameters to be suppressed from the current B matrix	MAIN	MAIN

NOTE

Before calling SUPRSS and LBLSUP, the total list of parameters to be suppressed is formed from the JBSUP and JCMSUP arrays; likewise for the variable NARG.

/LOCAL/

LOCAL is a common area used in each subroutine to reduce storage requirements of the program. Data transfer occurs only between the routines INVERT, MINV, EDIT, OPARC, OPGRAV, OPSTAT, where the array A(200, 200), a double precision array containing the matrix to be inverted and the inverted matrix, is transferred.

/PERM/

```
COMMON /PERM/B(500), BNEW(500), SIGTON, V1, V2, V3, V1C, V2C, V3C, IRT1, IDMAT,
        NROW, NCOL, NOB, ITYPE, BNAME(3), IRT2, LABS(500), LABSI(500),
        NRONI, LABSN(500), NROWN, NGRAV, NSTAT, NARC, JPAGE, LINE,
        NOREC, NOREC2, DBVS, ALPHA(8), IDCOMB, NROWC, NCOLC, NOBC,
        ITYPEC, BNAMEC(3), LABSC(501)
```

Variable	Type	Description	Program Where Defined	Program Where Used
B(500)	D	The right hand side of the normal equations	INVERT	MINV
BNEW(500)	D	The parameter set after backsubstitution	BACKSB	BACKSB
SIGTON	D	The predicted variance	MAIN	MAIN
V1, V2, V3 V1C, V2C, V3C	D	Arc variances and total variances after combining arcs	MAIN UPCOMB INVERT ELIM	MAIN MAIN
IRT1	I	Integer indicating the first record in a given matrix		
IDMAT	I	The ID number of a given matrix; on the header record of a matrix tape		
NROW	I	The number of rows in a given matrix; on the header record of a matrix tape		
NCOL	I	The number of columns in a given matrix; on the header record of a matrix tape		
NOB	I	The number of observations for each arc, and also for the combined arcs		

/PERM/ (continued)

Variable	Type	Description	Program Where Defined	Program Where Used
ITYPE	I	Matrix type identifier and used to indicate edit type		
BNAME(3)	R	An array containing the name of the B matrix being processed		
IRT2	D	Integer indicating the second record in a given matrix		
LABS(500)	I	An array containing the parameter labels	MAIN	UPCOMB SUPRSS LBLSUP
LABSI(500)	I	An array containing the parameter labels of the inverted matrix	INVERT	BACKSB
NROWI	I	Number of rows in the inverted matrix	INVERT	BACKSB
LABSN(500)	I	An array containing the parameter labels of the backsubstitution matrix	BACKSB	MAIN
NROWN	I	Number of rows in the back-substitution matrix	BACKSB	MAIN
NGRAV	I	Number of geopotential coefficient parameters	CALTYP	
NSTAT	I	Number of station position parameters	CALTYP	
NARC	I	Number of arc dependent parameters	CALTYP	
JPAGE	I	Page number of printed output	CHECK	
LINE	I	Line number of printed output	CHECK	

/PERM/ (continued)

Variable	Type	Description	Program Where Defined	Program Where Used
NOREC, NOREC2	I	Integers used to control backspacing of tapes	MAIN	BEDIT
OBVS	R	Set equal to NOB to compute sigma	MAIN	MAIN
ALPHA(8)	R	An array containing alphanumeric run identification printed at the head of each page of output	MAIN	CHECK
IDCOMB	I	ID number of the combined matrix	MAIN	MAIN COMB INCK
NROWC	I	Number of rows in combined matrix	UPCOMB	COMB
NCOLC	I	Number of columns in combined matrix	UPCOMB	COMB
NOBC	I	The total number of observations over all arcs used	UPCOMB	COMB
ITYPEC	I	The ITYPE value for the output combined matrix	MAIN	COMB
BNAMEC(3)	R	Output name of the combined matrix	MAIN	COMB
LABSC(501)	I	The label array for the combined matrix	UPCOMB	ELIM COMB

/SOGMA/

COMMON /SOGMA/XA(500)

Variable	Type	Description	Program Where Defined	Program Where Used
XA(500)	D	An array containing the standard deviations of the parameters	INVERT	OPARC OPSTAT OPGRAV

5.4 LUNGFISH—GEOSTAR-I SOLVE COMMON Block Cross Reference Table

The following table details the COMMON area structure of the GEOSTAR-I SOLVE program.

		SUBROUTINES																			
		MAIN	BACKSB	BEDIT	CALTYP	CHECK	COMB	EDIT	ELIM	ERROR	INCK	INVERT	LBLSUP	MATSUP	MINV	OPARC	OPGRAV	OBSTAT	SORTX	SUPRSS	UPCOMB
COMMON BLOCKS	EARTH	X																X			
	INPUT	X									X					X					
	LOCAL	X	X	X			X	X	X		X	X	X	X	X	X	X	X	X	X	X
	PERM	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
	SOGMA	X										X				X	X	X			

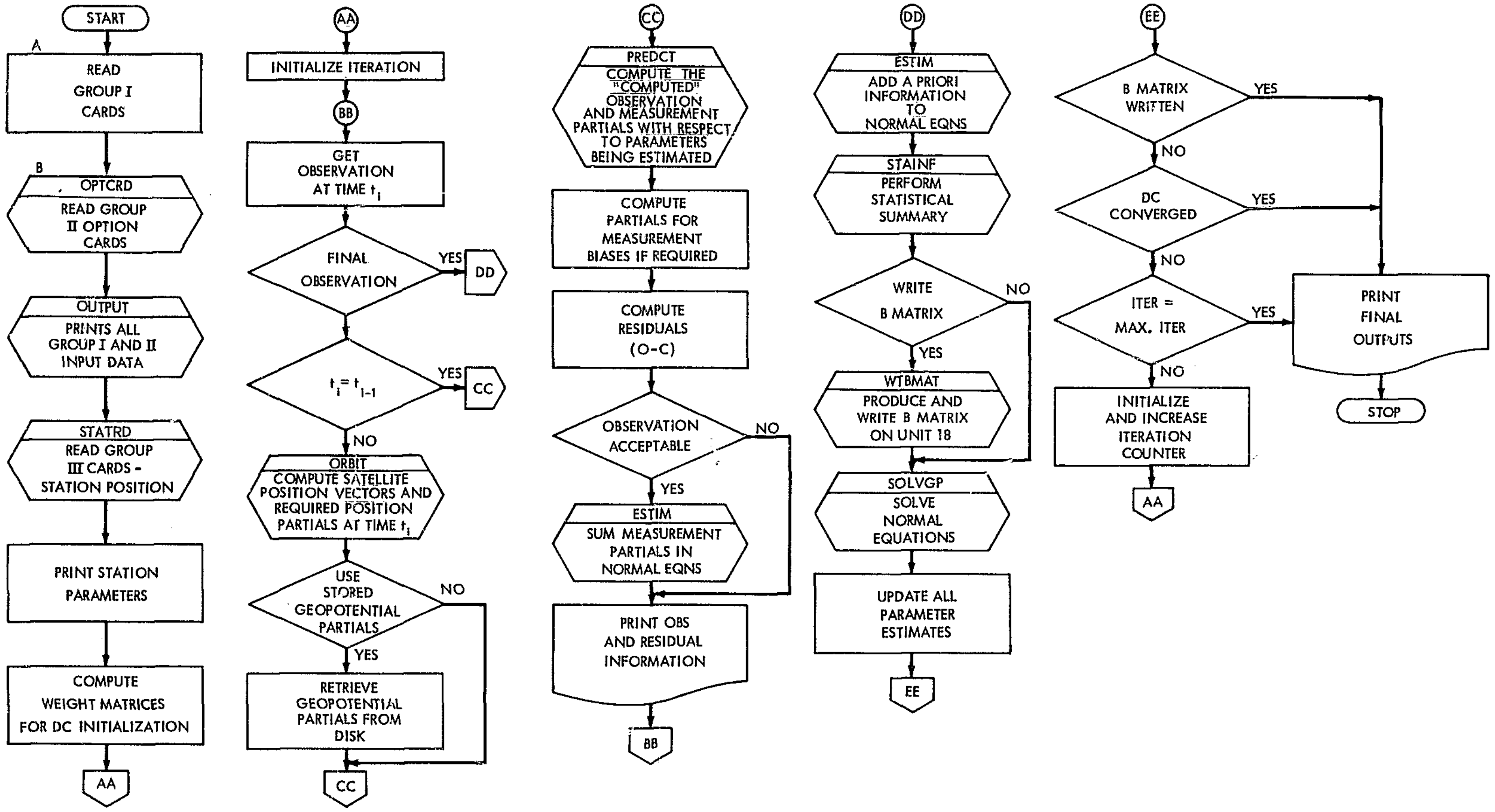
VI. PROGRAM FLOW CHARTS AND SUBROUTINE SUMMARIES

The following sections contain the flow diagrams of the MAIN or control routines in the GEOSTAR-I ODP and SOLVE programs. A summary of all the modules used in these programs and cross reference tables detailing the subroutine structure is also presented.

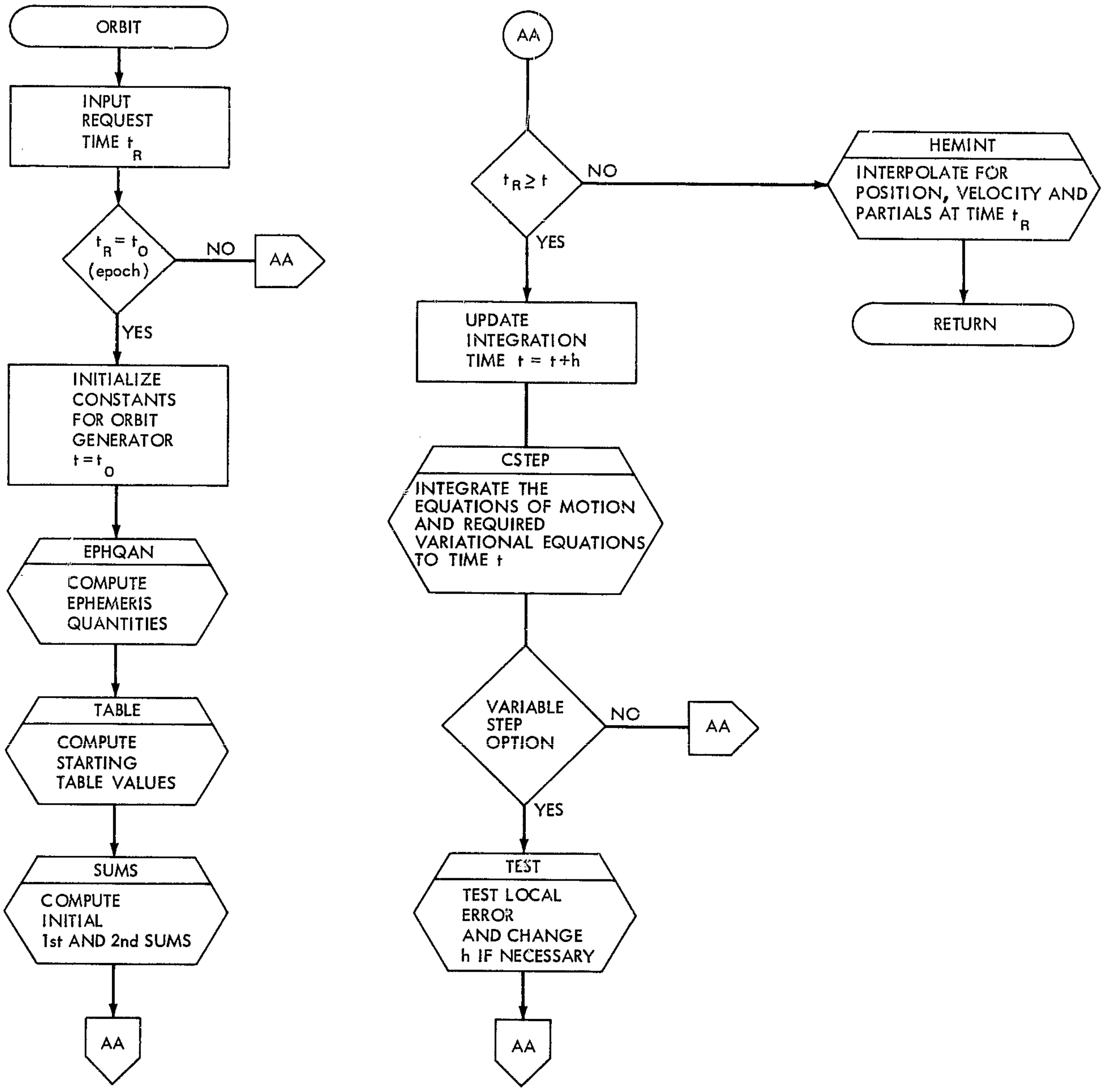
6.1 GEOSTAR-I ODP Flow Diagrams

This section contains detailed flow diagrams of the GEOSTAR-I ODP executive program MAIN and the orbit generator control subprogram ORBIT.

MAIN PROGRAM


START
A
READ GROUP I CARDS
B
OPTCRD
READ GROUP II OPTION CARDS
OUTPUT
PRINTS ALL GROUP I AND II INPUT DATA
STATRD
READ GROUP III CARDS - STATION POSITION
PRINT STATION PARAMETERS
COMPUTE WEIGHT MATRICES FOR DC INITIALIZATION
AA
AA
INITIALIZE ITERATION
BB
GET OBSERVATION AT TIME t_i
FINAL OBSERVATION
YES
DD
$t_i = t_{i-1}$
YES
CC
NO
ORBIT
COMPUTE SATELLITE POSITION VECTORS AND REQUIRED POSITION PARTIALS AT TIME t_i
USE STORED GEOPOTENTIAL PARTIALS
NO
YES
RETRIEVE GEOPOTENTIAL PARTIALS FROM DISK
CC
CC
PREDCT
COMPUTE THE "COMPUTED" OBSERVATION AND MEASUREMENT PARTIALS WITH RESPECT TO PARAMETERS BEING ESTIMATED
COMPUTE PARTIALS FOR MEASUREMENT BIASES IF REQUIRED
COMPUTE RESIDUALS (O-C)
OBSERVATION ACCEPTABLE
NO
YES
ESTIM
SUM MEASUREMENT PARTIALS IN NORMAL EQNS
PRINT OBS AND RESIDUAL INFORMATION
BB
DD
ESTIM
ADD A PRIORI INFORMATION TO NORMAL EQNS
STAINF
PERFORM STATISTICAL SUMMARY
WRITE B MATRIX
NO
YES
WTBMAT
PRODUCE AND WRITE B MATRIX ON UNIT 18
SOLVGP
SOLVE NORMAL EQUATIONS
UPDATE ALL PARAMETER ESTIMATES
EE
EE
B MATRIX WRITTEN
YES
NO
DC CONVERGED
YES
NO
ITER = MAX. ITER
YES
NO
INITIALIZE AND INCREASE ITERATION COUNTER
AA
PRINT FINAL OUTPUTS
STOP

ORBIT SUBROUTINE


ORBIT
INPUT REQUEST TIME t_R
$t_R = t_0$ (epoch)
NO
AA
YES
INITIALIZE CONSTANTS FOR ORBIT GENERATOR $t = t_0$
EPHQAN
COMPUTE EPHEMERIS QUANTITIES
TABLE
COMPUTE STARTING TABLE VALUES
SUMS
COMPUTE INITIAL 1st AND 2nd SUMS
AA
AA
$t_R \geq t$
NO
YES
HEMINT
INTERPOLATE FOR POSITION, VELOCITY AND PARTIALS AT TIME t_R
RETURN
UPDATE INTEGRATION TIME $t = t+h$
CSTEP
INTEGRATE THE EQUATIONS OF MOTION AND REQUIRED VARIATIONAL EQUATIONS TO TIME t
VARIABLE STEP OPTION
NO
AA
YES
TEST
TEST LOCAL ERROR AND CHANGE h IF NECESSARY
AA

6.2 GEOSTAR-I ODP Subroutine Summary

This section contains a brief summary of all modules used in the GEOSTAR-I ODP. Modules of the original NONAME system are included as well as those developed for GEOSTAR-I ODP. Complete documentation of the NONAME modules can be found in Reference 1.

ADDYMD Adds or subtracts an integral number of days from a date packed into one word in the form YYMMDD giving a new packed date.

AND2 A function which determines the logical "AND" of two arguments, each two bytes long. The FORTRAN function LAND is called.

APPER Computes apogee and perigee heights in kilometers given the Keplerian elements, radius of the earth, and inverse of flattening.

BLKSTA Sets up the working arrays of station positions merging initial values, prestored via the BLOCK DATA subprogram with card input station positions. The arrays are ordered with stations to be estimated first.

CLEAR Stores zeros in any two dimensional REAL*4 array.

CKDIFF* Computes the kth backward difference which is needed to estimate the local truncation error in Runge-Kutta generated points.

COEFF Estimates coefficients of a polynomial of a requested degree (maximum 7th degree) by the method of least squares using a maximum of 20 observations.

COEFL Prints non-zero coefficients of the spherical harmonic expansion of the geopotential model used by the system.

CSTEP Solves a set of 1st and 2nd order, ordinary differential equations using the summed Störmer/Adams-Bashforth predictor and the Cowell/Adams-Moulton corrector.

DARCTN Function returns the arc-tangent of (Y/X) in radians, between 0 and 2π. The FORTRAN function DATAN2 is called.

DATES Converts days elapsed from Jan. 0.0 of an input reference year into a three word date of the form YYMMDD, HHMM, SEC.

DAYEAR Converts days elapsed from Jan. 0.0 of the reference year into two words of integral number of days and integral number of seconds.

DENMUL Evaluates a 3rd degree polynomial given the coefficients of the polynomial and a value of the independent variable.

*Subroutines used only in the variable stepsize version of GEOSTAR-I ODP.

DENSTY A function which computes atmospheric density dependent on height and temperature. Temperature is derived from the Jacchia-Nicolet model considering: solar activity, semiannual variation, diurnal bulge, and geomagnetic activity.

DIFF Calculates difference between any two dates in the 20th centry, (input-dates input in two words of the form YYMMDD, HHMMSS; output-integral days and seconds of a day).

DINRAD Converts angles in arc measurements or time measurements to radians (input: integer degrees or hours, integer minutes, REAL * 8 seconds).

DJUL Converts input days since Jan. 0.0 of reference year to an adjusted Julian date (reference Jan. 0, 1950).

DNVERT Double precision matrix inversion using Gauss-Jordan method of condensation with partial (column) pivoting. No restrictions on dimension of matrix.

DODSRD Reads and processes observation data in the DODS system format to set up the system working arrays of input observations.

DOTPRD A function which computes the dot product of two three-dimensional vectors.

DRAG Computes accelerations in rectangular coordinates on a satellite due to drag forces.

EGRAV Computes acceleration in rectangular coordinates on a satellite due to geopotential forces, and acceleration partials (forcing functions) in rectangular coordinates due to geopotential coefficients.

ELEM Converts inertial position and velocity vector to osculating orbital elements. Inputs and outputs are in COMMON.

ELEMK Converts orbital elements as ELEM with input and output in the argument list.

EPHQAN Executive routine calling various subprograms which evaluate the three components of the moon's inertial unit vector and geocentric distance, the sun's inertial unit vector and geocentric distance, and the equation of equinoxes.

EQN Computes nutation in longitude, obliquity and right ascension; true obliquity of date.

EQUATR Transforms the rectangular coordinates of a vector referenced to either the mean or true equator and equinox of one epoch to either the mean or true equator and equinox of another epoch. The subroutines, PRECES and NUTATE are called.

ERROR Provides printed messages for error conditions.

ESTIM Sums all measurement partials into the normal equations matrix and right hand sides for each observation data point. When all observations have been processed, a priori parameter weights (input to the system as parameter sigmas or a full weight matrix) are applied.

FRCS Executive routine calling various subprograms which evaluate accelerations in rectangular coordinates on a satellite due to the various forces.

FIT Evaluates polynomial functions given the coefficient matrix and a value of the independent variable. There is no restriction on the maximum degree or number of the polynomials to be evaluated.

GDET Evaluates the determinant of a matrix (reduction to diagonal form using elementary row and column operations). There is no restriction on the dimension of the matrix.

GEOSRD Reads and processes input observation data in the NASA Science Data Center format and sets up the working arrays of observation data points. Data may be selected by measurement type, and/or station and/or time period.

GTIMIN, GTIMOT Measures the elapsed time of each iteration and computes the total elapsed time of a run in hundredths of seconds.

HEMINT Interpolates for position, velocity and position partials at a time between equally spaced points, using the Hermite interpolation formula.

INV.2/INV3 Inserts a matrix by using the Gauss-Jordan method of condensation with partial (column) pivoting.

MMATRX Multiplies an n × n matrix with an n × m matrix where n need not equal m.

MULMAT Multiplies two input matrices.

NUMBER Searches the entries of members of an array and compares with an input number or bit configuration. Index number of the array member whose entry is matched is returned. Zero is returned if no match is found. It is employed as a device to avoid excessive use of DO loops.

NUTATE Generates the nutation angles to transform from mean equator and equinox to true equator and equinox.

OBSDOT Calculates time derivatives of requested observation types. It is used in the computation of time biases.

OPTCRD Reads cards which redefine stored values of earth, satellite, and integration parameters as well as initializing various program options.

ORB1 Provides control to generate an ephemeris tape in the ORB1 format. ORBIT is called for the ephemeris points.

ORBIT Executive program which receives a state vector and its epoch, initializes required constants, and utilizes an integrator subprogram and an interpolation subprogram to find the state vector at a new epoch, as well as the position partials of any physical parameters being estimated.

OUTPUT Prints values of the earth, satellite, and integration parameters.

OUTRAD Converts radians to degrees or hours, minutes and seconds.

PLHOUT Converts a tracking station location and variance-covariance matrix in geodetic rectangular coordinates to geodetic latitude, longitude and height coordinates.

POSVEL Converts osculating orbital elements to inertial position and velocity vectors. Input and output is in COMMON.

PRECES Generates the angles for precession from mean equator and equinox of one epoch to mean equator and equinox of another. The year 1950 is used as a base year.

PREDCT

Converts inertial position and velocity to a computed observation required for the residuals (O-C). The partials of an observation with respect to state or parameters at epoch are computed as:

$$\frac{\partial M}{\partial x_0} = \frac{\partial M}{\partial x_t}\frac{\partial x_t}{\partial x_0}$$

where:

M is an input observation measurement at time t,

x represents state or parameters to be estimated in the DC, where the subscript zero denotes epoch time,

$\frac{\partial M}{\partial x_t}$ is computed in PREDCT,

$\frac{\partial x_t}{\partial x_0}$ is computed in ORBIT.

PRNTPR

Provides printed information when preprocessing of input observation data is requested.

PROCESS

Preprocesses a requested observation type.

READGP

Reads the input data cards and sets up working arrays for the geopotential estimation problem.

REFCOR

Transforms vectors to mean equator and equinox of reference year. The subroutine PRECES is called.

REFIMP

Computes nominal ionospheric refraction correction for range and range rate data.

RK

Solves a set of 1st order ordinary differential equations using an 8th order Runge-Kutta integration method.

ROTMAT

Generates the rotation matrix given the rotation angle and axis.

RYMDI

Separates date packed into one word in the form YYMMDD into the three words YY, MM, and DD.

SATCLC

Utilizes known differences in time between the GEOS-I satellite clock and UTC, and, assuming that the input time is the recorded UTC time of an observed active flash observation, calculates the time difference between the UTC of the actual flash and the input UTC.

SATCL2

Serves the same purpose as SATCLC for GEOS-2.

SDOLL Used with NASA Data Center format tapes to provide for the selection of input observation data. Reads up to 100 SELECT or DELETE cards, then reads records of observational data from an input tape or disk and writes these records onto a second tape or disk unit, selecting or deleting records by time interval, and/or station and/or type of observation as determined by the SELECT or DELETE cards.

SDTPRD Computes the single precision dot product of two three-dimensional vectors.

SOLVGP Computes the parameter corrections using matrix inversion given the weighted normal equations as produced by ESTIM. A gradient option is provided as an alternate method if normal matrix inverson is inadequate.

SQUANT Used in the initialization phase of a DC run to convert geodetic spherical coordinates (geodetic latitude, longitude and height above computational spheroid) to geodetic rectangular coordinates of a tracking station, and components of the station Zenith, East and North unit vectors for an array of stations (maximum 50). Also computes on first pass matrix of partial derivatives of geodetic rectangular coordinates with respect to geodetic spherical coordinates for tracking stations whose coordinates are to be adjusted (maximum 10). On subsequent calls to the program, geodetic rectangular coordinates of the stations being adjusted together with their variance-covariance matrices are transformed to geodetic spherical coordinates. Their local East, Zenith and North unit vectors are recomputed on the basis of the adjusted positions.

STAINF Computes statistical information at the end of each DC iteration; included are residual summaries by station and data tape.

STATRD Redefines and prints both the geodetic spherical coordinates and the geodetic rectangular coordinates of the stations (maximum 50).

STORGP Sets up the working arrays of the geopotential model used by VEVAL.

SUMS Computes the first and second sums necessary for the summed form of the predictor-corrector multistep integration formulas.

SUMTOB Provides for the shift of a specified matrix row and array variable in the subprogram WTBMAT.

SUN

Computes the solar position unit vector components and radius vector and the mean equinox and ecliptic of date.

SUNGRV

Computes the acceleration in rectangular coordinates due to solar and lunar gravity.

SWTEST

Computes indices which identify the forces being applied to the equations of motion and the forces being applied to the variational equations.

SYMMET

Computes the elements of given square matrix below the main diagonal on the assumption that the matrix is symmetrical.

TABLE

Executive routine calling other subprograms to produce the initial table of starting values for the Cowell integrator.

TABLEB*

Produces a new table of starting values for the Cowell integrator whenever a stepsize change occurs.

TDIF

A function that computes differences between the time systems A1, UTC, UT2 and UT1.

TEST*

Computes the local truncation error and tests this error against a set of tolerances to determine if an increase or decrease of the integration stepsize is necessary. If a stepsize change occurs, this subprogram computes the new stepsize and starting values by calling TABLEB.

VCONV

Converts a variance-covariance, VIN, matrix from one system to another, VOUT, by computing the matrix product $B = P^T AP$ from input matrices A and P where A and P are both 3 × 3. It is used for station information where:

B = VOUT, in spherical coordinates
A = VIN, in rectangular coordinates
P = Partials of VOUT variables with respect to VIN variables.

VEVAL

Computes the partials of a satellite's acceleration vector with respect to inertial position and velocity vectors and physical parameters to be estimated. The force model includes geopotential harmonics through 4th order zonal and 3rd order tesseral harmonics, drag, lunar gravity, solar gravity, and solar radiation.

*Subroutines used only in the variable stepsize version of GEOSTAR-I ODP.

WTBMAT

Used in multi-arc runs to reorder the normal equations and right hand sides, and to provide a parameter list and labels in the B matrix format (see Appendix B).

XEFIX, YEFIX,
XINERT, YINERT

Functions that convert earth-fixed rectangular coordinates to inertial rectangular coordinates and vice versa, specifically:

XEFIX - given inertial X and Y return earth fixed X

YEFIX - given inertial X and Y return earth fixed Y

XINERT - given earth fixed X and Y return inertial X

YINERT - given earth fixed X and Y return inertial Y.

YMDAY

Computes days elapsed from Jan. 0.0 of a reference year to input date in form YYMMDD, HHMM, SEC. Reference year is set by program to be equal to year of input data on first call to program.

6.3 GEOSTAR-I ODP Subroutine Cross Reference Tables

This section contains cross reference tables detailing the subroutine structure in the GEOSTAR-I ODP executive program MAIN and the orbit generation control subprogram ORBIT.

MAIN Program Cross Reference Table

CALLING SUBROUTINES

CALLED SUBROUTINES	MAIN	BLKSTA	COEFF	CSTEP	DATES	DAYEAR	DENSTY	DIFF	DJUL	DODSRD	DRAG	ELEM	EPHQAN	EQUATR	FRCS	GEOSRD	NUTATE	OBSDOT	OPTCRD	ORB1
ADDYMD					X															
AND2										X										
APPER	X																			
COEFF													X							
COEFL	X																			
DARCTN										X										
DATES	X					X														X
DAYEAR																				X
DENMUL							X													
DENSTY											X									
DIFF	X																			
DINRAD																X				
DJUL													X							X
DODSRD	X																			
DOTPRD	X														X					X
DRAG															X					
EGRAV															X					
ELEM	X																			
ELEMK	X																			
EQN													X				X			X
EQUATR										X						X				
ERROR	X											X							X	
ESTIM	X																			
FRCS				X																
GDET	X																			
GEOSRD	X																			
GTIMIN	X																			
GTIMOT	X																			
HEMINT				X																
INV2				X																
INV3	X		X																	
MMATRX				X																
MOONAD													X							
MULMAT																	X			
NUMBER	X	X								X						X			X	
NUTATE														X						
OBSDOT	X																			
OPTCRD	X																			
ORB1	X																			
ORBIT	X																			X
OUTPUT	X																			
POSVEL	X																			
PRECES														X						
PREDCT	X																			
READGP																			X	
REFCOR															X					
ROTMAT																	X			
RYMD1								X												X
SATCL2																X				
SATCLC																X				
SDOLL																X				
SDTPRD																		X		
SOLVGP	X																			
SQUANT	X																			
STAINF	X																			
STATRD	X																			
STORGP	X																			
SUN													X							
SUNGRV															X					
SYMMET																			X	
TDIF	X				X					X			X			X			X	X
VEVAL				X																
WTBMAT	X																			
XEFIX																		X		
XINERT																		X		
YEFIX																		X		
YINERT																		X		
YMDAY	X								X	X						X	X		X	

MAIN Program Cross Reference Table (Continued)

CALLING SUBROUTINES

CALLED SUBROUTINES	ORBIT	PLHOUT	PRECES	PREDCT	PROCES	REFCOR	REFIMP	RK	SATCL2	SATCLC	SOLVGP	SQUANT	STAINF	STATRD	TABLE	TABLEB	TDIF	TEST	VEVAL	WTBMAT	YMDAY
BLKSTA														X							
CKDIFF																		X			
CLEAR													X								
COEFL														X							
CSTEP	X																				
DARCTN				X																	
DATES														X							
DENSTY																			X		
DIFF																					X
DINRAD												X									
DNVRT1											X										
DOTPRD				X	X														X		
EGRAV																			X		
EPHQAN	X																				
FIT				X																	
FRCS	X							X								X					
HEMINT	X															X					
MULMAT			X			X															
NUMBER							X							X							
NUTATE						X															
OBSDOT					X																
OUTRAD												X									
PLHOUT												X		X							
PRECES						X															
PRNTPR														X							
PROCES				X																	
REFCOR	X																		X		
REFIMP					X																
RK															X						
ROTMAT			X																		
SQUANT														X							
SUMS	X															X					
SUMTOB																				X	
SWTEST	X																				
SYMMET											X									X	
TABLE	X																				
TABLEB																		X			
TEST	X														X						
VCONV		X												X							
VEVAL	X							X								X					
XEFIX				X																	
XINERT				X																	
YEFIX				X																	
YINERT				X																	
YMDAY			X						X	X							X				

Note... Subroutines CKDIFF, TABLEB, and TEST are used by the Variable Step Program

ORBIT Subprogram Cross Reference Table

CALLING SUBROUTINES

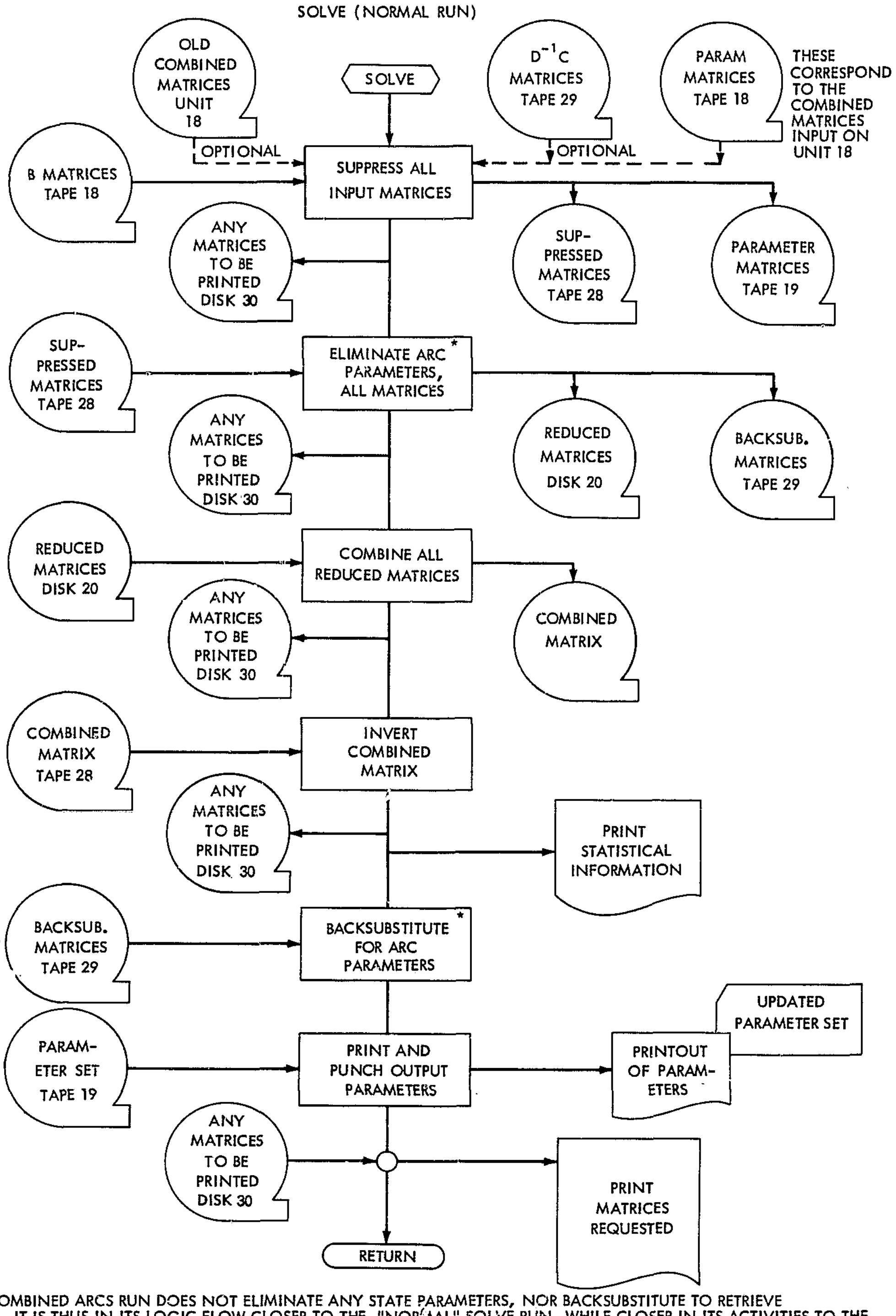
CALLED SUBROUTINES	ORBIT	COEFF	CSTEP	DENSTY	DIFF	DJUL	DRAG	EPHQAN	FRCS	NUTATE	PRECES	REFCOR	RK	TABLE	TABLEB	TEST	VEVAL	YMDAY	
CKDIFF																X			
COEFF								X											
CSTEP	X																		
DENMUL				X															
DENSTY							X										X		
DIFF																		X	
DJUL								X											
DOTPRD									X								X		
DRAG									X										
EGRAV									X								X		
EPHQAN	X																		
EQN								X		X									
FRCS	X		X										X		X				
HEMINT	X		X												X				
INV2			X																
INV3		X																	
MMATRX			X																
MOONAD								X											
MULMAT										X	X	X							
NUTATE												X							
PRECES												X							
REFCOR	X								X								X		
RK														X					
ROTMAT										X	X								
RYMDI					X														
SUMS	X														X				
SUN								X											
SUNGRV									X										
SWTEST	X																		
TABLE	X																		
TABLEB																X			
TDIF								X											
TEST														X					
VEVAL	X		X										X		X				
YMDAY						X				X	X								

Note... Subroutine CKDIFF, TABLEB, and TEST are used by the Variable Step Program

6.4 GEOSTAR-I SOLVE Flow Diagram

This section contains detailed flow diagrams of the GEOSTAR-I SOLVE executive programs in the normal execution mode and in the "invert only" mode.

SOLVE (NORMAL RUN)


SOLVE
OLD COMBINED MATRICES UNIT 18
OPTIONAL
B MATRICES TAPE 18
SUPPRESS ALL INPUT MATRICES
D^{-1}C MATRICES TAPE 29
OPTIONAL
PARAM MATRICES TAPE 18
THESE CORRESPOND TO THE COMBINED MATRICES INPUT ON UNIT 18
ANY MATRICES TO BE PRINTED DISK 30
SUP-PRESSED MATRICES TAPE 28
PARAMETER MATRICES TAPE 19
SUP-PRESSED MATRICES TAPE 28
ELIMINATE ARC PARAMETERS, ALL MATRICES *
ANY MATRICES TO BE PRINTED DISK 30
REDUCED MATRICES DISK 20
BACKSUB. MATRICES TAPE 29
REDUCED MATRICES DISK 20
COMBINE ALL REDUCED MATRICES
ANY MATRICES TO BE PRINTED DISK 30
COMBINED MATRIX
COMBINED MATRIX TAPE 28
INVERT COMBINED MATRIX
ANY MATRICES TO BE PRINTED DISK 30
PRINT STATISTICAL INFORMATION
BACKSUB. MATRICES TAPE 29
BACKSUBSTITUTE FOR ARC PARAMETERS *
UPDATED PARAMETER SET
PARAM-ETER SET TAPE 19
PRINT AND PUNCH OUTPUT PARAMETERS
PRINTOUT OF PARAM-ETERS
ANY MATRICES TO BE PRINTED DISK 30
PRINT MATRICES REQUESTED
RETURN

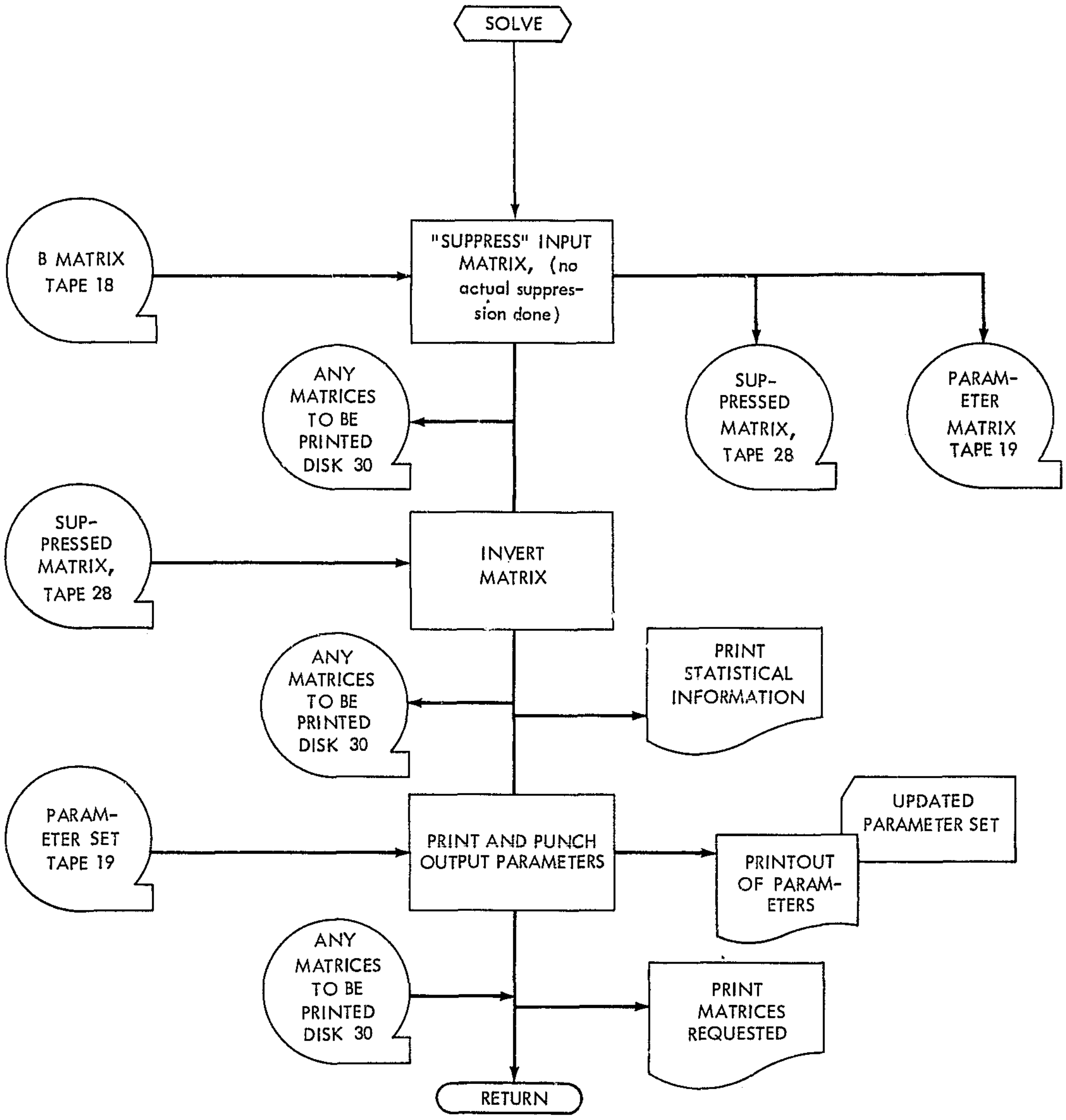
THIS TAPE CAN BE RE-INTRODUCED AS INPUT IN A LATER RUN IF A RELATED CB MATRIX AND PARAMETER SET ARE INTRODUCED AS WELL.
USE TAPE 29

THIS TAPE CAN BE RE-INTRODUCED AS INPUT IN A LATER RUN IF A RELATED CB MATRIX AND D^{-1}C MATRIX ARE INTRODUCED AS WELL.
USE TAPE 19

*NOTE: THE COMBINED ARCS RUN DOES NOT ELIMINATE ANY STATE PARAMETERS, NOR BACKSUBSTITUTE TO RETRIEVE THEM. IT IS THUS IN ITS LOGIC FLOW CLOSER TO THE "NORMAL" SOLVE RUN, WHILE CLOSER IN ITS ACTIVITIES TO THE INVERT ONLY RUN. THE COMBINED ARCS RUN BORROWS FROM BOTH.

THE FINAL COMBINED ARCS MATRIX HAS NO CORRESPONDING D^{-1}C MATRICES, AND HAS ITS PARAMETER SET WRITTEN OUT AFTER THE MATRIX ON THE SAME TAPE. HENCE, TO REINTRODUCE THE COMBINED ARCS MATRIX INTO A LATER RUN, TREAT IT AS A SELF-CONTAINED B MATRIX.

SOLVE (invert only)


SOLVE
B MATRIX TAPE 18
"SUPPRESS" INPUT MATRIX, (no actual suppression done)
SUP-PRESSED MATRIX, TAPE 28
PARAM-ETER MATRIX TAPE 19
ANY MATRICES TO BE PRINTED DISK 30
SUP-PRESSED MATRIX, TAPE 28
INVERT MATRIX
ANY MATRICES TO BE PRINTED DISK 30
PRINT STATISTICAL INFORMATION
PARAM-ETER SET TAPE 19
PRINT AND PUNCH OUTPUT PARAMETERS
UPDATED PARAMETER SET
PRINTOUT OF PARAM-ETERS
ANY MATRICES TO BE PRINTED DISK 30
PRINT MATRICES REQUESTED
RETURN

6.5 GEOSTAR-I SOLVE Subroutine Summary

This section contains a brief summary of all the modules used in the GEOSTAR-I SOLVE. Modules of the original LUNGFISH system are included as well as those developed for the GEOSTAR-I system. A more complete documentation of the LUNGFISH modules can be found in Reference 2.

ANDREE Computes the Penrose pseudoinverse of a given matrix. The computational rank of the given matrix is also determined.

BACKSB Solves the backsubstitution equations for the arc dependent parameters using the backsubstitution matrices generated during the elimination process, the associated right hand sides, and the arc independent parameter solution set.

BEDIT Writes the matrices to be edited onto unit 30 together with an edit format code.

CALTYP Calculates the number of gravity coefficient, station position and arc dependent parameter labels in a given SOLVE run.

CHECK Checks the number of lines printed out on any given page, and if enough lines have been written, a new page is begun, headed by the alphanumeric input of data card 1.

COMB Performs the matrix combination of the reduced matrices resulting from arc dependent elimination, and writes the resultant combined matrix on unit 28.

DARCTN Function returns the arc tangent of Y/X in radians between 0 and 2π. The FORTRAN function DATAN2 is called.

EDIT Prints the matrices to be edited (stored on unit 30) in the required edit format.

ELIM Performs a Gauss-Jordan elimination of the arc dependent parameters from a given matrix, producing a reduced matrix containing only arc independent parameters and a backsubstitution matrix for the eliminated parameters.

ERROR Prints an error message identifying a particular error condition and terminates the run.

GNORM Normalizes a given gravity coefficient.

INCK Checks the format of the various input options, values, and matrices. If these formats are not correct, ERROR is called.

INVERT Control routine for matrix inversion. Calls for either inversion or pseudoinversion of a given matrix.

LBLSUP Sets the labels corresponding to suppressed parameters to zero.

MATSUP Performs the matrix suppression by deleting rows and columns and writes the resulting suppressed matrix on unit [illegible].

MINV Inverts and solves a matrix equation using a Gauss-Jordan elimination method, with a pivot search on the diagonal elements.

OPARC Prints and punches the new estimates of the arc dependent parameters.

OPSTAT Prints and punches the new estimates of the station position parameters. Output is in both geodetic rectangular and spherical coordinates.

OUTRAD Converts radians to degrees, minutes and seconds.

PHLINN Converts station coordinates from geodetic rectangular coordinates to geodetic latitude, longitude and height above spheroid.

SORTX Sorts the gravity parameter labels and values so that all the C_{nm} geopotential parameters are in ascending order immediately preceding the S_{nm} geopotential parameters.

SUPRSS Control program for matrix suppression.

UPCOMB Produces the combined matrix identification and label records.

6.6 GEOSTAR-I SOLVE Subroutine Cross Reference Tables

The following table details the subroutine structure of the GEOSTAR-I SOLVE program.

	CALLING SUBROUTINES														
CALLED SUBROUTINES	MAIN	BACKSB	BEDIT	COMB	EDIT	ELIM	INCK	INVERT	MINV	OPARC	OPSTAT	OPGRAN	PHLINV	SUPRSS	UPCOMB
ANDREE								X							
BACKSB	X														
BEDIT	X														
CALTYP	X						X			X	X	X		X	X
CHECK	X				X		X		X	X	X	X			
COMB	X														
DARCTN													X		
EDIT	X														
ELIM	X														
ERROR	X	X	X	X	X	X	X			X					
GNORM												X			
INCK	X														
INVERT	X														
LBLSUP														X	
MATSUP														X	
MINV					X			X							
OPARC	X														
OPGRAV	X														
OPSTAT	X														
OUTRAD											X				
PLHINN											X				
SORTX												X			
SUPRSS	X														
UPCOMB						X									

REFERENCES

1. NONAME, An Orbit and Geodetic Parameter Estimation System, Vol. I-V, Prepared by Wolf R&D Corp. under NASA Contract No. NAS5-9756-71D, August 1968.

2. LUNGFISH, Lunar Gravitational Field in Spherical Harmonics, Books I-IV, Prepared by Computer Usage Co. Inc. under NASA Contract No. NAS1-4998, February 1966.

3. Finley, G., "Operating Guide GEOSTAR-M (multi-arc), GEOSTAR-I System, OLGA SYSTEM," Prepared by IBM Corp. under NASA Contract No. NAS5-10022, February 1970.

4. Douglas, B. and Williamson, R., "Final Report For the Harmonic Analysis of Perturbations Program (HAP)," Wolf R&D Corp. under NASA Contract No. 5-9756-147.

5. Rauch, H., "Advanced Orbit Mechanics Studies - Differential Correction and Least Squares Estimation," Lockheed Missiles and Space Co. Report No. C-64-67-1, March 1967.

6. Lefferts, E., "Least Squares and Pseudoinversion," NASA X-Document X-551-69-73, September 1968.

7 Deutsch, R., "Estimation Theory," Prentice Hall, 1965.

8. Maury, J. and Brodsky, G., "Cowell Type Numerical Integration As Applied To Satellite Orbit Computation," NASA X-Document X-533-59-46, April 1969.

9. Shanks, E., "Solution of Differential Equations by Evaluation of Functions," Math. of Computation, Vol. 20, pp. 21-38, 1966.

10. Anderle, R., "Computational Methods Employed in Deriving Geodetic Results from Doppler Observations of Artificial Earth Satellites," NWL Report No. 1977, April 1965.

11. "Specifications for R-Matrix Processing," Report Prepared by Computer Usage Co. Inc. for the Astronautics Branch of the Computation and Analysis Laboratory, U. S. Naval Weapons Laboratory, Dahlgren, Virginia. Contract No. N178-67-C-0030, March 1967.

Appendix A

Variable Stepsize Version of GEOSTAR-I ODP

New and Updated Modules in GEOSTAR-I ODP

This section describes the modifications and additions made to the subroutine structures of the GEOSTAR-I ODP for the variable stepsize version of the GEOSTAR-I ODP program.

To obtain the variable stepsize version, the following GEOSTAR-I ODP modules were modified:

MAIN
OPTCRD
OUTPUT
BLOCK DATA
ERROR
ORBIT
CSTEP
TABLE
FRCS
HEMINT
RK
SUMS
SWTEST

In addition, the following modules were developed specifically for the variable stepsize GEOSTAR-I ODP:

CKDIFF
TABLEB
TEST

It should be noted that this modified version of GEOSTAR-I ODP has all the capabilities of the original GEOSTAR-I ODP, including the multiple arc capability. The single exception is that the modified program can only solve for a total of 20 unknowns, as opposed to 50 for the original version.

In the following sections, those modules listed above which are new, or significantly modified existing subroutines will be documented in detail. The remaining modules which received only minor modifications are briefly outlined, with changes indicated.

A.1 Modifications to Existing GEOSTAR-I ODP Modules

The modifications made to the GEOSTAR-I ODP are designed to·

- Call the new subroutines (see Sec. 6.1 Orbit Flow)
- Extend computational algorithm to allow for stepsize modification in the integration of the equations of motion and the variational equations
- Extend tables to provide space for storing position', velocity and position partial vectors to allow for stepsize modification.

A summary of these modules, as modified for the variable stepsize GEOSTAR-I ODP, follows:

- MAIN — The ODP control program. Modified to determine when to use the variable stepsize option.
- FRCS
  HEMINT
  RK
  SUMS
  SWTEST — Modified to extend the position, velocity, and position partial vectors tables to allow for stepsize modification.
- OPTCRD
  OUTPUT
  ERROR — Modified to include the I/O required for the stepsize modification process.
- INPUT — Modified to extend the available program options to include variable stepsize and new data base constants required for this new option.
- ORBIT — The integration control program. Modified to control the new sub–routine TEST as well as the variable stepsize option logic.
- TABLE — The control program for the initial table of starting values to be used in the Cowell integrator. Modified to halve or double the stepsize of both the equations of motion and the variational equations if necessary. In addition, print statements have been included to indicate a stepsize change.
- CSTEP — The subroutine that solves a set of 1st and 2nd order differential equations using the predictor-corrector technique. Modified to save the predicted and corrected values of position which are used for local error computation.

A.2 New Variable Stepsize ODP Modules

CKDIFF

Purpose:

To compute the kth backward difference of the accelerations which is needed to estimate the local truncation error of the initial starting table for the equations of motion.

Called By:

TEST

Method:

To compute the kth difference of a table of function values, the formula

$$\nabla^k f(x) = \sum_{i=0}^{k} (-1)^i \binom{k}{i} f(x-ih)$$

is used.

Calling Sequence:

CALL CKDIFF (DIFF)

COMMON Blocks Used:

WORKER

Variables Not in COMMON:

FORTRAN Name	Format	Description
DIFF(3)	D	Kth backward difference vector
BINC(20)	D	Array of binomial coefficients

TEST

Purpose:

To compute and test the local truncation error to determine whether an increase or a decrease of the stepsizes of the equations of motion and the variational equations is required to satisfy the local error constraint equation.

Called By:

ORBIT
TABLE

Calls:

TABLEB
CKDIFF

Method:

The subroutine computes the local error Rn and determines whether the constraint equation

$$\text{TOL2} \leq \text{Rn} \leq \text{TOL1}$$

is satisfied, where TOL1 and TOL2 are specified error bounds. If this equation is satisfied, control returns to the calling program. If Rn > TOL1, the stepsize is decreased and if Rn > TOL2, the stepsize is increased. The methods used to compute the local error Rn, the new stepsize and the new tabular points depend on the calling program. Let k be the number of backpoints to be used; then we have:

Case I: Calling Program TABLE

In this case the test is made to determine whether the initial Cowell stepsize used in the starting table is adequate. The local error is estimated by the formula

$$\text{Rn} \cong \nabla^k \ddot{x}_k$$

where the kth difference ∇^k is computed by calling subroutine CKDIFF. The new stepsize computed is either half or double the original stepsize, and the new table of points are computed by returning to TABLE and restarting with the newly computed stepsize.

Case II: Calling Program ORBIT

In this case the test is made to determine whether the current Cowell stepsize at the nth integration step is adequate. The local error is estimated by the formula

$$Rn \cong C|\bar{x}_n^P - \bar{x}_n^C| ,$$

where C is an input error constant and $\bar{x}_n^P$, $\bar{x}_n^C$ are the predicted and corrected values of the satellite position vectors. The new stepsize is computed by the formula

$$h_{new} = h_{old} \left(\frac{TOL3}{Rn}\right)^{1/k}$$

where TOL3 is a specified value for the allowable local error, TOL2 ≤ TOL3 ≤ TOL1. In this case the new table of points at this new stepsize is computed by calling TABLEB and interpolating for the necessary values at the new stepsize.

The nominal values for the tolerances used in the program are:

$$TOL1 = .25 \times 10^{-4} \text{ meters}$$

$$TOL2 = .25 \times 10^{-10} \text{ meters}$$

$$TOL3 = .25 \times 10^{-7} \text{ meters}$$

which can be changed on option.

Calling Sequence

CALL TEST

COMMON Blocks Used:

ERR22
COWS
SSTEP
WORKER
LIMITS

Variables Not in COMMON:

FORTRAN Name	Format	Description
DIFF(3)	D	Kth backward difference
TP	D	Time span available at old stepsize

FORTRAN Name	Format	Description
TPP	D	Time span needed at new stepsize
OSTEP	D	Computed new stepsize
FN1	D	K

Error Constants

The error constants for the equations of motion when the calling program is ORBIT are:

Order	Value
4	0.00000000
5	.050000000
6	.052631579
7	.048721340
8	.044032445
9	.039713131
10	.035961476
11	.032748940
12	.029998465
13	.027632051
14	.025582503
15	.023794802

The error constants for the equations of motion when the calling program is TABLE are:

Order	Value
4	.0041666666
5	.0041666666
6	.0036541005
7	.0031415344
8	.0027086089
9	.0023553241
10	.0020677822
11	.0018320857
12	.0016369383
13	.0014736450
14	.0013356018
15	.0012177854

TABLEB

Purpose:

To produce a new starting table for both the equations of motion and the variational equations whenever a stepsize change occurs.

Called By:

TEST

Calls:

HEMINT
FRCS
SUMS
VEVAL

Method:

The subroutine produces a new set of position, velocity, and position partials for the equations of motion and the variational equations respectively at the new stepsize by:

(i) Computing the times at which points are to be produced;

(ii) Calling HEMINT with these times to interpolate for the necessary starting values.

A new starting table of acceleration vectors for the equations of motion and the variational equations is then obtained by using the newly computed position and velocity vectors in the FRCS and VEVAL subroutines. Next, new first and second sums for both the equations of motion and the variational equations using position, velocity, partials, and acceleration vectors at the new stepsize are computed by calling SUMS.

This subroutine also prints the value of the new stepsize and the time from epoch when the stepsize change occurred.

Calling Sequence:

CALL TABLEB

COMMON Blocks Used:

WORKER
LIMITS
SSTEP

GRBLOK
ANPART

Variables Not in COMMON:

FORTRAN Name	Format	Description
TPP	D	Time span needed at new stepsize
TMP	D	Time span for interpolation of points
TIME	D	Time of last good point computed
TI	D	Interpolation time
M	I	Interpolation order
K1P	I	Lower index of points to be interpolated
K1	I	Starting index of points to be interpolated

XS(15, 3, 20)
XDS(15, 3, 20)
	D	Storage arrays for interpolated position, velocity and position partials at the new stepsize

A.3 Variable Stepsize COMMON Block Variable Description

This section contains a detailed description of the COMMON areas used in the GEOSTAR-I ODP variable stepsize program which supplement those given in Section 5.1.

/COWS/

COMMON /COWS/TOL1, TOL2, TOL3, STPMIN, MODE

Variable	Type	Description	Program Where Defined	Program Where Used
TOL1	D	Upper bound on local error	BLOCK DATA OPTCRD	TEST
TOL2	D	Lower bound on local error	BLOCK DATA OPTCRD	TEST
TOL3	D	Desired local error at new integration stepsize	BLOCK DATA OPTCRD	TEST
STPMIN	D	Minimum stepsize allowed	BLOCK DATA OPTCRD	TEST
MODE	I	Variable indicating the mode of operation of the program. The values of this variable are: (2) Variable step mode (4) Fixed step mode.	MAIN BLOCK DATA OPTCRD	ORBIT TABLE

The nominal values of these variables built into the program are:

$$\text{TOL1} = .25 \times 10^{-4} \text{ meters}$$

$$\text{TOL2} = .25 \times 10^{-10} \text{ meters}$$

$$\text{TOL3} = .25 \times 10^{-7} \text{ meters}$$

$$\text{STPMIN} = 5.0 \text{ sec}$$

$$\text{MODE} = 2.$$

/ERR22/

COMMON /ERR22/SPO(3), SPOO(3)

Variable	Type	Description	Program Where Defined	Program Where Used
SPO(3)	D	Saved predicted value of position vector for local error test	CSTEP	TEST
SPOO(3)	D	Saved corrected value of position vector for local error test	CSTEP	TEST

/SSTEP/

COMMON /SSTEP/TEMP3, NNS, NORDER

Variable	Type	Description	Program Where Defined	Program Where Used
TEMP3	D	Old stepsize when a stepsize change occurs	TEST	TABLEB
NNS	I	Saved N(1) = P_1 - 2	TEST	TABLEB
NORDER	I	Saved ORDER(1)	TEST	TABLEB

where:

P_1 = order of the formula used to integrate the equations of motion

ORDER(1) = P_1

Appendix B

B Matrix Tape Format

Record	Record	
No.	Size	Description of Contents
1	60	Header — matrix size (N), total variance (V1), etc.
2	4N + 8	Matrix labels
3	8N + 12	Matrix data - row 1 (starting with right hand side)
4	8N + 12	Matrix data - row 2 (starting with right hand side)
N + 2	8N + 12	Matrix data - row N (starting with right hand side)
N + 3		Matrix parameter set identification
N + 4	4N + 4	Matrix parameter labels
N + 5	8N + 4	Matrix parameter values
N + 6	60	End of logical tape

Record No. 1 — Header

Item	Format	No. of Bytes	Value/Limits
Record Type	I * 4	4	10001
Matrix Identification Number	I * 4	4	1 to 99998
No. of Matrix Rows, N (= No. of Parameters)	I * 4	4	N ≤ 200
No. of Matrix Elements Per Row (Including the Right Hand Side)	I * 4	4	1 + N ≤ 201
Total Variance, V1	R * 8	8	
Weighted Variance, V2	R * 8	8	
Arc Variance, V3	R * 8	8	
No. of Observations	I * 4	4	
Matrix Type	I * 4	4	8
Matrix Name (alphanumeric)	R * 4	12	12 EBCDIC characters
	Total No. of Bytes	60	

Record No. 2 — Matrix Labels

Item	Format	No. of Bytes	Value/Limits
Record Type	I * 4	4	10002
Dummy	I * 4	4	0
Parameter Labels	I * 4	4N	
	Total No. of Bytes	4N + 8	

Record No. (J + 2) — Matrix Data — Row J (J = 1, 2, . . . , N)

Item	Format	No. of Bytes	Value/Limits
Record Type	I * 4	4	10003
Elements of Jth Matrix Row (Starting with the Right Hand Side)	R * 8	8(1 + N)	N 200
	Total No. of Bytes	8N + 12	

Record No. (N + 3) — Matrix Parameter Set Identification

Item	Format	No. of Bytes	Value/Limits
Record Type	I * 4	4	10011
Matrix Identification No.*	I * 4	4	1 to 99998
Dummy	I * 4	4	1
No. of Parameters*	I * 4	4	N ≤ 200
Dummy	I * 4	28	All 0
Code for Parameter Set	I * 4	4	8
Matrix Name*	R * 4	12	12 alphameric characters
	Total No. of Bytes	60	

*Same as in Record No. 1.

Record No. (N + 4) — Matrix Parameter Labels

Item	Format	No. of Bytes	Value/Limits
Record Type	I * 4	4	10012
Parameter Labels*	I * 4	4N	N ≤ 200
	Total No. of Bytes	4N + 4	

Record No. (N + 5) — Matrix Parameter Values

Item	Format	No. of Bytes	Value/Limits
Record Type	I * 4	4	10013
Parameter Values	R * 8	8N	N ≤ 200
	Total No. of Bytes	8N + 4	

Record No. (N + 6) — End of Logical Tape

Item	Format	No. of Bytes	Value/Limits
Record Type	I * 4	4	-19991
Dummy	I * 4	56	All 0
	Total No. of Bytes	60	