

MPAD REPORT CONTROL

MSC-01594

COPY
DO NOT REMOVE



NATIONAL AERONAUTICS AND SPACE ADMINISTRATION

MSC INTERNAL NOTE NO. 70-FM-26

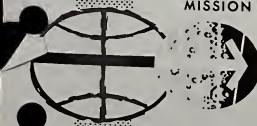
February 26, 1970

RTCC REQUIREMENTS FOR APOLLO 14:
TRAJECTORY COMPUTERS FOR TLI
AND MCC PROCESSORS

Lunar Mission Analysis Branch

MISSION PLANNING AND ANALYSIS DIVISION

MANNED SPACECRAFT CENTER
HOUSTON, TEXAS



MSC INTERNAL NOTE NO. 70-FM-26

PROJECT APOLLO

RTCC REQUIREMENTS FOR APOLLO 14: TRAJECTORY COMPUTERS
FOR TLI AND MCC PROCESSORS

By Brody O. McCaffety, William E. Moore, and Quentin A. Holmes
Lunar Mission Analysis Branch

February 26, 1970

MISSION PLANNING AND ANALYSIS DIVISION
NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
MANNED SPACECRAFT CENTER
HOUSTON, TEXAS

Approved:

Jerome D. Yonckman
for Ronald L. Berry, Chief
Lunar Mission Analysis Branch

Approved:

Robert E. Crumell
for John P. Mayer, Chief
Mission Planning and Analysis Division

CONTENTS

Section	Page
1.0 SUMMARY AND INTRODUCTION	1
1.1 The Trajectory Computers	2
1.2 Variables, Stopping Conditions	2
1.3 Lunar Orbits	3
2.0 ABBREVIATIONS	3
3.0 SUBROUTINES	4
3.1 Subroutine BURN	5
3.1.1 Function	5
3.1.2 Nomenclature	5
3.1.3 Method	6
3.2 Subroutine CTBODY	8
3.2.1 Function	8
3.2.2 Nomenclature	8
3.2.3 Method	9
3.3 Subroutine DGAMMA	14
3.3.1 Function	14
3.3.2 Nomenclature	15
3.3.3 Method	15
3.4 Subroutine EBETA	16
3.4.1 Function	16
3.4.2 Nomenclature	17
3.4.3 Method	17
3.5 Subroutine EPHM	18
3.5.1 Function	18
3.5.2 Remarks	18

Section	Page
3.6 Subroutine ELEMNT	18
3.6.1 Function	18
3.6.2 Nomenclature	19
3.6.3 Method	19
3.7 Subroutine FCOMP	20
3.7.1 Function	20
3.7.2 Nomenclature	20
3.7.3 Method	20
3.8 Subroutine LIERAT	22
3.8.1 Function	22
3.8.2 Nomenclature	22
3.8.3 Method	22
3.9 Subroutine LOPC	24
3.9.1 Function	24
3.9.2 Nomenclature	24
3.9.3 Method	25
3.10 Subroutine PATCH	26
3.10.1 Function	26
3.10.2 Nomenclature	26
3.10.3 Method	27
3.11 Subroutine RBETA	29
3.11.1 Function	29
3.11.2 Nomenclature	30
3.11.3 Method	30
3.12 Subroutine RNTSIM	33
3.12.1 Function	33
3.12.2 Nomenclature	33
3.12.3 Method	34
3.13 Subroutine RTASC	35
3.13.1 Function	35

Section	Page
3.14 Subroutine RVIO	35
3.14.1 Function	35
3.14.2 Nomenclature	35
3.14.3 Method	36
3.15 Subroutine TLIBRN	37
3.15.1 Function	37
3.15.2 Remarks	37
3.16 Subroutine TLMC	37
3.16.1 Function	37
3.16.2 Nomenclature	37
3.16.3 Method	38
3.17 Subroutine XBETA	39
3.17.1 Function	39
3.17.2 Nomenclature	39
3.17.3 Method	40
3.18 Subroutine PRCOMP	41
3.18.1 Function	41
3.18.2 Nomenclature	41
3.18.3 Method	45
3.19 Subroutine MCOMP	47
3.19.1 Function	47
3.19.2 Method	47
3.20 Subroutine PPC	48
3.20.1 Function	48
3.20.2 Nomenclature	48
REFERENCES	76

FLOW CHARTS

Flow Chart		Page
1	Subroutine PPC	50
2	Real time applications of the generalized iteritor	59
3	Translunar midcourse first guess trajectory computer	60
4	Functional flow of analytical trajectory computer for conic mission	62
5	Integrating trajectory computer	71

RTCC REQUIREMENTS FOR APOLLO 14: TRAJECTORY COMPUTERS

FOR TLI AND MCC PROCESSORS

By Brody O. McCaffety, William E. Moore, and Quentin A. Holmes

1.0 SUMMARY AND INTRODUCTION

This note is the last of a series documenting the Generalized Iterator used in the RTCC translunar injection and midcourse correction processors for Apollo 14. The mathematical formulation of the iterator itself is general and is documented in reference 1; the programs for the various mission options provided by the processors have been documented in references 2 and 3. This note gives the trajectory computers of the translunar injection and midcourse correction processors.

The term "Generalized Iterator" as used here refers to the whole program - supervisor, trajectory computer, and iterator. The iterator is a general formulation that applies to any problem which involves the solution of a minimum or maximum value of a given function. The technique has other RTCC applications in addition to its use in the TLI and MCC processors. The supervisor sets the dependent and independent variables to solve a desired problem. The trajectory computer indicates the sequence of events or computations needed to generate the desired trajectory. This note gives the functional and detailed information about the trajectory computer, and the subroutines used to construct a trajectory, their function, and their algorithms.

There are basically five types of trajectories generated by the TLI and MCC processors.

- a. Elliptical trajectories generated out of earth orbit (i.e., E-type ellipses and hybrid ellipses)
- b. x, y, z, and t return-to-nominal trajectories generated during translunar coast
- c. Free-return trajectories generated from EPO or translunar coast

d. Free-return, BAP reoptimized trajectories generated during translunar coast

e. Non-free-return, BAP reoptimized trajectories generated during translunar coast

These possibilities are shown in flow chart 3.

The calculation of each type involves the use of analytical and integrated computations. Conic, or analytical, trajectories are used in first guess routines to generate initial conditions and in optimizations to shorten computation time. Integrated calculations are necessary to provide precision target conditions. An explanation of the ways these computation modes are used together is contained in references 2, 3, 4, and 5.

1.1 The Trajectory Computers

Separate trajectory computers are used to provide first guesses for the midcourse correction, for the conic, and for the precision trajectory computations.

The MCC first guess trajectory computer solves Lambert's problem. Subroutine TLMC computes the first guess trajectory for the MCC. The flow diagram is shown in flow chart 4.

A functional flow diagram of the analytic trajectory computer for conic mission calculations is given in flow chart 5. This flow chart shows the general flow indicating the sequence of state vector calculations, the entry of the appropriate independent variables, the calculation of the dependent variables, and the sequence of the mass history calculations.

The precision propagation of an arc is done using the Herrick-Beta technique documented in the appendix of reference 5. Flow chart 5 shows the functional flow of the precision trajectory computer.

1.2 Variables, Stopping Conditions

Independent and dependent variables for the different trajectory computers are shown in tables I and II.

The stopping conditions for the computers are indicated in the flow diagrams but are also discussed here. Integrated trajectories that return to the nominal x , y , and z of the LOI node stop at the time of the node; the non-free-return BAP options also integrate the same arc and stop on the time of the node obtained from the conic optimization.

However, the precision transearth trajectory used in the lunar flyby stops on an entry flight-path angle. Finally, during the iteration process, before the height of LOI is completely correct, the position and velocity vectors at the start of LPO are scaled to circular conditions. These vectors are used to compute the rest of the trajectory; thus, the integrity of those independent variables based on the desired height of the orbit is retained; for example, ΔT_{10} , ΔT_{11s} .

1.3 Lunar Orbits

The initial lunar orbit may be either an ellipse or a circle. Subroutine PRCOMP is used with the LOI and DOI maneuvers. This subroutine integrates backward from the lunar landing site in an approximation to the initial LPO to obtain orbital elements at LOI and DOI times. This routine provides the link between the state vector at the start of LOI and the state vector at first pass over the lunar landing site.

2.0 ABBREVIATIONS

BAP	best adaptive path
EOI	earth orbit insertion
EMP	earth-moon plane
EPO	earth parking orbit
LIM	lunar landing mission
LOI	lunar orbit insertion
LOPC	lunar orbit plane change prior to lunar module ascent
LPO	lunar parking orbit
MCC	midcourse correction
RTCC	Real-Time Computer Complex
TEI	transearth injection
TLI	translunar injection

3.0 SUBROUTINES

The subroutines and computation modules used in the trajectory computers are listed in table III. The following subroutines are involved.

- a. BURN - simulates impulsive thrusting for application of a delta velocity magnitude, delta azimuth, and delta flight-path angle in the topocentric reference frame.
- b. CTBODY - used for propagation of a conic state vector for a specified time interval.
- c. DGAMMA - determines the universal conic variable from periapsis to the nearest specified flight-path angle.
- d. EBETA - determines the interval in the universal conic variable from a given state vector to periapsis.
- e. ELEMT - calculates a set of orbital elements from a given state vector, time, and central body constant.
- f. EPHM - obtains earth and moon states vectors relative to each other, solar position, and a precession-nutation-libration direction cosine matrix from the magnetic tape ephemeris.
- g. FCOMP - evaluates the universal conic functions for a specified value of the universal conic variable.
- h. LIBRAT - performs librations upon an input state vector and does a reference transformation.
- i. LOPC - computes the size and effect of the lunar orbit plane change (CSM2).
- j. PATCH - accomplishes patching of the geocentric and selenocentric vehicle state vectors at the sphere of action of the moon.
- k. RBETA - determines the value of the universal conic variable to propagate from a given state vector to a specified radial magnitude.
- l. RNTSIM - determines the landing conditions.
- m. RTASC - determines right ascension of the Greenwich meridian.
- n. RVIO - transforms a given set of coordinates in Cartesian or spherical form to the other form.

o. TLIBRN - simulates the translunar injection thrusting maneuver by evaluation precomputed polynomials.

p. TLMC - in control when first guesses for delta azimuth, delta velocity, and delta flight-path angle are determined for translunar abort of midcourse maneuvers.

q. XBETA - propagates a given state vector through a specified universal conic β to a desired state vector. The β is the stopping condition for XBETA.

r. PRCOMP - simulates lunar parking orbits from the start of LOI through first pass over the lunar landing site by integrating backwards in time in an approximation to the initial LPO.

s. MCOMP - computes the mass after each maneuver (accounting for maneuvers by the SPS or DPS) and returns an error message if all DPS fuel is used.

t. PPC - computes ΔV for plane changes in circular lunar parking orbits using conic propagation.

The remaining text of this internal note will be devoted to a detailed description of the input, output, and the mathematics needed for each of the subroutines listed above. All lunar orbit computations will be computed using the lunar radius at the landing site and not the mean radius of the moon.

3.1 Subroutine BURN

3.1.1 Function.- Subroutine BURN simulates impulsive thrusting of the vehicle. The ideal velocity equation is used to determine propellant consumption. This subroutine is used for the MCC, LOI, LOPC, and TEI burns.

3.1.2 Nomenclature.-

Symbol	Input (I), Output (O)	Definition
v_c	0	circular velocity
Δv_R	0	characteristic delta velocity

Symbol	Input (I), output (O)	Definition
Δv		change in scalar velocity during burn
v_{pl}	I	velocity at perilune of the desired ellipse (if ellipse is required)
$\Delta \gamma$	I	change in flight-path angle during burn
$\Delta \psi$	I	change in azimuth during burn
I_{sp}	I	specific impulse
m_f/m_o	O	ratio of mass after burn to mass before burn
g_o	I	constant used to convert pounds force to pounds mass
μ	I	gravitational constant of current reference body
R	I	initial position vector
\dot{R}	I	initial velocity vector
γ_o	I	initial flight-path angle
\dot{R}_1, \dot{R}_2	I	intermediate velocity vectors
R_f	O	final position vector
\dot{R}_f	O	final velocity vector

3.1.3 Method.- The vector R_f is the same as R ; that is, the routine assures that the position does not change during the maneuver. Compute the values for r and v according to equations (1) and (2)

$$r = \sqrt{R \cdot \dot{R}} \quad (1)$$

$$v = \sqrt{R \cdot \ddot{R}} \quad (2)$$

If a circular state vector after the burn is specified, enter

$$v_c = \sqrt{\frac{\mu}{r}}$$

$$\Delta v = v_c - v$$

$$\Delta \gamma = -\gamma_0$$

If an elliptical state vector is specified, enter

$$\Delta v = v_{pc} - v$$

$$\Delta \gamma = -\gamma_0$$

In the other more general option, Δv , $\Delta \gamma$, and $\Delta \psi$ are entered as input. Compute equations (3) through (7)

$$d = R \cdot \dot{R} \quad (3)$$

$$h = |R \times \dot{R}| \quad (4)$$

$$\dot{R}_1 = \dot{R} \cos \Delta \gamma + \frac{v^2 R - d \dot{R}}{h} \sin \Delta \gamma \quad (5)$$

$$\dot{R}_2 = \frac{2R(\dot{R} \cdot \dot{R}_1)}{r^2} \sin^2 \frac{\Delta\psi}{2} + \dot{R} \cos \Delta\psi - \frac{R \times \dot{R}_1}{r} \sin \Delta\psi \quad (6)$$

$$\dot{R}_f = \dot{R}_2 \left(1 + \frac{\Delta v}{v} \right) \quad (7)$$

Equation (7) represents the velocity vector part of the state S_f after the burn. The characteristic velocity can be determined from equation (8).

$$(\Delta v_R)^2 = \Delta v^2 + 4v(v + \Delta v) \left(\sin^2 \frac{\Delta\gamma}{2} + \frac{h^2 \cos \Delta\gamma - h\dot{a} \sin \Delta\gamma}{r^2 v^2} \sin^2 \frac{\Delta\psi}{2} \right) \quad (8)$$

The mass ratio is represented by equation (9).

$$\frac{m_f}{m_o} = \exp \left(\frac{-\sqrt{(\Delta v_R)^2}}{I_{sp} g_o} \right) \quad (9)$$

3.2 Subroutine CTBODY

3.2.1 Function.- Subroutine CTBODY determines the propagated state vector at a specified time, Δt , from a given epoch state vector. This is the classical problem of Kepler and must be solved iteratively because of the transcendental relationship between time and the anomalies.

3.2.2 Nomenclature.-

Symbols	Input(I), output (0)	Definition
K	I	central body indicator
r_o	I	position vector magnitude

Symbols	Input (I), output (O)	Definition
v_o	I	velocity vector magnitude
μ	I	gravity constant
α	O	square of universal variable divided by semimajor axis
$F_1, F_2, \left. \begin{array}{l} F_3, F_4 \end{array} \right\}$	O	functions of the universal variable
a		semimajor axis
R_o	I	initial position vector
\dot{R}_o	I	initial velocity vector
t_o	I	initial time
R_f	O	final position vector
\dot{R}_f	O	final velocity vector
r_m		radius of moon
J		$3/2 J_2$, second harmonic of moon's gravity
t_f	I	final time

3.2.3 Method.- Determine the interval of propagation with equation (10).

$$\Delta t = t_f - t_o \quad (10)$$

If $|\Delta t| < 10^{-12}$, the final state vector is the initial state vector, and the operation is complete; if not $|\Delta t| < 10^{-12}$, equations (11) and (12) result

$$\frac{1}{a} = \frac{2}{r_0} - \frac{v^2}{\mu} \quad (11)$$

$$D_0 = R_0 \cdot \dot{R}_0 \quad (12)$$

A first guess of the universal variable for the Newton-Raphson iteration is made from equations (13) and (14).

$$\beta = \frac{1}{5} \Delta t \frac{\sqrt{\mu}}{r_0} \quad (13)$$

$$\alpha = -\frac{\beta^2}{a} \quad (14)$$

Subroutine FCOMP is entered to obtain F_1 , F_2 , F_3 , and F_4 ; and the time equation is evaluated according to equations (15) and (16).

$$t = \left[\beta^2 F_1 + \frac{D_0}{\sqrt{\mu}} \beta F_2 + r_0 F_3 \right] \frac{\beta}{\sqrt{\mu}} \quad (15)$$

$$r = \frac{D_0}{\sqrt{\mu}} \beta F_3 + \beta^2 F_2 + r_0 F_4 \quad (16)$$

Increment β as defined by equation (17)

$$\beta = \beta + (\Delta t - t) \frac{\sqrt{D}}{r} \quad (17)$$

Equation (15) is evaluated with the new value of β , and the Newton-Raphson iteration [eq. (17)] continues until the convergence tolerance of 1×10^{-12} is met.

$$\left| \frac{t - \Delta t}{\Delta t} \right| < 10^{-12} \quad (18)$$

Exit with an error message if no convergence is obtained after, for example, 10 iterations.

As the iterations proceed, β will move in the same direction until it is very close to the answer. To protect against the tolerance of 10^{-12} in equation (18) being too small, the signs of successive values of $\Delta t - t$ are compared. If two successive iterations should have different signs before equation (18) is satisfied, β is replaced by the average of the two values associated with these iterations, and the process is repeated until the relative difference between two values being averaged is less than 10^{-12} .

With the universal variable determined, the state vector at the final time is determined from equations (19) through (24).

$$f = 1 - \frac{\beta^2 P_2}{r_0} \quad (19)$$

$$g = t - \frac{\beta^3 P_1}{\sqrt{D}} \quad (20)$$

$$\dot{f} = -\frac{\sqrt{\mu}\beta F_3}{r_o^2 r} \quad (21)$$

$$\dot{g} = 1 - \frac{\beta^2 F_2}{r} \quad (22)$$

$$\dot{R}_f = f\dot{R}_o + g\dot{R}_o \quad (23)$$

$$\ddot{R}_f = \dot{f}\dot{R}_o + g\ddot{R}_o \quad (24)$$

Check to see whether the ascending node is to be precessed. If not, \dot{R} , \ddot{R} are produced as output. Otherwise (at time t_p), rotate \dot{R} , \ddot{R} to selenographic coordinates \dot{G}_o , \ddot{G}_o . The components of \dot{G}_o and \ddot{G}_o will be x , y , z and \dot{x} , \dot{y} , \dot{z} , respectively.

Compute n_1 , n_2 , and n with equation (25) through (27).

$$n_1 = \dot{z}x - z\dot{x} \quad (25)$$

$$n_2 = \dot{z}y - z\dot{y} \quad (26)$$

$$n = \sqrt{n_1^2 + n_2^2} \quad (27)$$

If $n \leq 10^{-12}$, return without preprocessing the node. Otherwise, compute equations (28), (29), and (30).

$$\cos \Omega = \frac{n_1}{n} \quad (28)$$

$$\sin \Omega = \frac{n_2}{n} \quad (29)$$

$$H = \frac{G_o \times \dot{G}_o}{|G_o \times \dot{G}_o|} \quad (30)$$

The components of H will be h_1, h_2, h_3 . Then $\cos i = h_3$, and

$$\sin i = \sqrt{h_1^2 + h_2^2}$$

Compute equations (31) through (38)

$$\Delta \Omega = -Jr_m \sqrt{2} \cos i \left(\frac{1}{a}\right)^3 \left(\frac{1}{a}\right)^{1/2} \Delta t \quad (31)$$

$$= -1.14161 \times 10^{-5} \cos i \left(\frac{1}{a}\right)^3 \left(\frac{1}{a}\right)^{1/2} \Delta t$$

$$d = G_o \cdot \dot{G}_o \quad (32)$$

$$r^2 = R^2 = G_o^2 \quad (33)$$

$$v^2 = \dot{R}^2 = \dot{G}_O^2 \quad (34)$$

$$N = \begin{bmatrix} \cos \Omega \cos \Delta\Omega - \sin \Omega \sin \Delta\Omega \\ \sin \Omega \cos \Delta\Omega + \cos \Omega \sin \Delta\Omega \\ 0 \end{bmatrix} \quad (35)$$

$$M = \begin{bmatrix} -\cos i (\sin \Omega \cos \Delta\Omega + \cos \Omega \sin \Delta\Omega) \\ \cos i (\cos \Omega \cos \Delta\Omega - \sin \Omega \sin \Delta\Omega) \\ \sin i \end{bmatrix} \quad (36)$$

$$G = \frac{\dot{z}r^2 - zd}{n} N + \frac{z |G_O \times \dot{G}_O|}{n} M \quad (37)$$

$$\dot{G} = \frac{\dot{z}d - zv^2}{n} N + \frac{\dot{z} |G_O \times \dot{G}_O|}{n} M \quad (38)$$

With the same time t_f , rotate G, \dot{G} back into selenocentric coordinates R_p, \dot{R}_p .

3.3 Subroutine DGAMMA

3.3.1 Function.- Subroutine DGAMMA determines the value of the universal variable necessary to obtain a state vector at a desired flight-path angle, with a specific initial position magnitude and the reciprocal of the semimajor axis.

3.3.2 Nomenclature.-

Symbol	Input (I), output (O)	Definition
r_o	I	magnitude of position vector at periapsis
$1/a$	I	reciprocal of semimajor axis
γ	I	flight-path angle
H	O	hyperbolic eccentric anomaly
E	O	elliptic eccentric anomaly
β	O	universal variable
e	O	eccentricity

3.3.3 Method.- Because the given state vector is computed at periapsis, equations (39) and (40) are satisfied.

$$e = 1 - \frac{r_o}{a} \quad (39)$$

$$c = \sqrt{\left| \frac{2r_o}{a} - \frac{r_o^2}{a^2} \right|} \quad (40)$$

If $\frac{1}{a} < 0$, the orbit is hyperbolic.

$$H = \ln \left[\frac{c}{e} \tan \gamma + \sqrt{1 + \left(\frac{c \tan \gamma}{e} \right)^2} \right]$$

$$\beta = H \sqrt{|a|}$$

If $\frac{1}{a} > 0$, the orbit is elliptic.

$$\sin E = \frac{c \tan \gamma}{e}$$

$$E = \tan^{-1} \left(\frac{\sin E'}{\sqrt{1 - \sin^2 E'}} \right)$$

(where $-\frac{\pi}{2} < E < \frac{\pi}{2}$)

$$\beta = E\sqrt{a}$$

If $\frac{1}{a} = 0$, the orbit is parabolic.

$$\beta = (\sin \gamma / \cos \gamma) \sqrt{2r_0}$$

3.3.4 Remarks.- On an ellipse, the eccentric anomaly is double-valued with respect to the flight-path angle. It is apparent from the equation for E that the algorithm always given the solution nearer periapsis.

This formulation does not provide for optimization of the same trajectory arc from a hyperbolic energy through parabolic to an elliptical energy.

For the elliptic case, γ may be such that $|\sin E| > 1$. In this instance, γ cannot be achieved, and there is an error.

3.4 Subroutine EBETA

3.4.1 Function.- Subroutine EBETA determines the universal variable necessary to obtain the state vector at periapsis.

3.4.2 Nomenclature.-

Symbol	Input (I), output (O)	Definition
$1/a$	O	reciprocal of the semimajor axis
R_0	I	initial position vector
\dot{R}_0	I	initial velocity vector
r_0		magnitude of initial position vector
v_0		magnitude of initial velocity vector
β	O	universal variable
μ	I	gravitational constant
E		elliptical eccentric anomaly
H		hyperbolic eccentric anomaly
e		eccentricity

3.4.3 Method.- The universal variable and the state vector at periapsis are determined by equations (41) and (42).

$$D_0 = R_0 \cdot \dot{R}_0 \quad (41)$$

$$1/a = 2/r_0 - v_0^2/\mu \quad (42)$$

If $a > 0$, the orbit is elliptic.

$$e \cos E = 1 - \frac{r_0}{a}$$

$$e \sin E = D_0 / \sqrt{\mu a}$$

$$E = \tan^{-1}(e \sin E/e \cos E)$$

$$\beta = -E\sqrt{a}$$

If $1/a = 0$, the orbit is parabolic.

$$\beta = -\frac{D_0}{\sqrt{\mu}}$$

If $a < 0$, the orbit is hyperbolic.

$$e \cosh H = 1 - r_0/a$$

$$e \sinh H = D_0 \sqrt{\mu |a|}$$

$$H = \ln \left[\frac{e \cosh H + e \sinh H}{|(e \cosh H)^2 - (e \sinh H)^2|^{1/2}} \right]$$

$$\beta = -H\sqrt{|a|}$$

This formulation does not provide for optimization of the same trajectory arc from a hyperbolic energy through parabolic to an elliptic energy.

3.5 Subroutine EPHM

3.5.1 Function.- Ephemeris subroutines locate, transmit into core, and interpolate data from an ephemeris tape. From these data, earth and moon state vectors relative to each other, solar position, and a precession-nutation-libration direction cosine matrix are obtained.

3.5.2 Remarks.- The ephemeris subroutines used in the RTCC will be system subroutines.

3.6 Subroutine ELEMT

3.6.1 Function.- Subroutine ELEMT calculates a set of orbital elements from a given state vector, time, and central body constant.

3.6.2 Nomenclature.-

Symbol	Input (I), output (O)	Definition
R	I	position vector
\dot{R}	I	velocity vector
H	O	angular momentum vector per unit mass
μ	I	gravity constant
a	O	semimajor axis
e	O	eccentricity
i	O	inclination of conic
n	O	mean motion
P	I	period
η	O	true anomaly

3.6.3 Method.- With R , \dot{R} , t , μ specified, the following seven quantities are calculated.

$$\frac{1}{a} = \left(\frac{2}{|R|} - \frac{|\dot{R}|^2}{\mu} \right) \quad (43)$$

$$e = \sqrt{\left(1 - \frac{|R|}{a} \right)^2 + \frac{(R \cdot \dot{R})^2}{\mu a}} \quad (44)$$

$$H = R \times \dot{R} \quad (45)$$

$$i = \cos^{-1} \left(\frac{h_z}{|H|} \right) \quad (46)$$

$$n = \frac{\mu^{1/2}}{|a|^{3/2}} \quad (47)$$

$$\eta = \tan^{-1} \left(\frac{|H|(R \cdot \dot{R})}{|H|^2 - u|R|} \right) \quad (48)$$

$$p = \frac{2\pi a \sqrt{a}}{n} \quad (49)$$

Equations (43) through (47) apply for all conics; equation (48) does not apply to circular orbits, and equation (49) does not apply to parabolas and hyperbolas.

3.7 Subroutine FCOMP

3.7.1 Function.- Subroutine FCOMP determines the functions of the universal variable necessary to express two-body state vector quantities with a specific epoch state vector. The functions are well defined by circular and hyperbolic functions except as the universal variable approaches zero. To avoid numerical difficulty, the same series expansion is always used. FCOMP is used by XBETA and CTBODY to evaluate the functions of the universal constant.

3.7.2 Nomenclature.-

Symbol	Input (I), output (O)	Definition
F_i	0	functions of the universal variable
α	I	parameter needed to obtain F

3.7.3 Method.-

$$F_j = \sum_{i=0}^{\infty} \frac{\alpha^i}{(2i + 4 - j)!} \quad j = 1, 2 \quad (50)$$

Equation (50) is used to compute F_1 and F_2 ; F_3 and F_4 are computed from equations (51) and (52).

$$F_3 = aF_1 + 1 \quad (51)$$

$$F_4 = aF_2 + 1 \quad (52)$$

Determine n (the number of terms to be used in the series) as follows: For $|a| < x$, $n = y$.

x	y
2^{-7}	5
2^{-5}	6
2^{-3}	7
2^{-2}	8
2^{-1}	9
1	10
2	11
4	13
8	15
16	18
32	21
64	25
128	30
256	38
512	46

3.8 Subroutine LIBRAT

3.8.1 Function.- Subroutine LIBRAT obtains an appropriate transformation matrix and transforms input state vectors from selenographic coordinates to other coordinate systems.

3.8.2 Nomenclature.-

Symbol	Input (I), output (O)	Definition
R	I and O	position vector
\dot{R}	I and O	velocity vector
t	I	time of state vector
K	I	indicator
ME		moon with respect to earth

3.8.3 Method.- Six options exist from conversion of state vectors to different coordinate systems.

K = 1 - Earth-moon plane to selenographic

K = 2 - Selenographic to earth-moon plane

K = 3 - Earth-moon plane to selenocentric

K = 4 - Selenocentric to earth-moon plane

K = 5 - Selenocentric to selenographic

K = 6 - Selenographic to selenocentric

When the earth-moon plane is involved, a matrix is used to convert either to or from this coordinate system. This matrix is formed as follows.

With the specified position R_{ME} and velocity V_{ME} of the moon with respect to the earth at each given time, \vec{i} , \vec{j} , and \vec{k} can be determined from equations (53), (54), and (55).

$$\vec{i} = \frac{\vec{R}_{ME}}{|\vec{R}_{ME}|} \quad (53)$$

$$\vec{k} = \frac{\vec{R}_{ME} \times \dot{\vec{R}}_{ME}}{|\vec{R}_{ME} \times \dot{\vec{R}}_{ME}|} \quad (54)$$

$$\vec{j} = \vec{k} \times \vec{i} \quad (55)$$

Set $A = (\vec{i}, \vec{j}, \vec{k})$ and note that $\vec{i}, \vec{j}, \vec{k}$ are taken as column vectors. Let A^T denote the transpose of A . Then if the selenocentric coordinates in the equatorial system are R, \dot{R} , EMP coordinates are defined by equations (56a) and (56b).

$$R' = A^T R \quad (56a)$$

$$\dot{R}' = A^T \dot{R} \quad (56b)$$

or, equivalently,

$$R = AR' \quad (57a)$$

$$\dot{R} = A\dot{R}' \quad (57b)$$

When a conversion is made from the selenocentric coordinate system to the selenographic (moon-fixed) coordinate system, the libration matrix is used.

With the precession-nutation-libration matrix, B at each given time and the selenocentric coordinates R, \dot{R} , transform to the selenographic coordinates R', \dot{R}' by equations (58a) and (58b).

$$R'' = BR \quad (58a)$$

$$\dot{R}'' = B\dot{R} \quad (58b)$$

conversely,

$$R = B^T R'' \quad (59a)$$

$$\dot{R} = B^T \dot{R}'' \quad (59b)$$

A combination of the two preceding techniques can be used to transform vectors from moon orbit plane to selenographic coordinates and the reverse.

3.9 Subroutine LOPC

3.9.1 Function.- Subroutine LOPC determines the size and effect of the lunar orbit plane change maneuver (CSM2).

3.9.2 Nomenclature.-

Symbol	Input (I), output (O)	Definition
m	I	number of revolutions from first pass over lunar landing site (LLS) to (CSM2+ 1/4)
n	I	number of revolutions from (CSM2 + 1/4) to second pass over LLS
P	I	period of orbit adjusted by the rotational rate of the moon
S _o	I	state vector at lunar landing
t _o	I	time at lunar landing

Symbol	Input (I), output (O)	Definition
Δt_1		time from first pass over LLS to CSM2
S_1		state vector before CSM2
Δt_2		time from first pass over LLS to second pass over LLS
S_2		predicted state vector at second pass over LLS
t_L		time of second pass over LLS if no CSM2
S_3	O	state vector after CSM2
$\frac{m_f}{m_o}$	O	mass ratio of CSM2 maneuver
R_2		position vector at second pass over LLS in selenographic coordinates
\dot{R}_2		velocity vector at second pass over LLS in selenographic coordinates
L	I	selenographic components of unit vector pointing to the LLS

3.9.3 Method.- Compute equation (60).

$$\Delta t_1 = \left(m - \frac{1}{4}\right)P \quad (60)$$

Use CTBODY (regressed) to propagate S_o from t_o to $(t_o + \Delta t_1)$ to obtain S_1 . Then compute equation (61).

$$t_L = t_o + \Delta t_2 = t_o + (m + n)P \quad (61)$$

Use CTBODY (regressed) to propagate S_0 to t_L to obtain S_2 .

Call LIBRAT at time t_L to transform S_2 to selenographic coordinates R_2, R_2 .

$$\Delta\psi = -\sin^{-1}\left(\frac{R_2 \times \dot{R}_2}{|R_2 \times \dot{R}_2|} \cdot L\right)$$

Call BURN to obtain S_3 and $\frac{m_r}{m_o}$, using S_1 , $\Delta\psi$, and I_{sp} .

3.10 Subroutine PATCH

3.10.1 Function.- Subroutine PATCH finds a point at which there is a specific ratio between the spacecraft and the earth and the moon, and the spacecraft changes reference bodies at that point.

3.10.2 Nomenclature.-

Symbol	Input (I), output (O)	Definition
R	I and O	position vector
\dot{R}	I and O	velocity vector
t	I and O	time of vector
r		magnitude of position vector
i		reference body subscript: i = 1, primary body i = 2, secondary body
Q	I	direction of patch in time
ERROR	O	error return
μ	I	gravitational constant
a	I	acceleration with respect to body i
β	I	universal variable

Symbol	Input (I), output (O)	Definition
KREF	I	primary reference indicator
R_{21}		position of the secondary body with respect to the primary body
r_{21}		magnitude of R_{21}

3.10.3 Method.- In the following, if KREF = 1 (earth reference input), the earth is referred to as the primary body and the moon as the secondary body. If KREF = 2 (moon reference input), the moon is primary and the earth is secondary.

Subscripts 1 and 2 indicate primary and secondary bodies, respectively. The ratio is defined as follows.

$$\text{Ratio} = \frac{r_2}{r_1} = \frac{\text{distance of spacecraft from secondary body}}{\text{distance of spacecraft from primary body}}$$

Therefore, for a specific two-body orbit, Ratio is a function of the orbital parameters, the universal variable β , and the moon-earth ephemeris data. The procedure is to calculate a second order Taylor's expansion giving Ratio in terms of the following first and second partial derivatives.

$$\frac{d \text{Ratio}}{d\beta} \quad \text{and} \quad \frac{d^2 \text{Ratio}}{d\beta^2}$$

Begin with an initial value of β , a corresponding initial value of Ratio, and an increment, $\Delta\beta$ to β . Set Ratio ($\beta + \Delta\beta$) equal to the desired value of Ratio, and solve the resulting quadratic equation for $\Delta\beta$. If the discriminant is less than zero, set $d^2\text{Ratio}/d\beta^2 = 0$ and solve the linear equation instead. With an initial guessed value of β , propagate the initial state vector (by XBETA) to a final state vector at the patch with respect to the primary reference body. The position of the secondary body with respect to the primary body is obtained from EPHM. A reference change is made, and r_2 and d_2 are calculated.

$$\Delta \text{Ratio} = R - \text{Ratio} \quad (62)$$

$$\text{where } R = \begin{cases} \frac{1}{0.275} & \text{if the moon is the primary body} \\ 0.275 & \text{if the earth is the primary body} \end{cases}$$

$$\frac{d \text{Ratio}}{d\beta} = \frac{1}{r_2 \sqrt{\mu_1}} \left(d_2 - \frac{r_2^2 d_1}{r_1^2} \right) \quad (63)$$

$$\text{where } d_i = \dot{R}_i \cdot \dot{R}_i, \quad i = 1, 2$$

$$\begin{aligned} \frac{d^2 \text{Ratio}}{d\beta^2} &= \frac{r_1}{\mu_1} \cdot \frac{v_2^2 + R_2 \cdot A_2}{r_2} - \frac{d_1 d_2}{\mu_1 r_1 r_2} \\ &\quad - \frac{d_2^2 r_1}{\mu_1 r_2^3} - \frac{r_2 v_1^2}{r_1 \mu_1} + \frac{r_2}{r_1^2} + \frac{2d_1^2 r_2}{\mu_1 r_1^3} \end{aligned} \quad (64)$$

$$\text{where } v_i^2 = \dot{R}_i \cdot \dot{R}_i, \quad i = 1, 2 \text{ and } A_2 = \frac{\mu_1 R_1}{r_1^3} + \frac{(\mu_1 + \mu_2)}{r_2^3} R_{21}$$

$$\Delta \beta = \frac{2 \Delta \text{Ratio}}{\frac{d \text{Ratio}}{d\beta} + \text{sign} \left(\frac{d \text{Ratio}}{d\beta} \right) \sqrt{\left(\frac{d \text{Ratio}}{d\beta} \right)^2 + 2 \Delta \text{Ratio} \frac{d^2 \text{Ratio}}{d\beta^2}}}$$

Replace β by $\beta + \Delta\beta$ and repeat the process until ΔRatio is less than 1×10^{-12} . The last state vector and time with respect to the secondary body are the output state and time.

The initial first guesses for the earth and moon as primary bodies are the values of β needed to propagate to 40 e.r. and 10 e.r., respectively.

Upon further reference to the routine using a given primary body, the last value of distance in that particular primary body is used to derive a first guess for β . This method implies that two distances are saved, one for each primary body; the maximum distance will be 60 e.r. if the earth is the primary body, and the maximum distance will be 15 e.r. if the moon is the primary body.

3.10.4 Remarks.- The last variable in the calling sequence is an error indicator which is a logical variable and which will return a value of .TRUE. when an error has occurred in the routine. There are four situations in which .TRUE. will be set up on the error indicator.

- a. If the patch iterative procedure fails to converge within 10 iterations.
- b. If the ephemeris data table has not been initialized or the time calculated within the routine is outside the range of the ephemeris data.
- c. If the magnitude of the input position vector is greater than 40 e.r. when the earth is the primary body or 10 e.r. when the moon is the primary body and if the conic defined by the input state vector is such that the radius of periapsis is greater than 40 e.r. when the earth is the primary body or 10 e.r. when the moon is the primary body.
- d. Any error indicator from subroutine RBETA.

3.11 Subroutine RBETA

3.11.1 Function.- Subroutine RBETA determines the universal variable necessary to obtain a state vector at a desired radial magnitude, given an initial state vector.

3.11.2 Nomenclature.-

Symbol	Input (I), output (O)	Definition
R_0	I	initial position state vector
\dot{R}_0	I	initial velocity state vector
r_0	I	magnitude of initial position vector
v_0	I	magnitude of initial velocity vector
Q	I	direction indicator
β	O	universal variable
E		elliptic eccentric anomaly
H		hyperbolic eccentric anomaly
ERROR	O	indicator of error return
μ		gravity constant of reference body
r	I	desired radius magnitude
a		semimajor axis
e		eccentricity

3.11.3 Method.- Subroutine RBETA is restricted to cases in which the desired radius magnitude is greater than the initial magnitude. If an orbit is circular, the subroutine gives a return with the error indicator set .TRUE. because any β would suffice if the desired distance is the radius of the circle, and no β exists if the desired radius is not the circular radius. In general, the solution for a desired radius is double-valued; therefore an indicator Q is provided to select the desired solution. If $Q = +1$, the solution will be ahead of the initial position with respect to the direction of motion; if $Q = -1$, the solution will be behind the initial position.

Determine the dot produce of R_o and \dot{R}_o , the semimajor axis, and the eccentricity with equations (65) through (67).

$$D_o = R_o \cdot \dot{R}_o \quad (65)$$

$$\frac{1}{a} = \frac{2}{r_o} - \frac{v_o^2}{\mu} \quad (66)$$

$$e = \sqrt{\left(1 - \frac{r_o}{a}\right)^2 + \frac{D_o^2}{\mu a}} \quad (67)$$

If $1/a < 0$, the orbit is hyperbolic.

$$\cosh H_o = \frac{1}{e} \left(1 - \frac{r_o}{a}\right)$$

$$\cosh H = \frac{1}{e} \left(1 - \frac{r}{a}\right)$$

$$H_o = \pm \ln \left(\cosh H_o + \sqrt{\cosh^2 H_o - 1} \right)$$

where the sign is chosen to be the sign of D_o .

$$H = \ln \left(\cosh H + \sqrt{\cosh^2 H - 1} \right)$$

$$\theta = H_o - QH$$

$$B = Q|\theta| \sqrt{|a|}$$

If $1/a > 0$, the orbit is elliptic.

$$\cos E_o = \frac{1}{e} \left(1 - \frac{r_o}{a} \right)$$

$$\cos E = \frac{1}{e} \left(1 - \frac{r}{a} \right)$$

$$E_o = \pm \tan^{-1} \frac{\sqrt{1 - \cos^2 E_o}}{\cos E_o}$$

where the sign is chosen to be the sign of D_o .

$$E = \tan^{-1} \frac{\sqrt{1 - \cos^2 E}}{\cos E}$$

$$\theta = E_o - QE$$

$$\beta = Q|\theta|\sqrt{a}$$

If $1/a = 0$, the orbit is parabolic.

$$\beta = \frac{D_o}{\sqrt{\mu}} + Q\sqrt{\frac{D_o^2}{\mu} + 2(r - r_o)}$$

3.11.4 Remarks.- If any of the radicands involving r is less than zero, the distance r is impossible, and the calculation is suspended with error indicator set .TRUE.

3.12 Subroutine RNTSIM

3.12.1 Function.- Subroutine RNTSIM determines the landing conditions, conditions of delta time from entry to landing and longitude of landing.

3.12.2 Nomenclature.-

Symbol	Input (I), output (O)	Definition
λ	O	computed longitude of landing
λ_L	I	longitude of landing
$\Delta\lambda$	O	error in longitude of landing
R	I	position vector at reentry
R	I	velocity vector at reentry
r	I	magnitude of position vector at reentry
v	I	magnitude of velocity vector at reentry
t	I	time of entry
RR	I	entry range, n. mi.
Δt	I	time from reentry to landing
ϕ_L	O	latitude at landing
α_L	O	right ascension at landing
α_G	O	Greenwich right ascension at time of landing
γ	I	flight-path angle at reentry
θ	O	central angle between reentry and landing

3.12.3 Method.—With R , \dot{R} , and RR specified, the values of p , θ , S , ϕ_L , and α_L can be determined with equation (68) through (72).

$$P = \frac{\dot{R}}{v} \frac{1}{\cos \gamma} - \frac{R}{r} \tan \gamma \quad (68)$$

$$\theta = RR/3443.933585 \quad (69)$$

$$S = \frac{R}{r} \cos \theta + P \sin \theta \quad (70)$$

where S is the position at landing.

$$\phi_L = \tan^{-1} \frac{S_z}{\sqrt{S_x^2 + S_y^2}} \quad (71)$$

$$\alpha_L = \tan^{-1} \frac{S_y}{S_x} \quad (72)$$

Call RTASC at time $t + \Delta t$ to obtain α_G . Then

$$\Delta\lambda = \alpha_L - \alpha_G - \lambda_L$$

$$\lambda = \alpha_L - \alpha_G$$

Reduce $\Delta\lambda$ by any excess multiples of 2π . If the result is greater than π , subtract 2π ; if the result is less than or equal to $-\pi$, add 2π . Thus, $-\pi < \Delta\lambda \leq \pi$.

To allow partial derivatives to be obtained correctly despite the discontinuities inherent in this scheme, the following procedure is used when computing the trajectories involved in partial derivative calculations. After each nominal trajectory computation, the value of $\Delta\lambda$ is retained. During the perturbed trajectory computations, this value, called $\Delta\lambda_0$, is compared with the current value of $\Delta\lambda$. If $(\Delta\lambda - \Delta\lambda_0) < -\pi$, then $\Delta\lambda$ is replaced by $\Delta\lambda + 2\pi$; if $(\Delta\lambda - \Delta\lambda_0) > \pi$, $\Delta\lambda$ is replaced by $\Delta\lambda - 2\pi$.

3.13 Subroutine RTASC

3.13.1 Function.- Subroutine RTASC determines the right ascension of the Greenwich meridian. The computation for the right ascension is specified in reference 7.

3.14 Subroutine RVIO

3.14.1 Function.- Subroutine RVIO transforms a given set of coordinates in Cartesian or spherical form to the other form.

3.14.2 Nomenclature.-

Symbol	Input (I), output (O)	Definition
R	I and O	position vector
\dot{R}	I and O	velocity vector
r	I and O	position magnitude
v	I and O	velocity magnitude
x	I and O	x component of position vector
y	I and O	y component of position vector
z	I and O	z component of position vector
\dot{x}	I and O	x component of position vector
\dot{y}	I and O	y component of position vector
\dot{z}	I and O	z component of position vector

Symbol	Input (I), output (O)	Definition
θ	I and O	latitude
ϕ	I and O	right ascension angle
γ	I and O	flight-path angle
ψ	I and O	azimuth angle

3.14.3 Method.- To transform spherical coordinates to Cartesian coordinates, use equations (73) through (76)

$$x = r \cos \phi \cos \theta \quad (73)$$

$$y = r \cos \phi \sin \theta \quad (74)$$

$$z = r \sin \phi \quad (75)$$

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} \cos \phi \cos \theta & -\sin \theta & -\sin \phi \cos \theta \\ \cos \phi \sin \theta & \cos \theta & -\sin \phi \sin \theta \\ \sin \phi & 0 & \cos \phi \end{bmatrix} \begin{bmatrix} v \sin \gamma \\ v \cos \gamma \sin \psi \\ v \cos \gamma \cos \psi \end{bmatrix} \quad (76)$$

To transform Cartesian coordinates to spherical coordinates, use equations (77) through (82).

$$r = \sqrt{x^2 + y^2 + z^2} \quad (77)$$

$$\phi = \sin^{-1} \frac{z}{r} \quad (78)$$

$$\theta = \tan^{-1} \frac{y}{x} \quad (79)$$

$$v = \sqrt{\dot{x}^2 + \dot{y}^2 + \dot{z}^2} \quad (80)$$

$$\gamma = \sin^{-1} \frac{\dot{R} \cdot \dot{R}}{rv} \quad (81)$$

$$\psi = \tan^{-1} \left(\frac{h_z}{zr - zv} \right) = \tan^{-1} \left(\frac{\dot{x}\dot{y} - \dot{y}\dot{x}}{zr - z \frac{\dot{R} \cdot \dot{R}}{r}} \right) \quad (82)$$

3.15 Subroutine TLIBRN

3.15.1 Function.- Subroutine TLIBRN simulates the translunar injection thrusting maneuver by use of a precomputed polynomial.

3.15.2 Remarks.- The method of this subroutine is contained in references 7 and 8.

3.16 Subroutine TLMC

3.16.1 Function.- Subroutine TLMC determines the first guesses for delta azimuth, delta velocity, and delta flight-path angle for a translunar state at abort or midcourse.

3.16.2 Nomenclature.-

Symbol	Input (I), output (O)	Definition
S	I and O	state vector
t	I	time of state vector S
t _p	I	nominal time of node
x	I	x component of position vector
y	I	y component of position vector
z	I	z component of position vector

Symbol	Input (I), output (O)	Definition
r	I	desired radius at the perilune
λ		longitude of perilune in earth-moon plane system
v		velocity magnitude at perilune
γ		flight-path angle at perilune
ϕ	I	latitude of perilune in earth-moon plane system
ψ		azimuth of perilune in earth-moon system
Δt	I	amount of change in t_n (for non-free-return)
t_n		adjusted time of node
ERROR	O	flight indicating an error in TIMC
ΔV	O	change in scalar velocity for MCC
$\Delta \gamma$	O	change in flight-path angle for MCC
$\Delta \psi$	O	change in azimuth for MCC

3.16.3 Method.- Compute the adjusted time of node: $t_n = t_p + \Delta t$.

The earth-moon plane (EMP) matrix is obtained by subroutine LIBRAT at the time t_n for use in transforming the EMP coordinates at the node to the selenocentric system.

The next step sets the dependent variable limits, weights, and weight cuts. Three dependent variables, x, y, z, are defined as the components of the position vector at abort or midcourse. They are designated class 1 variables. The minimum and maximum required values of the

position components are found by addition and subtraction of a small tolerance (10^{-5} e.r.) to the abort position components. Flight-path angle at time of abort is a class two dependent variable with limits of 90-182°.

After the dependent variables have been described, the independent variables are set up and given a first guess. The first independent variable is the longitude of perilune, and the second independent variable is the velocity of perilune. First guesses for these two variables are obtained from the empirical equations given in reference 3. The third independent variable is the azimuth at the perilune, and the first guess is $\frac{\pi}{2}$ to obtain a retrograde lunar approach hyperbola.

With the perilune forced to lie at the required EMP latitude and to have the required height and flight-path angle, the above independent variables determine the state vector at perilune in the EMP. This trajectory is propagated backward to the initial time, t . The generalized iterator then finds the set of independent variables necessary to obtain the dependent variables at abort; that is, the abort position components. Once converged, the differences between the azimuth, flight-path angle, and velocity before abort, and the values after abort necessary to obtain the above node conditions are determined. These values become first guesses for the MCC maneuver.

3.17 Subroutine XBETA

3.17.1 Function.- Subroutine XBETA determines the state vector relative to the initial state vector for a desired value of the universal variable.

3.17.2 Nomenclature.-

Symbols	Input (I) output (O)	Definition
β	I	universal variable
K	I	central body indicator
F_1		functions of the universal variable
μ		gravity constant

Symbols	Input (I), output (O)	Definition
R_0	I	initial position vector
\dot{R}_0	I	initial velocity vector
r_0	I	magnitude of initial position vector
v_0	I	magnitude of initial velocity vector
t_0	I	initial time
R	O	fixed position vector
\dot{R}	O	fixed velocity vector
t	O	final time

3.17.3 Method.- From the initial state vector, the final state vector is determined as a function of β with equation (83) through (85).

$$D_0 = R_0 \cdot \dot{R}_0 \quad (83)$$

$$1/a = 2/r_0 + v_0^2/\mu \quad (84)$$

$$a = -\beta^2/a \quad (85)$$

Call subroutine FCOMP and determine the functions of the universal variable. Use equations (86) through (94).

$$t = \left(\beta^2 F_1 + \frac{D_0 \beta F_2}{\sqrt{\mu}} + r_0 F_3 \right) \beta / \sqrt{\mu} \quad (86)$$

$$t_f = t_o + t \quad (87)$$

$$r = \left(\frac{D_o F_3}{\sqrt{\mu}} + \beta F_2 \right) \beta + r_o F_4 \quad (88)$$

$$f = 1 - \frac{\beta^2 F_2}{r_o} \quad (89)$$

$$g = t - \beta^3 F_1 \mu \quad (90)$$

$$\dot{f} = -\sqrt{\mu} \beta F_3 / r_o r \quad (91)$$

$$\dot{g} = 1 - \beta^2 F_2 / r \quad (92)$$

$$R = f R_o + g \dot{R}_o \quad (93)$$

$$\dot{R} = \dot{f} R_o + \dot{g} \dot{R}_o \quad (94)$$

3.18 Subroutine PRCOMP

3.18.1 Function.- Subroutine PRCOMP constructs state vectors at LOI and DOI based upon the coordinates of the lunar landing site and the orientation and shapes of the LOI and DOI ellipses. This routine is called only twice during a real-time BAP computation: at the start of the first select mode and again at the beginning of the optimize mode.

3.18.2 Nomenclature.-

Symbol	Input (I), output (O)	Definition
ψ LLS		selenographic approach azimuth to the lunar landing site
ϕ LLS		selenographic latitude of the lunar landing site

Symbol	Input (I), output (O)	Definition
λ_{LLS}		selenographic longitude of the lunar landing site
r_{lls}		radius of the lunar landing site
h_{alls}	I	apolune altitude of DOI ellipse during landing rev
h_{p11s}	I	perilune altitude of DOI ellipse during landing rev
$\Delta\omega_1$	I	central angle between perilune of DOI ellipse and lunar landing site during the landing rev
n_1	I	estimate of true anomaly of LOI (on ellipse)
HA_{IP01}	I	apolune altitude of LOI ellipse
HP_{IP01}	I	perilune altitude of LOI ellipse
RVS2	I	number of complete revs spent in the DOI ellipse
RVS1	I	total number of revs spent in the LOI ellipse (a non-integer)
R1		integer part of RVS1
DR1		decimal part of RVS1 converted to radians
RA_{IP01}	O	radius of apolune on the LOI ellipse
RP_{IP01}		radius of perilune on the LOI ellipse
a_{lls}		semimajor axis on the DOI ellipse
T_{nd}		time of the nodal state on the approach hyperbola

Symbol	Input (I), output (O)	Definition
T_{LLS}		estimated time of first pass over the landing site
Δt_2		increment of time required for a conic trajectory to propagate from a true anomaly of n_1 forward through DR_1 radians along the LOI ellipse
n_2		true anomaly on the LOI ellipse at the position of DOI
\hat{u}_{lls}		selenocentric unit position vector of the landing site
\hat{u}_{pc}		selenocentric unit position vector to the hyperbolic perilune
DA		fraction of a rev (in radians) traveled in the DOI ellipse
$\{\phi_2, \lambda_2, R_2\}$ $\{V_2, \gamma_2, \psi_2\}$		polar components of a selenographic state vector over the lunar landing site at time T_{LLS}
T_L		approximate time of DOI obtained by backward integration
A_L	0	semimajor axis of the DOI ellipse at time T_L
E_L	0	eccentricity of the DOI ellipse at time T_L
\hat{u}_{p1}	0	unit selenocentric perilune position vector of the LOI ellipse at time T_L
RA2		apolute radius associated with the DOI ellipse at time T_L

Symbol	Input (I), output (O)	Definition
RP2		Perilune radius associated with the DOI ellipse at time T_L
$\left\{ \begin{array}{l} \phi, \lambda, R \\ V, \gamma, \psi \end{array} \right\}$		polar components of a selenocentric state at the start of DOI
\vec{R}_{LI}		position vector on the LOI ellipse obtained by backward integration to time T_{nd}
\vec{V}_{LI}		velocity vector on the LOI ellipse obtained by backward integration to time T_{nd}
\hat{u}'_{pl}		unit perilune position vector associated with R_{LI} , V_{LI}
RAI		radius of apolune associated with T_{LI} , V_{LI}
RP1		radius of perilune associated with R_{LI} , V_{LI}
Δw_p	0	change in perilune position of the LOI ellipse because of propagation
Δh_a	0	change in apolune altitude of the LOI ellipse because of propagation
Δh_p	0	change in perilune altitude of the LOI ellipse because of propagation
ΔT	0	time increment from LOI to DOI
SGSLOI	0	selenographic state vector at the end of LOI
\hat{u}_L	0	selenographic unit position vector of perilune on the DOI ellipse at time T_L

3.18.3 Method.- Input provides a nodal state on the approach hyperbola from the trajectory computer; the lunar landing site coordinates and the selenographic approach azimuth are taken from preflight data, while the size, orientation, and duration of the LOI and DOI ellipses are specified by MED quantities h_{alls} , h_{p11s} , $n1$, HA_{LP01} , HP_{LP01} , $RVS1$, and $RVS2$, and $\Delta\omega_1$.

Set $RA_{LP01} = HA_{LP01} + r_{11s}$, $RP_{LP01} = HP_{LP01} + r_{11s}$, and $a_{11s} = r_{11s} + (h_{alls} + h_{p11s})/2.0$; then use equation (95) to compute an estimate of the time of first pass over the lunar landing site.

$$T_{LLS} = T_{nd} + \frac{2\pi}{\sqrt{\mu}} \left\{ R1 \left[\frac{RA_{LP01} + RP_{LP01}}{2.0} \right]^{3/2} + RVS2 \left[a_{11s} \right]^{3/2} \right\} + \Delta t_2 \quad (95)$$

where Δt_2 is the time increment required for a conic to propagate from a true anomaly of $n1$ on the LOI ellipse forward through DRI radians. (The true anomaly at the end of this conic propagation is saved as η_2).

Next, form a unit position vector to the landing site, transform it to selenocentric coordinates at time T_{LLS} , and call it \hat{u}_{11s} . Let $\Delta\alpha' = \arccos(\hat{u}_{11s} \cdot \hat{u}_{pc})$ and assume $\Delta\alpha$ to be $\Delta\alpha'$ or $2\pi - \Delta\alpha'$ according to whether landing site passage occurs after or prior to $-\hat{u}_{pc}$ on the DOI ellipse, that is, if $(\hat{u}_{11s} \times \hat{u}_{pc}) \cdot \text{RXV}$ of hyperbola > 0 , $\Delta\alpha = 2\pi - \Delta\alpha'$. Form $DA' = \Delta\alpha = DRI$ and then use DA equal to DA' or $2\pi + DA'$ according to whether DA' is positive or negative. Increment T_{LLS} by the conic time required to travel DA radians from the landing site back along the DOI ellipse. Recompute a selenocentric unit position vector to the landing site with this updated time, and redetermine DA .

A selenographic state vector over the lunar landing site can be constructed in polar form according to $\phi_2 = \phi_{LLS}$, $\lambda_2 = \lambda_{LLS}$, $\psi_2 = \psi_{LLS}$ with equations (96) through (98).

$$R2 = a_{11s} (1 - e^2) / [1 + e \cos(\Delta\omega_1)] \quad (96)$$

$$v_2 = \left\{ \mu \left(\frac{2}{R^2} - \frac{1}{a_{11s}} \right) \right\}^{1/2} \quad (97)$$

$$\gamma_2 = -\text{sign}(\Delta\omega_1) \text{ arc cos} \left(\frac{\sqrt{\mu a_{11s} (1 - e^2)}}{(R^2)(v_2)} \right) \quad (98)$$

where $e = -1 + (h_{11s} + r_{11s})/a_{11s}$. Convert this state vector to Cartesian form, then transform to selenocentric form at time T_{LLS} and integrate backward through $2\pi \cdot RVS2 + DA$ radians. The time of resultant state vector will be used as T_L , the approximate time of DOI. Calculate the associated semimajor axis, A_L , eccentricity E_L , and unit perilune position vector \hat{u}_L . Compute the radius at apolune RA2, and the radius at perilune RP2.

A state vector at the end of the LOI ellipse can be constructed in polar form by noting that no plane change occurs during the DOI maneuver. With the abbreviations $A = \frac{1}{2} (RA_{LP01} + RP_{LP01})$ and $E = -1 + \frac{1}{a} (RA_{LP01})$ equations (99) through (101) can be evaluated.

$$R = A(1 - E^2)/[1 + E \cos(\eta_2)] \quad (99)$$

$$v = \mu \left(\frac{2}{R} - \frac{1}{A} \right)^{1/2} \quad (100)$$

$$\gamma = \text{sign}(\eta_2) \text{ arc cos} \left[\frac{\sqrt{\mu A(1 - E^2)}}{(R)(v)} \right] \quad (101)$$

with ϕ , λ , ψ , and time equal to their values at the start of the DOI ellipse (i.e., at the end of the backward propagation). Integrate this state vector backward in time to T_{nd} to obtain $\vec{R}II$, $\vec{V}II$. Compute the associated unit perilune position vector, \hat{u}_{p1} , radius at apolune, RA1, and radius at perilune RP1.

Evaluate $\Delta w_p = \arccos(\hat{u}_{p1} \cdot \hat{u}'_{p1}) \text{ sign}[\hat{u}'_{p1} \times \hat{u}_{pL} \cdot (R11 \times V1I)]$
 and form the differences $\Delta h_a = RA2 - RA1$, $\Delta h_p = RP2 - RP1$. Form
 $\Delta T = T_L - T_{nd}$; set the independent variable $\Delta T_{1st \text{ pass}} = T_{LLS} - T_{nd}$;
 and return to the main program.

3.19 Subroutine MCOMP

3.19.1 Function.- Subroutine MCOMP computes the mass of the spacecraft after each maneuver. The maneuver may be performed with either the SPS or the DPS. After the LM is undocked from the CSM, only SPS maneuvers are performed. An error message is returned if the LM fuel has been completely expended.

3.19.2 Method.- The subroutine is called with the maneuver ΔV , a docked or undocked indicator, a DPS or SPS indicator, the present spacecraft mass, the available LM fuel, the LM fuel spent in previous maneuvers, and the specific impulses of the SPS and DPS full throttle; and returns the mass after the maneuver.

An initial test is made to see whether the LM is still docked. If it is not docked, the final mass is computed as $m_o e^{-\Delta V/g_o I_{sp}}$ of SPS. If it is still docked, a test is made to see whether the maneuver is to be performed with the DPS. If it is not to be performed with the DPS, the final mass is computed as $m_o e^{-\Delta V/g_o I_{sp}}$ of SPS. If it is to be performed with the DPS, the mass expended on the maneuver is computed as $m_o (1 - e^{-\Delta V/g_o I_{sp}})$ of DPS. If the sum of this delta mass and any previous LM mass expended is greater than the LM fuel available for a maneuver, an error message is to be displayed which will indicate that this plan of maneuvers cannot be performed. If the sum of this delta mass and any previous expended LM fuel is less than the LM fuel available for maneuvers, a test is made to see whether the LM is to be dropped after this maneuver. If it is to be dropped, the final mass is computed as $m_o - (m_{LM} - \Delta m_{total})$. If it is not to be dropped, the final mass is computed as $m_o - \Delta m$ of the maneuver.

3.20 Subroutine PPC

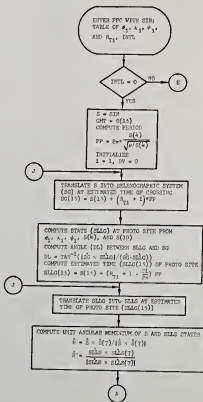
3.20.1 Function.- Subroutine PPC accepts a selenocentric LPO state vector and computes the delta V required to make a plane change which will place the spacecraft over a given photographic site R_{T1} revs later. The selenographic state vector at passage over the photographic site is output. Several plane changes can be handled sequentially.

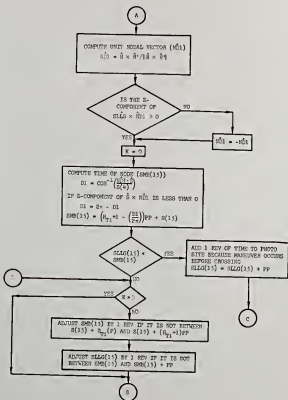
3.20.2 Nomenclature.-

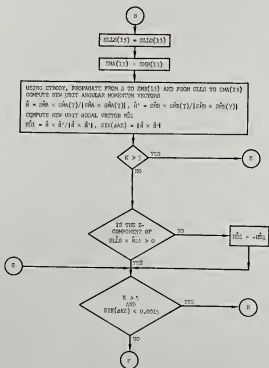
Symbol	Definition
D	time from SMA to ND1
DL	angle between ND1 and \hat{S} used for first guess computation
DL	angle between SLLG and SG used for first guess computation
DT	time from ND1 and SMB
DV	total accumulated velocity required
DVS	stored total velocity required (use for INTL \neq 0)
GMT	stored time from base time of original input state vector (used for INTL \neq 0)
\hat{H}	unit angular momentum vector associated with SMB
\hat{H}'	unit angular momentum vector associated with SMA
HSMA	negative angular momentum associated with SMA
i	counter for lunar photographic sites
INTL	initialization flag (INTL = 0 Iterate, INTL \neq 0 transform previous iterated result)
IMAX	maximum number of photographic sites
K	iteration counter
ND1	unit nodal vector of parking orbits defined by SMA and SMB
PP	period of LPO
PSMB	unit projection of SMB into plane defined by SMA
R_{T1}	number of passages over each photographic site
\hat{S}^a	current selenocentric base state vector (input state vector or previous site)
SIN	input selenocentric LPO state vector

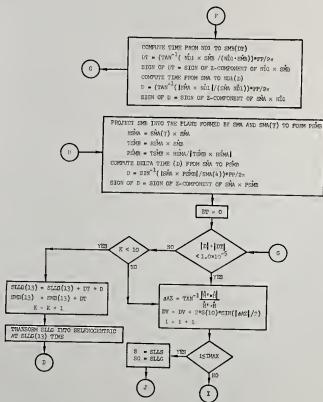
^aAll state vectors are of dimension 13 and contain X, Y, Z, r, r^2 , r^3 , X, Y, Z, V, V^2 , V^3 , and time from base time. Thus, $\hat{S}(7)$ represents the velocity vector associated with the state vector S, and so on.

$\hat{S}G$	selenographic state corresponding to \hat{S}
$\hat{S}LLG$	selenographic state vector associated with current $\lambda_1, \phi_1, \psi_1$ (This state vector is saved for INPL = 0 execution.)
$\hat{S}MA$	selenocentric state vector after the current maneuver
$\hat{S}MB$	selenocentric state vector before the current maneuver
$\hat{T}SMB$	$\hat{H}SMA \times \hat{S}MB$
ΔAZ	required plane change
λ_1	selenographic longitude of photographic sites
λ_{LLS}	selenocentric longitude associated with $\hat{S}LLS$
λ_{ND}	selenocentric longitude associated with $\hat{N}D1$
μ	gravitational constant of the moon
ϕ_1	selenographic latitude of photographic sites
ψ_1	selenographic approach azimuth of photographic sites

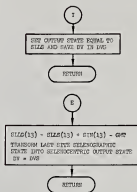








54



Page 5 of 5

Flow chart 1 - Subroutine PPC - Concluded.

TABLE I.- INDEPENDENT VARIABLES FOR
THE TRAJECTORY COMPUTERS

Variable	Reference frame	Use		
		Analytic MCC first guess	Analytic trajectory computer	Integrating trajectory computer
V_{p1}		✓		
λ_{p1}	EMP	✓		
ψ_{p1}	EMP	✓		
C_3			✓	✓
ΔT_{EPO}			✓	✓
δ			✓	✓
σ			✓	
ΔV_{MCC}			✓	✓
ΔY_{MCC}			✓	✓
$\Delta \psi_{MCC}$			✓	✓
$\Delta \psi_{LOI}$			✓	
Y_{LOI}			✓	
$\Delta t_{1st\ pass\ LLS}$			✓	
$T_{in\ lunar\ orbit}$			✓	
$\Delta \psi_{TEI}$			✓	
ΔV_{TEI}			✓	

TABLE II.- DEPENDENT VARIABLES FOR

THE TRAJECTORY COMPUTERS

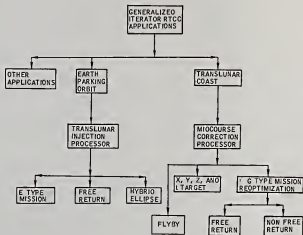
Variable	Reference frame	Use		
		Analytic MCC first guess	Analytic trajectory computer	Integrating trajectory computer
x_{mcpt}	GC or SC	✓		
y_{mcpt}	GC or SC	✓		
z_{mcpt}	GC or SC	✓		
$MASS_{TLI}$			✓	✓
$\Delta t_{TL \text{ Coast}}$			✓	✓
H_{ap}			✓	✓
H_{pc}			✓	✓
I_{pc}	EMP		✓	✓
ϕ_{pc}	EMP		✓	✓
$H_{fr-rtny}$			✓	✓
I_{fr}	EEP		✓	✓
H_{nd}			✓	✓
ϕ_{nd}	EMP		✓	✓
λ_{nd}	EMP		✓	✓
H_{LPO}			✓	
Δh_n			✓	
θ			✓	
ΔV_{LOI}			✓	
ΔV_{DOI}			✓	

TABLE II.- DEPENDENT VARIABLES FOR
THE TRAJECTORY COMPUTERS - Concluded

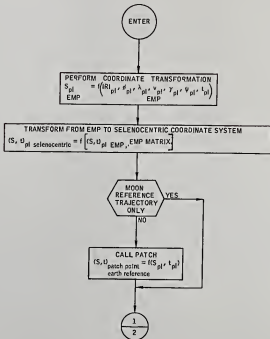
Variable	Reference frame	Use		
		Analytic MCC first guess	Analytic trajectory computer	Integrating trajectory computer
ϕ_{LLS}	SG		✓	
λ_{LLS}	SG		✓	
ψ_{LLS}	SG		✓	
MASS _{TEI}			✓	
$\Delta t_{TE \text{ Coast}}$			✓	

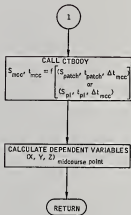
TABLE III.- BASIC MODULES USED IN TRAJECTORY COMPUTERS

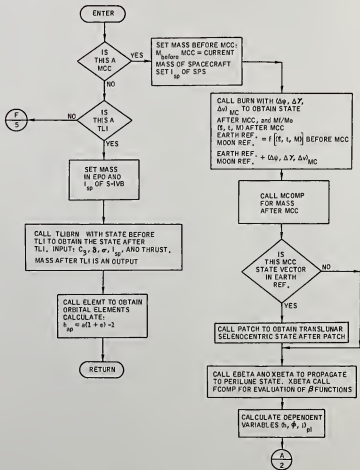
MCC first guess trajectory computer	TLI/MCC analytic trajectory computer	TLI/MCC integrated trajectory
EPHM (ephemeris)	DGAMMA	Integrator
RVIO (Cartesian to spherical, etc.)	XBETA (BETA series summation)	Forcing function
PATCH (both ways)	BURN-impulsive	Runge Kutta
EBETA	PATCH (both ways)	Predictor-corrector
RBETA	EBETA	Editor
XBETA (BETA series summation)	XBETA (BETA series summation)	EPHM
EPHM	RBETA	BETA series summation
CTBODY (BETA series summation)	EPHM (ephemeris)	RTASC
LIBRAT	CTBODY	TLIBRN
FCOMP	LIBRAT	
	EPHM	
	TLIBRN (calibrated)	LIBRAT
	ELEMT (orbital)	ELEMT (orbital)
	CTBODY (BETA series summation)	RVIO (Cartesian to spherical, etc.)
	EBETA	
	RTASC	
	RVIO (Cartesian to spherical, etc.)	
	FCOMP	

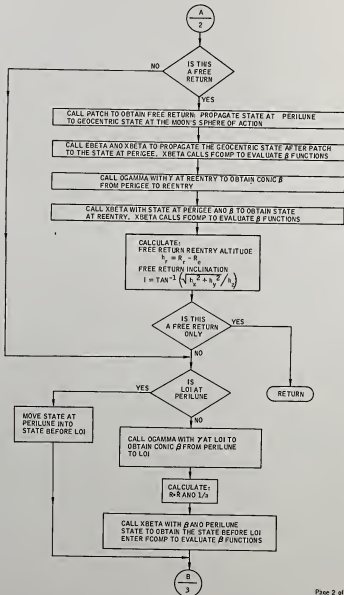


60

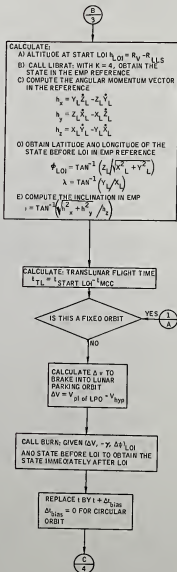






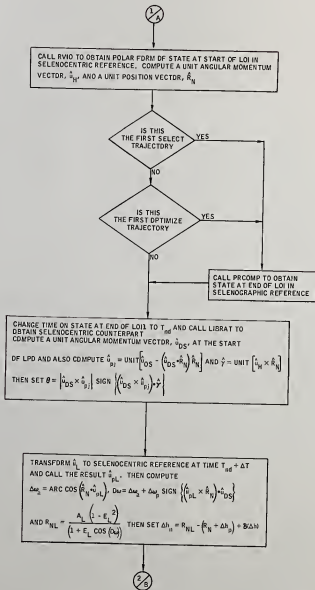


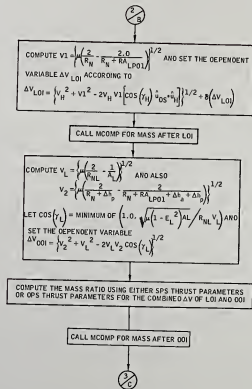
64

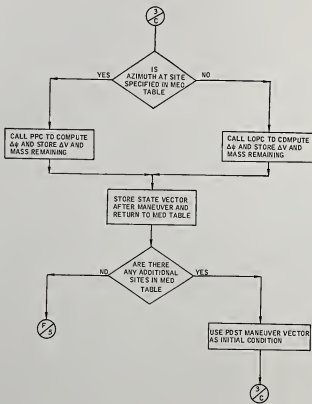


Page 3 of 9

Flow chart 4. - Functional flow of analytical trajectory computer for conic mission - Continued.







$$\frac{C}{4}$$

- A) LOI MASS RATIO: WHERE M_1 IS MASS BEFORE MANEUVER, WHERE M_2 IS MASS AFTER
 $M_1/M_2 = \exp\left[-(0.1V_e/R_{e1})\right]$
 B) MASS OF SC BEFORE LM SEPARATION
 $M_{SC} = M_{\text{before LOI}} (m1/m2)$
 C) ALTITUDE AT START OF LPD: $h_{L0} = R_0 - R_{LLS}$

- A) SCALE TO ACTUAL VELOCITY AND POSITION VECTORS SPECIFIED PARKING ORBIT VALUES STATE AT START OF LPD IS OBTAINED BY SCALING STATE AFTER LOI
 B) TIME OF SIMULATED LM LUNAR LANDING:
 $T_{LL} = T_1$ IMMEDIATELY AFTER LOI + ΔT_1 FIRST PASS OVER LLS

CALL CTBODY (REGRESSED) WITH STATE AT START OF LPD AND TIME OF LUNAR LANDING TO OBTAIN CSM STATE AT LANDING. ENTER FCOMP TO EVALUATE β FUNCTIONS.

CALL LIBRAT, WITH $K = 3$, AND USE THE PNL MATRIX TO ROTATE THE CSM POSITION VECTOR AT FIRST PASS OVER THE LUNAR LANDING SITE FROM SELENOCENTRIC INTO THE SELENOGRAPHIC, R_s . THE SELENOGRAPHIC LATITUDE AND LONGITUDE OF THE CSM AT LM LANDING TIME:

$$\phi_s = \tan^{-1}\left(\frac{Z_s}{\sqrt{X_s^2 + Y_s^2}}\right) \quad L = \frac{R_s}{|R_s|}$$

$$\lambda_s = \tan^{-1}\left(\frac{Y_s}{X_s}\right)$$

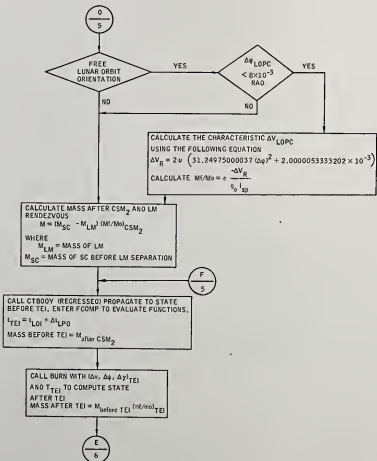
CALL RVIO WITH R_s AND V_s TO OBTAIN ϕ_s

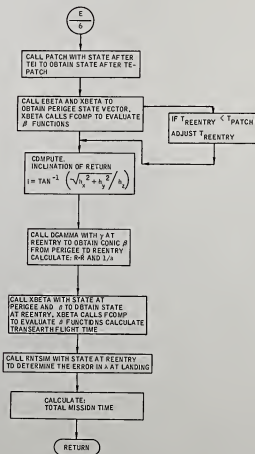
CALL ELEMNT COMPUTE PERIOD

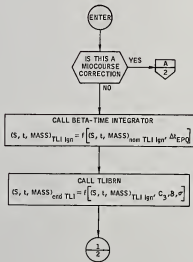
$$P = \frac{2\pi}{\left(\frac{\lambda}{R} + \mu\right)}$$

CALL LOPC COMPUTE STATE VECTOR AFTER LOPC, AND OBTAIN $\Delta\psi$ OF LOPC

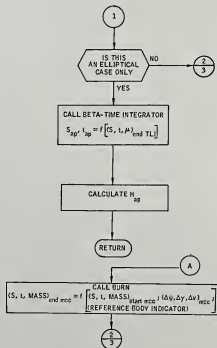
$$\frac{O}{5}$$





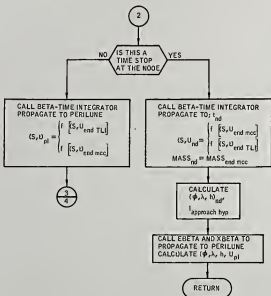


72

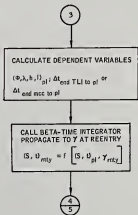


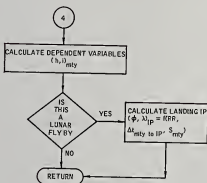
Page 2 of 5

Flow chart 5. - Integrating trajectory computer - Continued.



Change 1, August 28, 1966





REFERENCES

1. Moore, William E.: AS-503/504 Requirements for the RTCC: The Generalized Iterator. MSC IN 66-FM-131, Nov. 4, 1966.
2. Scheffman, David S.; Yencharis, Jerome D.; Zeiler, Kenneth T.: RTCC Requirements for Apollo 12 (Mission H-1): Translunar Injection Processor. MSC IN 69-FM-250, Sept. 25, 1969.
3. Holmes, Quentin A.; and Zeiler: RTCC Requirements for Apollo 14: XYZT and Free-Return Modes of the Translunar Midcourse Correction Processor. MSC IN 70-FM-15,
4. Holme's, Quentin A.; and Zeiler, Kenneth T. Zeiler: RTCC Requirements for Apollo 14: Non-Free-Return Modes of the Translunar Midcourse Correction Processor. MSC IN 70-FM-14, Feb. 16, 1970.
5. Zeiler, Kenneth T.; Holmes, Quentin A.: RTCC Requirements for Apollo 14: Flyby Modes of the Translunar Midcourse Correction Processor. MSC IN 70-FM-11, Feb. 6, 1970.
6. Moore, William E.: The Generalized Forward Iterator. MSC IN 66-FM-55, June 15, 1966.
7. Martin, D. T.; and Redwine, W. J.: Empirical Simulation of the Nominal and Alternate Mission Translunar Injection Burn Fit for an Extended Range of Independent Variables. TRW Note No. 67-FMT-506 (A-77), April 14, 1967.
8. Yencharis, J. D.: RTCC Requirements for Missions F and G: Empirical Equations for Simulating the Translunar Injection. MSC IN 68-FM-53, Feb. 26, 1968.
9. Flanagan, P. F: RTCC Requirements for Missions E, F, and G: Greenwich Hour Angle Formulation for the Predictor. MSC IN 68-FM-53, Feb. 26, 1968.

CHANGE SHEET


FOR

MSC INTERNAL NOTE 70-FM-26 (MSC-01594) DATED February 26, 1970
RTCC REQUIREMENTS FOR APOLLO 14: TRAJECTORY COMPUTERS FOR TLI
AND MCC PROCESSORS

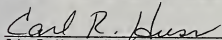
By Brady O. McCaffety, William E. Moore, and Quentin A. Holmes

Change 2

November 12, 1971



Ronald L. Berry, Chief
Planetary Mission Analysis Branch



John P. Mayer, Chief
Mission Planning and Analysis Division

NOTE: A black bar in the margin indicates the area of change.

After the attached enclosures, which are replacement pages, have been inserted and the pen-and-ink changes have been made, place this CHANGE SHEET between the cover and title page and write on the cover, "CHANGE 2 inserted."

Replace pages 43-44, 45-46, 65-66, and 67-68.

CHANGE HISTORY FOR 70-FM-26

Change no.	Date	Description
1	5/25/70	<p>Page 64: Pen-and-ink change to correct typographical error.</p> <p>Page 66: Pen-and-ink changes to eliminate redundant computation.</p> <p>Page 67: Deleted page because computation is no longer needed.</p> <p>Page 68: Pen-and-ink changes to eliminate redundant computation of mass ratios.</p> <p>Replace page 72: Added a call to MCOMP for computation of mass ratios.</p>
2	11/12/71	<p>These changes reflect the logic by which LPO-1 and LPO-2 are matched in the RTCC MCC processor.</p>

Symbol	Input (I), output (O)	Definition
T_{LLS}		estimated time of first pass over the landing site
Δt_2		increment of time required for a conic trajectory to propagate from a true anomaly of n_1 forward through DRI radians along the LOI ellipse
n_2		true anomaly on the LOI ellipse at the position of DOI
\hat{u}_{lls}		selenocentric unit position vector of the landing site
\hat{u}_{pc}		selenocentric unit position vector to the node on the approach hyperbola
DA		fraction of a rev (in radians) traveled in the DOI ellipse
$\left\{ \begin{array}{l} \phi_2, \lambda_2, R_2 \\ V_2, \gamma_2, \psi_2 \end{array} \right\}$		polar components of a selenographic state vector over the lunar landing site at time T_{LLS}
T_L		approximate time of DOI obtained by backward integration
A_L	0	semimajor axis of the DOI ellipse at time T_L
E_L	0	eccentricity of the DOI ellipse at time T_L
\hat{u}_{pl}	0	unit selenocentric perilune position vector of the LOI ellipse at time T_L
RA_2		apolune radius associated with the DOI ellipse at time T_L

Change 2, November 12, 1971

Symbol	Input (I), output (O)	Definition
RP2		Perilune radius associated with the DOI ellipse at time T_L
$\left\{ \begin{array}{l} \phi, \lambda, R \\ V, \gamma, \psi \end{array} \right\}$		polar components of a selenographic state at the start of DOI
\vec{R}_{II}		position vector on the LOI ellipse obtained by backward integration to time T_{nd}
\vec{V}_{II}		velocity vector on the LOI ellipse obtained by backward integration to time T_{nd}
\hat{u}_{pl}		unit perilune position vector associated with R_{II} , V_{II}
RAI		radius of apolune associated with T_{II} , V_{II}
RP1		radius of perilune associated with R_{II} , V_{II}
$\Delta\omega_p$	0	change in perilune position of the LOI ellipse because of propagation
Δh_a	0	change in apolune altitude of the LOI ellipse because of propagation
Δh_p	0	change in perilune altitude of the LOI ellipse because of propagation
ΔT	0	time increment from LOI to DOI
SGSLOI	0	selenographic state vector at the end of LOI
\hat{u}_L	0	selenographic unit position vector of perilune on the DOI ellipse at time T_L

3.18.3 Method.-- Input provides a nodal state on the approach hyperbola from the trajectory computer; the lunar landing site coordinates and the selenographic approach azimuth are taken from preflight data, while the size, orientation, and duration of the LOI and DOI ellipses are specified by MED quantities h_{alls} , h_{p11s} , η_1 , HA_{LP01} , HP_{LP01} , $RVS1$, and $RVS2$, and $\Delta\omega_1$.

Set $RA_{LP01} = HA_{LP01} + r_{11s}$, $RP_{LP01} = HP_{LP01} + r_{11s}$, and $a_{11s} = r_{11s} + (h_{alls} + h_{p11s})/2.0$; then use equation (95) to compute an estimate of the time of first pass over the lunar landing site.

$$T_{LLS} = T_{nd} + \frac{2\pi}{\sqrt{\mu}} \left\{ R1 \left[\frac{RA_{LP01} + RP_{LP01}}{2.0} \right]^{3/2} + RVS2 \left[a_{11s} \right]^{3/2} \right\} + \Delta t_2 \quad (95)$$

where Δt_2 is the time increment required for a conic to propagate from a true anomaly of η_1 on the LOI ellipse forward through DR1 radians. (The true anomaly at the end of this conic propagation is saved as η_2).

Next, form a unit position vector to the landing site, transform it to selenocentric coordinates at time T_{LLS} , and call it \hat{u}_{11s} . Let $\Delta\alpha' = \arccos(\hat{u}_{11s} \cdot \hat{u}_{pc})$ and assume $\Delta\alpha$ to be $\Delta\alpha'$ or $2\pi - \Delta\alpha'$ according to whether landing site passage occurs after or prior to $-\hat{u}_{pc}$ on the DOI ellipse, that is, if $(\hat{u}_{11s} \times \hat{u}_{pc}) \cdot \text{RXV of hyperbola} > 0$, $\Delta\alpha = 2\pi - \Delta\alpha'$. Form $DA' = \Delta\alpha = DR1$ and then use DA equal to DA' or $2\pi + DA'$ according to whether DA' is positive or negative. Increment T_{LLS} by the conic time required to travel DA radians from the landing site back along the DOI ellipse. Recompute a selenocentric unit position vector to the landing site with this updated time, and redetermine DA .

A selenographic state vector over the lunar landing site can be constructed in polar form according to $\phi_2 = \phi_{LLS}$, $\lambda_2 = \lambda_{LLS}$, $\psi_2 = \psi_{LLS}$ with equations (96) through (98).

$$R2 = a_{11s}(1 - e^2)/[1 + e \cos(\Delta\omega_1)] \quad (96)$$

Change 2, November 12, 1971

$$v_2 = \left\{ \mu \left(\frac{2}{R_2} - \frac{1}{a_{11s}} \right) \right\}^{1/2} \quad (97)$$

$$\eta_2 = -\text{sign}(\Delta\omega_1) \text{ arc cos} \left(\frac{\sqrt{a_{11s}(1-e^2)}}{(R_2)(v_2)} \right) \quad (98)$$

where $e = -1 + (h_{11s} + r_{11s})/a_{11s}$. Convert this state vector to Cartesian form, then transform to selenocentric coordinates at time T_{LLS} and integrate backward in time through $2\pi \cdot RVS2 + DA$ radians. The time of the resulting state vector will be used as T_L , the approximate time of DOI. Calculate the associated semi-major axis, A_L , eccentricity E_L , and radius R_L . Compute the radius at apolune, RA_2 , the radius at perilune, RP_2 , flight-path angle γ_L , and velocity V_L .

A state vector at the start of DOI can be constructed in polar form by noting that no plane change occurs during the DOI maneuver, that the true anomaly on LP01 is η_2 at the impulsive position of DOI and that the altitude on LP01 and LPO2 differ by $\delta(\Delta h)$. We have

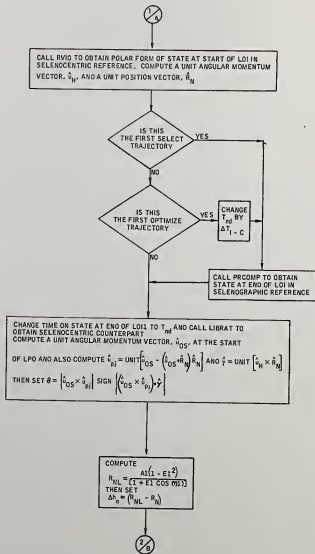
$$E = \{RA_{LP01} - [R_L - \delta(\Delta h)]\} / \{RA_{LP01} + [R_L - \delta(\Delta h)] \cos(\eta_2)\} \quad (99a)$$

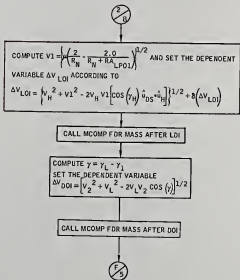
$$A = RA_{LP01} / (1 + E) \quad (99b)$$

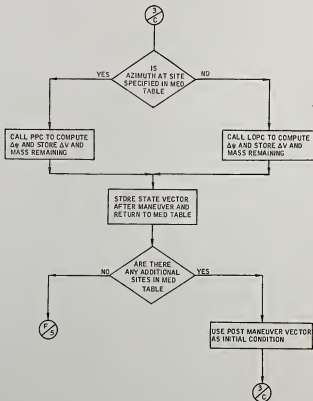
$$V_1 = \mu \left(\frac{2}{[R - \delta(\Delta h)]} - \frac{1}{A^2} \right)^{1/2} \quad (100)$$

$$\eta_1 = \text{sign}(\eta_2) \text{ arc cos} \left[\frac{\sqrt{A(1-E^2)}}{[R_L - \delta(\Delta h)] V_1} \right] \quad (101)$$

with ϕ , λ , ψ , and time equal to their values at the start of the DOI ellipse (i.e., at the end of the backward integration). Integrate this new state vector backward in time to T_{nd} to obtain R_{II} , and V_{II} . Store R_{II} , V_{II} in selenographic coordinates and compute the associated true anomaly η , unit perilune position vector \hat{u}_{p1} , radius at apolune RA_1 , radius at perilune RP_1 , semimajor axis A_1 , and eccentricity E_1 . Project \hat{u}_{pc} onto the LP01 plane and compute the angle $\Delta\eta'$ between it and R_{II} . Use subroutine DELTAT to compute the time increment, $DTCORR$, required to travel from η to $\eta + \Delta\eta'$ on the LOI ellipse. Transform \hat{u}_{11s} to selenographic coordinates update T_{LLS} by $DTCORR$ and repeat the backward integration of LPO2 and LP01 just once.







68

$$\frac{C}{4}$$

A) SCALE TO ACTUAL VELOCITY AND POSITION VECTORS SPECIFIED PARKING ORBIT VALUES STATE AT START OF LPO IS OBTAINED BY SCALING STATE AFTER LOI
 B) TIME OF SIMULATED LM LUNAR LANDING: $T_{LL} = T_{IMMEDIATELY AFTER LOI} + \Delta T$ FIRST PASS OVER LLS

CALL CTBODY (REGRESSED) WITH STATE AT START OF LPO AND TIME OF LUNAR LANDING TO OBTAIN CSM STATE AT LANDING, ENTER FCOMP TO EVALUATE β FUNCTIONS.

CALL LIBRAT, WITH $K = 5$, AND USE THE PNL MATRIX TO ROTATE THE CSM POSITION VECTOR AT FIRST PASS OVER THE LUNAR LANDING SITE FROM SELENOCENTRIC INTO THE SELENOGRAPHIC, R_s .
 THE SELENOGRAPHIC LATITUDE AND LONGITUDE OF THE CSM AT LM LANDING TIME:

$$\phi_s = \tan^{-1} \left(\frac{z}{\sqrt{x_s^2 + y_s^2}} \right) \quad L = \frac{R_s}{|R_s|}$$

$$\lambda_s = \tan^{-1} \left(\frac{y_s}{x_s} \right)$$

CALL RVIO WITH R_s AND V_s TO OBTAIN ψ_s

CALL ELEM COMPUTE PERIOD

$$p = \frac{2\pi}{\frac{v}{R} + \omega}$$

CALL LOPC COMPUTE STATE VECTOR AFTER LOPC, AND OBTAIN ϕ_p OF LOPC

$$\frac{D}{5}$$

Page 7 of 9

Flow chart 4. - Functional flow of analytical trajectory computer for cenic mission - Continued.

Change 4, December 1974



Mission Planning and Analysis Division
NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
MANNED SPACECRAFT CENTER
HOUSTON, TEXAS 77058

IN REPLY REFER TO: 70-FM54-123

15 1970

MEMORANDUM TO: See attached list

FROM : FM5/Chief, Lunar Mission Analysis Branch

SUBJECT : RTCC requirements for Apollo 14 (H-3): Trajectory computers for TLI and MCC processors - Change 1

Enclosed is Change 1 to MSC Internal Note No. 70-FM-26. This change reflects elimination of some redundant paths in the trajectory computers and some minor corrections to the original logic.

Ronald L. Berry
Ronald L. Berry

APPROVED BY:

John P. Mayer
John P. Mayer
Chief, Mission Planning
and Analysis Division

The Flight Software Branch concurs with the above recommendations.

James C. Stokes, Jr.
James C. Stokes, Jr., Chief
Flight Software Branch

Enclosure

CHANGE HISTORY FOR 70-FM-26

Change no.	Date	Description
1	5/25/70	<p>Page 64: Pen-and-ink change to correct typographical error.</p> <p>Page 66: Pen-and-ink changes to eliminate redundant computation.</p> <p>Page 67: Deleted page because computation is no longer needed.</p> <p>Page 68: Pen-and-ink changes to eliminate redundant computation of mass ratios.</p> <p>Replace page 72: Added a call to MCOMP for computation of mass ratios.</p>

CHANGE SHEET

FOR

MSC INTERNAL NOTE 70-FM-26 DATED FEBRUARY 26, 1970
RTCC REQUIREMENTS FOR APOLLO 14: TRAJECTORY COMPUTERS
FOR TLI AND MCC PROCESSORS

By Brody O. McCaffety, William E. Moore, and Quentin A. Holmes

Change 1

May 25, 1970

Ronald L. Berry
Ronald L. Berry, Chief
Lunar Mission Analysis Branch

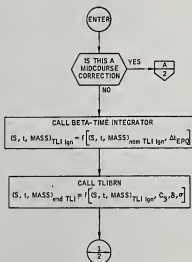
William A. Sullivan
for John P. Mayer, Chief
Mission Planning and Analysis Division

Page 1 of 3
(with enclosures)

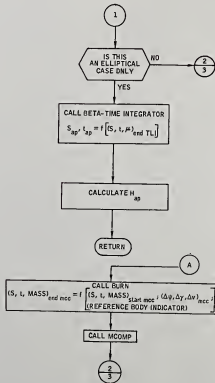
NOTE: A black bar in the margin indicates the area of change.

After the attached enclosures, which are replacement pages, have been inserted and after the following pen-and-ink changes have been made, place this CHANGE SHEET between the cover and title page and write on the cover, "CHANGE 1 inserted".

1. Page 64: reverse the direction of the arrow to $\left(\frac{1}{A}\right)$ so that it points toward $\left(\frac{1}{A}\right)$.
2. Page 66: delete the block which reads "Compute the mass ratio using either SPS thrust parameters or DPS thrust parameters for the combined ΔV of LOI and DOI".
3. Page 66: replace $\left(\frac{3}{C}\right)$ with $\left(\frac{F}{5}\right)$.
4. Page 68: delete the topmost block on page 68 which begins with "A) LOI MASS RATIO: WHERE M_0 IS MASS BEFORE...".



72



Page 2 of 5

Flow chart 5. - Integrating trajectory computer - Continued.

Change 1, May 25, 1970

NASA-MSC