2 (mix)

NASA CR-111852

TR-DA2154
31 December 1969

USER'S MANUAL FOR THE MARK IV ERROR PROPAGATION PROGRAM

CONTRACT NAS1-9307

Prepared by:
SATELLITE APPLICATIONS DEPARTMENT

Prepared for:
NATIONAL AERONAUTICS & SPACE ADMINISTRATION
LANGLEY RESEARCH CENTER
HAMPTON, VIRGINIA

PHILCO Ford

SPACE & RE-ENTRY SYSTEMS DIVISION
Philco-Ford Corporation
Palo Alto, California 94303

USER'S MANUAL
FOR THE
MARK IV ERROR PROPAGATION PROGRAM

December 1969

Contract NAS 1-9307

Prepared by

Satellite Applications Department
Philco-Ford Corporation
Palo Alto, California

for

NASA Langley Research Center
Hampton, Virginia

FOREWORD

The Mark IV Error Propagation Program was developed under Contract NAS1-9307 for the National Aeronautics and Space Administration, Langley Research Center by Philco-Ford Corporation. Two tasks were undertaken in this development. They were:

1. Conversion of the Mark II Error Propagation Program to a form acceptable to the CDC 6600 computer, and
2. Addition of capabilities for start-up and search for suitable interplanetary trajectories.

The work was performed in the period June 30, 1969 through December 31, 1969. The Philco-Ford personnel responsible for the program development and writing of this manual were:

W. S. Bjorkman	Senior Engineering Specialist
M. J. Brooks	Senior Programmer

The effort was managed by R. C. Jensen.

This manual contains a description of the Mark IV Error Propagation Program and detailed instructions for its utilization. For additional details about the implemented theory, refer to "User's Manual for the Mark II Error Propagation Program", WDL-TR2758, and to "Subroutine Descriptions and Listings for the Mark II Error Propagation Program", WDL-TR2757, Volume I.

It is Philco-Ford's hope and expectation that the Mark IV program will prove helpful in understanding the interrelationships of the various parameters which affect mission success and will, therefore, become a useful tool in the planning of future space missions.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

INTRODUCTION

Space mission planning requires an understanding of the interrelationships among the numerous parameters affecting mission success. In most instances, no simple equation defining these interrelationships exists. The only practical way of studying the effects of many of these parameters is to simulate the mission on a digital computer, using the parameters together with laws of physics and principles of statistics in the simulation.

The Mark IV Error Propagation Program is a computer program which enables the study of parameter interrelationships in the realms of lunar or interplanetary trajectories, navigation and guidance. It is a fourth-generation program which was developed for LRC by Philco-Ford. Three earlier error propagation programs were developed by Philco-Ford under NASA contracts for GSFC. The Mark IV program is a version of the GSFC Mark II Error Propagation Program converted for operation in Fortran IV on the CDC 6600 computer and modified by the addition of start-up and search capabilities. The Mark II program, delivered in December 1965, was written partly in Fortran IV and partly in machine language (MAP) for the IBM 7094 computer. It lacked the start-up and generalized search capabilities, presuming the initial trajectory conditions to be obtained from some other program. A restricted search capability for integrated trajectories was available in the Mark II program, however. The Mark IV program retains all of the capability of the Mark II program, is generally more efficient, and has been simplified in its programming.

This manual is divided into two sections. The first contains a brief description of the capabilities of the Mark IV program and includes a summary of the theory, assumptions and equations which have been implemented. The second section contains instructions for using the program along with input and output examples. Tables of user-relevant information are provided for easy reference once a program familiarity has been developed. The user is referred to the user's manual (Ref. 1) and subroutine descriptions (Ref. 2) for the Mark II program for additional details not found in this manual.

SECTION I

PROGRAM CAPABILITIES

The capabilities of the Mark IV Error Propagation Program fall into two major categories which are:

1. trajectory generation, and
2. error propagation.

The error propagation capability requires a nominal trajectory which must be generated and stored on tape or disc sometime prior to propagation of errors. We will, therefore, begin this section with a description of the program's capabilities for trajectory generation and conclude with error propagation capabilities.

1.1 TRAJECTORY GENERATION

The sections of the program concerned with trajectory generation are called START-UP, SEARCH, CONW and PINT. START-UP provides approximate initial conditions for interplanetary transfer trajectories using the matched-conic assumption. SEARCH is a "generalized" search routine which iterates by the steepest descent method to differentially correct initial conditions in order to satisfy imposed constraints at the end of the trajectory. The trajectory model used by SEARCH may be chosen to be either patched-conic or precision-integrated. CONW takes specified initial conditions for a patched-conic trajectory and writes interpolation coefficients for that trajectory on tape or disc for later use in error propagation. PINT's primary function is to write integrated trajectory interpolation coefficients on tape just as CONW does for patched-conic trajectories. But, PINT can also perform a restricted search on initial conditions to satisfy end constraints as SEARCH does, using an integrated trajectory model. Another PINT capability is to compute and integrate variational equations for equation of motion parameter sensitivities or the state transition matrix. CONW and PINT were sections of the Mark II Error Propagation Program and will, therefore, be discussed only briefly.

1.1.1 START-UP

This capability requires the user to specify the launch and target planets plus the departure and arrival dates for interplanetary transfer. The program then interrogates the planetary ephemeris for positions and velocities of the launch body at departure date and target body at arrival date. The heliocentric conic section which joins the two (massless) planets in the specified flight time is then determined. The vector difference between the initial velocity on the heliocentric conic and the launch body velocity at departure date is taken to be the hyperbolic excess velocity at launch.

The next assumption used in the determination of approximate initial conditions is that the trajectory originates from a circular parking orbit about the launch body. The user must specify four parameters which describe this parking orbit, namely: insertion latitude, insertion longitude, insertion velocity azimuth and orbit altitude. If the launch body is the Earth, the START-UP assumes the parking orbit parameters to be geographic (Earth-fixed) and uses them to find the time of day at which the hyperbolic excess velocity vector lies in the plane of the parking orbit. If the latitude of the hyperbolic excess velocity vector is greater than the inclination of the parking orbit, the program finds the time of day which minimizes the latitude of the hyperbolic excess velocity vector relative to the parking orbit. The time of day thus found is interpreted to be the time of insertion into parking orbit (new departure date) and is always later than the input departure date. If two solutions for time of day exist, the one nearer the input departure date is selected by the program. If the launch body is not the Earth, the parking orbit parameters are assumed to represent an "inertial" parking orbit which does not rotate with the launch body. The program next proceeds to calculate the time in parking orbit which results in the minimum injection velocity requirement and yet attains the desired hyperbolic excess velocity. The injection maneuver is a velocity impulse. The quantities output from (i.e., provided by) START-UP for use by SEARCH, CONW or PINT are:

Date (and time, UMT) of park orbit insertion

Date (and time, UMT) of injection onto the transfer hyperbola

Launch parameters - park time, azimuth of the velocity impulse relative to the park orbit track at injection, elevation of the velocity impulse at injection, and velocity impulse magnitude

Cartesian state - three components of position and three of velocity at injection referred to the mean equator and equinox of 1950.0 coordinate system

Spherical state - radius, declination, right ascension, speed, flight path angle and azimuth at injection referred to EE50.

The conditions provided by START-UP are precise enough to initiate a patched-conic or integrated trajectory to the vicinity of the target body. An option for "matching" conics is provided if a better solution is desired. Under this option, the program utilizes the previously-computed departure trajectory to calculate the point and time of "patch" to the sun. The hyperbolic excess velocity and specified desired miss vector at arrival are used to compute the arrival hyperbola and thus the point and time of "patch" to the target body. The heliocentric trajectory is then re-computed as the conic section which transfers between the sun-patch and target-patch points in the adjusted transfer time. Of course, the launch and target bodies move during the departure and arrival phases. The new heliocentric conic results in new hyperbolic excess velocities at departure and arrival which may then be used to initiate another iteration. Otherwise, the hyperbolic excess velocity at departure is simply used to provide an improved set of launch control parameters and injection conditions according to the parking orbit assumptions. A summary of the mathematical formulation of START-UP may be found in Appendix A.

1.1.2 SEARCH

This program performs a systematic search to satisfy a specified set of end constraints by differentially correcting a set of initial control parameters. The control parameters (from one to six in number) may be selected from any one of the following sets.

Set 1	Set 2	Set 3
Cartesian	Spherical	Launch Parameters
X	R	DLAZ (insertion azimuth)
Y	LAT	DTL (time of insertion)
Z	LON	PRKT (park time)
VX	V	DVAZ (injection impulse azimuth)
VY	γ	DVEL (injection impluse elevation)
VZ	AZ	DELV (injection impulse magnitude)

The cartesian and spherical controls may be referred to the Earth's equator and equinox or to the ecliptic and equinox (both mean of either 1950.0 epoch or date). Their origin may be a time other than when initial conditions are specified, making it possible to target from a midcourse maneuver time. Choice of the launch parameter set as control variables requires that the parking orbit and insertion date be specified and that the search take place at injection from parking orbit into the transfer orbit.

The constraints may be selected from:

1.- 6. cartesian end state components

7.-12. spherical end state components

13. B.T., asymptotic miss vector component

14. B.R., asymptotic miss vector component

15. v_∞, hyperbolic excess speed

16. r_p, radius at periapsis

17. i, inclination

18. Ω, longitude of the ascending node

19. ω_p, argument of perifocus

20. t_f, time of flight

The number of constraints cannot exceed the number of controls and is always six or less. The frame-relative constraints may be referred to:

1. Earth's mean equator and equinox of 1950.0,
2. target orbital coordinates (1. along the radius vector from the target's central body, 2. normal to the other axes in the right-handed sense, and 3. along the target's orbit normal),
3. targetographic coordinates, or
4. mean ecliptic and equinox of date.

The trajectory model (or plant) by which constraint errors are computed from controls may be either patched-conic or precision-integrated to include perturbations. In the integrated formulation, Encke's Method is used for trajectory calculation and Adams' Fourth-Order Method is used for numerical integration. Sensitivities of constraints to controls are computed by the secant or difference method. That is, finite increments are added to each control in turn and the trajectory and constraints are re-calculated. The sensitivities are then taken to be the constraint differences divided by the control increment. This technique is costly, but reliable and conceptually simple. The Method of Steepest Descent is used to predict control increments which will reduce the constraint errors.

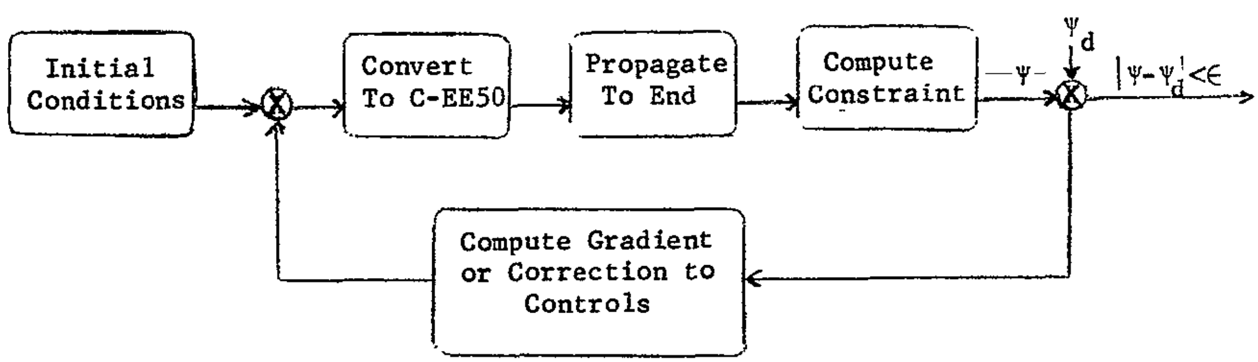


Figure 1-1 Search Logic

Figure 1-1 is a block diagram of the computational logic of the SEARCH program. The symbol Ψ represents the computed constraint vector and Ψ_D the desired constraint vector.

A scanning option has been provided in SEARCH whereby any control variable may be automatically incremented. This option is useful for assessing the nature of local constraint behavior due to control variation.

Appendix B contains a brief description of the mathematical formulation of SEARCH. A description of the acceleration equations used in the integrated trajectory model is found in Appendix B of Reference 1. Appendix C of this manual describes the method of numerical integration used here.

1.1.3 CONW

This section of the Mark IV program accepts initial trajectory conditions and computes a patched-conic trajectory. Interpolation coefficients for the trajectory are written on tape or disc for later use by ERP, the error propagation section. In addition to trajectory interpolation coefficients, CONW computes the matrix of sensitivity of end conditions (B.T, B.R, time to periapsis, v_∞, r_p, i) to end cartesian state for later use in guidance and prediction calculations. The advantage of using a patched-conic nominal trajectory lies in the speed and simplicity of its generation - either type trajectory gives rise to similar answers in error propagation. Initial conditions for CONW may be obtained from START-UP, SEARCH or other sources. CONW accepts cartesian or spherical components of state or orbital elements referred to equatorial, ecliptic or body-fixed coordinates of 1950.0 or date.

1.1.4 PINT

This section of the Mark IV program accepts initial trajectory conditions and performs the calculations for a precision-integrated trajectory. The input state options and the tape- or disc-stored quantities are the same as in CONW.

The targeting option of PINT, REFINE, serves the same basic purpose as SEARCH - to try to satisfy end constraints by systematic variation of initial conditions. There are some major differences, however. On the positive side, REFINE's sensitivities of constraints to controls are generated by integration of variational equations along with the trajectory. This method is faster than the secant method used in SEARCH. On the negative side, the available control set is limited to the launch parameters listed in Section 1.1.2. The constraint set has been expanded over that of the Mark II program to include B·T and B·R or only radius of closest approach, but is still very restricted relative to SEARCH.

Another PINT option enables the user to calculate the sensitivity of the trajectory to variations (uncertainties) in constants in the equations of motion (e.g., planetary masses, gravitational harmonic coefficients). These sensitivities are calculated by integration of variational equations along the nominal trajectory. The same sensitivities are computed, but not available for output, in error propagation when treating equation of motion parameter uncertainties. Calling out yet another PINT option, the state transition matrix may also be calculated by integrating variational equations. The implemented variational equations are described fully in Appendix B of Reference 1.

Some improvements have been made to the set of stopping functions used when integrating trajectories in both PINT and SEARCH. The stopping functions are computed along with a trajectory and are used to terminate the integration. The five functions which are computed (in subroutine FSUB) are:

1. $\left|\frac{\Delta r}{r}\right| - .03$, the Encke rectification criterion

2. $r \text{ sign } (R \cdot V) - r_{patch}$, the patch-away criterion

3. $f(r_{patch})$, the patch-to or closest approach criterion defined in subroutine PATCH (Ref. 2)

4. $t - t_{stop}$, the time limit criterion

5. $f(r_{stop})$, the criterion for closest approach or given radius from the target body.

In the Mark II program, the patch-away criterion was not signed, the closest-approach function did not exist, and the patch-to and radius-stop functions were merely distance differences. The newer formulations eliminate ambiguities and permit lengthening the integration step size for interplanetary transfer trajectories.

1.2 ERROR PROPAGATION

The fundamental questions answered by the Mark IV Error Propagation Program are, "Given a space trajectory and a set of measurements of a specified type and quality,

1. How well can the trajectory be determined? (navigation)

2. How well can measurement biases and equation of motion parameter uncertainties be determined?

3. How do measurement biases and equation of motion parameter uncertainties affect the quality of trajectory determination?

4. What effects do navigation errors have on midcourse guidance requirements and how do guidance execution errors affect navigation?"

In order to answer these and other questions, the Mark IV Error Propagation Program utilizes statistical principles to determine ensemble characteristics of trajectories in the vicinity of the nominal trajectory. A covariance matrix of state estimation errors is assumed to represent these ensemble characteristics. This covariance matrix is manipulated by the program according to Schmidt-Kalman filter theory using optimal weighting of all measurements. A brief summary of these manipulations as implemented in the Mark IV program will be presented in this section. More detail and derivations will be found in References 1, 2 and 3.

The state vector X, consists of 6 cartesian components of position and velocity, k equation of motion or dynamic parameters and ℓ measurement biases. (The term "measurement biases" includes location errors and time biases.) If X represents the estimate of state and $x = X-\hat{X}$ represents the error in estimate of state, the covariance matrix of state estimation errors, P, is

$$P = E(xx^T)$$

where E is the statistical expectation operator and X^T means X-transposed. The user must supply P to the program initially. Except for the part of P which represents the trajectory estimation error distribution (upper left 6 x 6 matrix) the program expects P to be initially diagonal. That is, the dynamic and measurement biases are assumed to be uncorrelated initially.

The answers to the questions posed earlier are to be found through interpretation of the P-matrix. We assume that the mean state error is zero. The elements of P represent statistically

$$P_{ij} = \rho_{ij} \ \sigma_i \ \sigma_j$$

where σ_i is the standard deviation of state error component x_i about a zero mean value and the ρ_{ij} are coefficients denoting the correlation between x_i and x_j. In general, the smaller the standard deviation of a component of state error, the better that state component is estimated. The probability that a sample state error vector would lie within the hyper-ellipsoid defined by the covariance matrix could be (but is not) computed.

The program computes the trajectory state and its corresponding portion of the covariance matrix referred to the mean equator and equinox of 1950.0 coordinate system. This system is not very meaningful for output, so the program supplies simply-computed performance measures for positions, RMSP, and for velocity, RMSV. These are defined by

$$\mathrm{RMSP} = \sqrt{P_{11} + P_{22} + P_{33}} = \sqrt{\sigma_x^2 + \sigma_y^2 + \sigma_z^2} \ , \quad \text{and}$$

$$\mathrm{RMSV} = \sqrt{P_{44} + P_{55} + P_{66}} = \sqrt{\sigma_{vx}^2 + \sigma_{vy}^2 + \sigma_{vz}^2} \ .$$

The i-th diagonal element of P represents the variance of the i-th state error component about a zero mean value. At the end of a processing interval and in special output the covariance matrix is transformed into other coordinate systems than EE50 and printed out.

Between measurements or other events, P is propagated according to the equation

$$P(t_n) = \Phi(t_n; t_{n-1}) P(t_{n-1}) \Phi^T(t_n ; t_{n-1})$$

where P(t) means "P at time t" and $\Phi(t_n; t_{n-1})$ is the state transition matrix from time t_{n-1} to time t_n. The state transition matrix is partitioned as shown.

$$\Phi(t_n; t_{n-1}) = \begin{bmatrix} \varphi_x & \varphi_u & 0 \\ 0 & I & 0 \\ 0 & 0 & I \end{bmatrix}$$

In this partitioning, φ_x represents the trajectory state transition matrix, dimensioned 6X6. It is computed in closed-form as an average conic transition matrix between the states $X(t_{n-1})$ and $X(t_n)$. For details about the computation of φ_x, see subroutine PHIZ in Reference 2. The 6xk matrix φ_u represents the sensitivity of the trajectory state to variations in the dynamic bias parameters. It is computed by integrating variational equations along the nominal trajectory from t_{n-1} to t_n. The other elements of φ are either null matrices or identities. The null matrix in the first row represents the fact that the trajectory error is insensitive to measurement biases in the interval between measurements. The remaining elements of Φ represent the assumption that the non-trajectory state elements are time-invariant biases.

At a measurement, P is changed according to the equation

$$P^{+} = P^{-} - P^{-}H^{T}(HP^{-}H^{T}+Q)^{-1}HP^{-}$$

where (+) means "after processing the measurement" and (-) means "before processing the measurement". H is the gradient of the measurement with respect to the state at the time of the measurement (see Appendix A of Reference 1 for complete derivations of H) and Q is the variance of the measurement's random error. It is assumed that the measurement errors are uncorrelated. If there are several measurements at the same time, these are processed individually so that the inverse indicated in the above equation is computed as a scalar reciprocal and the gradient, H, is a single-row matrix or vector.

The Mark IV program contains an option for considering the effects of dynamic or measurement biases on the state estimation without including them as additional state components. The "consider" option is implemented by retaining and manipulating the correlations between the "solved-for" state and the "considered" state. If the correlation matrix for the "considered" dynamic biases is denoted C_{ux} and for measurement biases, C_{vx}, the manipulation equations between events are:

$$P(t_n) = \Phi P(t_{n-1}) \Phi^T + \Phi C_{ux}(t_{n-1})U^T + UC_{ux}(t_{n-1}) \Phi^T + UDU^T$$

$$C_{ux}(t_n) = \Phi C_{ux}(t_{n-1}) + UD$$

$$C_{vx}(t_n) = \Phi C_{vx}(t_{n-1})$$

In the above equations, U is the sensitivity of the state at t_n to the vector of "considered" dynamic biases over the interval from t_{n-1} to t_n. Φ is the state transition matrix from t_{n-1} to t_n. D is the (diagonal and constant) covariance matrix of "considered" dynamic biases. A derivation of the above equations may be found in Reference 3. Both C_{ux} and C_{vx} are considered to be zero initially - indicating that the initial state and bias errors are statistically uncorrelated. At a measurement, the "consider" option is implemented by the following equations:

$$\overline{Y} = HP^-H^T + HC_{vx}^- G^T + GC_{vx}^{-T} H^T + GWG^T + Q$$

$$P^+ = P^- - (P^-H^T + C_{vx}^- G^T)(\overline{Y})^{-1}(P^-H^T + C_{vx}^- G^T)^T$$

$$C_{vx}^+ = C_{vx}^- - (P^-H^T + C_{vx}^- G^T)(\overline{Y})^{-1}(HC_{vx} + GW)$$

$$C_{vx}^+ = C_{vx}^- - (P^-H^T + C_{vx}^- G^T)(\overline{Y})^{-1}(HC_{vx})$$

In these equations, G is the gradient of the measurement with respect to the "considered" measurement bias vector and W is the (diagonal and constant) covariance matrix of considered measurement bias errors. A complete derivation of these equations may be found in Reference 3. The above equations for updating P, C_{ux} and C_{vx} in time and changing them at measurements are the "heart" of the Mark IV Error Propagation Program. Most of the other computations in the program are, relatively speaking, "programming details".

The only event at which P is changed other than the measurement event is a simulated midcourse guidance maneuver. It is necessary, in considering midcourse guidance, to carry along another 6 x 6 covariance matrix, PAR, in addition to P. PAR represents the distribution of trajectory errors about the nominal trajectory. These trajectory errors are assumed to result from random errors in guiding to the desired nominal trajectory. The PAR matrix is updated in time by means of the state transition matrix, but naturally does not change at measurements. Both P and PAR are changed at a midcourse guidance maneuver simulation. The assumption of an impulsive velocity maneuver is made according to one of three available guidance laws. This assumption leads to a change only in the velocity portion of the P-matrix. Specifically, the P-matrix is changed at a midcourse maneuver only by the addition of a 3x3 matrix representing the velocity uncertainty due to errors in executing the maneuver.

$$P^+ = P^- + \begin{bmatrix} 0_{3x3} & 0_{3x3} \\ 0_{3x3} & k^2 \cdot E(\epsilon\,\epsilon^T) \end{bmatrix}$$

The factor k^2 defines the accuracy presumed in monitoring the correction. The PAR matrix after the maneuver is computed to be the sum of the trajectory navigation uncertainty prior to the maneuver and the execution errors.

$$PAR^+ = P^- + \begin{bmatrix} 0_{3x3} & 0_{3x3} \\ 0_{3x3} & E(\epsilon\,\epsilon^T) \end{bmatrix}$$

Formulation of $E(\epsilon\epsilon^T)$ is accomplished by transforming resolution, pointing and proportional maneuver errors into the program's cartesian system. The theory and equations for guidance calculations are described fully in Reference 1.

The measurement simulation capabilities of the Mark IV program include observations from Earth-based tracking stations, beacons located on the Moon or on planets, and devices on board the spacecraft. Storage dimensions limit the number of stations to 12 and beacons to 10 for any one case. The beacons must all be located on the same body. A list of the available measurements will be found in the capability summary which concludes this section. Formulae for the various measurements in terms of the state and station or beacon locations will be found in Appendix A of Reference 1. The station and beacon view times are computed by the program from the stored nominal trajectory before proceeding with error propagation calculations. During error propagation the stations or beacons are assumed to observe (take measurements on) the spacecraft when it comes into view and then at regular intervals specified by input until the spacecraft is occulted. The simulated onboard measurements also occur at regular intervals determined by input.

Midcourse guidance maneuvers (as many as five) are simulated at pre-set times along the trajectory. The available guidance laws are:

1. fixed time of arrival,
2. fixed energy at arrival, and
3. minimum fuel, variable time and energy.

Another option provides the capability of (linearly) propagating the P-matrix (and, if guiding, the PAR-matrix) to the end of the trajectory to predict the covariance matrix of miss vector (B·T, B·R, time of arrival, v_∞, r_p and i) errors. The resulting miss vector errors are those which would be realized if no more observations (or, for guidance, no more correction maneuvers) were made. The expected midcourse velocity correction requirement and direction of the critical plane normal vector are provided at each output point when the guidance option is selected. The theory for these computations is found in Reference 1.

The program's storage requirements set the P-matrix dimension at 900 cells. This permits the treatment of a 30-element state vector for any one case ($30 \times 30 = 900$). The number of dynamic or measurement biases considered but not solved-for may be 100 or less. The principal limitation on the number of state components imposed by the 900-element P-matrix is

(number of solved-for) x (number of solved for + number considered) ≤ 900

The special output capability (SPOUT) performs calculations not normally required or performed in error propagation. A special output file is written on tape or disc by option in ERP. This file contains the trajectory history and the P-matrix history relative to cartesian EE50 coordinates, as well as RMSP and RMSV. When SPOUT is called, the file is read in and transformed according to input option specification for selective output. RMSP and RMSV may be printer-plotted versus time as may be any of the solved-for bias standard deviations. A case and record number connects the SPOUT file with the ERP output and permits the user to be selective in his SPOUT calculations.

The following summary provides a short but comprehensive list of the capabilities of the Mark IV Error Propagation Program.

CAPABILITY SUMMARY

I Trajectory Generation

A. Start-Up

1. Determines interplanetary transfer trajectory
   - a. massless planet solution
   - b. matched-conic finite influence solution
2. Determines necessary injection conditions
   - a. circular parking orbit assumed
   - b. computes launch control parameters (see B.1. below)
   - c. computes cartesian and spherical injection state

B. Search

1. Controls
   - a. Cartesian state (x, y, z, vx, vy, vz)
   - b. spherical state (r, lat, lon, v, γ, az)
   - c. launch control parameters (insertion azimuth, insertion time, park time, Δv azimuth, Δv elevation, Δv magnitude)
2. Constraints
   - a. end cartesian state (6 components)
   - b. end spherical state (6 components)
   - c. miss vector (B·T, B·R)
   - d. orbital elements (v_∞, r_p, i, Ω, ω_p)
   - e. time of flight
3. Search run options
   - a. steepest descent constraint satisfaction
   - b. secant method gradient with hold-fixed option
   - c. patched-conic trajectory model
   - d. perturbed, integrated trajectory model (see D.4. below)
   - e. midcourse targeting
   - f. multiple control state coordinate systems

I.B.3 (continued)

g. multiple constraint coordinate systems
h. automatic constraint scan in one control

C. Patched-Conic Trajectory-Write (CONW)

1. Generates and writes interpolation coefficients on tape or disc file
2. Writes critical records of patch-points and end points
3. Generates and writes end point miss sensitivity matrix

D. Precision Trajectory Integration (PINT)

1. Integrates perturbed trajectory
   a. writes interpolation coefficients (see C.1., 2., 3. above)
   b. integrates variational equations for state transition matrix
2. Refines initial conditions
   a. steepest descent constraint satisfaction
   b. integrates variational equations for gradient
   c. launch parameter controls (see B.1. above)
   d. constraint options (B·T, B·R, r_p, v_∞, time of flight, target vector, earth-return point, minimum injection Δv)
   e. target body orbital constraint reference system
3. Computes motion parameter sensitivities
   a. interpolates trajectory from tape
   b. integrates variational equations for sensitivities
4. Trajectory calculations
   a. Encke's Method of trajectory calculation
   b. Adams' fourth-order method of numerical integration
   c. interpolated stopping functions (Encke rectification, patch-away, patch-to, closest approach, stopping radius; time of flight)
   d. perturbations considered:

      4 zonal harmonics of Earth's gravity
      5 longitudinal harmonics of Earth's gravity
      7-body gravitational attraction
      lunar triaxiality
      Earth's atmospheric drag
      solar radiation pressure
      tangential thrust

I.D.4 (continued)

e. center-shift at fixed spheres of influence

E. General

1. Optional lunar and planetary ephemeris
    a. approximate, mean-element package
    b. JPL ephemeris tape interpolation package
2. Units of computation
    a. time in seconds
    b. positions in kilometers
    c. velocities in kilometers/second
3. Coordinate system for computation
    a. Earth's mean equator and equinox of 1950.0
    b. cartesian

II Error Propagation

A. Measurements (Random Error Sources)

1. Earth-based tracking stations (up to 12)
    a. range
    b. azimuth and elevation
    c. right ascension and declination
    d. Minitrack (direction cosines)
    e. range rate
    f. azimuth and elevation rates
    g. right ascension and declination rates
    h. direction cosine rates
2. Moon- or planet-based beacons (up to 10)
    a. range
    b. range rate
    c. azimuth and elevation from the vehicle

II.A (continued)

3. On-board star and planet measurements
    a. height
    b. height rate
    c. planet subtended angle
    d. latitude and longitude (from inertial platform)
    e. sextant (star-planet angles)

B. Deterministic Error Sources

1. Equation of Motion Parameters
    a. astronomical unit conversion
    b. lunar and planetary masses
    c. zonal harmonics of Earth's gravity
    d. longitudinal harmonics of Earth's gravity
    e. harmonics of the Moon's gravity
    f. Earth's atmospheric drag coefficients
    g. solar radiation pressure coefficient
    h. venting thrust magnitude
2. Deterministic Error Sources in Measurement
    a. tracking station measurement biases
    b. tracking station location errors
    c. tracking station time bias
    d. beacon measurement biases
    e. beacon location errors
    f. vehicle time bias
    g. onboard height and height rate biases
    h. velocity of light uncertainty

C. Program Implementation

1. Minimum variance estimation technique
2. Linear propagation of the covariance matrix
    a. closed-form conic transition matrix
    b. computed in inertial Cartesian coordinates

II.C (continued)

3. Automatic search for station on-off times
    a. occultations
    b. artificial horizons and zenith limits
4. Nominal trajectory interpolated from tape
    a. patched conic
    b. precision integrated
5. Extra output tape
    a. normal output of rms position and velocity without tape
    b. covariance matrix output
    c. state output various coordinate systems
    d. automatic plotting of specific parameters

D. Treatment of Deterministic Errors

1. Treat as though solving for the error
    a. 900-element covariance matrix
2. Treat as though only considering the error's influence
    a. consider up to 100 error sources
3. Equation of motion error sources
    a. sensitivities obtained by integrating variational equations about nominal trajectory

E. Prediction and Guidance

1. Prediction to the end point
    a. uncertainty in miss parameters due to navigation errors (P)
    b. uncertainty in miss parameters due to trajectory errors (PAR)
2. Midcourse guidance
    a. Guidance laws:

        fixed time of arrival
        constant target-relative energy
        minimum velocity correction

II.E.2 (continued)

b. Correction errors:

pointing

resolution

proportional

monitoring

SECTION 2

PROGRAM USAGE

The Mark IV Error Propagation Program consists of six related sub-programs which may be run individually or serially. These sub-programs are:

1. PLANET (START-UP) for generating approximate interplanetary trajectory information and initial conditions
2. SEARCH for improving initial conditions in order to satisfy specified end constraints
3. CONW for writing an ephemeris file of the vehicle's patched-conic trajectory
4. PINT for writing an ephemeris file of the vehicle's integrated trajectory
5. ERP for performing error propagation calculations
6. SPOUT for calculating and displaying selected quantities after error propagation.

The organization of the program by subroutine is shown in Figure 2-1. The purpose of each subroutine is listed in Appendix E. Descriptions of the subroutines are to be found in Reference 2 as amended under this contract.

Each sub-program has need of planetary ephemeris information. This information may be supplied to the program by means of a self-contained, mean-element ephemeris (subroutines ANTRI, EXPAND, EL2EX and UPDATE as shown in Figure 2-1) or by means of JPL's ephemeris tape (interpolated with subroutines ANTR1, DEPHEM, BUFFIL and INTCOF). The taped ephemeris slows program execution and requires more storage, but is more precise

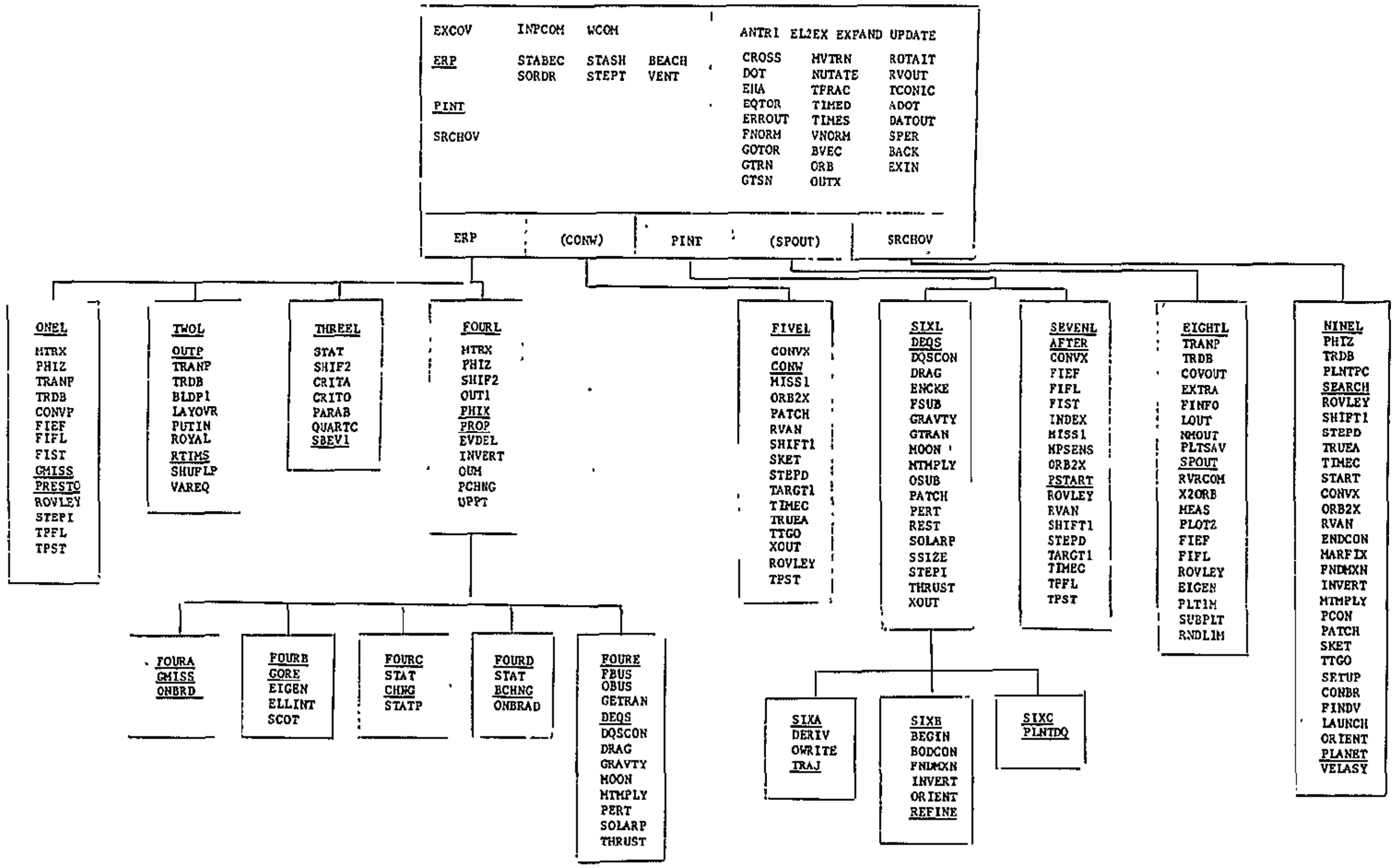


Figure 2-1. PROGRAM ORGANIZATION BY SUBROUTINE

than the self-contained ephemeris. Replacement of decks is required in order to change ephemeris options.

2.1 INPUT DATA

A data run consists of one or more cases submitted to computer operations with the Mark IV program deck for the purpose of obtaining printed output information computed by the program. Each case consists of a call of one of the six main sub-programs. The first card of any case contains a number (punched into column 5) which specifies which of the main sub-programs is to be called for that case. The code for this card is:

0	end of run
1	PINT
2	ERP
3	SPOUT
4	CONW
5	PLANET
6	SEARCH

Several precautions must be taken in ordering cases. First, a PINT or CONW case must be executed sometime prior to an ERP case so that the vehicle's ephemeris file required by ERP has been written. Second, a tape or disc file must be written by ERP before SPOUT can be run. These files may be written in a previous run if stored on tape. In this event, the user must be sure to have the right tape mounted for the later run. Another precaution is to avoid executing PINT, PLANET or SEARCH cases after ERP, SPOUT or CONW cases in any one run. The first three sub-programs use WCOM as an input array and the latter three use WCOM as a working array. Incorrect usage could erase expected input data, causing strange results.

Case data follow the first card for each case. These data replace or augment standard case data stored in the program by BLOCK DATA routines.

The stored data values may be found in computer listings of INPCOM and WCOM or in Reference 2. Data contained in the program at the beginning of a run will remain and be used until changed by input. In other words, if a great many input quantities are specified in case no. 1 and desired for other cases in the run, they need not be specified again for later cases in the run. When changed for any case, the changed data values will remain in use for the remainder of the run or until changed again. The only exception to this philosophy is in ERP.

Input data are read into the two COMMON arrays, INPCOM and WCOM. Generally, the user must supply the address or location of the array to be filled and the numerical value to fill that address or location. Input description for the Mark IV program consists primarily of lists relating locations to parameters required by the program. Appendix D of this manual is a summary of input requirements of the modified Mark II program's input requirements and sample input/output may be found in Section 5 of Reference 1. The remainder of the present section will be given to input and output examples of the modifications: the start-up and search capabilities.

2.1.1 INPUT FOR START-UP

The start-up option is called with "5" in column 5 of the first card. Case data are then read into the WCOM array through subroutine ROVLEY. An input example is shown in Figure 2-2. The launch and target planets are specified by fixed-point inputs and the body code of Table 1.

IN(1) launch body number (1 for Earth)

IN(2) target body number (5 for Mars)

The remaining case data are input by locations and corresponding numerical values in a 4(I3,E12.8) format. (See Table 2.) These are primarily:

FORTRAN CODING FORM

Program	Punching Instructions: Graphic / Punch	Card Form # *	Page of
Programmer	Date		Identification 73–80

C FOR COMMENT

STATEMENT NUMBER (1–5) | Cont (6) | FORTRAN STATEMENT (7–72)

```
    5     START-UP FOR LRC CASE 1 LEAVING 7/24/73, ARRIVING 2/16/74
   1           2                                        I
     1    5
121 28.5      122 -80.5     123 67.74     124 184.0
373 -4930.0   374 7200.0
401 7307.24   402 0.0       403 7402.16   404 0.0
405 1.0
 ——— Blank card to end case data read-in ———
    6     SEARCH FOR LRC CASE 1 LEAVES 7/24/73, ARRIVES 2/16/74
111 0.0       112 21000.0   113 3400.0
146 0.0001    175 100.0
301 10.0      302 2.0       303 1.0       304 2.0
305 0.0       306 2.0       307 -1.0      308 0.0
309 3.0       310 11001.0
344 100.0     345 5.0       348 0.01
351 13.0      373 -4930.0   393 10.0
352 14.0      374 7200.0    394 10.0
353 20.0      380 20614.30  400 100.0
354 0.0
418 0.0       419 3.0
 ——— Blank card to end case data read-in ———
```

* A standard card form, IBM electro 888157, is available for punching source statements from this form.

WDL-2673

Figure 2-2. SAMPLE INPUT FOR THE START-UP AND SEARCH OPTIONS departure date (7307.24, 0. means 0^h, July 24, 1973)

arrival date (7402.16, 0. means 0^h, February 16, 1974)

although it is also necessary to have specified the number of matched-conic iterations to be performed (X(405) = 1.0) and the desired miss vector components at the target (B·T = X(373) =-4930.0 km and B·R = X(374) = 7210. km). Parking orbit parameters at launch must also be specified.

insertion latitude (X(121) = 28.5°)
insertion longitude (X(122) =-80.5°)
insertion azimuth (X(123) = 67.74°)
insertion altitude (X(124) = 184. km)

These parameters enable the program to compute specific injection conditions which are meaningful relative to the parking orbit assumption.

An output sample resulting from the input example of Figure 2-2 is shown in Figure 2-3 and described in section 2.2.1.

2.1.2 INPUT FOR SEARCH

The search option is called with "6" in column 5 of the first card. Case data are then read into the WCOM array through subroutine ROVLEY. The principal data required for a search case are:

initial date and time (locations 99 and 100 or 101 and 102)
initial trajectory state (311-316, 317-322, or 323-328)
control selector (310)
control limit levels (331-336, 337-342, or 343-348)
constraint selectors (351-356)
desired constraint values (361-380)
constraint error tolerances (381-400)
option selectors (scattered locations)

The complete list of input data required by SEARCH is found in Table 3.

An input example is shown in Figure 2-2. The user must supply the launch and target body numbers unless these have been supplied to the program in an earlier PINT, PLANET or SEARCH case of the same run. The input example for PLANET shows how to set IN(1) = 1 and IN(2) = 5 which tells the program that the launch body is Earth and the target is Mars. The SEARCH case shown was run after a PLANET case, so the launch and target bodies were not re-specified. The next line (card) shows locations 111, 112 and 113. These represent (respectively) the time (0. DH.MS) from initial date at which the variation or search is to begin, the precautionary stop time (21000. DH.MS = 210 days), and the precautionary distance (3400. km) from the target at which the trajectory must stop if neither stop time nor closest approach has been reached. The next card contains the fraction of control limits to use in generating partials (X(146) = .0001) and an upper constraint tolerance factor (X(175) = 100.) used in iterative testing. The next two cards specify eight options.

(301) 10. means that the iteration is to be terminated after 10 iterations even if convergence has not been achieved

(302) 2. means that the program is to compute its own factors for scaling the gradient

(303) 1. means that the gradient is to be computed by finite differences in FNDMXN rather than being supplied

(304) 2. means that the iterative steps are to be controlled by numerical estimation of curvature when minimizing

(305) 0. asks for no extra output

(306) 2. means that the gradient is to be used for two more iterations in addition to the one in which it was computed

(307) -1. cancels extra output of initial conditions at each step

(308) 0. asks for the patched-conic trajectory model

The next card sets 3. into location 309 and 11001. into location 310. This 3. means that the control set is to be the launch parameter set for which initial conditions are located in 323-238. These initial conditions are omitted from input because they have been set by the earlier PLANET case. Other parameters which would have to be input for SEARCH if they were not set in the PLANET case are the park orbit insertion date (locations 99 and 100) and the park orbit specifications (locations 121-124). The 11001. in location 310 specifies the control variables to be launch time, park time and injection velocity impulse. The next card specifies the control limit levels on launch time (100. seconds), park time (5. seconds) and injection velocity impulse (.01 km/sec). The next four cards specify the constraint set, desired values and tolerances.

Selection	Desired Value	Tolerance
(351) 13. for B·T	(373) B·T is -4930. km	(393) B·T is 10. km
(352) 14. for B·R	(374) B·R is 7200. km	(394) B·R is 10. km
(353) 20. for flight time	(380) t_f is $206^d\ 14^h\ 30^m\ 0^s$	(400) t_f is 100. seconds
(354) 0. ends the constraint set length at three		

The next card selects the search option (S(418) = 0.) as opposed to the scanning option and selects the target-fixed constraint coordinate reference system (X(419) = 3.). A blank card ends the input and leads to execution of the case. An output example resulting from this input example is shown in Figure 2-3 and described in Section 2.2.2.

2.2 PROGRAM OUTPUT

Print-out from the Mark IV Error Propagation Program is basically the same as that of the Mark II program except, of course, for the addtion of start-up and search output. Examples of the Mark II program's print-out are given and explained in Section 5 of Reference 1. Examples of the start-up and search print-outs will now be given.

2.2.1 OUTPUT FOR THE START-UP CAPABILITY

The start-up capability's output is computed and printed primarily in PLANET. Figures 2-3 and 2-4 show this print-out. The first block of print-out for any case shows the data input for that case (see 2.1.1) and is labeled "OVERLAY INPUT." The massless planet solution is then headed "APPROXIMATE TRAJECTORY DATA." Later iterations are headed "MATCHED-CONIC ITERATIVE

```
                                   OVERLAY INPUT
INTEGERS FROM     1 TO     2
   1      5
  121   2.85000000E+01     122  -8.05000000E+01     123   6.77400000E+01     124   1.84000000F+02
  373  -4.93000000E+03     374   7.20000000E+03      -0  -0                   -0  -0.
  401   7.30724000E+03     402   0.                 403   7.40216000E+03     404   0.
  405   1.00000000E+00      -0  -0.                  -0  -0.                  -0  -0.

                                   APPROXIMATE TRAJECTORY DATA

                                                                   MEAN ECLIPTIC OF LAUNCH DATE
   DEPARTS FROM EARTH  (7387.24,    0.00) = JD 2441887.500     RAD   1.016    LAT   -.000    LON  -59.040
     ARRIVES AT MARS   (7402.16,    0.00) = JD 2442094.500     RAD   1.577    LAT   1.266    LON   92.527

                                   HELIOCENTRIC ORBIT DATA

                         TRANSFER TIME         2 7.000 DAYS
                         TRANSFER ANGLE       151.541 DEGS
                         APHELION RADIUS        1.658 A.U.
                         PERIHELION RADIUS      1. 15 A.U.
                         ORBITAL PERIOD       564.518 DAYS
                         TRUE ANOMALY          -4.654 DEGS
                         ASCENDING NODE       - 9.037 DEGS
                         INCLINATION            2.657 DEGS
                         ARG. OF PERIHELION     -.003 DEGS
                         INC WRT LAUNCH ORB     2.656 DEGS
                         INC WRT TARGET ORB     3.685 DEGS

                                   ASYMPTOTIC DATA
                 LAUNCH                                              ARRIVAL

       HYP EXCESS SPEED        3.88  KM/S           HYP EXCESS SPEED       2.876 KM/S
       C3 (ENERGY)            14.100 K2/S2          C3 (ENERGY)            8.118 K2/S2
       LATITUDE, ECLIPTIC     23.149 DEGS           LATITUDE, ECLIPTIC   -28.620 DEGS
       LATITUDE, ORBITAL      23. 79 DEGS           LATITUDE, ORBITAL    -28.273 DEGS
       DECLINATION            35.4   DEGS           DECLINATION          -12.6 2 DEGS
       LONGITUDE, ECL,EQU     36.943 DEGS           LONGITUDE, ECL,EQU    39. 31 DEGS
       LONGITUDE, ORB,RAD     95.98  DEGS           LONGITUDE, ORB,RAD   -54.405 DEGS
       RIGHT ASCENSION        25.51  DEGS           RIGHT ASCENSION       45.65  DEGS
       SUN-ASYMPTOTE ANG.     84.50  DEGS           SUN-ASYMPTOTE ANG.   128.763 DEGS

                                   SPECIFIC LAUNCH DATA
  INSERTION DATE AND PARK ORBIT DESCRIPTION
555 JUL 24, 1973,  8 HRS, 33 MIN, 21.00  SEC                                     JULIAN DATE  2441887.8565 347
   ALT 1.84000000E+02  LAT 2.85000000F+01  LON-8.05000000E+01  VAZ 6.77400000E+01  INC 3.55781455F+01  GHA 7.02635960E+01

  INJECTION CONDITIONS AT EARTH
    JUL 24,  973,  9 HRS, 33 MIN, 48.329 SEC                                     JULIAN DATE  2441887.898476-4
  DLAZ 0.             ATL 0.             PRKT 3.62642995E+03 DVAZ-6.08372755E-14 DVEL-2.16617572F-13 DELV 3.85420124E+00
     X-2.19226738E+03     Y-5.27320672F+03     Z-3.23732326E+03   DX 9.29520820E+00   DY-6.05152129E+00   DZ 3.55712069E+00
     R 6.56216500E+03   DEC-2.95597933F+01    RA-1.12585795F+02    V 1.16479575E+01  PTH-4.73746518E-09   AZ 6.94464798E+01
   SMA-2.80911406E+04   ECC 1.23362265F+00   INC 3.54048832E+01  LAN 3.00178066E+02  APF 3.01758644E+02  RCA 6.56216500E+03
    C3 1.41896410E+01 THET-8.57097428F-09 PERV 1.16479575F+01   SLR 1.46572692E+04 IMPV 3.85420124F+00 TPER-9.75418947E-13
```


Figure 2-3. OUTPUT FOR START-UP

```
                        MATCHED-CONIC ITERATIVE SOLUTION

                                                        MEAN ECLIPTIC OF LAUNCH DATE
DEPARTS FROM EARTH  (7307.24,    0.00) = JD 2441887.500      RAD   1.016    LAT   -.000    LON  -56.193
  ARRIVES AT MARS   (7402.16,    0.00) = JD 2442094.500      RAD   1.575    LAT    1.240   LON   91.447

                           HELIOCENTRIC ORBIT DATA

                        TRANSFER TIME         201.806 DAYS
                        TRANSFER ANGLE        147.385 DEGS
                        APHELION RADIUS         1.658 A.U.
                        PERIHELION RADIUS       1. 15 A.U.
                        ORBITAL PERIOD        564.6 7 DAYS
                        TRUE ANOMALY           -1.531 DEGS
                        ASCENDING NODE        -56. 93 DEGS
                        INCLINATION             2.6 7 DEGS
                        ARG. OF PERIHELION      - 001 DEGS
                        INC WRT LAUNCH ORB      2.645 DEGS
                        INC WRT TARGET ORB      3.6 3 DEGS

                             ASYMPTOTIC DATA
              LAUNCH                                        ARRIVAL

    HYP EXCESS SPEED       3.863 KM/S           HYP EXCESS SPEED       2.874 KM/S
    C3 (ENERGY)            14. 64 K2/S2         C3 (ENERGY)            8.1 8 K2/S2
    LATITUDE, ECLIPTIC     2 .156 DEGS          LATITUDE, ECLIPTIC   -28.556 DEGS
    LATITUDE, ORBITAL      23.154 DEGS          LATITUDE, ORBITAL    -28 213 DEGS
    DECLINATION            35.434 DEGS          DECLINATION          - 2.593 DEGS
    LONGITUDE, ECL,EQU     36.777 DEGS          LONGITUDE, ECL,EQU    39.135 DEGS
    LONGITUDE, ORB,RAD     92.979 DEGS          LONGITUDE, ORB,RAD   -53 3 9 DEGS
    RIGHT ASCENSION        25.33  DEGS          RIGHT ASCENSION       45.7 6 DEGS
    SUN-ASYMPTOTE ANG.     87. 77 DEGS          SUN-ASYMPTOTE ANG.   1 1.771 DEGS

                              SPECIFIC LAUNCH DATA
INSERTION DATE AND PARK ORBIT DESCRIPTION
  JUL 24,  973,  8 HRS, 36 MIN, 43.939 SEC                                   JULIAN DATE  2441887.85884189
 ALT 1.84000000E+02  LAT 2.85000000E+01  LON-8.05000000E+01  VAZ 6.77400000E+01  INC 3.55781455E+01  GHA 7.11077322E+01

INJECTION CONDITIONS AT EARTH
  JUL 24, 1973,  9 HRS, 36 MIN, 56. 23 SEC                                   JULIAN DATE  2441887.90064957
DLAZ 0.             DTL 0.          PRKT 3.61219369E+03 DVAZ-5.81535991E-14 DVEL-2.01229914E-13 DELV 3.84878950E+00
  X-2.20352443E+03   Y-5.24475221F+03  Z-3.27094234E+03  DX 9.31418175E+00  DY-6.06960156F+00  DZ 3.45757520E+00
  R 6.56216500E+03 DEC-2.98978199F+01  RA-1.12789171E+02  V 1.16425457E+01  PTH-5.20503818E-09  AZ 6.99665914E+01
 SMA-2.83429017E+04 ECC 1.23152764F+00 INC 3.54658713E+01 LAN 3.01024531E+02  APF 3.00786337E+02 RCA 6.56216500E+03
  C3 1.40635989E+01 THET-9.43204724F-09 PERV 1.16425457E+01 SLR 1.46436525E+04 IMPV 3.84878950F+00 TPER-1.07391222E-12
```


Figure 2-4. OUTPUT FOR START-UP

SOLUTION." The next two lines show the departure and arrival dates and planetary positions at those dates. The coordinate system for position output here is the mean ecliptic and equinox of launch date. Radii are printed in a.u. and latitudes and longitudes in degrees. Heliocentric orbit data are printed next to describe the heliocentric transfer trajectory. Transfer time and transfer angle refer to the interval between departure patch and arrival patch - and the sphere of influence has zero radius for the massless planet solution. The only other explanations needed for this block are that "TRUE ANOMALY" refers to the heliocentric orbit at departure patch and that the ascending node, inclination and argument of perihelion are referred to the mean ecliptic and equinox of launch date. The "ASYMPTOTIC DATA" describe the hyperbolic excess velocities at departure (LAUNCH) and arrival. Declination and right ascension of the asymptotic (hyperbolic excess velocity vector) are referred to the Earth's mean equator and equinox of launch date. The only other explanation needed is that the sun-asymptotic angle is the angle between the planet-to-sun line and the hyperbolic excess velocity vector.

"SPECIFIC LAUNCH DATA" describe the park orbit, insertion and injection conditions which connect the parking orbit with the departure hyperbola. Insertion date is the time of park orbit initiation. Injection conditions refer to the initial conditions for the departure hyperbola. The reference frame is EE50 for all cooordinatized data. The following list explains the remaining symbols.

ALT	Parking orbit altitude (km)
LAT	Parking orbit insertion latitude (deg)
LON	Parking orbit insertion longitude (deg)
VAZ	Parking orbit insertion velocity azimuth (deg)
INC	Parking orbit inclination (equator of date, deg)
GHA	Greenwich hour angle at insertion date
DLAZ	Incremental insertion azimuth control (rad)
DTL	Incremental launch time (sec) measured from insertion date

PRKT	Time in parking orbit (sec)
DVAZ	Azimuth of the injection impulse measured CCW from the parking orbit at injection (rad)
DVEL	Elevation of the injection impulse measured up from local horizontal at injection (rad)
DELV	Magnitude of the injection velocity impulse (km/sec)
X,Y,Z	Cartesian position components at injection (km)
DX,DY,DZ	Cartesian velocity components at injection (km/sec)
R,DEC,RA	Spherical position components at injection (km,deg)
V,PTH,AZ	Spherical velocity components at injection (km/sec, deg)
SMA	Semi-major axis (km)
ECC	Eccentricity (no units)
INC	Inclination (deg)
LAN	Longitude of the ascending node (deg)
APF	Argument of perifocus (deg)
RCA	Radius of closest approach (km)
C3	Vis-viva energy (km^2/sec^2)
THET	True anomaly at injection (deg)
PERV	Periapsis velocity (km/sec)
SLR	Semi-latus rectum (km)
IMPV	Impulsive velocity required to circularize (km/sec)
TPER	Time after periapsis passage (days)

2.2.2 OUTPUT FOR THE SEARCH CAPABILITY

The standard output for the search capability is shown in Figure 2-5. Input case data are printed in the block labeled "OVERLAY INPUT." The next few lines are printed in SETUP and describe the control-constraint parameters in effect. For the case shown, the control variables are time of launch (DTL), park time (PRKT) and velocity impulse (DELV). Initial values and limit steps corresponding to the control variables are shown

```
                                OVERLAY INPUT
111    .                  112   2.10000000E+04    113   3.40000000E+03      -0  -0.
146    1.00000000E-04     175   1.00000000E+02     -0  -0.                   -0  -0.
301    1.00000000E+01     302   2.00000000E+00    303   1.00000000E+00     304  2.00000000E+00
305    .                  306   2.00000000E+00    307  -1.00000000E+00     308  0.
309    3.00000000E+00     310   1.10010000E+04     -0  -0.                   -0  -0.
344    1.00000000E+02     345   5.00000000E+00    348   1.00000000E-02      -0  -0.
351    1.30000000E+01     373  -4.93000000E+03    393   1.00000000E+01      -0  -0.
352    1.40000000E+01     374   7.20000000E+03    394   1.00000000E+01      -0  -0.
353    2.00000000E+01     387   2.06143000E+04    400   1.00000000E+02      -0  -0.
354    .                   -    -0.                -0  -0.                   -0  -0.
-18    .                  .19   3.10000000E+00     -0  -0.                   -0  -0.

                   TARGET THE TR JECTORY FROM EARTH  TO MARS

          CONTROL PARAMETERS         .  INITIAL VALUES            LIMIT LEVELS
                 DTL                        .    .                1.00000000E+02
                 PRKT                   3.61218369E+03            5.00000000E+00
                 DEL                    3.84878950E+00            1.00000000E-02
                    .
            CONSTRAINTS                 DESIRED V LUES            TOLER NCES
                 RDI                   -4.93000000E+03          . 1.00000000E+01
                 BDN                    7. 0000000E+03            1.00000000E+01
                 TFL                    2 06143000E+04     .      1.00000000E+02


  JUL 24, 973,  8 HRS, 36 MIN, 43.030 SEC                                   JULIAN DATE  2441887.85884189
  JUL 24, 1973,  9 HRS, 36 MIN, 56.123 SEC                                  JULIAN DATE  2441887.90064957
DLAZ 0.          .        DTL 0.           PRKT 3.61218369E+03 DVAZ-5.81535991E-14 DVEL-2.11229914E-13 DELV 3.84878950E+00
  X-2.20352500E+03   Y-5.24475172E+03   Z-3.27074213E+03   DX 9.31410075E+00   DY-6.06060360E+00   DZ 3.45757431E+00
  R 6.56216500E+03  DEC-2.98978178E+01  RA-1.12749186E+02   V 1.16425457E+01  PTH-5.20549293E-09   AZ 6.99665972E+01
SMA-2.83429017E+04  ECC 1.23152764E+00  INC 3.54056666E+01  LAN 3.01024523E+02  APF 3.00786332E+02  RCA 6.56216500E+03
 C3 1.40635989E+01 THET-9.43323371E-09 PERV 1.16425457E+01  SLR 1.46436525E+04 IMPV 3.84878950E+00 TPER-1.07404731E-12

ITER. NO.  0     CONTROLS      0.00000000  3612.18369054     3.8487895
               CONSTRAINTS     -517.2982   -21549.8421       6643.3125
               CONTROL INC     86.22052403   -4.2270839      -.00032891

ITER. NO.  1     CONTROLS      86.22052403  3607.95011015    3.84846159
               CONSTRAINTS       39.9275      -29 .7345       847.5896
               CONTROL INC    -21.67043382    1.1715 149     .00002387

ITER. NO.  2     CONTROLS      64.54989 21  3609.2705174     3.84848446
               CONSTRAINTS      -5.3740       135.37 9       -61.5628
       .       CONTROL INC     .10105013     -.0025 0516     .00000115

ITER. NO.  3     CONTROLS      64.65094134  3609.2225658     3.84848501
               CONSTRAINTS      -.3185         -.7553        -9.0895


                     ***** IT CONVERGES IN  3 ITERATIONS *****

                        INITIAL CONDITIONS EARTH -CENTERED
  JUL 24, 1973,  9 HRS, 37 MIN, 7.612 SEC          .                        JULIAN DATE  2441887.90136125
DLAZ 0.                   DTL 6.46509403E+01 PRKT 3.60912226E+03 DVA -5.81535991E-14 DVEL-2.11229914E-13 DELV 3.84848561E+00
  X-2.19853238E+03   Y-5.24225713E+03   Z-3.27829300E+03   DX 9.32786807E+00   DY-6.06046953E+00   DZ 3.43560492E+00
  R 6.56216500E+03  DEC-2.99718804E+01  RA-1.12752524E+02   V 1.16422418E+01  PTH-5.27325028E-09   AZ 7.00833003E+01
SMA-2.83571696E+04  ECC 1.23141114E+00  INC 3.54061880E+01  LAN 3.01295380E+02  APF 3.00570289E+02  RCA 6.56216500E+03
 C3 1.40565228E+01 THET-9.55585741E-09 PERV 1.16422418E+01  SLR 1.46428881E+04 IMPV 3.84848561E+00 TPER-1.08803738E-12

                    .

                        TERMINAL CONDITIONS MARS  -CENTERED
. FEB 16, 1974,   HRS,  8 MIN,  6.772 SEC                                   JULIAN DATE  2442094.50563312
  X 4.77601957E+03   Y-7.75845309E+02   Z 8.25203127E+02   DX 1.06165184E+00   DY 4.60308009E+00   DZ-1.81675496E+00
  R 4.91848848E+03  DEC 9.67838827E+00  RA-9.22087224E+00   V 5.06123004E+00  PTH-4.54747351E-13   AZ 1.11354544E+02
SMA-5.30300326E+03  ECC 1.92560540E+00  INC 2.33516961E+01  LAN 1.94040315E+02  APF 1.54903842E+02  RCA 4.90848848E+03
 C3 8.11442647E+00 THET-7.64854187E-13 PERV 5.06123004E+00  SLR 1.43603004E+04 IMPV 2.10220798E+00 TPER-1.49842142E-16
```


Figure 2-5. OUTPUT FOR SEARCH

in their appropriate units, i.e., seconds, seconds and kilometers/second respectively. The constraints for this case are $B \cdot \hat{T}$, $B \cdot \hat{R}$ and time of flight. The desired values and tolerances for $B \cdot \hat{T}$ and $B \cdot \hat{R}$ are in units of kilometers, while the desired value of flight time is printed out in the (days)(hours).(minutes)(seconds) format used for input. The flight time constraint tolerance is in units of seconds. Only flight time of the 20 constraints has mixed units for desired value and tolerance.

The insertion date is next printed if launch control parameters are selected, followed by the initial estimate of injection date and the initial values of the entire launch control parameter set (see 2.2.1 for explanation of symbols). If either cartesian or spherical controls had been selected, the insertion date would not have appeared and the starting date would be printed in the calendar format used for input. The input initial state would be printed out next. In either case, initial cartesian, spherical and orbital components of the trajectory are next printed out in EE50 coordinates. (See 2.2.1 for symbol definition and units.)

A number of blocks are next printed to show the iteration history. The iteration number is followed by the control values used for that iteration. The order and meaning of these values follows that of the earlier print-out of control variables in effect, i.e. DTL, PRKT and DELV. The constraint errors are printed next in the earlier-stated order. The error is defined as "desired value minus current value." Units of the constraint errors are the same as input units except for flight time, where seconds are seen. The third line in the block shows the control increments as calculated by the program to improve the constraint error. These increments are in the same order and units as the control values. When a control increment is the same size as the limit level, a limited control step will be taken. The iteration is considered to be convergent when the constraint errors are each smaller than the corresponding tolerances.

The final two blocks of print-out show (1) the initial date and initial state which correspond to the solution control set, and (2) the final date and final state which were determined by the solution control set and the trajectory model in effect. The coordinate system for both of these blocks is EE50. The symbols and units for these blocks are identified in 2.2.1 although the comment "at injection" no longer necessarily applies.

The user may obtain extra output by setting the key in location 305 non-zero. In this case the time and state conditions at trajectory initiation and at closest approach to the target body are printed out. The latter are printed in the selected constraint coordinate frame at the end of each trajectory calculation. Another line then prints out the computed values of the quantities listed below.

BDT	$B \cdot \hat{T}$ (km)
BDR	$B \cdot \hat{R}$ (km)
TFL	Time of flight (sec)
DECS	Latitude of the arrival asymptote (deg)
RAS	Longitude of the arrival asymptote (deg)
RAT	Longitude of the $\hat{T}$-vector (deg)

The extra output also includes the scaled gradient,

$$H = \frac{\partial \psi}{\partial X}$$

and the factors by which it is scaled. The elements of the gradient may be obtained by dividing the scale factor into each element of the column to which it belongs.

Extra output may also be obtained for integrated trajectories by setting the parameters in locations 115 and/or 116 non-zero. This extra output is primarily used for de-bugging purposes.

REFERENCES

(1) Users Manual for Mark II Error Propagation Program, Philco WDL-TR2758, 15 February 1966.

(2) Subroutine Descriptions and Listings for Mark II Error Propagation Program and Powered Flight Optimization and Error Analysis Programs, Philco WDL-TR2757, Volumes I and II, 15 February 1966.

(3) "The Application of State Space Methods to Navigation Problems," Stanley F. Schmidt, Philco WDL Guidance and Control System Engineering Technical Report 4, July 1964.

(4) Program Description and Theoretical Basis for the Orbit Determination Program, Philco-Ford TR-DA1508 dated December 1967.

TABLE 1

CELESTIAL BODY NUMBER CODE

1	Earth
2	Moon
3	Sun
4	Venus
5	Mars
6	Saturn
7	Jupiter

TABLE 2

INPUTS FOR START-UP

Case Card: Put 5 in column 5, anything or nothing thereafter

ROVLEY cards: Integer or real data ended by a blank card

Location	Name		Units
IN(1)	IB1	Launch body number (see Table 1)	None
2	IB2	Target body number (see Table 1)	None
X(121)	BOLAT	Park orbit insertion latitude	Degrees
122	BOLON	Park orbit insertion longitude	Degrees
123	BOVAZ	Park orbit insertion azimuth	Degrees
124	BOALT	Park orbit insertion altitude	Kilometers
373	BT	Desired miss vector component, $B \cdot \hat{T}$	Kilometers
374	BR	Desired miss vector component, $B \cdot \hat{R}$	Kilometers
401	DL1	Launch date (year, month, day)	YM.D
402	DL2	Launch date (hour, minute, second)	HM.S
403	DA1	Arrival date (year, month, day)	YM.D
404	DA2	Arrival date (hour, minute, second)	HM.S
405	YET	Number of matched-conic iterations	None
412	DLAZ	Incremental insertion azimuth	Radians

Table 3

INPUTS FOR SEARCH

Case card: Put 6 in column 5, anything or nothing thereafter

ROVLEY cards: Integer or real data ended by a blank card

Location	Name	Meaning	Units
IN(1)	IB1	Launch body number (see Table 1)	none
2	IB2	Target body number (see Table 1)	none
X(101)	YW	Initial date (year, month, day)	YM.D
102	YF	Initial date (hour, minute, second)	HM.S
111	TSECI	Time from input date above at which the search is to originate (used only for cartesian or spherical searches)	DH.MS
112	TSTP	Time at which the trajectory is to stop even if the target body has not yet been encountered	DH.MS
113	RSTP	Distance from the target body at which the trajectory computation ends even if closest approach or stop time have not yet occurred	kilometers
146	ESX	Fraction of control limits to be used in generation of partials	none
175	EPSM	Upper constraint tolerance factor	none
301	TRIES	Maximum number of iterations before giving up the search or the number of scanning steps	none
302	SCOPT	Scaling option key for gradient computation (use 2. for automatic scaling)	none
303	GROPT	Gradient option key (use 1. for automatic differencing)	none
304	COPT	Convergence control option key (use 2.)	none

Table 3 (continued)

Location	Name	Meaning	Units
305	XTRA	Extra output key for gradient and target periapsis conditions (0. to suppress, 1. to obtain output)	none
306	SAVG	Number of trials to use the gradient after the trial on which it is computed	none
307	XWOUT	Extra output key for initial conditions (0. for extra, -1. for none)	none
308	HOW	Trajectory option key (0. for patched-conic model, 1. for perturbed, integrated)	none
309	TYPE	Control set option key (1. cartesian, 2. spherical, 3. launch parameters)	none
310	XNI	Control selector key ($\sum_1^6 N_i (10)^{i-1}$ where N_1 is assigned by the following chart:	none

N_i	TYPE = 1.	TYPE = 2.	TYPE = 3.
N_6	X	r	DLAZ
N_5	Y	lat	DTL
N_4	Z	lon	PRKT
N_3	VX	v	DVEL
N_2	VY		DVAZ
N_1	VZ	az	DELV

N_i = 0. omits the i-th variable from the control set and N_i = 1. includes it. For example, 11001. selects the second, third and sixth controls.)

Table 3 (continued)

IF TYPE = 1.

Location	Name	Meaning	Units
311	XIN	Initial cartesian state values	kilometers
312			kilometers
313			kilometers
314			km/sec
315			km/sec
316			km/sec
331	XLIM	Cartesian control limit levels (see XNI)	kilometers
332			kilometers
333			kilometers
334			km/sec
335			km/sec
336			km/sec

IF TYPE = 2.

Location	Name	Meaning	Units
317	XIN	Initial spherical state values	kilometers
318			degrees
319			degrees
320			km/sec
321			degrees
322			degrees
337	XLIM	Spherical control limit levels (see XNI)	kilometers
338			degrees
339			degrees

Table 3 (continued)

Location	Name	Meaning	Units
340			km/sec
341			degrees
342			degrees
		IF TYPE = 3.	
99	YWIN	Park orbit insertion date (year, month, day)	YM.D
100	YFIN	Park orbit insertion date (hour, minute, second) (YW, YF in locations 101, 102 need not be set)	HM.S
121	BOLAT	Park orbit insertion latitude	degrees
122	BOLON	Park orbit insertion longitude	degrees
123	BOVAZ	Park orbit insertion azimuth	degrees
124	BOALT	Park orbit insertion altitude	kilometers
323	XIN	Initial launch parameter values	radians
324			seconds
325			seconds
326			radians
327			radians
328			km/sec
343	XLIM	Launch control limit levels (see XNI)	radians
344			seconds
345			seconds
346			radians
347			radians
348			km/sec

Table 3 (continued)

CONSTRAINT SPECIFICATION

Location	Name	Meaning	Units
351	PNI	Constraint selector key (a number between 1. and 20. inclusive selects each constraint, 0. ends the list - see folded sub-table below)	none
352			
353			
354			
355			
356			
357			

PNI	Name	Constraint	Desired Value (PSID) Location	Tolerance (OK) Location	Units
1.	X	Cartesian state	361	381	kilometers
2.	Y		362	382	kilometers
3.	Z		363	383	kilometers
4.	VX		364	384	km/sec
5.	VY		365	385	km/sec
6.	VZ		366	386	km/sec
7.	R	Spherical state	367	387	kilometers
8.	Lat		368	388	degrees
9.	Lon		369	389	degrees
10.	V		370	390	km/sec
11.	PTH		371	391	degrees
12.	AZM		372	392	degrees

Table 3 (continued)

PNI	Name	Constraint	Desired Value (PSID) Location	Tolerance (OK) Location	Units
13.	BT	Miss vector components	373	393	kilometers
14.	BR		374	394	kilometers
15.	HEV	Hyperbolic excess velocity	375	395	km/sec
16.	RCA	Radius of closest approach	376	396	kilometers
17.	INC	Inclination	377	397	degrees
18.	LAN	Longitude of the asc. node	378	398	degrees
19.	APF	Argument of peri-focus	379	399	degrees
20.	TFL	Flight time	380	400	DH.MS, sec

Location	Name	Meaning	Units
419	COORD	Constraint coordinate selector key (1. EE50 2. target's orbital system, R, RxVxR, RxV 3. target's body-fixed 4. ecliptic and equinox)	none

INTEGRATED TRAJECTORIES (HOW = 1.)

Location	Name	Meaning	Units
6	CDEQ	Output interval (set it large)	seconds
13	TSTEP	Factor for computing heliocentric integration step size (try .005)	none
115	EXTRA	Extra de-bug output key (use 0.)	none
116	XTRREF	Extra output key for rectification and patch conditions (0. suppress, 1. include)	none

Table 3 (continued)

Location	Name	Meaning	Units
		SCANNING OPTION	
418	SCAN	Scanning option key (0. for normal search, n. for stepping the n-th selected control (see TYPE, XNI) TRIES times in steps of limit level (see XLIM)	none

APPENDIX A

MATHEMATICAL DESCRIPTION OF THE START-UP CAPABILITY

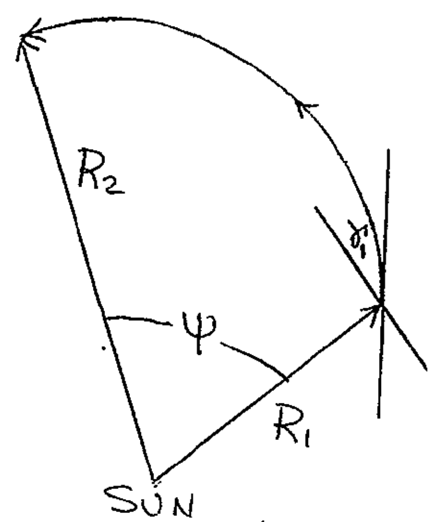


Figure A-1 Heliocentric Transfer Geometry

The fundamental equation solving Lambert's Problem for the heliocentric conic joining two (massless) planets is (A.1).

$$p = \frac{r_1 r_2 (1 - \cos\psi)}{(r_1 - r_2 \cos\psi + r_2 \sin\psi \tan\gamma_1)} \tag{A.1}$$

In the equation, p is the orbit's semi-latus rectum, r_1 and r_2 the heliocentric radii of the launch and target planets, ψ is the transfer angle and γ_1 is flight path angle at launch. The flight path angle, γ_1, is varied until p results in the correct transfer time through Kepler's Equation. The hyperbolic excess velocity, S, at launch is computed by A.2.

$$S = V_1 - V_L \tag{A.2}$$

where V_1 is the heliocentric transfer orbit's velocity at launch and

V_L is the launch planet's velocity. Launch time (if the launch body is Earth) is the time when S is contained in the Earth-fixed parking orbit plane, i.e., the time when (by Earth's rotation),

$$H \cdot S = 0, \tag{A.3}$$

where H is normal to the parking orbit. The true anomaly, θ_S, of the hyperbolic excess velocity vector, S, is given by

$$\cos \theta_S = \frac{1}{e} \tag{A.4}$$

where e is eccentricity,

$$e = 1 - \frac{r_p}{a} \tag{A.5}$$

where r_p is parking orbit radius and where a is semi-major axis of the departure hyperbola.

$$a = - \frac{\mu}{|S|^2 - \frac{2\mu}{r_{patch}}} \tag{A.6}$$

Knowing θ_S, then, the angle and time in parking orbit are easily determined. The radius vector, R, relative to the departure hyperbola is next calculated and used in the determination of the velocity, V, required at injection to attain S.

$$V = \frac{\sqrt{C_3}}{2} \left\{ \left[\sqrt{1 - \frac{4a}{r(1+\hat{R}\cdot\hat{S})}} - 1 \right] \hat{R} + \left[\sqrt{1 - \frac{4a}{r(1+\hat{R}\cdot\hat{S})}} + 1 \right] \hat{S} \right\} \tag{A.7}$$

In A.7, C_3 is $-\frac{\mu}{a}$, r is $|R|$ and $\hat{R}$ and $\hat{S}$ are unit R and S. The injection impulse, ΔV, is the difference between V and the parking orbit velocity, V_p.

$$\Delta V = V - V_p \tag{A.8}$$

The launch control parameter set is completed by computation of the azimuth, elevation and magnitude of ΔV.

$$DVAZ = \tan^{-1}\left(\frac{\Delta V \cdot \hat{H}}{\Delta V \cdot (\hat{H} x \hat{R})}\right) \tag{A.9}$$

$$DVEL = \tan^{-1}\left(\frac{\Delta V \cdot \hat{R}}{\sqrt{(\Delta V \cdot \hat{R}^2) + (\Delta V \cdot \hat{H})^2}}\right) \tag{A.10}$$

$$DELV = |\Delta V| \tag{A.11}$$

The launch control parameters may be used to re-compute cartesian components of R and V through subroutine START.

The matched-conic iteration scheme uses R and V to compute the state, R^* and V^*, and time, t_1^*, at the sphere of influence. The heliocentric radius vector, R_1^*, is the sum of R^* and the launch body's heliocentric position at patch time, R_L.

$$R_1^* = R^* + R_L \tag{A.12}$$

At the other end of the heliocentric trajectory, the desired values of B·T and B·R are used in computing radius of closest approach and eccentricity of the arrival hyperbola. These permit computation of the time, t_2^*, of patch to the target's sphere of influence and the target body's heliocentric position at that time, R_T. The hyperbolic excess velocity at arrival, S, enables computation of unit vectors, $\hat{T}$ and $\hat{R}$, in the miss plane. The desired miss-vector, B, and the target-centered position vector, R^*, at target-patch are then computed.

$$B = (B \cdot \hat{T})\,\hat{T} + (B \cdot \hat{R})\,\hat{R} \qquad (A.13)$$

$$R^{*} = -\sin(\theta+\alpha)\,\hat{B} + \cos(\theta+\alpha)\,\hat{S} \qquad (A.14)$$

In A.14, θ is the negative true anomaly at patch on the arrival hyperbola and α is the half-angle between the asymptotes. The heliocentric position at patch, R_2^{*}, is computed by A.15.

$$R_2^{*} = R^{*} + R_T \qquad (A.15)$$

By using R_1^{*}, R_2^{*}, t_1^{*} and t_2^{*} instead of R_1, R_2 and the original departure and arrival dates, the iteration on equation A.1 finds the heliocentric trajectory between spheres of influence (patch points). This trajectory provides improved estimates of departure and arrival hyperbolic excess velocities which may then be used to calculate launch control parameters and perhaps initiate another conic-matching iteration.

APPENDIX B

MATHEMATICAL DESCRIPTION OF THE SEARCH CAPABILITY

The control variables for the search capability are selected from one of the three sets shown below.

AVAILABLE CONTROLS

Cartesian	Spherical	Launch Parameters
X	R	DLAZ
Y	LAT	DTL
Z	LON	PRKT
VX	V	DVEL
VY	PTH	DVAZ
VZ	AZ	DELV

The initial conditions of the trajectory are completely specified by any one of these sets appropriately transformed into cartesian EE50 components. Let S(t) represent the cartesian EE50 state vector at time t, let X_i represent the control set for the i-th trial and let Y (a constant vector) complete the set required to compute $S(t_o)$. That is,

$$S_i(t_o) = F(X_i, Y) \tag{B.1}$$

where F represents the transformation which maps $X_i \cup Y$ into S at t_o. The transformation is performed by subroutine CONVX for the cartesian or spherical sets and by START for the launch control set (see Reference 2 for details).

The state at t is derived from $S(t_o)$ by patched-conic trajectory calculation or by numerical integration of the perturbed equations of motion.

Let this trajectory generation function be denoted by G.

$$S(t) = G(S(t_o)) \tag{B.2}$$

The final time, t, may represent the occurrence of a significant trajectory event such as closest approach to the target as well as a specified time after t_o. See Appendix B of Reference 1 and Appendix C of this report for details relative to equation B.2.

The constraint vector, ψ , is a function of $S(t)$.

$$\psi = P(S(t)) \tag{B.3}$$

The components of ψ are user-selected from an available set of 20 constraint functions. Subroutine ENDCON computes the relationship, P, indicated by B.3. The constraint functions are related to the controls, X, through the transformation or "plant" indicated by equations B.1, B.2 and B.3. The gradient, H, which represents (linearly) the sensitivity of ψ to changes in X is computed by the method of finite differences.

$$H^j = \frac{\partial\psi}{\partial X^j} = \frac{\psi(X+\Delta X^j) - \psi(X)}{\Delta X^j} \tag{B.4}$$

Equation B.4 symbolically represents the method of computing the j-th column of H which is the sensitivity of ψ to the j-th control variable, X^j. X^j is incremented by X^j to form a new control vector, $X+ X^j$. This control vector is used, through B.1, B.2 and B.3, to compute a new $\psi(X+\Delta X^j)$. The sensitivity, H^j is then computed as in B.4. When H has been completed by computation of each of its columns, it is used as shown in B.5 to compute the control set for the (i+1)-th trial.

$$X_{i+1} = X_i + H^T(HH^T)^{-1}[\psi_D - \psi(X_i)] \tag{B.5}$$

In B.5, ψ_D is the desired constraint vector as specified by the user.

The available constraints are shown below.

AVAILABLE CONSTRAINTS

Cartesian		Spherical		Other			
1	X	7	r	13	B·T	19	ω_p
2	Y	8	lat	14	B·R	20	t_F
3	Z	9	lon	15	v_∞		
4	VX	10	v	16	r_p		
5	VY	11	γ	17	i		
6	VZ	12	az	18	Ω		

Most of the available constraints are defined relative to some specific coordinate frame. The coordinatization is implied in B.3 and must be uniform for all elements of ψ. Available coordinates include Earth's mean equator and equinox of 1950.0, target body orbital, target-fixed and ecliptic-equinox.

APPENDIX C

NUMERICAL INTEGRATION

This appendix describes the mathematical theory of numerical integration used in the Mark IV Error Propagation Program. It is implemented in subroutine DEQS. This appendix is taken directly from Reference 4 and still contains equation sequence numbers from that document.

We now consider the integration of the equations of motion and variational equations for a given set of parameters, U, and given initial conditions $R(t_1)$, $V(t_1)$. All the equations may be considered as the vector equation

$$\ddot{X} = f(X, \dot{X}) \tag{D. 4-1}$$

In any numerical integration process, we approximate the integral $X(t)$ at a sequence of points, t_i, on the integration interval, (t_o, t_n) obtaining the $X(t_i)$ from some approximation of the Taylor's series

$$X\left(t_{i+1}\right) = X(t_i) + h\dot{X}(t_i) + \frac{1}{2}h^2\ddot{X}(t_i) + \frac{1}{6}h^3\dddot{X}(t_i) + \ldots$$

$$h = t_{i+1} - t_i \tag{D. 4-2}$$

At any t_i, the second derivative may be determined from the differential equation, and the higher order derivatives must be developed implicitly from the known derivative at neighboring points. The various methods differ in the way in which the series (D. 4-2) is approximated.

The ODP uses Adams' method, which approximates the series using the values of $f(X, \dot{X})$ computed at the previous integration points, t_{i-1}, t_{i-2}, etc., for long-term integration, and a generalized Kutta method for short-term integration and for starting the Adams' integration. The Kutta method uses values of $f(X, \dot{X})$ at suitably chosen points on the internal (t_i, t_{i+1}). The two methods are described below.

D. 4. 1 Adams' Method

We assume that the quantities

$$X_i = X(t_i)$$

$$\dot{X}_i = \dot{X}(t_i) \qquad \text{(D. 4-3)}$$

$$f_i = f(X_i, \dot{X}_i)$$

have been determined at the sequence of equally spaced points

$$t_{n-m} = t_n - mh$$

$$m = 0, 1, \ldots, N \qquad \text{(D. 4-4)}$$

We write the Taylor's series

$$f(t_n + sh) = f_n + sh\,\dot{f}(t_n) + \frac{1}{2} s^2 h^2 \ddot{f}(t_n) + \ldots \qquad \text{(D. 4-5)}$$

truncating after terms in $(sh)^N$. The coefficients of the resulting Nth degree polynomial is s may be determined to satisfy the $N + 1$ conditions.

$$f(t_n - mh) = f_{n-m} \qquad \text{(D. 4-6)}$$

The polynomial is usually written in terms of the backward differences

$$\nabla f_n = f_n - f_{n-1}$$

$$\nabla^2 f_n = \nabla f_n - \nabla f_{n-1} \qquad \text{(D. 4-7)}$$

$$\nabla^{p+1} f_n = \nabla^p f_n - \nabla^p f_{n-1}$$

and hence

$$f^{(o)}_{n+s} = \sum_{k=0}^{N} a_k(s)\, \nabla^k f_n$$

$$a_o = 1 \qquad \text{(D. 4-8)}$$

$$a_k = \frac{1}{k!} s(s+1) \ldots (s+k-1), \quad k \geq 1$$

The error in approximation on the interval (t_i, t_{i+1}) is

$$f(t_n + sh) - f^{(o)}_{n+s} = a_{N+1}(s)\, h^{N+1}\, f^{(N+1)}(\xi) \qquad \text{(D. 4-9)}$$

$$t_n - N \le \xi \le t_n + sh$$

D. 4. 1. 1 Integration Formulas. If we substitute the polynomial (D. 4-8) into the integral relationships

$$\dot{X}_{n+s} = X_n + \int_{t_n}^{t_n+sh} f\left(X(t), \dot{X}(t)\right) dt .$$

$$X_{n+s} = \dot{X}_n + \int_{t_n}^{t_n+sh} \int_{t_n}^{t} f\left(X(t), \dot{X}(t)\right) dt\ dt \qquad \text{(D. 4-10)}$$

we obtain the approximations

$$\dot{X}^{(o)}_{n+s} = \dot{X}_n + h \sum_{k=0}^{N} A_k(s)\, \nabla^k f_n$$

$$X^{(o)}_{n+s} = X_n + sh\dot{X}_n + h^2 \sum_{k=0}^{N} B_k(s)\, \nabla^k f_n \qquad \text{(D. 4-11)}$$

where

$$A_k(s) = \int_0^s a_k(t)\, dt$$

$$B_k(s) = \int_0^s A_k(t)\, dt \qquad (D. 4-12)$$

with errors

$$\dot{X}_{n+s} - \dot{X}_{n+s}^{(o)} = A_{N+1}(s)\, h^{N+2} f^{(N+1)}(\xi_o)$$

$$X_{n+s} - X_{n+s}^{(o)} = B_{N+1}(s)\, h^{N+3} f^{(N+1)}(\eta_o) \qquad (D. 4-13)$$

$$t_{n-N} \leq \xi_o,\ \eta_o \leq t_n + sh$$

since $a_{N+1}(s)$, $A_{N+1}(s)$ do not change sign on (0, 1). These formulas resulting from extrapolation are termed open. An alternative form of the polynomial

$$f_{n+s}^{(1)} = \sum_{k=o}^{N} a_k(s-1)\, \nabla^k f_{n+1} \qquad (D. 4-14)$$

yields the closed formulas

$$X_{n+s}^{(1)} = X_n + h \sum_{k=o}^{N} C_k(s)\, \nabla^k f_{n+1}$$

$$X_{n+s}^{(1)} = X_n + sh\dot{X}_n + h^2 \sum_{k=o}^{N} D_k(s)\, \nabla^k f_{n+1}$$

$$C_k(s) = \int_o^s a_k\,(t-1)\,dt \qquad \text{(D. 4-15)}$$

$$D_k(s) = \int_o^s C_k\,(t)\,dt$$

with errors

$$\dot{X}_{n+s} - \dot{X}_{n+s}^{(1)} = C_{N+1}(s)\; h^{N+2}\, f^{(N+1)}(\xi_1)$$

$$X_{n+s} - X_{n+s}^{(1)} = D_{N+1}(s)\; h^{N+3}\, f^{(N+1)}(\eta_1) \qquad \text{(D. 4-16)}$$

The closed formulas require knowledge of $f(X_{n+1}, \dot{X}_{n+1})$ for the determination of X_{n+1}, $\dot{X}_{n+1}$, and hence may be used directly only in simple quadrature. For the integration of differential equations, they must be used in conjunction with formulas for the prediction of X_{n+1}, $\dot{X}_{n+1}$. The obvious solution is to use the open formulas as predictors to compute estimates $X_{n+1}^{(o)}$, $\dot{X}_{n+1}^{(o)}$, and to use the closed formulas as correctors. Evaluating the coefficients at $s = 1$,

$$X_{n+1} = \dot{X}_{n+1}^{(o)} + A_{n+1}\, h^{N+2} f^{(N+1)}(\xi_o)$$

$$= X_{n+1}^{(1)} + C_{N+1}\, h^{N+2}\, f^{(N+1)}(\xi_1) + \sum_{k=o}^{N} C_k\, h(f_{n+1} - f_{n+1}^{(o)})$$

$$X_{n+1} = X_{n+1}^{(o)} + B_{N+1} h^{N+3} f^{(N+1)}(\eta_o)$$

(D. 4-17)

$$= X_{n+1}^{(1)} + D_{N+1} h^{N+3} f^{(N+1)}(\eta_1) + \sum_{k=o}^{N} D_k h^2 (f_{n+1} - f_{n+1}^{(o)})$$

If we assume that h is sufficiently small that $h(f_{n+1} - f_{n+1}^{(o)})$ is negligible compared with $h^{N+2} f^{(N+1)}(\xi)$, and that $f^{(N+1)}(\xi)$ varies only slowly with ξ, we may eliminate $\dot{X}_{n+1}$, X_{n+1}, obtaining

$$h^{N+2} f^{(N+1)}(\xi) = (X_{n+1}^{(1)} - X_{n+1}^{(o)}) / (A_{N+1} - C_{N+1}) \tag{D. 4-18}$$

Using the easily established relations

$$A_k(s) = A_{k+1}(s) - C_{k+1}(s)$$

(D. 4-19)

$$\nabla^k f_n = \nabla^k f_{n+1} - \nabla^{k+1} f_{n+1}$$

we have

$$h^{N+1} f^{(N+1)}(\xi) \; \nabla^{N+1} f_{n+1} \tag{D. 4-20}$$

and hence our best estimate of the integrals is

$$\dot{X}_{n+1} = \dot{X}_{n+1}^{(o)} + hA_{N+1} \nabla^{N+1} f_{n+1}^{(o)}$$

$$X_{n+1} = X_{n+1}^{(o)} + h^2 B_{N+1} \nabla^{N+1} f_{n+1}^{(o)}$$

The integration coefficients are listed through k=8 in Table D-2, below.

TABLE D-2

ADAMS' INTEGRATION COEFFICIENTS

k	A_k	B_k	C_k	D_k
0	1	$\frac{1}{2}$	1	$\frac{1}{2}$
1	$\frac{1}{2}$	$\frac{1}{6}$	$-\frac{1}{2}$	$-\frac{1}{3}$
2	$\frac{5}{12}$	$\frac{3}{24}$	$-\frac{1}{12}$	$-\frac{1}{24}$
3	$\frac{9}{24}$	$\frac{38}{360}$	$-\frac{1}{24}$	$-\frac{7}{360}$
4	$\frac{251}{720}$	$\frac{135}{1440}$	$-\frac{19}{720}$	$-\frac{17}{1440}$
5	$\frac{475}{1440}$	$\frac{863}{10080}$	$-\frac{27}{1440}$	$-\frac{82}{10080}$
6	$\frac{19087}{60480}$	$\frac{9625}{120960}$	$-\frac{863}{60480}$	$-\frac{731}{120960}$
7	$\frac{36799}{120960}$	$\frac{135812}{1814400}$	$-\frac{1375}{120960}$	$-\frac{8563}{1814400}$
8	$\frac{1070017}{3628800}$	$\frac{515529}{7257600}$	$-\frac{33953}{3628800}$	$-\frac{17719}{7257600}$

D.4.1.2 Interpolation. To obtain f, $\dot{X}$, X at points other than integration points, we may use the polynominals (D.4-8), (D.4-11). Setting

$$t = t_n + sh$$

$$F_k = h^k \frac{d^k f(t_n)}{dt^k} = \frac{d^k f(t_n)}{ds^k} \qquad \text{(D. 4-22)}$$

we obtain

$$f(t) = \sum_{k=o}^{N} F_k s^k/k!$$

$$\dot{X}(t) = X_n + h\sum_{k=o}^{N} F_k s^{k+1}/(k+1)! \qquad (D.\,4\text{-}23)$$

$$X(t) = X_n + sh\,X_n + h^2 \sum_{k=o}^{N} F_k s^{k+2}/(k+2)!$$

and for s on the interval (-1, 0), the derivatives F_k are obtained from

$$\begin{bmatrix} F_o \\ F_1 \\ F_2 \\ F_3 \\ F_4 \\ F_5 \\ F_6 \\ F_7 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & \frac{1}{2} & \frac{1}{3} & \frac{1}{4} & \frac{1}{5} & \frac{1}{6} & \frac{1}{7} \\ 0 & 0 & 1 & 1 & \frac{11}{12} & \frac{5}{6} & \frac{137}{180} & \frac{7}{10} \\ 0 & 0 & 0 & 1 & \frac{3}{2} & \frac{7}{4} & \frac{15}{8} & \frac{29}{15} \\ 0 & 0 & 0 & 0 & 1 & 2 & \frac{17}{6} & \frac{7}{2} \\ 0 & 0 & 0 & 0 & 0 & 1 & \frac{5}{2} & \frac{25}{6} \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 3 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} f_n \\ \nabla f_n \\ \nabla^2 f_n \\ \nabla^3 f_n \\ \nabla^4 f_n \\ \nabla^5 f_n \\ \nabla^6 f_n \\ \nabla^7 f_n \end{bmatrix} \qquad (D.\,4\text{-}24)$$

where all differences after the N^{th} are to be set zero.

D.4.1.3 Change of Interval Size. For a set of differences $\nabla^k f_n$ for the spacing h, we may compute an equivalent set $\nabla^k \bar{f}_n$ for any spacing sh, so that the interpolation polynominals for the two sets are identical in t. Two particular changes may be made rather simply, for s = 1/2 and s = 2, and these changes provide all the spacing flexibility required.

Using (D.4-22), we have for s = 1/2,

$$
\begin{bmatrix} \bar{f}_n \\ \nabla \bar{f}_n \\ \nabla^2 \bar{f}_n \\ \nabla^3 \bar{f}_n \\ \nabla^4 \bar{f}_n \\ \nabla^5 \bar{f}_n \\ \nabla^6 \bar{f}_n \\ \nabla^7 \bar{f}_n \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ & \frac{1}{2} & \frac{1}{8} & \frac{1}{16} & \frac{5}{128} & \frac{7}{256} & \frac{21}{1024} & \frac{33}{2048} \\ & & \frac{1}{4} & \frac{1}{8} & \frac{5}{64} & \frac{7}{128} & \frac{21}{512} & \frac{33}{1024} \\ & & & \frac{1}{8} & \frac{3}{32} & \frac{9}{128} & \frac{7}{128} & \frac{45}{1024} \\ & & & & \frac{1}{16} & \frac{1}{16} & \frac{7}{128} & \frac{3}{64} \\ & & & & & \frac{1}{32} & \frac{5}{128} & \frac{10}{256} \\ & & & & & & \frac{1}{64} & \frac{3}{128} \\ & & & & & & & \frac{1}{128} \end{bmatrix} \begin{bmatrix} f_n \\ \nabla f_n \\ \nabla^2 f_n \\ \nabla^3 f_n \\ \nabla^4 f_n \\ \nabla^5 f_n \\ \nabla^6 f_n \\ \nabla^7 f_n \end{bmatrix} \quad \text{(D.4-25)}
$$

and for S = 2,

$$
\begin{bmatrix} \bar{f}_n \\ \nabla \bar{f}_n \\ \nabla^2 \bar{f}_n \\ \nabla^3 \bar{f}_n \\ \nabla^4 \bar{f}_n \\ \nabla^5 \bar{f}_n \\ \nabla^6 \bar{f}_n \\ \nabla^7 f_n \end{bmatrix}
=
\begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 & 2 & -1 & 0 & 0 & 0 & 0 & 0 \\
 & & 4 & -4 & 1 & 0 & 0 & 0 \\
 & & & 8 & -12 & 6 & -1 & 0 \\
 & & & & 16 & -32 & 24 & -8 \\
 & & & & & 32 & -80 & 80 \\
 & & & & & & 64 & -192 \\
 & & & & & & & 128
\end{bmatrix}
\begin{bmatrix} f_n \\ \nabla f_n \\ \nabla^2 f_n \\ \nabla^3 f_n \\ \nabla^4 f_n \\ \nabla^5 f_n \\ \nabla^6 f_n \\ \nabla^7 f_n \end{bmatrix}
\qquad (D.4\text{-}26)
$$

D.4.1.4 Ordinate Formulas. The use of difference formulas has some computational disadvantages. At each integration point, a complete set of differences must be computed, and the old set must be retained until the integration accuracy is verified. More efficient computation results from direct use of the computed ordinates. The corresponding formulas may be obtained from the relations

$$
\nabla^k f_n = \sum_{m=0}^{k} \frac{(-1)^m \, m!}{k! \, (k-m)!} f_{n-m} \qquad (D.4\text{-}27)
$$

The various coefficients depend upon N as well as on k. For N = 5, the integration formulas are:

$$
\dot{X}^{(0)}_{n+4} = \dot{X}_n + \frac{h}{10080} \Big[29939 f_n - 55461 f_{n-1} + 69874 f_{n-2} - 51086 f_{n-3} + 20139 f_{n-4} - 3325 f_{n-5} \Big] \qquad (D.4\text{-}28)
$$

$$X_{n+1}^{(0)} = X_n + h\dot{X}_n + \frac{h^2}{10080}\Big[10852f_n - 15487f_{n-1} + 18752f_{n-2}$$

$$- 13474f_{n-3} + 5260f_{n-4} - 863f_{n-5}\Big]$$

$$\dot{X}_{n+1} = \dot{X}_{n+1}^{(0)} + \frac{19087h}{60480}\nabla^6 f_{n+1}$$

$$X_{n+1} = X_{n+1}^{(0)} + \frac{9625h^2}{120960}\nabla^6 f_{n+1}$$

$$\nabla^6 f_{n+1} = f_{n+1}^{(0)} - 6f_n + 15f_{n-1} - 20f_{n-2} + 15f_{n-3} - 6f_{n-4} + f_{n-5} \qquad \text{(D.4-28) (Contd.)}$$

The interpolation formulas are

$$\begin{bmatrix} F_0 \\ F_1 \\ F_2 \\ F_3 \\ F_4 \\ F_5 \end{bmatrix} = \frac{1}{60}\begin{bmatrix} 60 & 0 & 0 & 0 & 0 & 0 \\ 137 & -300 & 300 & -200 & 75 & -12 \\ 225 & -770 & 1070 & -780 & 305 & -50 \\ 255 & -1065 & 1770 & -1470 & 615 & -105 \\ 180 & -840 & 1560 & -1440 & 660 & -120 \\ 60 & -300 & 600 & -600 & 300 & -60 \end{bmatrix}\begin{bmatrix} f_n \\ f_{n-1} \\ f_{n-2} \\ f_{n-3} \\ f_{n-4} \\ f_{n-5} \end{bmatrix} \qquad \text{(D.4-29)}$$

D.4.2 Generalized Kutta Method

The various methods called Kutta or Runge-Kutta methods are based on a process suggested by Runge (Reference 7) and developed for first order equations by Kutta (Reference 8). Applied to second order equations, the method requires evaluation of the derivative f at the sequence of points:

$$t_n = t_o + C_n h$$

$$\dot{X}_n = \dot{X}_o + h \sum_{i=0}^{n-1} C_{ni} f_i$$

$$X_n = X_o + h \left(b_n \dot{X}_o + h \sum_{i=0}^{n-1} d_{ni} f_i \right)$$

$$C_n = \sum_{i=0}^{n-1} C_{ni} \tag{D. 4-30}$$

where t_o is the initial point on the integration interval. The process (D. 4-30) is repeated through N substitutions $(n = 0, 1, \ldots, N-1)$, and $\dot{X}(t_o+h)$, $X(t_o+h)$ are then approximated by the N + 1 st values in the sequence. Appropriate values of the coefficients are determined by matching as many as possible of the leading terms of the series (D. 4-2) with those of the series obtained by substituting the Taylors series for $f(X, \dot{X})$ into the sequence (D. 4-30).

Several methods have been developed for integrating (D. 4-1) and for using two adjacent intervals for computing truncation error, interpolating between interval end-points, etc. Miachin (Reference 9) treated the special case $\ddot{X} = f(X)$, obtaining accuracy through terms in h^5 and an expression for the truncation error in $X(t_o+h)$ using derivatives computed on the two intervals (t_o, t_o+h) and (t_o+h, t_o+2h). In two unpublished communications, T. W. Hinton* treated the case $\ddot{X} = f(X, \dot{X})$, obtaining accuracy through h^5 for the case $\partial f/\partial \dot{X} = 0$ and through h^4 for the general case. Hinton also gave an expression for the truncation error in $X(t_o+h)$ and equations for interpolating on the interval pair (t_o, t_o+h) and (t_o+h, t_o+2h).

* Lockheed Missiles and Space Co., Sunnyvale, California, August 1963.

D.4.2.1 Integration Formulas. The coefficients given by Hinton are:

$$C_1 = b_1 = 3/10 \qquad d_1 = 9/200$$

$$C_2 = b_2 = 3/4 \qquad d_2 = 9/32$$

$$C_3 = b_3 = C_4 = b_4 = 1 \qquad d_3 = d_4 = 1/2$$

$$C_{20} = -21/32 \qquad d_{20} = 0$$

$$C_{21} = 45/32 \qquad d_{21} = 9/52$$

$$C_{30} = 83/27 \qquad d_{30} = 10/27$$

$$C_{31} = -280/81 \qquad d_{31} = 7/162$$

$$C_{32} = 112/81 \qquad d_{32} = 14/81$$

$$C_{40} = 5/54 \qquad d_{40} = 5/54$$

$$C_{41} = 250/567 \qquad d_{41} = 25/81$$

$$C_{42} = 32/81 \qquad d_{42} = 8/81$$

$$C_{43} = 1/14 \qquad d_{43} = 0 \qquad \text{(D. 4-31)}$$

If we denote by $f_{i,1}$ and $f_{i,2}$ the values calculated for $f(X_i, \dot{X}_i)$ on the intervals (t_o, t_o+h) and (t_o+h, t_o+2h), respectively, the truncation error in $X(t_o+2h)$, assuming no error in $X(t_o+h)$ is

$$T = \frac{h^2}{34020} \Big[81f_{3,2} + 112f_{2,2} - 550f_{1,2} - 2478f_{0,2} + 1134f_{3,4} + 3248f_{2,1} - 2450f_{1,1} + 903f_{0,1} \Big] \qquad \text{(D. 4-32)}$$

This term is of order h^5 and the error in the approximation is of order h^6 for the general case.

D.4.2.2 Interpolation. Linear combinations of the f_n may be used to interpolate for $f, \dot{X}, X$ on the interval (t_o, t_o+2h). We again set

$$S = \left(t - [t_o+2h]\right)/h$$

$$F_k = \frac{d^k f}{dS^k} \qquad \text{(D. 4-33)}$$

obtaining the interpolation formulas (D. 4-23). The F_k are given by

$$\begin{bmatrix} F_o \\ F_1 \\ \frac{1}{2}F_2 \\ \frac{1}{6}F_3 \end{bmatrix} = \frac{1}{1134}\begin{bmatrix} 486 & 1344 & -1200 & 1638 & -486 & -1344 & 1200 & -504 \\ 1620 & 2912 & -8900 & 8904 & -2106 & -5600 & 5900 & -2730 \\ 1458 & 2016 & -9900 & 9828 & -1944 & -4704 & 6900 & -3654 \\ 324 & 448 & -2200 & 1428 & -324 & -448 & 2200 & -1428 \end{bmatrix}\begin{bmatrix} f_{3,2} \\ f_{2,2} \\ f_{1,2} \\ f_{0,2} \\ f_{3,1} \\ f_{2,1} \\ f_{1,1} \\ f_{0,1} \end{bmatrix} \qquad \text{(D. 4-34)}$$

The expression yields accuracy through h^3, h^4, h^5 for f, $\dot{X}$, X respectively for the general case $f(X, \dot{X})$.

D.4.2.3 Conversion to Adams' Ordinates. As we noted earlier, some starting process is required to accumulate the necessary ordinates for the Adams' integration. The necessary ordinates are computed by the ODP by interpolation on a single interval pair integrated in the Kutta mode. To avoid extrapolation beyond the interval pair in computing $\nabla^3 f_n$, the highest significant difference obtainable, we take $S = 1/2$ and set $\nabla^i f_n = 0$ for all $j \geq 4$. We have

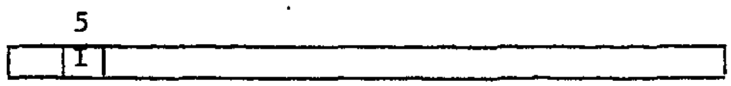
$$\begin{bmatrix} f_n \\ \nabla f_n \\ \nabla^2 f_n \\ \nabla^3 f_n \end{bmatrix} = \frac{1}{756} \begin{bmatrix} 324 & 896 & -800 & 1092 & -324 & -896 & 800 & -336 \\ 324 & 672 & -1500 & 1449 & -405 & -1120 & 1000 & -420 \\ 324 & 448 & -2200 & 2562 & -486 & -1344 & 1200 & -504 \\ 162 & 224 & -1100 & 714 & -162 & -224 & 1100 & -714 \end{bmatrix} \begin{bmatrix} f_{3,2} \\ \cdot \\ \cdot \\ f_{0,1} \end{bmatrix} \tag{D. 4-35}$$

APPENDIX D

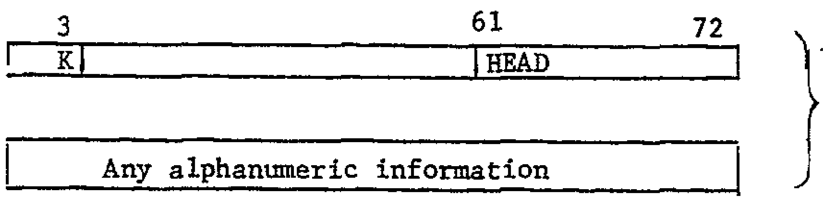
INPUT SUMMARY

I CARD FORMATS

D.1.1 General Formats

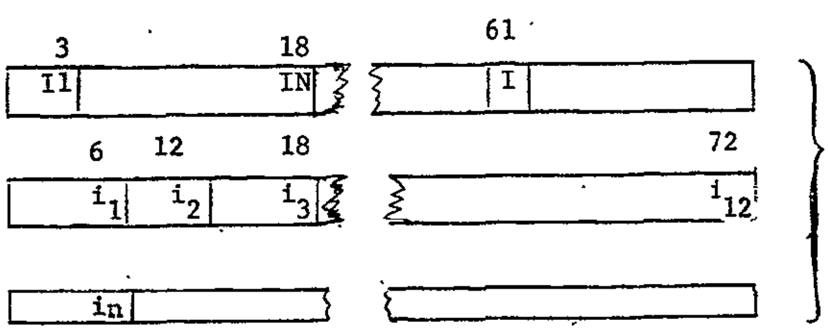


EXEC
I = 1 PINT
= 2 ERP
= 3 SPOUT
= 4 CONW
= 5 START-UP
= 6 SEARCH
= 0 terminate run and wrap up tapes.



HEADER Cards

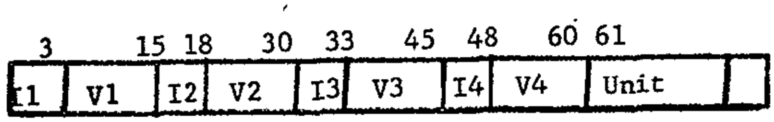
K = 0 or 1



Integer cards

I1 = 1st subscript
IN = end subscript

OVERLAY cards



(4(I3,E15.8),A6)

I_i = subscript for
V_i = value.

Unit = FT
FT/SEC
NM
NM/SEC
SQUARE
(or blank).

D.1.2 Special ERP Formats

Overlay card with I1 = NCH
V1 = TSTART
V2 = TSTOP
V3 = OINTV

NCH Control times card

NCH ≥0, process measurements from TSTART to TSTOP.

NCH >0, read processing options or changes.

NCH =0, no changes to read.

NCH <0, no measurements to be processed.

NCH=111, stop.

1		7	9	11	13	15	…		55	57	
N	NAME	i_1	i_2	i_3	i_4		…		i_{24}	i_{25}	

Change card
N = type of change

1, stations
2, beacons
3, onboard
4, equation of motion

i_i = error source treatment code.

(See ERP section, tables 1, 2, 4, 6, 8, 10).

Overlay Card

Data Cards

(Station, beacon, etc.)

(See ERP section, tables 3, 5, 7, 9).

II INPUT ORDER AND DESCRIPTIONS

D.2.1 PINT

Input for a PINT case consists of:

1. EXEC card (1 in column 5)
2. C-array data, ending with blank. This includes the header for the vehicle ephemeris tape both when reading and writing this tape (K = 0 format).
3. IN, X-array data ending with blank.

Both blanks must be included, whether or not any data is being read in.

INPCOM (C-ARRAY) PARAMETERS RELEVANT TO PRECISION INTEGRATION

Address	Name	Dimension	Description
C(8)	ASTU	1	Conversion factor, kilometers/a.u.
C(21)	UM	10	Gravitational constants (km^3/sec^2).
C(31)	RPL	10	Planetary semi-major axes or radii (km), Earth's radius value used in gravity and park orbit calculations.
C(51)	RPAT	10	Sphere of influence radii (km) or distance at which transfer is made to or from each body.
C(71)	WE	1	Earth's sidereal rotation rate (rad/sec).

C(585)	EMP	24	Indicators for equation of motion parameters to consider for computing sensitivities. These indicators are set 1. in INPCOM block data A zero in EMP(I) asks for the sensitivity of state to EMP(I).
C(609)	ZH	4	Zonal harmonic coefficients, J_i, of the Earth's gravitational field (dimensionless).
C(613)	TH	10	The TH array contains 10 tesseral harmonics ordered J_{21}, λ_{21}, J_{22}, λ_{22}, J_{31}, λ_{31}, J_{32}, J_{33}, λ_{33}. J_{nm} is the m^{th} tesseral harmonic coefficient of the n^{th} order, (no units); and λ_{nm} is the geographic longitude corresponding to J_{nm}.
C(623)	ABC	3	Mass-normalized moments of inertia of the moon used for computing the triaxial gravity perturbation (km^2).
C(626)	DRAGC	2	Atmospheric drag coefficients of the model $-c_1 e^{-c_2 h} \mid V \mid \vec{V}$. Both have units of ($km^{-1}$).
C(628)	TMAG	1	Magnitude of thrusting acceleration (km/sec^2).
C(629)	SPK	1	Solar radiation pressure coefficient (km^3/sec^2).
C(630)	DUTD	1	Discrepancy between ephemeris time and universal time (days).

C(672)	ICASE	1	Case counter or trajectories generated and written on the vehicle ephemeris tape by a PINT run. Set initially zero by INPCOM BLOCK DATA, ICASE is incremented automatically and written on tape, so that subsequent tape-using routines may select a desired trajectory by this identification number.
C(675)	HEAD	12	Alphanumeric header written on the vehicle ephemeris tape, and used by reading routines to identify which tape to read.

FIXED-POINT WCOM (IN-ARRAY) PARAMETERS FOR THE PRECISION INTEGRATION OPTION

Address	Name	Dimension	Description
IN(1)	IB1	1	Launch body number or initial central body (1 Earth, 2 Moon, 3 Sun, 4 Venus, 5 Mars, 6 Saturn, 7 Jupiter).
IN(2)	IB2	1	Intended target body number.
IN(3)	KREF	1	KREF < 0 signals no refine, but compute the state transition matrix. KREF = 0 signals no refine or state transition matrix, but trajectory calculation and maybe tape-write. KREF = 1 signals refinement option but do not save the solution. KREF = 2 signals refine and save.

IN(4)	NRIPBO		Tape-write or sensitivity computation indicator (0) for neither, (1) for writing a precision trajectory on tape, (-K) for integrating variational equations from case #K on tape for motion parameter sensitivities.
IN(10)	NDEQ	9	Integration package indicators. (see DEQ writeup for definitions--need not be changed for normal operation.)
IN(19)	IBC	10	Bodies to be considered as perturbing force centers, (1) consider, (0) omit.
IN(29)	NH	5	Tesseral harmonic gravity indicators NH(1) is the order of the highest zonal harmonic to be used (for J_{50} use 5). NH(2-5) is the highest degree longitudinal harmonic to be included for each order (NH(3) = 2 means use J_{31} and J_{32}).

FLOATING-POINT WCOM (X-ARRAY) PARAMETERS FOR THE PRECISION INTEGRATION OPTION

Address	Name	Dimension	Description
X(1)	CDEQ	8	Integration parameters CDEQ(1) = current step size computed by program. CDEQ(2) = first (next) time at which output is desired (sec), reset each case. CDEQ(3) = initial step size, set by program. CDEQ(4) = doubling limit, set by program. CDEQ(5) = halving limit, set by program. CDEQ(6) = output time interval (sec) starting at CDEQ(2). CDEQ(7) = minimum divisor for relative error. CDEQ(8) = upper bound allowed on truncation error, run time increasing as CDEQ(8) is decreased.
X(9)	RECT	1	Conic rectification tolerances. Reference conic section for Encke's method is changed when ratio of position deviation to radius from central body exceeds RECT.
X(10)	RNØØB	1	Radius from the Earth at which oblateness ceases to be considered (km).
X(11)	TSTEP	10	Fraction of a radian to use in computing the initial integration step size (Δt_o = TSTEP$\cdot\left(\frac{r}{v}\right)$ rounded to 2^n for each central body).

X(21)	ERRTØL	1	Relative error tolerance for interpolating when writing vehicle ephemeris tape.
X(22)	XTRTAP	1	Key for extra output when writing ephemeris: (0.) no extra output, (1.) extra output.
X(23)	DTPREC	1	Time interval over which Earth's precession matrix may be assumed constant (sec).
X(24)	DTNUT	1	Time interval over which Earth's nutation matrix may be assumed constant (sec).

NORMAL TRAJECTORY INPUT DATA

Address	Name	Dimension	Description
X(101)	YW	1	Year, month and day of injection (YM.D, e.g. 6311.16 is Nov 16, 1963.)
X(102)	YF	1	Hour, minute and second of injection (HM.S. e.g. 1408.1409 is $14^h 8^m$ 14.09^s).
X(103)	BØDNØ	1	Body number to which the injection state is relative (1. Earth, 2.Moon, etc).
X(104)	TYPIN	1	Input state coordinate type in the form $A.10^2 + B.10 + C$ where A, B and C are interpreted as follows: A = 0, Cartesian; A = 1, Spherical, A = 2, Orbital Elements; B = 0, Equator, B = 1, Ecliptic; B = 2, Body-fixed; C = 0, Epoch of 1950.0; C = 1, Epoch of Date (101. means Spherical, Equator and Equinox of Date).
X(105)	RZ	3	Cartesian position vector (km) if TYPIN < 100. Radius (km), latitude (deg), longitude (deg) if 100. ≤ TYPIN < 200. Orbital elements if TYPIN > 200: Semi-major axis (km), eccentricity, true anomaly (deg).

X(108)	VZ	3	Cartesian velocity vector (km/sec) if TYPIN < 100. Spherical velocity if 100. ≤ TYPIN < 200: Speed (km/sec), path angle (deg), azimuth (deg). Orbital elements if TYPIN ≥ 200: Longitude of the node (deg), inclination (deg), argument of periapsis (deg).
X(111)	TSECI	1	Trajectory starting time from YW and YF. (Days.10^2 + Hours + Minutes.10^{-2} + Seconds.10^{-4}).
X(112)	TSTP	1	Trajectory stopping time from YW and YF. (Days.10^2 + Hours + Minutes.10^{-2} + Seconds.10^{-4}).
X(113)	RSTP	1	Trajectory stopping radius from IB2, the target body (km).
X(114)	OYTP	1	Trajectory output indicator: (0.) minimum output, (1.) Equator and Equinox of 1950.0,(-1.) Equator and Equinox of date.
X(115)	XTRA	1	Extra output key for stopping functions in FSUB and DERIV: (0.) omit, (1.) print extra output.
X(116)	XTRREF	1	Extra output key for REFINE information:(0.) none,(1.) print extra information.

REFINE INPUT DATA

Parking Orbit Data

Location	Name	Description
X(120)	STARTK	(1.) Asks for "Earth-fixed" park orbit. (2.) Asks for "Inertial" park orbit.
X(121)	DLAT	Insertion latitude ($\pm$ 90 deg) measured positive northward from the equator of insertion epoch.
X(122)	DLON	Insertion longitude ($\pm$ 180 deg) measured positive eastward from the X-axis or Greenwich Meridian.
X(123)	DVAZ	Insertion velocity azimuth ($\pm$ 180 deg) measured positive clockwise from local north at insertion.
X(124)	PALT	Altitude at insertion and of the circular parking orbit (km).
X(125)	ORBN	Number of whole parking orbits after insertion and before injection (floating-point integer).
X(128)	YWINS	Year, month and day of park orbit insertion (format YM.D, two digits each), for example 6608.03 is Aug. 3, 1966).
X(129)	YFINS	Hour, minute and second of parking orbit insertion (format HM.S; for example 1348.0533 is $13^h\ 48^m\ 5.33^s$UMT).

Control Parameters

X(130)	TL	Insertion time increment from nominal insertion date (sec).
X(131)	PARKT	Time in parking orbit before injection (sec).
X(132)	AZIM	Azimuth of the injection velocity impulse (rad) relative to the parking orbit plane at injection measured counter-clockwise about the injection radius vector.

Location	Name	Description
X(133)	PTH	Elevation of the injection velocity impulse (rad) measured up from the horizontal at injection.
X(134)	DELTAV	Magnitude of the injection velocity impulse (km/sec).

Control Limit Levels

Location	Name	Description
X(140)	TIMAX	Insertion time, maximum increment (sec).
X(141)	PMAX	Park time, maximum increment (sec).
X(142)	AZMAX	Injection impulse azimuth, maximum increment (rad).
X(143)	ELMAX	Injection impulse elevation, maximum increment (rad).
X(144)	ENMAX	Post-injection energy (vis-viva), maximum increment (km^2/sec^2).
X(145)	TRY	Number of trials (iterations) allowed in REFINE if convergence is not obtained earlier.
X(146)	SX	Fraction of control limit levels to be used for computing numerical partials.
X(147)	SCALE	Distance to which end point variational positions are to be scaled before using the variational state to compute constraint partials (km).

Desired Constraints

Location	Name	Description
X(150)	FLTIME	Indicator: (0.) neither flight time nor target-relative energy are constrained, (1.) constrain flight time, (-1.) constrain arrival energy.
X(151)	TFL	Desired time of flight to closest approach or impact from park orbit insertion (Days Hours. Minutes Seconds).
X(152)	C3D	Desired energy (Vis-Viva) relative to the target body at arrival (km^2/sec^2).

Location	Name	Description
X(153)	TLAT	Target vector latitude measured positive toward the z-axis of the target coordinate system (± 90 deg).
X(154)	TLON	Target vector longitude measured positive toward the y-axis from the x-axis of the target coordinate system (± 180 deg).
X(155)	PRD	Periapsis radius desired at radius of closest approach (km).
X(156)	PACTIM	Indicator: (0.) no impact of the target body, (1.) impact the target, ignoring PRD above.
X(157)	HOWB	Indicator: constrain -1., only flight time or energy (see FLTIME) 0., TLAT, TLON, PRD, see FLTIME 1., PRD, see FLTIME 2., BDTD, BDRD, see FLTIME
X(158)	BDTD	Desired B•T component if HOWB=2. (km)
X(159)	BDRD	Desired B•R component if HOWB=2. (km)

Earth Return Constraints

Location	Name	Description
X(160)	DLATE	Latitude of the desired touchdown point (± 90 deg).
X(161)	DLONE	Geographic longitude of the desired touchdown point (± 180 deg).
X(162)	DVAZE	Desired inertial velocity azimuth at the touchdown point, measured positive clockwise from north (±180 deg).
X(163)	PERH	Altitude of virtual perigee desired (km).
X(164)	AVETIM	Reciprocal average angular rate from virtual perigee to touchdown (sec/rad).
X(165)	RMIN	Indicator to choose between two possible range angles from entry to touchdown (+min, 0 max).

Location	Name	Description
		Constraint Satisfaction Criteria
X(170)	VOK	Injection velocity impulse being minimized is close enough if within VOK of minimum (km/sec).
X(171)	BDTOK	B•T computed from target input constraints and that of the test trajectory must be closer than BDTOK (km).
X(172)	BDRØK	B•R computed must be closer than BDROK (km) to B•R of the test trajectory for convergence.
X(173)	FLTØK	Flight time satisfaction criterion (sec).
X(174)	ENØK	Energy (vis-viva) satisfaction criterion (km^2/sec^2).
X(175)	EPSM	Factor for testing convergence in FNDMXN.
X(176)	VST	Starting improvement when minimizing velocity impulse magnitude (km/sec).
X(177)	PARKK	Indicator for inertial parking orbits: (1) minimize velocity impulse by varying park time, (0.) fix park time.

2.2 ERP

Input for an ERP case consists of:

1. EXEC card (2 in column 5).

2. C, IW data, ending in a blank. This includes 2 headers, K = 0 for the vehicle ephemeris tape and K = 1 for the case heading (and special output tape if requested). (C(699) should be set ≠ 0 if a tape is not wanted.)

   The blank following this data must appear.

3. NCH control times card.

   If NCH = 111, the next card will be a new EXEC card for the next case.

   If NCH ≤ 0, the next card will be a new control times card, unless TSTOP ≥ the flight time of the run. In the latter case, the next card will be a new EXEC card.

   If NCH > 0 (but not = 111) the next card is a change card.

4. Change card. (Station, beacon, etc).

4a. Appropriate (station, beacon, etc) data cards, ending with a blank. This blank is necessary whether or not there are any data cards.

   The combination of change and data cards may be repeated as often as necessary. The sequence must be terminated by a blank to signify the end of the change cards.

If NCH > 0 but no changes are desired, then the deck will appear:

NCH Control times card
Blank (end of change cards)
NCH Control times card or EXEC card as necessary.

If changes are included, the deck may look like:

NCH Control time card
Beacon change card
Beacon data card
Blank (end of data)
Blank (end of changes)
NCH control times card or EXEC card, as necessary.

Table 1 Error Source Treatment Code

Code	Meaning
00	Omit any treatment of the quantity
2	Treat the quantity as having only random errors (applicable only to measurements, i.e., where measurement has no bias)
01	Consider the quantity as a deterministic error source (and with random errors, if the quantity is a measurement, i.e., when a measurement has a bias)
-1	Consider the quantity as a deterministic error source in the sense that the error would be determined

Table 2 Station Changes

Columns	Quantity
1	Station change indicator (1)
2-7	Station name
8-9	Consider or omit station ($\pm$ 1 means consider, 0 means omit)
10-11	Station identification number (1 to 12)
12-13	Type of angles to measured
14-15	Range measurement
16-17	Range rate measurement
18-19	Azimuth, right ascension, or ℓ-direction cosine measurement
20-21	Elevation, declination, or m-direction cosine measurement
22-23	Azimuth, right ascension, or ℓ-direction cosine rate measurement
24-25	Elevation, declination, or m-direction cosine rate measurement
26-27	Latitude error in station location
28-29	Longitude error in station location
30-31	Altitude error in station location
32-33	Time bias error in the station clock

Table 3 Station Information

Index	Quantity	Input Units
2	Period of Observation	(Days)(Hours).(Min.) (Sec.) or (-Sec.) (e.g. -.5 = .5 sec)
3	Station Latitude(geodetic)	Degrees
4	Station Longitude	Degrees
5	Station Altitude	Meters
6	Artifical Horizon	Degrees
7	Maximum Elevation	Degrees
8	Range Error(Random)	Meters
9	Range Rate Error(Random)	Meters/Second
10	Azimuth Error(Random) or Right Ascension Error(Random) or ℓ- direction Cosine Error(Random)	Milliradians Milliradians Unitless
11	Elevation Error(Random) or Declination Error(Random) or m-direction Cosine Error(Random)	Milliradians Milliradians Unitless
12	Azimuth(etc.) Rate Error(Random)	Milliradians/Sec
13	Elevation(etc.)Rate Error(Random)	Milliradians/Sec
14	Range Error(Bias)	Meters
15	Range Rate Error(Bias)	Meters/Sec
16	Azimuth(etc)Error(Bias)	Milliradians
17	Elevation (etc.) Error(Bias)	Milliradians
18	Azimuth (etc.) Rate Error (Bias)	Milliradians/Second
19	Elevation (etc.) Rate Error (Bias)	Milliradians/Second
20	Latitude Location Error (Northing)	Meters
21	Longitude Location Error (Easting)	Meters
22	Altitude Location Error (Down)	Meters
23	Time Error (Bias)	Seconds

Table 4 Beacon Measurement Changes

Column	Quantity
1	2 means this is a beacon change card
2-7	Mnemonic message
8-9	01 means "include", 00 means "delete" all beacons
10-11	00 means this is the measurement type change card
12-13	Number of the body on which beacons are found (1 - Earth, 2 - Moon, 5 - Mars)
14-15	Range measurement
16-17	Range rate measurement
18-19	Angle 1 measurement
20-21	Angle 2 measurement
22-23	Time bias error of the onboard clock

Table 5 Beacon Measurement Change Data

Index	Quantity	Input Units
1	Period of observations	(Days)(Hours).(Min.) (Sec.) or (-Sec.)
2	(Spare)	
3	Range error(random)	meters
4	Range rate error(random)	meters/second
5	Angle 1 error (random)	milliradians
6	Angle 2 error (random)	milliradians
7	Range error(bias)	meters
8	Range rate error (bias)	meters/second
9	Angle 1 errors (bias)	milliradians
10	Angle 2 error (bias)	milliradians
11	Time or clock error (bias)	seconds

Table 6 Individual Beacon Change Card

Column	Quantity
1	2 means this is a beacon change card
2-7	Mnemonic for beacon identification
8-9	01 means "include", 00 means "delete"
10-11	Beacon number (must be 1-10)
12-13	Number of the body on which beacon is located (same for all beacons)
14-15	Latitude error in beacon location (Northing)
16-17	Longitude error in beacon location (Easting)
18-19	Altitude error in beacon location (Down)

Table 7 Beacon Specification Data

Index	Quantity	Input
2	Latitude	Degrees
3	Longitude	Degrees
4	Altitude	Meters
5	Artificial Horizon	Degrees
6	Latitude Location Uncertainty	Meters Northing
7	Longitude Location Uncertainty	Meters Easting
8	Altitude Uncertainty	Meters

Table 8 Onboard Change Card

Column	Quantity
1	3 means this is an onboard change card
2-7	Mnemonic for onboard identification
8-9	01 means "include", 00 means "delete" onboard consideration
10-11	Number of the body for onboard radar-type measurements (1-Earth, 2-Moon, 4-Venus, 5-Mars, 6-Saturn, 7-Jupiter)
12-13	Height measurement
14-15	Height rate measurement
16-17	Time bias error of the onboard clock
18-19	Number of angular measurements of the first type in the cycle
20-21	First type of angular measurement
	00 means no measurements in the stated interval
	01 means subtended angle
	02 means right ascension or longitude-type angle and
	declination or latitude-type angle
	03 means maximum line-of-sight change star-planet angle
	04 means minimum line-of-sight change star-planet angle
22-23	Number of the body on which the first type measurement is made (1-Earth, 2-Moon, 3-Sun, 4-Venus, 5-Mars, 6-Saturn, 7-Jupiter)
24-25	Number of angular measurements of the second type in the cycle.
26-27	Second type of angular measurement
28-29	Number of the body for second type angular measurement
30-55	More specification of angular measurements

Table 9 Onboard Measurement Data

Index		
1	Height error (random)	Meters
2	Height rate error (random)	Meters/second
3	Height error (bias)	Meters
4	Height rate error (bias)	Meters/second
5	Time bias uncertainty	Seconds
6	Altitude to cease radar observations	Kilometers
7	Period of radar type measurements	(Days)(Hours) · (Minutes)(Seconds) or (-Seconds)
8	Angle 1 error k_1 (random) (of error model $\sigma_e^2 = k_1^2 + k_2^2(2\sin^{-1}\frac{r_p}{r})^2$)	Arc seconds
9	Angle 1 error k_2 (random)	Unitless
10	Angle 2 error k_1 (random)	Arc seconds
11	Angle 2 error k_2 (random)	Unitless
12	Subtended angle error k_1 (random)	Arc Seconds
13	Subtended angle error k_2 (random)	Unitless
14	Period of angular measurements	(Days)(Hours) · (Minutes)(Seconds) or (-Seconds)
15	Right ascension of reference star	Degrees
16	Declination of reference star	Degrees

Table 10 Equation of Motion Parameters

Index	Column	Quantity	Units
	1	4 means "equation of motion parameter"	
	2-7	Mnemonic message	
1	8-9	Astronomical Unit	
2	10-11	Earth's Mass	Fraction of Sun's Mass
3	12-13	Moon's Mass	Fraction of Sun's Mass
4	14-15	Venus' Mass	Fraction of Sun's Mass
5	16-17	Mars' Mass	Fraction of Sun's Mass
6	18-19	Jupiter's Mass	Fraction of Sun's Mass
7	20-21	Saturn's Mass	Fraction of Sun's Mass
8	22-23	Mercury's Mass (not included)	Fraction of Sun's Mass
9	24-25	Second Zonal Harmonic	Dimensionless
10	26-27	Third Zonal Harmonic	Dimensionless
11	28-29	Fourth Zonal Harmonic	Dimensionless
12	30-31	Fifth Zonal Harmonic	Dimensionless
13	32-33	First Longitudinal Harmonic	Radians
14	34-35	Second Longitudinal Harmonic	Radians
15	36-37	Third Longitudinal Harmonic	Radians
16	38-39	Fourth Longitudinal Harmonic	Radians
17	40-41	First Lunar Gravity Parameter	$(\text{Kilometers})^2$
18	42-43	Second Lunar Gravity Parameter	$(\text{Kilometers})^2$
19	44-45	Third Lunar Gravity Parameter	$(\text{Kilometers})^2$
20	46-47	First Drag Parameter	$(\text{Kilometers})^{-1}$
21	48-49	Second Drag Parameter	$(\text{Kilometers})^{-1}$
22	50-51	Solar Radiation Pressure Factor	$(\text{Kilometers})^3/(\text{seconds})^2$
23	52-53	Venting Thrust Magnitude	$\text{Kilometers}/(\text{seconds})^2$
24	54-55	Speed of Light Error Factor	Dimensionless

/INPCOM/C-ARRAY VALUES

LOCATION	NAME	COMPILED VALUE	DEFINITION
Constants			
1	HPI	1.57079632679497	$\pi/2$
2	PI	3.14159265358979	π
3	TPI	6.28318530717959	2π
4	RTD	57.2957795130823	Conversion factor, radians to degrees.
5	DTR	.017453292519943	Conversion factor, degrees to radians.
6	SPMSD	86400.	Seconds per mean solar day.
7	RSPMSD	1.1574074074074E-5	Reciprocal seconds per mean solar day = 1/86400.
8	ASTU	.149599E9	Kilometers per astronomical unit.
9		299774.	Speed of light, km/sec.
10		.10	SBEV1 constant for step-size formula.
Body Constants			
11-90	BODC(I,J)		J^{th} body constant for body # I, where J = 1 to 8 and I = 1 to 7 is loaded, per Table 1: BØDC(I,J). I = 8 to 10 is available for three extra bodies with constants J ordered as per Table 1 (page 10.)
91-98*			
Trajectory Initial Conditions			
99	SECO	0.0	Starting time, seconds from epoch.
100	TARG	5.0	Target body number, (Mars).

* Unused cells.

LOCATION	NAME	COMPILED VALUE	DEFINITION
101	FLTIM	30000.	Maximum flight time in days x 10^2 + hours + minutes x 10^{-2} + seconds x 10^{-4}. (300 days to radius of closest approach to Mars).
102	DATE	7502.10	Starting date in (Year-1900) x 10^2 + month + whole day x 10^{-2} (February 10, 1975).
103	FDATE	201.25146	Starting time of day, Greenwich Mean Time, hour x 10^2 + minute + second x 10^{-2} (1 minute and 25.146 seconds past 2AM).
104	BCEN	1.	Initial central body number (earth).
105	TYPEX	0.0	Type of coordinates in X (0.0 for cartesian position and velocity, mean equator and equinox of 1950.0).
106-111	X		Vehicle initial conditions in TYPEX coordinates relative to body # BCEN.
106	X(1)	-5194.0522	x position, km; Earth-Mars Trajectory 2/10/75
107	X(2)	-3371.4096	y position, km; Earth-Mars Trajectory 2/10/75
108	X(3)	-2175.8862	z position, km; Earth-Mars Trajectory 2/10/75
109	X(4)	9.7623319	$\dot{x}$ velocity, km/sec; Earth-Mars Trajectory 2/10/75
110	X(5)	-11.540528	$\dot{y}$ velocity, km/sec; Earth-Mars Trajectory 2/10/75
111	X(6)	-5.4222573	$\dot{z}$ velocity, km/sec; Earth-Mars Trajectory 2/10/75
Initial 6x6	Covariance Matrix		
112-133	PI		Specification for the symmetric upper left 6x6 portion of the initial P covariance matrix.
112	PI(1)	0.0	Type of coordinates to be described in PI(2 thru 22), 0.0 to signify mean equator and equinox of 1950.0.

PHILCO Ford
SPACE & RE-ENTRY SYSTEMS DIVISION
Philco-Ford Corporation

LOCATION	NAME	COMPILED VALUE	DEFINITION
113	PI(2)	100.	Diagonal elements P(1,1)
114	PI(3)	100.	P(2,2)
115	PI(4)	100.	P(3,3)
116	PI(5)	.0001	P(4,4)
117	PI(6)	.0001	P(5,5)
118	PI(7)	.0001	P(6,6)
119-123	PI(8-12)	0.0	From left to right starting 1 beyond the diagonal: the remaining 5 elements of row 1.
124-127	PI(13-16)	0.0	The remaining 4 elements of row 2.
128-130	PI(17-19)	0.0	Remaining 3 elements of row 3.
131-132	PI(20-21)	0.0	Remaining 2 elements of row 4.
133	PI(22)	0.0	Remaining element of row 5.
Guidance Specification			
134-155	PARI		Specifications for the symmetric initial PAR covariance matrix.
134	PARI(1)	-1.	Type of coordinates to be described in PARI (2 thru 22), -1. to signify the initial PAR is identical to the initial upper left 6x6 of P. (The using program would thus ignore PARI (2 thru 22)).
156	PRED	0.0	Prediction key (no prediction for zero or negative value unless guidance is included).
157	GUID	0.0	Guidance law; set $\leq$ 0. for no guidance 1. fixed time of arrival 2. constant target energy $\geq$ 3. minimum energy.

LOCATION	NAME	COMPILED VALUE	DEFINITION
158-162	GUIDT		Five chronologically ordered guidance times referenced to starting epoch, in format: days x 10^2 + hours + minutes x 10^{-2} + seconds x 10^{-4}.
158	GUIDT(1)	1000.	10 days from epoch or Feb 20, 1975 at $2^h\ 1^m\ 25.146^s$.
159	GUIDT(2)	10000.	100 days from epoch.
160	GUIDT(3)	15000.	150 days.
161	GUIDT(4)	20000.	200 days.
162	GUIDT(5)	25000.	250 days, (never to be executed for the above Earth → Mars sample because actual flight time is around 235 days).
163	(EXER)	30000.	Percentage error for monitoring guidance correction. The numerical value loaded by this block data is erroneous. The user should overlay an appropriate value when using the guidance option.
164	GUIDI(1)	10.	Resolution error standard deviation, (10 meters/second).
165	GUIDI(2)	1.	Proportional error standard deviation, (1 %).
166	GUIDI(3)	1.	Pointing error standard deviation, (1 degree).
167-584		0.0	Certain of the following arrays are herein set to zero for initialization.
167-196	XTRB	0.0	XTRB(I,J) to contain 10 data values, I = 1 to 10, for up to three extra bodies J = 1 to 3. I is date, fdate, body center, coordinate type, and 6 coordinates, same in order and units as C(102 thru 111). (XTRB(3,J)=0. signals no extra bodies included.)

LOCATION	NAME	COMPILED VALUE	DEFINITION
197-199*			
200-475	S		S(I,J) to contain 23 data values, I = 1 to 23 for each station # J, J = 1 to 12.
476-566	B		B(I) to contain up to 91 beacon data values.
567-584	OB		OB(I) to contain up to 18 onboard data values
585-608	EMP	1.	The EMP array serves 3 purposes in the Mark II Error Propagation Program, whereas in the Patched Conic Program only the first purpose is nominally served. 1. An ordered array of variances on equation of motion error sources for use in error propagation. (The Patched Conic Program is unable to execute equation of motion error propagation.) 2. An ordered array of keys for subroutine VAREQ, where zero values set logic for Subroutine FBUS to make certain omissions; e.g., omissions of related perturbations in the gradient calculation. (ERP portion of Mark II). 3. An ordered array of keys for subroutine MPSENS, where for each EMP(I) = 0.0 the sensitivities of the state with respect to equation of motion error source I are computed. (PINT portion of Mark II).
609	ZH(1)	-.1082E-2	ZH_i are zonal harmonic coefficients, $J_{i+1,o}$ of the Earth's gravitational field (dimensionless).
610	ZH(2)	-.23E-5	
611	ZH(3)	-.18E-5	
612	ZH(4)	0.	

LOCATION	NAME	COMPILED VALUE	DEFINITION
613-622	TH		Tesseral harmonics where: $J_{n,m}$ is the m^{th} longitudinal harmonic coefficient of the n^{th} order, (no units); and $\lambda_{n,m}$ is the geographic longitude reference of $J_{n,m}$, where $-\pi < \lambda_{nm} \leq \pi$, radians.
613	TH(1)	0.	J_{21}
614	TH(2)	0.	λ_{21}
615	TH(3)	-.120E-5	J_{22}
616	TH(4)	-.460	λ_{22}
617	TH(5)	-1.9E-6	J_{31}
618	TH(6)	.080	λ_{31}
619	TH(7)	-.14E-6	J_{32}
620	TH(8)	-.293	λ_{32}
621	TH(9)	-.10E-6	J_{33}
622	TH(10)	.743	λ_{33}
623	ABC		Mass-normalized moments of inertia of the moon used for computing the triaxial gravity perturbation, (km^2).
623	ABC(1)	1.20926E6	a
624	ABC(2)	1.20952E6	b
625	ABC(3)	1.21003E6	c
626	DRAGC(1)	.109	Atmospheric drag coefficients of the model:
627	DRAGC(2)	.14294	$-C_1 e^{-C_2 h} \mid V \mid \vec{V}$, (both have units of Km^{-1}.)
628	TMAG	0.	Magnitude of thrusting acceleration (km/sec^2).
629	SPK	.94E7	Solar radiation pressure coefficient (km^3/sec^2).

LOCATION	NAME	COMPILED VALUE	DEFINITION
630	DUTD	.4050926E-3	Discrepancy between Ephemeris Time and Universal Time, (days).
631-700			The remainder of INPCOM are herein set to zero for initialization of the following data.
631-671*			
672	ICASE	0	Case counter for trajectories generated and written on binary tape by the PINT portion of the Mark II Program. (ICASE is logically identical to KASE, see C(674).**
673	IKAS	0	Case counter for ERP cases.**
674	KASE	0	Case counter for trajectories generated and written on binary tape by Subroutine CONW. Each trajectory generated is numbered consecutively and written on tape, so that subsequent tape-using routines may select the trajectory desired by examining the tape for this identification number. Initially , the case counter must be zero to properly position the binary tape.**
675-686	HEAD(1-12)	0.0	12 word alphanumeric header on trajectory tape, # 10. This header is written by CONW or PINT, and used by ERP to identify the trajectory tape required. HEAD is required input.
687-698	HEAD(13-24)	0.0	12 word alphanumeric header 1. Describing the ERP case being computed. 2. Written on the special output tape if ITAPE = 0.0. 3. Used by SPOUT to identify the special output tape required.

** Refer to footnote on following page.

LOCATION	NAME	COMPILED VALUE	DEFINITION
699	ITAPE	0.0	ITAPE = 0, causes special output tape # 12 to be written during execution of ERP.
700	IOCAS	0	Case counter for cases written on special output tape. Subsequently read by SPOUT to identify the desired case to process.**

** Program logic requires that ICASE, IKAS, KASE, and IOCAS be initially zero, as provided herein. These case counters are automatically incremented by the Mark II Program and should never be altered by an alternate BLOCK DATA or by ROVLEY input.

TABLE 1: BODC(I,J)

J	I:	1	2	3	4	5	6	7
1	Name	EARTH	MOON	SUN	VENUS	MARS	SATURN	JUPITR
2	UM(I), gravity constants, km^3/sec^2	398603.2	4900.7588	.13271545E12	324769.5	42977.8	3.791870E7	1.267106E8
3	RPL(I), semi-major axis, km	6378.165	1738.	695500.	6200.	3400.	60400.	71350.
4	Semi-minor axis,km	6356.5838	1738.	695500.	6200.	3400.	54050.	66600.
5	RPAT(I) sphere of influence, km	925000.	66000.	1.E10	616000.	565000.	.546E8	.48E8
6	Rotation rate,rad/sec	.72921152E-4	.266169952E-5	0.0	0.0	.70882177E-4	.170553347E-3	.177491110E-3
7	Max step size, seconds	43200.	21600.	864000.	43200.	43200.	86400.	172800.
8	Interpolation interval, days	40.	40.	40.	40.	40.	40.	40.

D.2.3 SPOUT

Input for a SPOUT case consists of:

1. EXEC card (3 in column 5).

2. Header for special output tape and IC data, ending with the blank. (Header is type K = 0).

   Blank card must appear.

Input Data For the SPOUT Option

Nominal Value	Program Name	Location	Description
1		IC(1)	Case (from tape) to use for output.
1		IC(2)	Regular output point number (counter IKY) at which to start outputting.
241	KEND	IC(3)	Regular output point number at which to end output.
10	KDEL	IC(4)	Output interval (SPOUT will output every KDEL points of type KTYP from IC(2) to KEND).
2	IPLT	IC(5)	Plotting interval (relative to IC(4)).
5	NTYPE	IC(6)	No. of output types (This should not be changed unless more coding is added).
1	KTYP	IC(7)	Critical event type which will control output. IC(7) = 0, time control 1, regular output point control K, type K event control (See KEY in description of tape)
	ITBLE		This is the output choice table. To choose a type of output, set:
1		IC(10)	= 1 to get time and state. Details of output are set in IC(20).
1		IC(11)	= 1 to get matrix output. Details of output are set in IC(40).
0		IC(12)	(Space reserved for measurement outputs.)
1		IC(13)	= 1 to plot. Selection of variables to be plotted is in IC(60).
0		IC(14)	(Space reserved for guidance outputs.)
1, 2, 3, 0		IC(20)	Details of state output to be printed.
0, 1, 111, 0		IC(30)	Up to ten different combinations of body center and coordinate system may

Nominal Value	Program Name	Location	Description
			be chosen. The vehicle position and velocity are output in the coordinate system specified in the IC(30) list centered at the body set in the corresponding slot of the IC(20) list. A zero in the IC(20) list terminates the state outputs. The body center numbers are: 1 Earth 2 Moon 3 Sun 4 Venus 5 Mars 6 Saturn 7 Jupiter The coordinate system code is made up by summing: 0 (1950) or 1 (date) + 0 (equator) 10 (ecliptic) or 20 (selenographic) + 0 (Cartesian) 100 (spherical) or 200 (elements)
		IC(40)	Matrix output options. Zero terminates the list. Each entry is made up according to the code: 0 (inertial) 10 (N.V.W.) or 20 (elements) + 1 (matrix + STD) or 2 (Eigen values + Euler)

Nominal Value	Program Name	Location	Description
0 0 0 0 0 0 1 0	ITEMP	IC(50)	This is an array which allows output at any critical event of a given type, even though the run is regularly searching for some other type, or time. Set ITEMP (K) $\neq$ 0 to get output at every critical event type K, K $\neq$ IC(7). K is as described under record 1, KEY. (This is independent of IC(7) except that if IC(7) > 1, search for events of type K will not start until after a record with sequence number IC(2). If IC(7) $\leq$ 1, output at type K events will start at the beginning of the tape.)
25 -26 -25 -26 107 0		IC(60)	Plot table. The numbers (indices) of variables to be plotted. A zero terminates the list. Entries = 25 (RMSP), 26 (RMSV), or the MPO number of any RMS on the tape (See Section 5.1 for description of MPO's). Each entry is + for normal variable, - for last variable of a plot. (For instance, 25, -26, 25, 0 is the same as 25, -26, -25, 0 and will give two plots - the first of RMSP + RMSV, the second of RMSP alone.
0.0	TSTART	WC(1)	only applicable if IC(7) = 0: T start
0.0	TEND	WC(2)	only applicable if IC(7) = 0: T end
0.0	DELT	WC(3)	only applicable if IC(7) = 0: Δ t
0.0	TTOL	WC(4)	only applicable if IC(7) = 0: T tolerance (program will accept any value within Δt of T_1 as the value being searched for).

D.2.4 CONW

Input for a CONW case consists of:

1) EXEC card (4 in column 5).

2) C-array data and headers of both types, ending with a blank.

The blank card must appear.

CONW Input Data

Location	Definition
C(99)	Zero reference time, from epoch, in the form: Days $\cdot 10^2$ + hours + minutes $\cdot 10^{-2}$ + seconds $\cdot 10^{-4}$.
C(100)	Target body number.
C(101)	Stop time, from epoch, in the same form as zero time.

Vehicle Initial Conditions

C(102)	Year (of 20th century) $\cdot 10^2$ + month + day $\cdot 10^{-2}$.
C(103)	Hours $\cdot 10^2$ + minutes + seconds $\cdot 10^{-2}$.
C(104)	Initial body center number.
C(105)	Input type in the form $A \cdot 10^2 + B \cdot 10 + C$, where A, B and C are interpreted as follows: A = 0, Cartesian; B = 0, Equator; C = 0, 1950 Epoch; A = 1, Spherical; B = 1, Ecliptic; C = 1, Epoch of date; A = 3, Orbital elements; B = 2, Body fixed.

C(106-111) Specify initial state consistent with input type. All lengths are in km, all times are in seconds, all angles are in degrees.

	Cartesian	Spherical	Orbital Elements
C(106)	X	R	Semi-major axis
C(107)	Y	Lat	Eccentricity
C(108)	Z	Long	True Anomaly
C(109)	$\dot{X}$	V	Long. of ascending node
C(110)	$\dot{Y}$	Path Angle	Inclination to X-Y plane
C(111)	$\dot{Z}$	Azimuth	Arg of periapsis from ascending node

Extra Body Initial Conditions are specified in exactly the same form as the vehicle initial conditions. The input locations are indicated below.

Location	Definition
C(167-176)	Extra body one.
C(177-186)	Extra body two.
C(187-196)	Extra body three.

APPENDIX E

MARK IV: PINT

Precision Integration

Subroutine Name	Description
ADOT	Computes the angle between two given vectors.
AFTER	Performs the case calculations which follow trajectory integration.
ANTR1	Provides cartesian components of position and velocity of sun, moon, and planets on given date and time. Two versions of this subroutine are available: 1. PANTRY*** computes from mean orbital elements, and 2. DEPHEM** reads a JPL Ephemeris Tape and interpolates.
BACK	Backspaces a binary tape N logical records or a BCD tape N physical records.
BEGIN	Computes cartesian injection state as a function of the controls in subroutine REFINE.
BLOCK DATA	/DQSCON/ Loads data into DQSCON common block for use by subroutine DEQS.
BLOCK DATA	/INPCOM/ Loads data into INPCOM common block for use in the Mark IV Program.
BLOCK DATA	/WCOM/ Loads data into WCOM common block.
BODCON	Supplies values of target constraints to subroutine REFINE.
BUFFIL**	Locates the required epoch on the JPL Ephemeris Tape and reads the appropriate data for use in subroutine DEPHEM.
BVEC	Computes the miss-vector components of the orbit, relative to the target body, from the cartesian state.
CONVX	Converts input state to cartesian, equatorial, 1950 system and outputs results.
CROSS	Computes the vector cross product.
DATOUT	Computes and outputs calendar and Julian dates.

---

* Refer to the bracketed subroutine for further description; e.g., DOT is described in the ADOT subroutine writeup.

** Not required by approximate ephemeris package, ANTR1-PANTRY.

*** Not required by JPL Tape Ephemeris package, ANTR1-DEPHEM.

MARK IV: PINT

Subroutine Name	Description
DEPHEM**	Computes position and velocity of planets and moon at any Julian date, using Everett's formula to interpolate from a JPL Ephemeris Tape.
DEQS	Integrates n first or second order differential equations.
DERIV	Computes the second derivatives of the sensitivities of the state to motion parameters.
DOT	Computes the vector dot product, (ADOT*).
DRAG	Computes acceleration due to atmospheric drag, the gradient thereof, and partial derivatives of same wrt the drag coefficients.
EHA	Computes the Greenwich hour angle and the earth's angular velocity.
EL2EX***	Converts mean orbital elements to cartesian position and velocity.
ENCKE	Calculates the inverse-square perturbing acceleration due to a central body.
EQTOR	Computes the transformation from mean equator and equinox of 1950.0 to mean equator and equinox of date; computes mean obliquity of date.
ERROUT	Provides programmed response to anticipated errors.
EXCOV	Directs the flow of the entire Mark IV Error Propagation Program.
EXIN	Reads interpolation coefficients from binary tape and reconstructs an N dimensional vector X at time T.
EXPAND***	Solves Kepler's equation by series in terms of eccentricity and mean anomaly.
FIEF	Spaces a binary tape forward over N files, or backward over N + 1 files.
FIFL	Locates a specified case on binary tape, and reads Record 1 of this case.
FIST	Rewinds the binary input tape, and reads and checks the header.
FNDMXN	Finds the maximum or minimum of a function.
FNORM	Computes the magnitude of a vector, (ADOT*).

MARK IV: PINT

Subroutine Name	Description
FSUB	Computes the second derivatives of the perturbing accelerations, the integrated transition matrix, and stopping functions for trajectory integration.
GOTOR	Solves Kepler's equation.
GRAVTY	Computes the acceleration due to the central body's gravitational field.
GTRAN	Computes the coordinate transformation from equator and equinox of 1950 to selenographic or true equator and Greenwich of date.
GTRN	Generates one or a sequence of rotational transformations from input angles.
GTSN	Generates one or a sequence of rotational transformations from input sines and cosines.
INDEX	Sets initial controls, constraint tolerances, and indicators for refining the trajectory.
INTCOF**	Computes interpolation coefficients for subroutine DEPHEM.
INVERT	Inverts a matrix.
MISS1	Computes and outputs the matrix of partials at the target vector wrt the state at time t, (normally the end point).
MOON	Computes the perturbing accelerations (and gradient thereof) caused by the triaxiality of the lunar gravity.
MPSENS	Initializes integration of Motion Parameter Sensitivities from a vehicle ephemeris tape.
MTMPLY	Forms the product of any two matrices.
MVTRN	Computes the product of a 3 x 3 matrix and a 3 x N matrix.
NUTATE	Computes the transformations from earth's mean equator, equinox to earth's true equator, equinox and/or moon's true equator, node.
ORB	Computes and outputs orbital elements.
ORB2X	Converts orbital elements to cartesian position and velocity.
ORIENT	Finds the angle through which one vector must be rotated about a second vector in order that the first vector should form a given angle with a third vector.
OSUB	Prints trajectory information, writes a binary tape of the trajectory, and calls for a center shift at patch points.

MARK IV: PINT

Subroutine Name	Description
OUTX	Writes out cartesian and spherical position and velocity components.
OWRITE	Writes output and spaces tape during integration for motion parameter sensitivities.
PATCH	Calculates a constraint function for lunar and planetary approach.
PERT	Computes the n-body perturbing acceleration and gradient thereof.
PINT	Controls the generation of precision integration trajectories.
PSTART	Initializes the Mark IV Error Propagation Program for the precision integration option.
REFINE	Iterates on a set of launch controls until the resulting trajectory satisfies a prescribed set of target arrival conditions.
REST	Performs the trajectory shifting required at rectification.
ROTAIT	Rotates two vectors in a plane.
ROVLEY	Reads fixed, floating, and alphanumeric input data into core.
RVAN	Converts spherical coordinates to cartesian.
RVOUT	Converts cartesian position and velocity to spherical coordinates.
SEVENL	Calls subroutine PSTART or AFTER. (SEVENL is the PROGRAM for the (7,0) primary overlay.)
SHIFT1	Calculates position and velocity relative to ephemeris bodies and extra bodies.
SIXA	Calls subroutine TRAJ. (SIXA is the PROGRAM for the (6,1) secondary overlay.)
SIXB	Calls subroutine REFINE. (SIXB is the PROGRAM for the (6,2) secondary overlay.)
SIXL	Calls one of 3 secondary overlays. The (6,0) primary overlay makes the DEQS precision integration package available to TRAJ or REFINE, (or to PLNTDQ for SEARCH). (SIXL is the PROGRAM for the (6,0) primary overlay.)
SOLARP	Computes the acceleration due to solar radiation pressure and the gradient thereof, or the partial derivative of same.
SPER	Computes spherical coordinates of a cartesian 3-vector.

MARK IV: PINT

Subroutine Name	Description
SSIZE	Computes a starting integration step size.
STEPD	Computes new cartesian state on a conic, given the old state plus incremental time or true anomaly.
STEPI	Computes the array of orbital elements to define a conic for subroutine STEPT.
STEPT	Computes cartesian state at given time on given conic.
TARGT1	Computes the transformation matrix to target coordinates.
TCONIC	Calculates conic time as a function of true anomaly.
TFRAC	Updates time in whole and fractional days from epoch.
THRUST	Calculates the acceleration due to thrust, the gradient thereof, or the partial derivative of same.
TIMEC	Converts calendar date and time to whole and fractional days from January 1, 1950.
TIMED	Converts time from (days, hours, minutes, seconds) to seconds.
TIMES	Converts time from seconds to alphanumeric days, hours, minutes, and seconds.
TPFL	Writes Record 1 of each case on binary tape.
TPST	Rewinds a binary tape, writes a header, then an End of File on this tape.
TRAJ	Drives the integration for 1) regular trajectories or tape-write, 2) computing equation of motion sensitivities.
UPDATE***	Updates mean orbital elements in time from one epoch to another.
VNORM	Normalizes a vector and also computes the magnitude, (ADOT*).
XOUT	Computes interpolation coefficients and writes them on a binary tape.

MARK IV: ERP

Error Propagation

Subroutine Name	Description
ANTR1	Provides cartesian components of position and velocity of sun, moon, and planets on given date and time. Two versions of this subroutine are available: 1. PANTRY*** computes from mean orbital elements, and 2. DEPHEM** reades a JPL Ephemeris Tape and interpolates.
BACK	Back spaces a binary tape N logical records or a BCD tape N physical records.
BCHNG	Makes beacon measurements and appropriately updates the expanded covariance matrix.
BEACH	Supplies information pertaining to the next beacon critical event.
BLDP1	Builds the expanded P covariance matrix and outputs the error sources included.
BLOCK DATA	/INPCOM/ Loads data into INPCOM common block for use in the Mark IV Program.
BLOCK DATA	/DQSCON/ Loads data into DQSCON common block for use by subroutine DEQS.
BUFFIL**	Locates the required epoch on the JPL Ephemeris tape and reads the appropriate data for use in Subroutine DEPHEM.
CHNG	Makes earth-based tracking station measurements and appropriately updates the expanded covariance matrix.
CONVP	Outputs the input covariance matrix and converts it to a 6x6 in equator of 1950.0.
CRITA	Outputs when a body starts and stops occulting the vehicle.
CRITO	Outputs station and beacon in-view, out-of-view critical events.
CROSS	Computes the vector cross product.

* Refer to the bracketed subroutine for further description; e.g., DOT is described in the ADOT subroutine writeup.

** Not required by approximate ephemeris package, ANTR1-PANTRY.

*** Not required by JPL Tape Ephemeris package, ANTR1-DEPHEM.

MARK IV: ERP

Error Propagation

Subroutine Name	Description
DATOUT	Computes and outputs calendar and Julian dates.
DEPHEM**	Computes position and velocity of planets and moon at any Julian date, using Everett's formula to interpolate from a JPL Ephemeris Tape.
DEQS	Integrates n first or second order differential equations.
DOT	Computes the vector dot product, (ADOT*).
DRAG	Computes acceleration due to atmospheric drag, the gradient thereof, and partial derivatives of same wrt the drag coefficients.
EHA	Computes the Greenwich hour angle and earth's rate.
EIGEN	Computes eigenvalues and eigenvectors of a real symmetric matrix.
ELLINT	Evaluates the incomplete elliptic integrals of the first and second kinds.
EL2EX***	Converts mean orbital elements to cartesian position and velocity.
EQTOR	Computes the transformation from mean equator and equinox of 1950.0 to mean equator and equinox of date; computes mean obliquity of date.
ERP	Controls the logical flow of error propagation computations.
ERROUT	Provides programmed response to anticipated errors.
EVDEL	Orders the critical events chronologically and determines the advancing step size.
EXCOV	Directs the flow of the entire Mark IV Error Propagation Program.
EXIN	Reads interpolation coefficients from binary tape and reconstructs an N dimensional vector X at time t.
EXPAND***	Solves Kepler's equation by series in terms of eccentricity and mean anomaly.
FBUS	Supplies the DEQS integration routine with the derivatives to integrate the variational equations.
FIEF	Spaces a binary tape forward over N files, or backward over N + 1 files.
FIFL	Locates a specified case on a binary tape, and reads record 1 of this case.

MARK IV: ERP

Error Propagation

Subroutine Name	Description
FIST	Rewinds the binary input tape, and reads and checks the header.
FNORM	Computes the magnitude of a vector, (ADOT*).
FOURA	Calls subroutine ONBRD at onboard measurement critical event, or calls GMISS at body center change. (FOURA is the PROGRAM for the (4,1) secondary overlay.)
FOURB	Calls subroutine GORE at a guidance correction critical event. (FOURB is the PROGRAM for the (4,2) secondary overlay.)
FOURC	Calls subroutine CHNG as required at a station measurement critical event. (FOURC is the PROGRAM for the (4,3) secondary overlay.)
FOURD	Calls subroutine ONBRAD or BCHNG at a critical event requiring onboard radar or beacon measurements, respectively. (FOURD is the PROGRAM for the (4,4) secondary overlay.)
FOURE	Calls subroutine DEQS when required by subroutine PHIX. (FOURE is the PROGRAM for the (4,5) secondary overlay.)
FOURL	Performs error propagation and calls subroutine OUT1 to output results at critical events. (FOURL is the PROGRAM for the (4,0) primary overlay).
GETRAN	Computes the transformation from EE50 coordinates to body-fixed coordinates of instant.
GMISS	Computes the guidance sensitivity matrix from a specified patch point to the end point.
GORE	Makes a guidance correction.
GOTOR	Solves Kepler's equation.
GRAVTY	Computes the acceleration due to the central body's gravitational field.
GTRN	Generates one or a sequence of rotational transformations from input angles.
GTSN	Generates one or a sequence of rotational transformations from input sines and cosines.
INTCOF**	Computes interpolation coefficients for subroutine DEPHEM.
INVERT	Inverts a matrix.
LAYOVR	Reads station, beacon, onboard, and equation of motion data cards and converts to proper units.

MARK IV: ERP

Error Propagation

Subroutine Name	Description
MOON	Computes the perturbing accelerations (and gradient thereof) caused by the triaxiality of the lunar gravity.
MTMPLY	Forms the product of any two matrices.
MTRX	Multiplies a 6 x 6 matrix times the upper left 6 x 6 of a given matrix of equal or larger dimension.
MVTRN	Computes the product of a 3 x 3 matrix and a 3 x N matrix.
NUTATE	Computes the transformation(s) from earth's mean equator and equinox to earth's true equator and equinox, and/or to moon's true equator and node.
OBUS	Prints a message when called by DEQS.
ONBRAD	Makes onboard radar measurements and appropriately updates the expanded covariance matrix.
ONBRD	Makes onboard optical measurements and appropriately updates the expanded covariance matrix.
ONEL	Calls subroutine PRESTO and, if necessary, GMISS to initialize the trajectory and other parameters for error propagation. (ONEL is the PROGRAM for the (1,0) primary overlay.)
ORB	Computes and outputs orbital elements.
OUM	Outputs the RMS uncertainty in the miss vector.
OUTP	Outputs the P and Par covariance matrices in Darboux coordinates.
OUT1	Outputs the state, calculates and outputs RMS values and target miss vector on output tape and binary tape.
PARAB	Fits a parabola through three points.
PCHNG	Updates the expanded covariance matrix at an observation.
PERT	Computes the n-body perturbing acceleration and gradient thereof.
PHIX	Computes the state transition matrix from state at current time to state at specified time, and calls FBUS if equation of motion error sources are included.
PHIZ	Computes the state transition matrix (on one conic).
PRESTO	Reads overlay input for common and initializes the tape read and various parameters for the Error Propagation.

MARK IV: ERP

Error Propagation

Subroutine Name	Description
PROP	Controls the updating of the state vector and the covariance matrix.
PUTIN	Reads in measurement error source keys.
QUARTC	Finds the solutions to the quadratic equation.
ROVLEY	Reads fixed, floating, and alphanumeric input data into core.
ROYAL	Outputs input random, bias, and time errors associated with station and beacon measurement error sources.
RTIMS	Controls input of control times and changes in measurement treatment.
SBEV1	Computes vehicle in-view and out-of-view critical events for earth-based tracking stations or beacons.
SCOT	Finds the expected value of a given matrix.
SHIF2	Calculates position and velocity relative to ephemeris bodies and extra bodies.
SHUFLP	Rearranges the expanded covariance matrix when error sources are added, deleted, or considered differently.
SOLARP	Computes the acceleration due to solar radiation pressure and the gradient thereof, or the partial derivative of same.
SORDR	Sorts an array X in ascending order, while preserving the correspondence between array X and array NX.
SPER	Computes spherical coordinates of a cartesian 3-vector.
STABEC	Sets up logic and loads critical event arrays for observations made by stations, beacons, and on board optical.
STASH	Supplies information for next earth-based tracking station critical event.
STAT	Computes inertial coordinates of a tracking station and the orthogonal transformation relating inertial cartesian plane to local tangent plane North-East-Down.
STATP	Calculates the partials of earth-based tracking station measurements with respect to the extended state vector.
STEPI	Computes an array of orbital elements to define a conic for subroutine STEPT.
STEPT	Computes cartesian state at a given time on a given conic.
TCONIC	Calculates conic time as a function of true anomaly.

MARK IV: ERP

Error Propagation

Subroutine Name	Description
TFRAC	Updates time in whole and fractional days from epoch.
THREEL	Calls subroutine SBEV1 to compute on-off critical events for stations or beacons, as required. (THREEL is the PROGRAM for the (3,0) primary overlay).
THRUST	Calculates the acceleration due to thrust, the gradient thereof, or the partial derivative of same.
TIMED	Converts time from (days, hours, minutes, seconds) to seconds.
TIMES	Converts time from seconds to alphanumeric days, hours, minutes, and seconds.
TPFL	Writes Record 1 of each case on binary tape.
TPST	Rewinds a binary tape, writes a header, then an End of File on this tape.
TRANP	Transforms the covariance matrix from one inertial frame to another via a given transformation matrix.
TRDB	Computes Darboux or local tangent plane transformation matrix.
TWOL	Calls subroutine RTIMS to read in and organize measurement requirements, and/or calls OUTP to print the covariance matrix. (TWOL is the PROGRAM for the (2,0) primary overlay).
UPDATE***	Updates mean orbital elements in time from one epoch to another.
UPPT	Propagates the expanded covariance matrix in time.
VAREQ	Performs initializations for the integration of variational equations when equation of motion error sources are considered.
VENT	Initializes the KEV and EVNT critical event arrays and other parameters.
VNORM	Normalizes a vector and also computes the magnitude, (ADOT*).

MARK IV: SPOUT

Special Output

Subroutine Name	Description
ADOT	Computes the angle between two given vectors.
ANTR1	Provides cartesian components of position and velocity of sun, moon, and planets on given date and time. Two versions of this subroutine are available: 1. PANTRY*** computes from mean orbital elements, and 2. DEPHEM** reads a JPL Ephemeris Tape and interpolates.
BLOCK DATA	/INPCOM/ Loads data into INPCOM common block for use in the MARK IV Program.
BLOCK DATA	/RVR/ Loads data into RVR common block.
BUFFIL**	Locates the required epoch on the JPL Ephemeris Tape and reads the appropriate data for use in Subroutine DEPHEM.
COVOUT	Outputs the covariance matrix as requested by options in the special output program.
CROSS	Computes the vector cross product.
DATOUT	Computes and outputs calendar and Julian dates.
DEPHEM**	Computes position and velocity of planets and moon at any Julian date, using Everett's formula to interpolate from a JPL Ephemeris Tape.
DOT	Computes the vector dot product, (ADOT*).
EIGEN	Computes eigenvalues and eigenvectors of a real symmetric matrix.
EIGHTL	Calls subroutine SPOUT. (EIGHTL is the PROGRAM for the (8,0) primary overlay.)
EL2EX***	Converts mean orbital elements to cartesian position and velocity.
EQTOR	Computes the transformation from mean equator and equinox of 1950.0 to mean equator and equinox of date; computes mean obliquity of date.

* Refer to the bracketed subroutine for further description; e.g., DOT is described in the ADOT subroutine writeup.

** Not required by approximate ephemeris package, ANTR1-PANTRY.

*** Not required by JPL Tape Ephemeris package, ANTR1-DEPHEM.

MARK IV: SPOUT

Subroutine Name	Description
ERROUT	Provides programmed response to anticipated errors.
EXCOV	Directs the flow of the entire MARK IV Error Propagation Program.
EXPAND***	Solves Kepler's equation by series in terms of eccentricity and mean anomaly.
EXTRA	Reads the extended output binary tape produced by an ERP run, processes this data according to input options, and saves this data for plotting.
FIEF	Spaces a binary tape forward over N files, or backward over N + 1 files.
FIFL	Locates a specified case on the extended output binary tape, and reads Record 1 of this case.
FINFO	Locates a specified output set on the extended output binary tape.
FNORM	Computes the magnitude of a vector, (ADOT*).
GTRN	Generates one or a sequence of rotational transformations from input angles.
GTSN	Generates one or a sequence of rotational transformations from input sines and cosines.
INTCOF**	Computes interpolation coefficients for subroutine DEPHEM.
LOUT	Normalizes and outputs the P covariance matrix.
MEAS	Outputs measurements and associated RMS quantities.
MVTRN	Computes the product of a 3x3 matrix and a 3xN matrix.
NMOUT	Outputs the vehicle state in the coordinate system(s) requested by input.
NUTATE	Computes the transformation(s) from earth's mean equator and equinox to earth's true equator and equinox, and/or to moon's true equator and node.
PLOTZ	Plots the variables saved during a run of the special output program.
PLTIM	Makes rough plots of selected variables for output by the printer.

MARK IV: SPOUT

Subroutine Name	Description
PLTSAV	Accumulates an array of variables for plotting.
RNDLIM	Produces rounded limits for a plot axis.
ROVLEY	Reads fixed, floating, and alphanumeric input data into core.
SPER	Computes spherical coordinates of a cartesian 3-vector.
SPOUT	Controls the flow of the special output portion of the MARK IV Error Propagation Program.
SUBPLT	Prepares one line of output for the printer plotter subroutine PLT1M.
TFRAC	Updates time in whole and fractional days from epoch.
TIMES	Converts time from seconds to alphanumeric days, hours, minutes, and seconds.
TRANP	Transforms the covariance matrix from one inertial frame to another via a given transformation matrix.
TRDB	Computes Darboux or local tangent plane transformation matrix.
UPDATE***	Updates mean orbital elements in time from one epoch to another.
VNORM	Normalizes a vector and also computes the magnitude, (ADOT*).
X2ORB	Computes orbital elements from cartesian positions and velocity.

MARK IV: CONW

Patched Conic Tape Generation

Subroutine Name	Description
ADOT	Computes the angle between two given vectors.
ANTR1	Provides cartesian components of position and velocity of sun, moon, and planets on given date and time. Two versions of this subroutine are available: 1. PANTRY*** computes from mean orbital elements, and 2. DEPHEM** reads a JPL Ephemeris Tape and interpolates.
BLOCK DATA	/INPCOM/ Loads data into INPCOM common block for use in the Mark IV Program.
BUFFIL**	Locates the required epoch on the JPL Ephemeris Tape and reads the appropriate data for use in Subroutine DEPHEM.
BVEC	Calculates the target miss vector.
CONVX	Converts input state to cartesian, equatorial, 1950 system and outputs results.
CONW	Generates a patched conic trajectory and stores it on a binary tape for use by other programs.
CROSS	Computes the vector cross product.
DATOUT	Computes and outputs calendar and Julian dates.
DEPHEM**	Computes position and velocity of planets and moon at any Julian date, using Everett's formula to interpolate from a JPL Ephemeris Tape.
DOT	Computes the vector dot product, (ADOT*).
EHA	Computes the Greenwich hour angle and Earth's rate.
EL2EX***	Converts mean orbital elements to cartesian position and velocity.

---

* Refer to the bracketed subroutine for further description; e.g., DOT is described in the ADOT subroutine writeup.

** Not required by approximate ephemeris package, ANTR1-PANTRY.

*** Not required by JPL Tape Ephemeris package, ANTR1-DEPHEM.

MARK IV: CONW

Subroutine Name	Description
EQTOR	Computes the transformation from mean equator and equinox of 1950.0 to mean equator, equinox of date; computes mean obliquity of date.
ERROUT	Provides programmed response to anticipated errors.
EXCOV	Directs the flow of the entire Mark IV Error Propagation Program.
EXPAND***	Solves Kepler's equation by series in terms of eccentricity and mean anomaly.
FIVEL	Calls CONW to generate a patched conic trajectory. (FIVEL is the PROGRAM for the (5,0) primary overlay.)
FNORM	Computes the magnitude of a vector, (ADOT*).
GOTOR	Solves Kepler's equation.
GTRN	Generates one or a sequence of rotational transformations from input angles.
GTSN	Generates one or a sequence of rotational transformations from input sines and cosines.
INTCOF**	Computes interpolation coefficients for subroutine DEPHEM.
MISS1	Computes and outputs the matrix of partials of the target vector wrt the state at time t (normally the endpoint).
MVTRN	Computes the product of a 3x3 matrix and a 3xN matrix.
NUTATE	Computes the transformation(s) from earth's mean equator and equinox to earth's true equator and equinox, and/or to moon's true equator and node.
ORB	Computes and outputs orbital elements.
ORB2X	Converts orbital elements to cartesian position and velocity.
OUTX	Writes out cartesian and spherical position and velocity components.
PATCH	Calculates a constraint function for lunar and planetary approach.
ROTAIT	Rotates two vectors in a plane.
ROVLEY	Reads fixed, floating, and alphanumeric input data into core.
RVAN	Converts spherical coordinates to cartesian.
RVOUT	Converts cartesian position and velocity to spherical coordinates.

MARK IV: CONW

Subroutine Name	Description
SHIFT1	Calculates position and velocity relative to ephemeris bodies and extra bodies.
SKET	Determines the trajectory type and sets an index accordingly.
SPER	Computes spherical coordinates of a cartesian 3-vector.
STEPD	Calculates cartesian state on a conic by stepping time or true anomaly.
TARGT1	Computes the transformation matrix to target coordinates, (TARGT*).
TCONIC	Calculates conic time as a function of true anomaly.
TFRAC	Updates time in whole and fractional days from epoch.
TIMEC	Converts calendar date and time to whole and fractional days from January 1, 1950.
TIMED	Converts time from (days, hours, minutes, seconds) to seconds.
TIMES	Converts time from seconds to alphanumeric days, hours, minutes, and seconds.
TPST	Rewinds a binary tape, writes a header, then an End of File on this tape.
TRUEA	Computes true anomaly from semi-latus rectum, eccentricity, and radius in the orbit.
TTGO	Computes time-to-go function for patching conics.
UPDATE***	Updates mean orbital elements in time from one epoch to another.
VNORM	Normalizes a vector and also computes the magnitude, (ADOT*).
XOUT	Computes interpolation coefficients and writes them on a binary tape.

MARK IV: START-UP

Subroutine Name	Description
ADOT	Computes the angle between two given vectors.
ANTR1	Provides cartesian components of position and velocity of sun, moon, and planets on given date and time. Two versions of this subroutine are available: 1. PANTRY*** computes from mean orbital elements, and 2. DEPHEM** reads a JPL Ephemeris Tape and interpolates.
BLOCK DATA	/INPCOM/ Loads data into INPCOM common block.
BLOCK DATA	/WCOM/ Loads data into WCOM common block.
BUFFIL**	Locates the required epoch on the JPL Ephemeris Tape and reads the appropriate data for use in Subroutine DEPHEM.
CONBR	Computes the conic trajectory connecting two points in a given time.
CROSS	Computes the vector cross product.
DATOUT	Computes and outputs calendar and Julian dates.
DEPHEM**	Computes position and velocity of planets and moon at any Julian date, using Everett's formula to interpolate from a JPL Ephemeris Tape.
DOT	Computes the vector dot product, (ADOT*).
EHA	Computes the Greenwich hour angle and the earth's angular velocity.
EL2EX***	Converts mean orbital elements to cartesian position and velocity.
EQTOR	Computes the transformation from mean equator and equinox of 1950.0 to mean equator and equinox of date.

* Refer to the bracketed subroutine for further description; e.g., DOT is described in the ADOT subroutine writeup.

** Not required by approximate ephemeris package, ANTR1-PANTRY.

*** Not required by JPL Tape Ephemeris package, ANTR1-DEPHEM.

MARK IV: START-UP

Subroutine Name	Description
ERROUT	Provides programmed response to anticipated errors.
EXCOV	Directs the flow of the entire MARK IV Error Propagation Program.
EXPAND***	Solves Kepler's equation by series in terms of eccentricity and mean anomaly.
FINDV	Minimizes or finds zeroes of a function of a scalar variable.
FNORM	Computes the magnitude of a vector, (ADOT*).
GOTOR	Solves Kepler's equation.
GTRN	Generates one or a sequence of rotational transformations from input angles.
GTSN	Generates one or a sequence of rotational transformations from input sines and cosines.
INTCOF**	Computes interpolation coefficients for subroutine DEPHEM.
LAUNCH	Computes the control values which join the parking orbit to the hyperbolic excess velocity.
MVTRN	Computes the product of a 3x3 matrix and a 3xN matrix.
NINEL	Calls subroutine PLANET then stores solution in WCOM common, or calls subroutine SEARCH. (NINEL is the PROGRAM for the (9,0) primary overlay).
NUTATE	Computes the transformation(s) from earth's mean equator and equinox to earth's true equator and equinox, and/or to moon's true equator and node.
ORB	Computes and outputs orbital elements.
ORIENT	Finds the angle through which one vector must be rotated about a second vector in order that the first vector form a given angle with a third vector.
OUTX	Writes out cartesian and spherical position and velocity components.
PLANET	Start-up driver - computes approximate interplanetary trajectory solutions.
ROTAIT	Rotates two vectors in a plane.
ROVLEY	Reads fixed, floating, and alphanumeric input data into core.

MARK IV: START-UP

Subroutine Name	Description
RVOUT	Converts cartesian position and velocity to spherical coordinates.
SHIFT1	Calculates position and velocity relative to ephemeris bodies and extra bodies.
SPER	Computes spherical coordinates of a 3-vector.
START	Computes cartesian injection state as a function of launch parameter controls.
STEPD	Calculates cartesian state on a conic by stepping time or true anomaly.
TCONIC	Calculates conic time as a function of true anomaly.
TFRAC	Updates time in whole and fractional days from epoch.
TIMEC	Converts calendar date and time to whole and fractional days from January 1, 1950.
TRDB	Computes the transformation to local tangent plane or Darboux coordinates.
TRUEA	Computes true anomaly from semi-latus rectum, eccentricity, and radius in the orbit.
UPDATE***	Updates mean orbital elements in time from one epoch to another.
VELASY	Computes the velocity vectors and differences at the end - points of the conic section which connects two radii in a given time.
VNORM	Normalizes a vector and also computes the magnitude, (ADOT*).

MARK IV: SEARCH

Subroutine Name	Description
ADOT	Computes the angle between two given vectors.
ANTR1	Provides cartesian components of position and velocity of sun, moon, and planets on given date and time. Two versions of this subroutine are available: 1. PANTRY*** computes from mean orbital elements, and 2. DEPHEM** reads a JPL Ephemeris Tape and interpolates.
BLOCK DATA	/DQSCON/ Loads data into DQSCON common block for use by subroutine DEQS.
BLOCK DATA	/INPCOM/ Loads data into INPCOM common block.
BLOCK DATA	/WCOM/ Loads data into WCOM common block.
BUFFIL**	Locates the required epoch on the JPL Ephemeris Tape and reads the appropriate data for use in Subroutine DEPHEM.
BVEC	Computes the miss vector components of the orbit, relative to the target body, from the cartesian state.
CONVX	Converts input state to cartesian, equatorial, 1950 system and outputs results.
CROSS	Computes the vector cross product.
DATOUT	Computes and outputs calendar and Julian dates.
DEPHEM**	Computes position and velocity of planets and moon at any Julian date, using Everett's formula to interpolate from a JPL Ephemeris Tape.
DEQS	Integrates n first or second order differential equations.
DOT	Computes the vector dot product, (ADOT*).
DRAG	Computes acceleration due to atmospheric drag, the gradient thereof, and partial derivatives of same wrt the drag coefficients.

* Refer to the bracketed subroutine for further description; e.g., DOT is described in the ADOT subroutine writeup.

** Not required by approximate ephemeris package, ANTR1-PANTRY.

*** Not required by JPL Tape Ephemeris package, ANTR1-DEPHEM.

MARK IV: SEARCH

Subroutine Name	Description
EHA	Computes the Greenwich hour angle and the earth's rotation rate.
EL2EX***	Converts mean orbital elements to cartesian position and velocity.
ENCKE	Calculates the inverse-square perturbing acceleration due to a central body.
ENDCON	Computes the end constraints for SEARCH.
EQTOR	Computes the transformation from mean equator and equinox of 1950.0 to mean equator and equinox of date; computes mean obliquity of date.
ERROUT	Provides programmed response to anticipated errors.
EXCOV	Directs the flow of the entire Mark IV Error Propagation Program.
EXPAND***	Solves Kepler's equation by series in terms of eccentricity and mean anomaly.
FNDMXN	Finds the maximum or minimum of a function.
FNORM	Computes the magnitude of a vector, (ADOT*).
FSUB	Computes the second derivatives of the perturbing accelerations, the integrated transition matrix, and stopping functions for trajectory integration.
GOTOR	Solves Kepler's equation.
GRAVTY	Computes the acceleration due to the central body's gravitational field.
GTRAN	Computes the transformation from equator and equinox of 1950 to selenographic or true equator and Greenwich of date.
GTRN	Generates one or a sequence of rotational transformations from input angles.
GTSN	Generates one or a sequence of rotational transformations from input sines and cosines.
INTCOF**	Computes interpolation coefficients for subroutine DEPHEM.
INVERT	Inverts a matrix.
MARFIX	Computes the transformation from mean equator and equinox of 1950.0 to mars fixed coordinates.

MARK IV: SEARCH

Subroutine Name	Description
MOON	Computes the perturbing accelerations (and gradient thereof) caused by the triaxiality of the lunar gravity.
MTMPLY	Forms the product of any two matrices.
MVTRN	Computes the product of a 3 x 3 matrix and a 3 x N matrix.
NINEL	Calls subroutine PLANET then stores results, or calls subroutine SEARCH. (NINEL is the PROGRAM for the (9,0) primary overlay.)
NUTATE	Computes the transformation(s) from earth's mean equator and equinox to earth's true equator and equinox, and/or to moon's true equator and node.
ORB	Computes and outputs orbital elements.
ORB2X	Converts orbital elements to cartesian position and velocity.
OSUB	Prints integrated trajectory information, writes a binary tape of the trajectory.
OUTX	Writes out cartesian and spherical position and velocity components.
PATCH	Calculates a constraint function for lunar and planetary approach.
PCON	Computes points on a patched conic trajectory.
PERT	Computes the n-body perturbing acceleration and gradient thereof.
PHIZ	Computes the closed-form conic state transition matrix.
PLNTDQ	Drives the integrated trajectory calcuation for SEARCH.
PLNTPC	Drives the patched conic trajectory calculation for SEARCH.
REST	Performs the trajectory shifting required at rectification and patch points.
ROTAIT	Rotates two orthonormal vectors through an angle in their mutual plane.
ROVLEY	Reads fixed, floating, and alphanumeric input data into core.
RVAN	Converts spherical coordinates to cartesian.
RVOUT	Converts cartesian position and velocity to spherical coordinates.
SEARCH	Searches to satisfy selected end constraints by varying selected trajectory controls.

MARK IV: SEARCH

Subroutine Name	Description
SETUP	Interprets the input options, sets up the various indicators and parameters required by SEARCH, and prints out requested options.
SHIFT1	Calculates position and velocity relative to ephemeris bodies and extra bodies.
SIXC	Calls subroutine PLNTDQ to provide an integrated trajectory for SEARCH. (SIXC is the PROGRAM for the (6,3) secondary overlay).
SIXL	Calls one of three secondary overlays. The (6,0) primary overlay makes the DEQS precision integration package available to PLNTDQ for use in SEARCH, or to TRAJ or REFINE as required by PINT. (SIXL is the PROGRAM for the (6,0) primary overlay.)
SKET	Determines the trajectory type and sets an index accordingly.
SOLARP	Computes the acceleration due to solar radiation pressure and the gradient thereof, or the partial derivative of same.
SPER	Computes spherical coordinates of a cartesian 3-vector.
SRCHOV	Driver for the generalized search capability.
SSIZE	Computes a starting integration step size.
START	Computes cartesian injection state as a function of launch parameter controls.
STEPD	Computes new cartesian state on a conic, given the old state plus incremental time or true anomaly.
STEPI	Computes the array of orbital elements to define a conic for subroutine STEPT.
STEPT	Computes cartesian state at a given time on a given conic.
TCONIC	Computes the time from periapsis for a given true anomaly on a Keplerian conic.
TFRAC	Updates time in whole and fractional days from epoch.
THRUST	Calculates the acceleration due to thrust, the gradient thereof, or the partial derivative of same.
TIMEC	Converts calendar date and time to whole and fractional days from January 1, 1950.
TIMED	Converts a time interval from days, hours, minutes, and seconds to seconds.

MARK IV: SEARCH

Subroutine Name	Description
TIMES	Converts a time interval from seconds to days, hours, minutes and seconds and sets up an alphanumeric array for subsequent output.
TRDB	Computes the transformation to Darboux or local tangent plane.
TRUEA	Computes true anomaly, given semi-latus rectum, eccentricity, and radius in the orbit.
TTGO	Computes time-to-go function for patching conics.
UPDATE***	Updates mean orbital elements in time from one epoch to another.
VNORM	Normalizes a vector and also computes the magnitude, (ADOT*).
XOUT	Computes interpolation coefficients and writes them on a binary tape.

NOTES

NOTES

NOTES

PHILCO

SPACE & RE-ENTRY SYSTEMS DIVISION
Philco-Ford Corporation
Palo Alto, California 94303