NATIONAL AERONAUTICS AND SPACE ADMINISTRATION

MSC INTERNAL NOTE NO. 70-FM-19

JANUARY 30, 1970

RTCC REQUIREMENTS FOR APOLLO 14 (H-3) MISSION: MOON-CENTERED RETURN-TO-EARTH CONIC SUBPROCESSOR

MISSION PLANNING AND ANALYSIS DIVISION
MANNED SPACECRAFT CENTER
HOUSTON, TEXAS

Addressees:
IBM/J. Hoskins (5)
S. James (2)
FC/C. Charlesworth
FS5/J. Stokes (3)
★ J. Dungan
FM/J. P. Mayer
C. R. Huss
R. P. Parten
Branch Chiefs
FM6/R. Regelbrugge
FM5/R. Ernull

★ = Hand carried - R. S. Davis

Printed by RW - No release approved

cc:
Bellcomm/K. Martersteck
IBM Library
IBM/D. R. Procter
★ G. E. Myers
★ J. Yonkers
TRW Library (4)
TRW/F. A. Evans (7)
D. P. Johnson (3)
D. M. Gafford (2)
T. P. Garrison
F. M. Northcutt (5)
R. E. Utterback
W. R. Lee
FM6/R. L. Phelps (2)
CF/W. J. North
EG/D. C. Cheatham
R. G. Chilton
EA2/R. A. Gardiner
KA/R. F. Thompson
KM/W. B. Evans
PA/J. A. McDivitt
PD/O. E. Maynard
PD8/J. P. Loftus, Jr.
PD7/R. H. Kohrs
PD12/R. J. Ward
FA/S. A. Sjoberg
R. G. Rose
C. C. Critzos
FC/E. F. Kranz
M. P. Frank
J. S. Llewellyn
C. F. Deiterich
T. E. Weichel
FL/J. B. Hammack (2)
★ FS5/E. G. Clayton
★ M. D. Dixon
FM15/Report Control Files (25) 28
★ FM5/J. D. Yencharis
★ R. S. Davis (5)

Dist 2/10/70 2:15 gh
2/10/70 2:25 m.p.

FM5:RSDavis:rw

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

IN REPLY REFER TO: 70-FM54-33

Feb. 2, 1970

MEMORANDUM TO: See attached list

FROM : FM5/Chief, Lunar Mission Analysis Branch

SUBJECT : RTCC requirements for Apollo 14 (H-3) mission: Return-to-Earth Abort Processor

References:

1. MSC Internal Note 70-FM-19, "RTCC requirements for Apollo 14 (H-3) mission: Moon-centered Return-to-Earth Conic Subprocessor," by F. M. Northcutt, TRW Systems, dated January 30, 1970.

2. MSC Internal Note 68-FM-27, "RTCC requirements for Mission G: Return-to-Earth Abort Conic Subprocessor for moon-centered aborts," by D. M. Gafford, F. M. Northcutt, and J. D. Monroe, TRW Systems, dated February 1, 1968.

3. MSC Internal Note 70-FM-23, "RTCC requirements for Apollo 14 (H-3) mission: Earth-centered Return-to-Earth Conic Subprocessor," by D. R. Davis and T. P. Garrison, TRW Systems.

4. To be published MSC Internal Note, "RTCC requirements for Apollo 14 (H-3) mission: Return-to-earth abort processor supervisory and precision computation logic," by R. S. Davis.

5. MSC Internal Note 68-FM-190, "RTCC requirements: Revised Target Lines and Reentry Range Functions for the Lunar Landing Program - Return-to-Earth Abort Processor," by James W. Tolin, Jr., dated August 5, 1968.

6. Flight Software Branch RTCC Transmittal No. 12905, "Return-to-Earth reentry functions for the G and subsequent mission systems," by James C. Stokes, Jr., dated February 24, 1969.

7. MSC Memorandum No. 69-FM23-36, "RTCC requirements: Reentry coefficients for the RTCC return-to-earth abort processor for Apollo Mission G," by Floyd V. Bennett, dated February 13, 1969.

The enclosed MSC Internal Note 70-FM-19 (Reference 1) presents the detailed program logic for the RTCC Return-to-Earth (RTE) Abort Processor. Reference 1 supersedes MSC Internal Note 68-FM-27 (Reference 2).

The complete specification of RTE for H-3 is presented in three documents, one of which is the enclosed internal note. The two remaining internal notes are References 3 and 4.

The enclosed internal note performs three functions: presents logic that is in the current RTCC program or has been previously specified for H-3, presents corrections to logic presented in earlier documents, and presents new logic. The majority of the logic in the internal note has been specified or agreed upon by either the original internal note; change sheets to the original internal note; conversations among IMAB, FSB, TRW, and IBM; or RTCC transmittals. Included in this category of logic are the overlapped conic trajectory formulation, the new fuel-critical unspecified area optimization logic, the specified inclination option of the alternate target point (ATP) mode, and the state vector offset technique. The second type of logic presented by the internal note is the correction of typographical and logical errors in the original documents. The location of these corrections is denoted by a vertical bar in the margin adjacent to the altered logic. Routines which contain such corrections are MCUA, INRFV, PSTATE, KEPLER, and TFPCR. The last function is the presentation of new logic. The ATP solution logic was reformulated which required a new version of routine CLL.

The reentry simulation presented by the enclosed internal note is <u>not</u> the RTCC specification for the reentry logic. It is presented here for completeness only. The information presented by the internal note does correspond to that presented by the current defining documents for the reentry target lines and reentry computations. The reentry (V-γ) target lines are defined in Reference 5 and presented in routine URBRER. The reentry logic and coefficients are defined in References 6 and 7 and presented in routine RENTRY. The following table presents the correspondence between the nomenclature in References 7 and 1.

<u>Reentry Coefficients</u>

Reference 1 Label	Reference 7 Label	Reference 1 Label	Reference 7 Label
$b_1 - b_6$	$b_o - b_5$	$jj_{21} - jj_{26}$	$ll_o - ll_5$
cc	cc_o	$qq_{11} - qq_{16}$	$q_o - q_5$
$ee_1 - ee_3$	$ee_o - ee_2$	$qq_{21} - qq_{26}$	$r_o - r_5$
$ff_1 - ff_3$	$ff_o - ff_2$	pp_{11}, pp_{12}	pp_o, pp_1
$jj_{11} - jj_{16}$	$jj_o - jj_5$	pp_{21}, pp_{22}	mm_o, mm_1

The logic specified by the enclosed Reference 1, excluding the reentry simulation noted above, represents Apollo 14 (H-3) program requirement for the Return-to-Earth Abort Processor.

J. D. Yencharis
for Ronald L. Berry

APPROVED BY:

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

The Flight Software Branch concurs with the above recommendations.

L. J. Dungan for
James C. Stokes, Jr., Chief
Flight Software Branch

Enclosure

MSC INTERNAL NOTE NO. 70-FM-19

PROJECT APOLLO

RTCC REQUIREMENTS FOR APOLLO 14 (H-3) MISSION: MOON-CENTERED RETURN-TO-EARTH CONIC SUBPROCESSOR

By Frances M. Northcutt
Analytic Mechanics Section
TRW Systems Group

January 30, 1970

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

MSC Task Monitor
R. S. Davis

Approved: J. D. Yencharis
for R. L. Berry, Chief
Lunar Mission Analysis Branch

Approved: [signature]
for John P. Mayer, Chief
Mission Planning and Analysis Division

CONTENTS

RTCC REQUIREMENTS FOR APOLLO 14 (H-3) MISSION:

MOON-CENTERED RETURN-TO-EARTH CONIC SUBPROCESSOR

By Frances M. Northcutt
TRW Systems Group

1. SUMMARY

This note describes the program logic for the analytic conic subprocessor to be used in the Real Time Computer Complex to calculate return-to-earth maneuvers within the moon's sphere of action for the Apollo 14 (H-3) mission. This logic has been developed by TRW Systems with MSC direction under Task MSC/TRW A-60.1 of the Mission Trajectory Control Program, Contract NAS 9-8166. The program is based on the overlapped conics model and calculates single impulse return-to-earth maneuvers. It is designed primarily to provide first-guess solutions for aborts within the moon's sphere of action and nominal transearth injection maneuvers. In addition, it can be used to evaluate transearth midcourse requirements inside the moon's sphere of action and lunar flyby trajectories for alternate missions.

2. INTRODUCTION

This note is one of a set of reports which describes the RTCC return-to-earth requirements for Apollo 14 (H-3) mission. The requirements for the earth - centered return-to-earth conic subprocessor are contained in Reference 1. The requirements for the return-to-earth processor supervisory and precision computation logic are contained in Reference 2. This note describes the requirements for the moon - centered return-to-earth conic subprocessor.

The logic computes three types of solutions within the moon's sphere of action:

a. Tradeoff display data are used to demonstrate the relationship between ΔV and maneuver time for various times of landing.

b. Discrete solutions at a specified maneuver time serve as target conditions and first guesses for the impulsive precision trajectories displayed in the Abort Scan Table.

c. Search solutions use maneuver time as an additional degree of freedom to determine the analytic minimum ΔV return trajectory. The minimum solutions are then used as target conditions and first guesses for the impulsive precision solutions displayed in the Abort Scan Table.

For each of these types of solutions the processor provides the option to return to a specified landing area. The landing area may be defined as a primary target point (PTP) or an alternate target point (ATP). The PTP is a particular earth latitude and longitude. The ATP is made up of line segments connecting specific points on the surface of the earth. The discrete and search modes can also be used to return to an unspecified area (UA) in which the return trajectory is not constrained by geographical considerations.

All of these solutions minimize propellant consumption required for the maneuver and all trajectories are subject to the following constraints:

a. Reentry velocity less than some maximum value

b. Reentry altitude equal to a given value

c. Reentry flight-path angle a function of reentry speed

d. Downrange* and crossrange distance from reentry to landing a function of reentry velocity, reentry azimuth, and reentry inclination

*Downrange distance may be fixed by input.

e. Reentry inclination* less than a given maximum

f. Time of landing between some given maximum and minimum values

g. Time of the maneuver between a given maximum and minimum time

h. Pericynthion altitude greater than or equal to some given value

i. Change in velocity required for the maneuver less than some given maximum

j. Reentry must be posigrade

A brief description of the model and each submode of operation is included in the text. The appendixes contain detailed flow diagrams and descriptions of the routines used in the program. Appendix A contains the six main control routines. Appendix B contains the routines which construct the overlapped conic trajectories. The input and output routines are in Appendix C, and the utility routines are in Appendix D.

This logic basically has the same capabilities as the return-to-earth logic used in prior missions. However, there are four major changes which significantly improve the quality of the analytic solutions. These changes are

a. The overlapped conics model described in Reference 3 has been incorporated into the program. Previously, the patched conics model described in Reference 4 was used. The overlapped conics model produces analytic trajectories which better approximate three-body motion and, therefore, improve the optimization and require fewer iterations to converge in the precision mode.

b. The unspecified area logic contained in routine MCUA of Appendix B has been rewritten to better minimize ΔV as a function of return inclination and landing time. This change is completely described in Reference 5.

c. The offset vector method of Reference 6 has been included to further improve the UA solutions and make the logic more capable of defining the small ΔV maneuvers required for transearth midcourse.

*Reentry inclination may be input as a fixed value when returning to an ATP.

d. Return inclination can be specified by the user for returns to an ATP. This provides the flight controller with better landing site control to avoid bad weather in the landing area and offers him a manual iteration capability for return-to-earth solutions.

The above changes have all been previously specified for the Apollo 14 (H-3) mission. In addition to these changes, this note contains some logic changes which have not been previously specified. The area of these changes is denoted in the flow charts of the appendices by a black bar in the margin. The most significant of these changes is in routine CLL. This logic has been rewritten to better minimize ΔV required for return to an alternate target point. In addition, there are minor changes in routines MCUA, KEPLER, TFPCR, PSTATE, and INRFV. These changes were made to correct errors found in the previously specified logic.

3. NOMENCLATURE

The notation used in this program is essentially indentical to that used in the Analytic Return-to-Earth Program - Moon Reference (Reference 7). Because many parameters are involved, a general notation is given indicating the type of parameters, and the subscripts are used to identify the points at which the parameters are specified.

Parameters

A	azimuth angle
a	semimajor axis
D	day of month
e	eccentricity
F	thrust
h	altitude from the surface
i	inclination
I_{sp}	specific impulse of the engine
M	month of year
r	radial distance, used only for reentry radius (r_r), earth radius (r_e), and moon radius (r_m)
T	time measured from perifocus (single subscript) or, more commonly, the time difference between two points (double subscript)
t	time measured from the abort day (0 hour Greenwich mean time) and written with a single subscript
$\overline{u}$	geocentric velocity vector
u	magnitude of $\overline{u}$
$\overline{v}$	selenocentric velocity vector
v	magnitude of $\overline{v}$
W	weight
$\dot{W}$	weight flow rate

$\bar{x}$	geocentric position vector
x	magnitude of $\bar{x}$
Y	year
$\bar{y}$	selenocentric position vector
y	magnitude of $\bar{y}$
α	right ascension; α_{SIDO} is used for siderial time
β	flight-path angle measured from the vertical
δ	declination
ϵ	tolerance value
η	angle measured in the trajectory plane
θ	included angle between two position vectors
λ	longitude
μ	latitude; μ_e, μ_m are used as gravitational constants
Ψ	included angle between a position vector and a final velocity vector
Ω	longitude of the ascending node
ω	argument of perigee; ω_e is used as the rotational rate of the earth

Superscripts

-	denotes a vector
~	denotes a quantity referenced to the geographic reference frame
v	denotes a unit vector

Subscripts

a	postmaneuver
e	earth parameter
h	selenocentric conic element

m	moon parameter
r	earth reentry
x	usually refers to the postpericynthion pseudostate; may also refer to exit from the pseudostate transformation sphere
z	earth landing site
0	premaneuver

Geocentric conic elements are usually written without subscripts (e.g., a, e, Ω). The exception is the inclination i_r.

Abbreviations

AST	Abort Scan Table
ATP	alternate target point
MSA	moon's sphere of action
PTP	primary target point
PTS	pseudostate transformation sphere
RTCC	Real Time Computer Complex
UA	unspecified area

4. PROGRAM DESCRIPTION

4.1 Analytic Model

The analytic program is based on the overlapped conics model, which results from application of the three-body pseudostate theory described in Reference 3. In this model, earth-moon space is divided into two regions by a pseudostate transformation sphere (PTS) which is centered on and moves with the moon. The radius of the PTS is 24 earth radii, and it is used only for the purpose of computing analytic trajectories. This definition of the PTS size has no effect on the size of the moon's sphere of action (MSA), which has a radius of approximately 10 earth radii and is used as the criterion for switching central bodies during numerical integration.

The description which follows is taken directly from Reference 3:

> "In the overlapped conic technique ... lunar perturbations of the geocentric conic are ignored outside the PTS .... In contrast to the patched conic method, the geocentric conic is extended into the sphere, rather than being terminated at the PTS surface. Instead of defining a region of exclusive lunar influence, the PTS defines a region wherein geocentric conic states are regarded as pseudostates that are related to their real counterparts by a reversible transformation algorithm. ..
>
> The transformation of a geocentric prepericynthion pseudostate to its real selenocentric counterpart is a three-step process requiring (1) conversion of the pseudostate from geocentric to selenocentric coordinates, (2) linear propagation of the converted state backward to the PTS surface, and (3) selenocentric conic propagation forward to the original pseudostate time. Transformation of a geocentric postpericynthion pseudostate is similar, except that the direction of propagation is reversed in steps (2) and (3).

For a return-to-earth trajectory which involves both prepericynthion and postpericynthion motion, two geocentric conics are used to simulate the actual trajectory: one is used to describe prepericynthion motion, and another is used to describe postpericynthion motion. The elements of both geocentric conics are fixed by the definition of an initial state vector and remain constant as long as no non-gravitational forces such as engine thrust are applied to the spacecraft.

4.2 Submodes

4.2.1 Primary target point. - The PTP logic is designed to return the spacecraft to a particular landing site within the constraints imposed upon the problem. If the site is not accessible within the constraints, the maneuver which returns as near as possible to the site can be computed.

With the requirement that the return trajectory must land at a particular site, there is only one possible solution each 24 hours. No tradeoff is made between ΔV required and miss distance from the site. The latitude width of the landing site accessibility contour is directly dependent on the downrange distance from reentry to landing; consequently, when the inertial downrange distance is less than about 1800 nautical miles, the contour has become so narrow that a very small change in latitude at landing can result in a large change in ΔV. This makes the alternate target point logic more desirable for short downrange distances.

4.2.2 Alternate target point. - The ATP logic is designed to return the spacecraft to an alternate target line within the constraints imposed upon the problem. An alternate target line is composed of straight line segments on the surface of the earth. Each line segment is defined by the latitude and longitude of its end points.

This logic minimizes ΔV required for the maneuver as a function of return inclination for each possible day of landing. The ΔV minimization makes this mode particularly useful when the downrange distance from reentry to landing is short. An option is available for the user to specify the return inclination he desires. This removes the optimization capability but provides the user landing site control to avoid bad weather and return plane control to provide better visual monitoring conditions. The inclination is input as a signed parameter with negative values indicating that the azimuth at reentry is greater than the reentry azimuth associated with the minimum possible return inclination. Positive values indicate that the azimuth at reentry is less than that associated with the minimum possible return inclination.

4.2.3 Unspecified area. - The UA logic is designed to minimize the ΔV for the maneuver within the constraints imposed upon the problem. The whole of the landing site accessibility contour is considered in this problem. This logic is designed to calculate transearth midcourse inside the MSA and fuel critical aborts. In order to better optimize the precision solution, the offset vector method described in Reference 6 is used with this submode.

4.3 Inputs

4.3.1 Basic input requirements. - The following quantities are required input to the subprocessor for all problems:

SMODE1
- = 12, PTP discrete option
- = 14, ATP discrete option
- = 16, UA discrete option
- = 22, PTP tradeoff display
- = 24, ATP tradeoff display
- = 32, PTP search option
- = 34, ATP search option
- = 36, UA search option

STAVEC	an array of premaneuver states spaced along the preabort trajectory
JUMPI	the number of states in STAVEC
EPOCHI	the base time from which input and output times are measured
BY0I	reference year
BM0I	reference month of year
DAY0I	reference day
TZMAXI	maximum allowable landing time measured from EPOCHI; this is used in tradeoff displays and the UA submode
or	
TZMINI	approximate landing time measured from EPOCHI for PTP and ATP discrete and search cases; also used as the minimum allowable landing time for tradeoff displays and all UA submodes

4.3.2 General program options. - In order to retain program flexibility while keeping the input requirements as simple as possible, most constraint boundaries and program options are set internally to their usual values. The following quantities are preset and may be overridden by the user:

IRMAXI = 40 deg	maximum allowable return inclination
URMAXI = 36,323 fps	maximum allowable reentry speed
RRBI = 0 n mi	relative range override
CIRI = 0	= 0, postmaneuver direction of motion is selected internal to the program = 1, only noncircumlunar motion is allowed = 2, only circumlunar motion is allowed
HMINI = 50 n mi	minimum altitude allowed at closest approach to the moon
EPI = 2	= 0, guided reentry to the steep target line = 1, manual reentry to the shallow target line = 2, manual reentry to the steep target line
L2DI = 0.29332	lift-to-drag ratio
DVMAXI = 10,000 fps	maximum allowable ΔV

4.3.3 Landing area identification. - Landing area information is required for the PTP and ATP submodes. For the PTP submodes (SMODEI = 12, 22, and 32) the following inputs are required:

LAMZI	longitude of the desired site
MUZI	latitude of the desired site

In addition, the miss distance option may be requested by inputting

MDMAXI	the maximum allowable miss distance from the target site

For the ATP submodes (SMODEI = 14, 24, and 34) the desired alternate target line is defined by

LINE	the latitude, longitude coordinates of the target line; the order of input is latitude of point one, longitude of point one, latitude of point two, longitude of point two, etc; as many as five points may be used

An ATP option is available to control the return plane

IRKI	the desired return inclination is |IRKI|; when IRKI < 0 the azimuth at reentry is greater than the azimuth at reentry for the minimum possible return inclination; when IRKI > 0, the reentry azimuth is less than that for the minimum possible return inclination

4.4 Outputs

4.4.1 Tradeoff displays. - The following parameters are output to the supervisory logic for the ATP and PTP tradeoff displays:

ΔV	change in velocity required for the maneuver
t_0	time of the maneuver measured from zero hour Greenwich mean time of the reference day
t_{0get}	time of the maneuver measured from epoch (EPOCH) of the reference day
q_m	= 0, noncircumlunar motion = 1, circumlunar motion

t_{zget}	time of landing measured from epoch time (EPOCH) of the reference day

4.4.2 Discrete and search cases. - The following parameters are putput to the supervisory logic for ATP, PTP, and UA discrete and search cases:

λ_{z1}	longitude of impact
μ_{z1}	latitude of impact
$\overline{v}_a$	postmaneuver velocity vector
ΔV	change in velocity required for the maneuver
t_z	time of landing measured from zero hour Greenwich mean time of the reference day
T_{ar}	time from maneuver to reentry
T_{ax}	time from maneuver to PTS exit
$\overline{x}_x$	geocentric pseudostate position vector
$\overline{u}_x$	geocentric pseudostate velocity vector
$\overline{y}_x$	selenocentric PTS exit position vector
$\overline{v}_x$	selenocentric PTS exit velocity vector
η_{rz}	transfer angle from reentry to landing
β_r	reentry flight-path angle
t_{zget}	time of landing measured from epoch time of the reference day
λ_r	longitude of reentry
μ_r	latitude of reentry
i_r	return inclination
u_r	reentry velocity
MD	miss distance
q_a	apogee passage flag
q_d	pericynthion passage flag

η_{xr}	transfer angle from the geocentric postpericynthion pseudostate to reentry
ψ	angle from the premaneuver position vector to the PTS exit velocity vector
r_p	radius of pericynthion
t_{0get}	ignition time measured from epoch time of the reference day

For search cases the following are also output:

$\overline{r}_0$	premaneuver position vector of the optimum solution
$\dot{\overline{r}}_0$	premaneuver velocity vector of the optimum solution

APPENDIX A

ANALYTIC CONTROL ROUTINES

CONTENTS

A-1 MASTER

Purpose

This routine is the main control routine for the Moon-Centered Return-to-Earth Conic Subprocessor.

Input

STAVEC	a table of premaneuver states spaced along the premaneuver trajectory

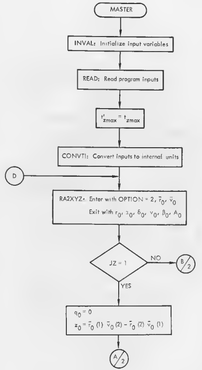
Output

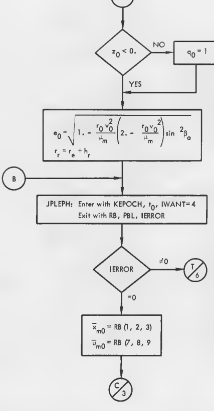
This routine outputs the desired solutions as first guesses and target conditions for the impulsive precision solutions displayed in the AST. If tradeoff display data are desired, the solutions are output to the supervisory logic for plotting.

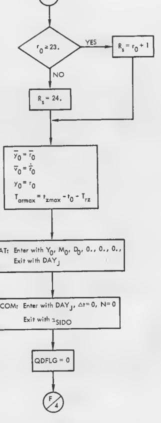
Discussion

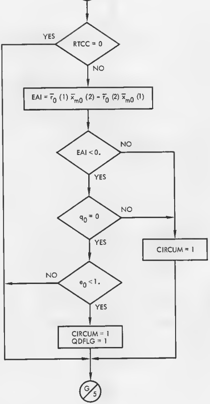
This routine begins by initializing the program variables to their customary values. Then the program inputs are read and converted to internal units. If the postmaneuver direction of motion is to be determined internally, the premaneuver position vector is checked to see whether the state is on the transearth side of the earth - moon line. If so, only noncircumlunar motion is allowed. If not, the eccentricity of the trajectory is computed to see whether the trajectory is elliptic. If so, only retrograde motion with respect to the moon is allowed. If not, both return directions of motion are allowed.

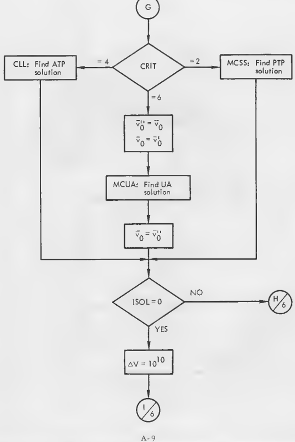
For discrete cases, control is returned to the supervisory logic as soon as a solution for the case is computed and stored. For the search and tradeoff modes, new state vectors are taken from a table of state vectors and the appropriate solutions found for each. When the table of state vectors is exhausted, control is returned to the supervisory logic.

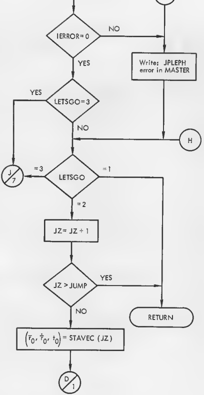

MASTER
INVAL: Initialize input variables
READ: Read program inputs
$t'_{zmax} = t_{zmax}$
CONVTI: Convert inputs to internal units
D
RA2XYZ: Enter with OPTION = 2, $\bar{r}_0$, $\bar{v}_0$
Exit with r_0, α_0, δ_0, v_0, β_0, A_0
JZ = 1
NO
B/2
YES
$q_0 = 0$
$z_0 = \bar{r}_0(1)\ \bar{v}_0(2) - \bar{r}_0(2)\ \bar{v}_0(1)$
A/2

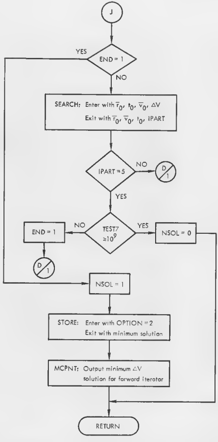

A
$z_0 < 0.$
NO
$q_0 = 1$
YES
$e_0 = \sqrt{1. - \frac{r_0 v_0^2}{\mu_m}\left(2. - \frac{r_0 v_0^2}{\mu_m}\right)\sin^2\beta_o}$
$r_r = r_e + h_r$
B
JPLEPH: Enter with KEPOCH, t_0, IWANT=4
Exit with RB, PBL, IERROR
IERROR
≠0
T/6
=0
$\bar{x}_{m0}$ = RB (1, 2, 3)
$\bar{u}_{m0}$ = RB (7, 8, 9
C/3


C
$r_0 \geq 23.$
YES
$R_s = r_0 + 1$
NO
$R_s = 24.$
$\bar{y}_0 = \bar{r}_0$
$\bar{v}_0 = \dot{\bar{r}}_0$
$y_0 = r_0$
$T_{armax} = t_{zmax} - t_0 - T_{rz}$
TJUDAT: Enter with Y_0, M_0, D_0, 0., 0., 0.,
Exit with DAY_J
SIDCOM: Enter with DAY_J, $\Delta t = 0$, N= 0
Exit with α_{SIDO}
QDFLG = 0
F/4


F
RTCC = 0
YES
NO
EAI = r̄0 (1) x̄m0 (2) - r̄0 (2) x̄m0 (1)
EAI < 0.
NO
YES
q0 = 0
NO
YES
CIRCUM = 1
e0 < 1.
NO
YES
CIRCUM = 1
QDFLG = 1
G/5


G
CRIT
= 4
= 2
= 6
CLL: Find ATP solution
MCSS: Find PTP solution
$\bar{v}_0'' = \bar{v}_0$
$\bar{v}_0 = \bar{v}_0'$
MCUA: Find UA solution
$\bar{v}_0 = \bar{v}_0''$
ISOL = 0
NO
H/6
YES
$\Delta V = 10^{10}$
I/6


I
T
IERROR = 0
NO
YES
Write: JPLEPH error in MASTER
YES
LETSGO = 3
NO
H
J/7
= 3
LETSGO
= 1
= 2
JZ = JZ + 1
JZ > JUMP
YES
NO
RETURN
$(\bar{r}_0, \dot{\bar{r}}_0, t_0)$ = STAVEC (JZ)
D/1


J
END = 1
YES
NO
SEARCH: Enter with $\overline{r}_0$, t_0, $\overline{v}_0$, ΔV
Exit with $\overline{r}_0$, $\overline{v}_0$, t_0, IPART
IPART = 5
NO
D/1
YES
TEST7 $\geq 10^9$
NO
YES
END = 1
D/1
NSOL = 0
NSOL = 1
STORE: Enter with OPTION = 2
Exit with minimum solution
MCPNT: Output minimum ΔV
solution for forward iterator
RETURN

A-2 MCSS

Purpose

This routine contains the control logic for primary target point solutions. A primary target point solution lands at a particular longitude and latitude. If a solution cannot be found which satisfies the constraints, then a miss distance solution may be found.

Input

h_{min}	minimum altitude allowed at closest approach to the moon
CIRCUM	= 0, both postmaneuver directions of motion are considered = 1, retrograde or noncircumlunar motion as determined in MCDRIV = 2, circumlunar motion only
μ_z	latitude of the desired landing site
λ_z	longitude of the desired landing site
u_{rmax}	maximum reentry velocity allowed
i_{rmax}	maximum reentry inclination allowed
MD_{max}	maximum miss distance allowed for solutions to be used in the selection of an optimum
T_{armax}	maximum return time allowed from maneuver to reentry
t_{zmin}	minimum landing time allowed

Output

The desired solutions for the primary target point submode are output, plus

ISOL	= 0, no solution found = 1, solution found

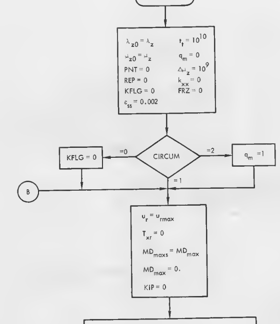
Discussion

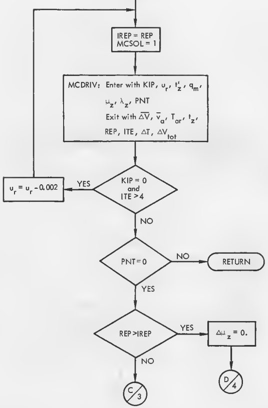
After initializing the problem, MCSSLM is called to determine the accessibility of the desired site. When control is returned to MCSS, the flag REP and the variable ΔT indicate what was determined in MCSSLM. If ΔT = 0, the landing site is probably accessible. Otherwise, the flag REP = 1 indicates that miss distance solutions were found in MCSSLM. If REP = 0, no solution is available for the given direction of motion. In the latter two instances, the problem is begun anew for the other direction of motion.

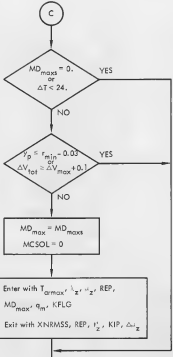
If MCSSLM has indicated that the site may be accessible, MCDRIV is called to construct a trajectory which lands at a specified time and returns so that the desired offset landing site is in the plane of the trajectory. An offset site is used to account for out-of-plane motion from reentry to landing. MCDRIV determines whether or not the impact point of the trajectory is sufficiently close to the actual PTP site. If it is not, the error in landing time is estimated and returned to MCSS. Fifteen calls to MCDRIV are allowed to find a trajectory which returns to the site.

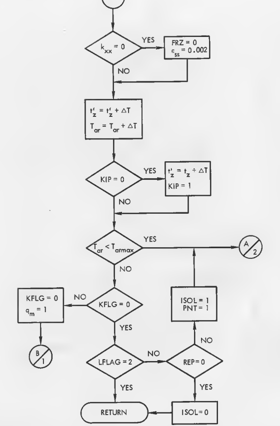
If MCDRIV produces a trajectory which returns to the site and satisfies all of the constraints, the landing time is incremented by 24 hours, and another iteration begins in MCDRIV. This procedure continues until the landing time constraint is exceeded. If required, the problem is begun anew for the other direction of motion.

If MCDRIV fails to produce a satisfactory solution, a miss distance solution may be found in MCSSLM.


MCSS
$\lambda_{z0} = \lambda_z$ $t_f = 10^{10}$
$\mu_{z0} = \mu_z$ $q_m = 0$
PNT = 0 $\Delta\mu_z = 10^9$
REP = 0 $k_{xx} = 0$
KFLG = 0   FRZ = 0
$\varepsilon_{ss} = 0.002$
CIRCUM
=0
=2
=1
KFLG = 0
$q_m = 1$
B
$v_r = v_{rmax}$
$T_{xr} = 0$
$MD_{maxs} = MD_{max}$
$MD_{max} = 0.$
KIP = 0
MCSSLM: Enter with KFLG, v_{rmax}, i_{rmax}, T_{armax}, λ_z, μ_z, REP, MD_{max}, q_m
Exit with XNRMSS, REP, t_z', KIP, $\Delta\mu_z$
D/4


A
IREP = REP
MCSOL = 1
MCDRIV: Enter with KIP, υ_r, t'_z, q_m, μ_z, λ_z, PNT
Exit with $\overline{\Delta V}$, $\bar{v}_a$, T_{ar}, t_z, REP, ITE, ΔT, ΔV_{tot}
KIP = 0 and ITE > 4
YES
$\upsilon_r = \upsilon_r - 0.002$
NO
PNT = 0
NO
RETURN
YES
REP > IREP
YES
$\Delta\mu_z = 0.$
NO
C/3
D/4


C
$MD_{maxs} = 0.$ or $\Delta T < 24.$
YES
NO
$y_p \le r_{min} - 0.03$ or $\Delta V_{tot} \ge \Delta V_{max} + 0.1$
YES
NO
$MD_{max} = MD_{maxs}$
MCSOL = 0
MCSSLM: Enter with T_{armax}, λ_z, μ_z, REP,
MD_{max}, q_m, KFLG
Exit with XNRMSS, REP, t'_z, KIP, $\Delta\mu_z$
D 4


D
$k_{xx} = 0$
YES
FRZ = 0
$\varepsilon_{ss} = 0.002$
NO
$t_z' = t_z' + \Delta T$
$T_{ar} = T_{ar} + \Delta T$
KIP = 0
YES
$t_z' = t_z + \Delta T$
KIP = 1
NO
$T_{ar} < T_{armax}$
YES
A/2
NO
KFLG = 0
NO
KFLG = 0
$q_m = 1$
B/1
YES
ISOL = 1
PNT = 1
NO
LFLAG = 2
NO
REP = 0
YES
YES
RETURN
ISOL = 0

A-3 MCSSLM

Purpose

This routine is used by MCSS to determine the accessibility of the PTP landing site. If the site cannot be reached due to constraint boundaries, a miss distance solution can be generated in this routine.

Input

q_m	= 0, if used in conjunction with QDFLG = 1, retrograde motion about the moon is specified; otherwise motion is specified to be noncircumlunar = 1, circumlunar motion
u_{rmax}	maximum reentry velocity allowed
i_{rmax}	maximum return inclination allowed
t_{zmax}	maximum time of landing allowed
λ_z	longitude of the desired site
μ_z	latitude of the desired site
t'_z	approximate time of landing associated with the site
KIP	= 0, u_r is the independent variable = 1, t'_z is the independent variable
t_{zmin}	minimum landing time allowed
MD_{max}	= 0., determine if the site lies in the accessibility contour ≠ 0., an acceptable solution could not be found; minimize the miss distance from the site with this value being the maximum allowable miss distance
MD_{maxs}	saved value of MD_{max}
LFLAG	= 1, output tradeoff display solutions = 2, output discrete or search solutions

ΔV_{max}	maximum allowable ΔV
h_{min}	minimum altitude allowed for closest approach to the moon
REP	solution flag transferred from MCSS

Output

XNRMSS	miss distance for the solution
MCSOL	= 0, no solution was found = 1, a miss distance solution was found within the constraints
t'_z	approximate landing time at the site, used in MCSS and MCDRIV to initialize the iteration on the site
ΔT	= 0., the site may be accessible; try to find hit solution in MCSS = 10^{10}, the site is not accessible; continue problem for other direction of motion as required

Discussion

There are two types of calls made to MCSSLM. In the first call, MCSS requests information on the accessibility of the PTP site. In the second type of call, MCSS and MCDRIV have failed to converge on a satisfactory solution, and a miss distance solution is requested.

The accessibility of the site is evaluated by determining the latitude bounds of the landing site accessibility contour. These bounds are not constrained by pericynthion altitude or ΔV. The bounds are defined by the extremities of the impact latitudes, μ_{min} and μ_{max}, obtained from four trajectories which land at the minimum and maximum possible times. All trajectories reenter with the maximum allowable return inclination. For a given time of landing there is a trajectory in each of the return planes. If the latitude μ_z of the PTP site is within 0.03 earth radius of these bounds, a PTP solution is likely to exist. Control is returned to MCSS with $\Delta T = 0$ and a predicted landing time at the site. This time is computed to be

$$t'_z = t'_{z1} - \Delta\lambda/\omega_e$$

where

t'_{z1} = minimum landing time

$\Delta\lambda$ = difference between the desired and actual landing longitudes

ω_e = rotational rate of the earth

If the latitude of the PTP site is not within the contour boundaries, a miss distance solution may be obtained if

$$\mu_{min} - 0.03 - MD_{maxs} \leq \mu_z \leq \mu_{max} + 0.03 + MD_{maxs}$$

where

MD_{maxs} = maximum miss distance allowed

When this inequality is satisfied, control remains in MCSSLM to compute the miss distance solutions for all allowable days of landing. If a miss distance solution is not available, control returns to MCSS with $\Delta T = 10^{10}$ and MCSS will either terminate the problem or reinitialize the problem for the other direction of motion.

When finding a miss distance solution, the logic defines the bounds of the accessibility contour at the approximate landing time of the PTP site. This definition of the bounds includes the limitations imposed by the ΔV constraint and the pericynthion altitude constraint. If these constraints are limiting the accessibility, then return inclination is the independent variable iteratively used to define the bounds. A trajectory is computed which returns to the boundary of the accessibility contour nearest to the latitude of the desired landing site. An iteration on landing time is then performed to drive the longitude of the impact point to the longitude of the site. If necessary, another iteration on inclination is used to define the boundary at the new landing time. The miss distance is computed to be

$$\Delta\mu_1 = \arccos\left(\sin\mu_z \sin\mu_{z1} + \cos\mu_z \cos\mu_{z1} \cos(\lambda_z - \lambda_{z1})\right)$$

where

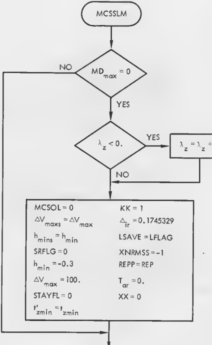
μ_z = latitude of the PTP site

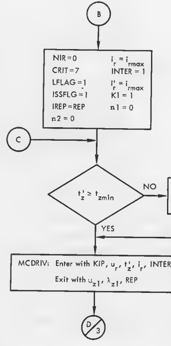
μ_{z1} = latitude of impact

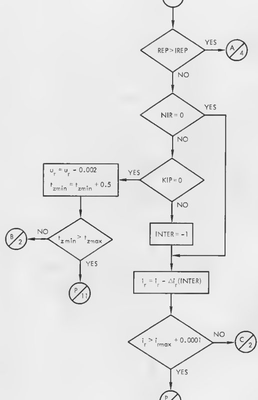
λ_z = longitude of the PTP site

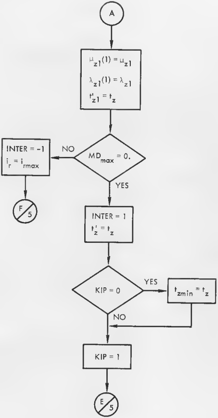
λ_{z1} = longitude of impact

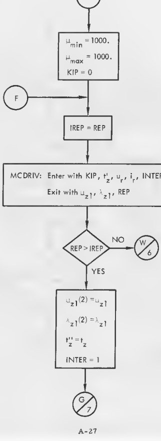
If this miss distance is less than the maximum allowed, the solution is acceptable and will be treated as a PTP solution by the control logic. If earlier computations indicated that all solutions for this direction of motion must be miss distance solutions, time will be incremented by 24 hours, and a new miss distance solution will be sought. Otherwise, control returns to MCSS.

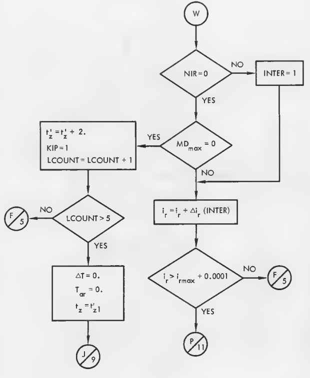

MCSSLM
NO
MD max = 0
YES
λz < 0.
YES
λz = λz + 2 π
NO
MCSOL = 0
ΔV maxs = ΔV max
h mins = h min
SRFLG = 0
h min = -0.3
ΔV max = 100.
STAYFL = 0
t'zmin = tzmin
KK = 1
Δir = 0.1745329
LSAVE = LFLAG
XNRMSS = -1
REPP = REP
T ar = 0.
XX = 0
B
2

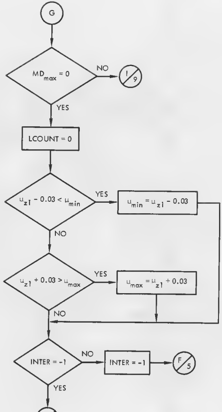

B
NIR = 0
CRIT = 7
LFLAG = 1
ISSFLG = 1
IREP = REP
n2 = 0
$i_r = i_{rmax}$
INTER = 1
$i'_r = i_{rmax}$
K1 = 1
n1 = 0
C
$t'_z \geq t_{zmin}$
NO
KIP = 0
$u_r = u_{rmax}$
YES
MCDRIV: Enter with KIP, u_r, t'_z, i_r, INTER
Exit with u_{z1}, λ_{z1}, REP
D
3

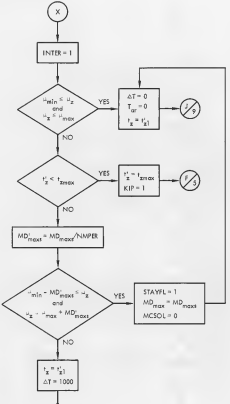

D
REP > IREP
YES
A/4
NO
NIR = 0
YES
NO
KIP = 0
YES
NO
u_r = u_r - 0.002
t_zmin = t_zmin + 0.5
t_zmin > t_zmax
NO
B/2
YES
P/11
INTER = -1
i_r = i_r - Δi_r(INTER)
i_r > i_rmax + 0.0001
NO
C/2
YES
P/11

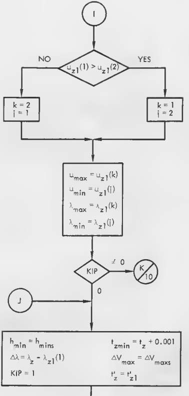

A
$\mu_{z1}(1) = \mu_{z1}$
$\lambda_{z1}(1) = \lambda_{z1}$
$t'_{z1} = t_z$
$MD_{max} = 0.$
NO
INTER = -1
$i_r = i_{rmax}$
F/5
YES
INTER = 1
$t'_z = t_z$
KIP = 0
YES
$t_{zmin} = t_z$
NO
KIP = 1
E/5

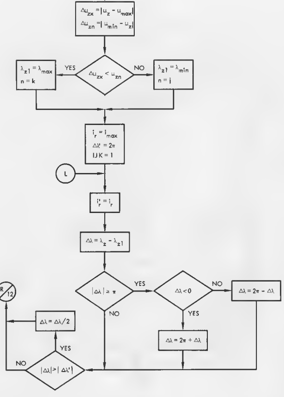

E
μ_{min} = 1000.
μ_{max} = 1000.
KIP = 0
F
IREP = REP
MCDRIV: Enter with KIP, t'_z, u_r, i_r, INTER
Exit with u_{z1}, λ_{z1}, REP
REP > IREP
NO
W/6
YES
$u_{z1}(2) = u_{z1}$
$\lambda_{z1}(2) = \lambda_{z1}$
$t''_z = t_z$
INTER = 1
G/7

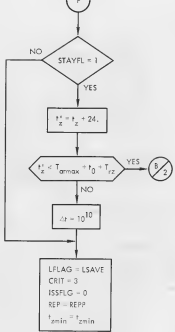

W
NIR = 0
NO
INTER = 1
YES
t'z = t'z + 2.
KIP = 1
LCOUNT = LCOUNT + 1
YES
MDmax = 0
NO
NO
LCOUNT > 5
F 5
ir = ir + Δir (INTER)
YES
ΔT = 0.
Tar = 0.
tz = t'z1
ir > irmax + 0.0001
NO
F 5
YES
J 9
P 11

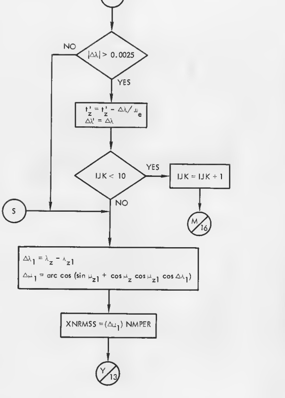

G
MD max = 0
NO
I/9
YES
LCOUNT = 0
u z1 - 0.03 < u min
YES
u min = u z1 - 0.03
NO
u z1 + 0.03 > u max
YES
u max = u z1 + 0.03
NO
INTER = -1
NO
INTER = -1
F/5
YES
X/8

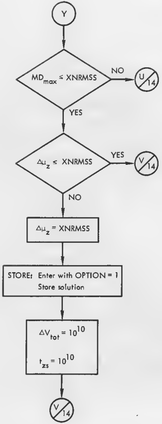

X
INTER = 1
$\mu_{min} \le \mu_z$ and $\mu_z \le \mu_{max}$
YES
$\Delta T = 0$
$T_{ar} = 0$
$t_z = t'_{z1}$
J/9
NO
$t'_z < t_{zmax}$
YES
$t'_z = t_{zmax}$
KIP = 1
F/5
NO
$MD'_{maxs} = MD_{maxs}/NMPER$
$\mu_{min} - MD'_{maxs} \le \mu_z$ and $\mu_z - \mu_{max} + MD'_{maxs}$
YES
STAYFL = 1
$MD_{max} = MD_{maxs}$
MCSOL = 0
NO
$t_z = t'_{z1}$
$\Delta T = 1000$
J/9

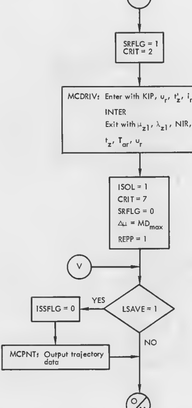

I
NO
$u_{z1}(1) > u_{z1}(2)$
YES
k = 2
j = 1
k = 1
j = 2
$u_{max} = u_{z1}(k)$
$u_{min} = u_{z1}(j)$
$\lambda_{max} = \lambda_{z1}(k)$
$\lambda_{min} = \lambda_{z1}(j)$
KIP
$\neq 0$
K/10
0
J
$h_{min} = h_{mins}$
$t_{zmin} = t_z + 0.001$
$\Delta\lambda = \lambda_z - \lambda_{z1}(1)$
$\Delta V_{max} = \Delta V_{maxs}$
KIP = 1
$t'_z = t'_{z1}$
T/15

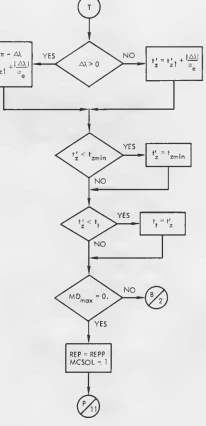

K
$\Delta u_{zx} = |u_z - u_{max}|$
$\Delta u_{zn} = |u_{min} - u_z|$
$\Delta u_{zx} < u_{zn}$
YES
NO
$\lambda_{z1} = \lambda_{max}$
n = k
$\lambda_{z1} = \lambda_{min}$
n = j
$i_r = i_{max}$
$\Delta\lambda' = 2\pi$
IJK = 1
L
$i'_r = i_r$
$\Delta\lambda = \lambda_z - \lambda_{z1}$
$|\Delta\lambda| \geq \pi$
YES
NO
$\Delta\lambda < 0$
NO
YES
$\Delta\lambda = 2\pi - \Delta\lambda$
$\Delta\lambda = 2\pi + \Delta\lambda$
R
12
$\Delta\lambda = \Delta\lambda/2$
YES
NO
$|\Delta\lambda| > |\Delta\lambda'|$

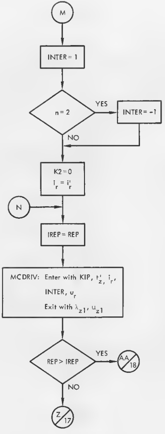

P
STAYFL = 1
NO
YES
t'z = tz + 24.
t'z ≤ Tarmax + t0 + Trz
YES
B/2
NO
Δt = 10^10
LFLAG = LSAVE
CRIT = 3
ISSFLG = 0
REP = REPP
tzmin = tzmin
RETURN

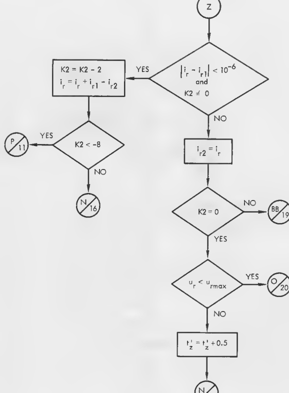

R
NO
$|\Delta\lambda| > 0.0025$
YES
$t_z' = t_z' - \Delta\lambda / \omega_e$
$\Delta\lambda' = \Delta\lambda$
$IJK < 10$
YES
$IJK = IJK + 1$
M 16
S
NO
$\Delta\lambda_1 = \lambda_z - \lambda_{z1}$
$\Delta\mu_1 = \text{arc cos} (\sin \mu_{z1} + \cos \mu_z \cos \mu_{z1} \cos \Delta\lambda_1)$
XNRMSS = $(\Delta\mu_1)$ NMPER
Y 13

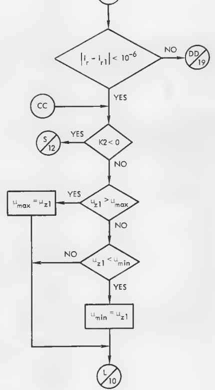

Y
MDmax ≤ XNRMSS
NO
U/14
YES
Δμz ≤ XNRMSS
YES
V/14
NO
Δμz = XNRMSS
STORE: Enter with OPTION = 1
Store solution
ΔVtot = 10^10
tzs = 10^10
V/14

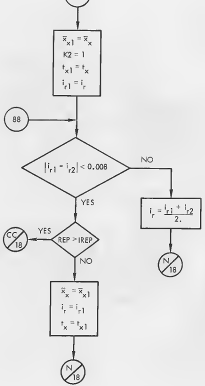

U
SRFLG = 1
CRIT = 2
MCDRIV: Enter with KIP, υr, t'z, ir, INTER
Exit with μz1, λz1, NIR, REP, tz, Tar, υr
ISOL = 1
CRIT = 7
SRFLG = 0
Δμ = MDmax
REPP = 1
V
LSAVE = 1
YES
ISSFLG = 0
NO
MCPNT: Output trajectory data
O
11

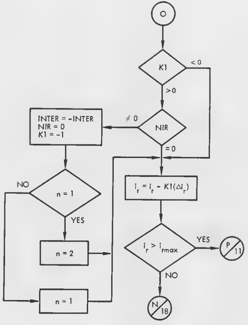

T
Δλ > 0
YES
NO
Δλ = 2π − Δλ
t'z = t'z1 + |Δλ| / ωe
t'z = t'z1 + |Δλ| / ωe
t'z < tzmin
YES
t'z = tzmin
NO
t'z < tf
YES
tf = t'z
NO
MDmax = 0.
NO
B/2
YES
REP = REPP
MCSOL = 1
P/11


M
INTER = 1
n = 2
YES
INTER = -1
NO
K2 = 0
$i_r = i'_r$
N
IREP = REP
MCDRIV: Enter with KIP, t'_z, i_r,
INTER, u_r
Exit with λ_{z1}, u_{z1}
REP > IREP
YES
AA/18
NO
Z/17


Z
|i_r - i_r1| < 10^-6
and
K2 ≠ 0
YES
K2 = K2 - 2
i_r = i_r + i_r1 - i_r2
NO
i_r2 = i_r
K2 < -8
YES
P/11
NO
N/16
K2 = 0
NO
BB/19
YES
u_r < u_rmax
YES
O/20
NO
t'_z = t'_z + 0.5
N/16


AA
$|i_r - i_{r1}| < 10^{-6}$
NO
DD/19
YES
CC
YES
S/12
K2 < 0
NO
YES
$u_{max} = u_{z1}$
$u_{z1} > u_{max}$
NO
NO
$u_{z1} < u_{min}$
YES
$u_{min} = u_{z1}$
L/10


DD
$\bar{x}_{x1} = \bar{x}_x$
K2 = 1
$t_{x1} = t_x$
$i_{r1} = i_r$
88
$|i_{r1} - i_{r2}| < 0.008$
NO
$i_r = \frac{i_{r1} + i_{r2}}{2.}$
N/18
YES
YES
CC/18
REP > IREP
NO
$\bar{x}_x = \bar{x}_{x1}$
$i_r = i_{r1}$
$t_x = t_{x1}$
N/18


O
K1
< 0
> 0
≠ 0
NIR
= 0
INTER = -INTER
NIR = 0
K1 = -1
NO
n = 1
YES
n = 2
n = 1
$i_r = i_r - K1(\Delta i_r)$
$i_r > i_{rmax}$
YES
P/11
NO
N/18

A-4 CLL

Purpose

This routine contains the control logic for alternate target point solutions. This logic is designed to return the spacecraft to an alternate target line within the constraints imposed upon the problem. An alternate target line is composed of straight line segments on the surface of the earth. Each line segment is defined by the latitude and longitude of its end points.

Input

LFLAG	= 1, output flag for the tradeoff display = 2, output flag for an AST solution
CIRCUM	= 0, both postmaneuver directions of motion are used = 1, either retrograde or noncircumlunar solutions are sought as determined in MASTER = 2, only circumlunar motion is allowed
LINE	the latitude, longitude coordinates of the target line
i_{rmax}	maximum return inclination
T_{armax}	maximum time from abort to reentry
t_0	time at abort
T_{rz}	time from reentry to landing
h_{min}	minimum altitude allowed at closest approach to the moon
ΔV_{max}	maximum available abort velocity
ω_e	rotational rate of the earth
u_{rmax}	maximum reentry speed
i_{rk}	the desired return inclination is $\|i_{rk}\|$; when $i_{rk} < 0$, A_{zr} is greater than the azimuth at reentry for the minimum possible return plane; when $i_{rk} > 0$, A_{zr} is less than that for the minimum possible inclination

q_d $\begin{cases} = 0, \text{ no pericynthion passage} \\ = 1, \text{ pericynthion passage} \end{cases}$

h_p pericynthion altitude

Output

The output of this routine is the trajectories to the alternate target line and the solution flag

ISOL $\begin{cases} = 0, \text{ no solution found} \\ = 1, \text{ a solution was found} \end{cases}$

Discussion

A solution which returns with minimum return time to an unspecified area is first generated without regard for the pericynthion altitude and ΔV_{max} constraints. By determining how close the longitude of impact is to the target line at that latitude, the landing time is adjusted to hit near the ATP using the equation

$$t_z{}' = t_z{}' + \frac{\lambda_{zl} - \lambda_t}{\omega_e}$$

where

$t_z{}'$ = time of landing

λ_{zl} = longitude of impact

λ_t = longitude of ATP at the latitude of impact

ω_e = rotational rate of the earth

A trajectory is computed using the time of landing, and the resulting impact longitude is compared with the longitude of the target line. This process is continued until the two longitudes are within a tolerance of 0.05 earth radius.

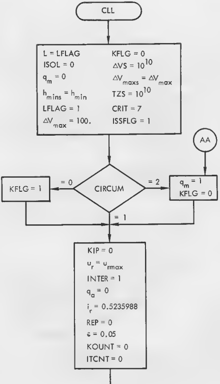
The constraints are now put back on the problem, and iteration begins to find the minimum ΔV solution. Assuming that

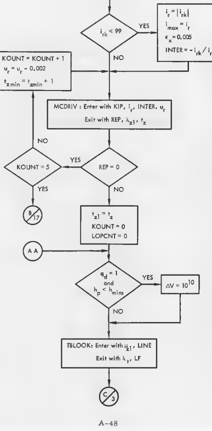
$$\frac{\partial^2 \Delta V}{\partial i_r{}^2} \approx \text{constant}$$

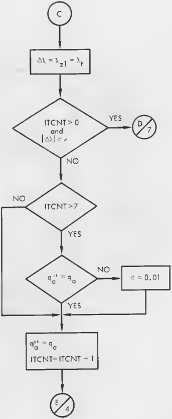
at a fixed landing time, trajectories are generated which return with maximum inclination (both planes) and with the minimum possible inclination. MINMIZ is called to predict the value of i_r which results in minimum ΔV requirements. This minimum ΔV is also estimated by MINMIZ. Using the new i_r, another trajectory is constructed in MCDRIV and the actual ΔV required is determined. When the estimated and actual ΔV differ by less than 0.0001 earth radius per hour, the iteration ceases. Otherwise the inputs to MINMIZ are updated and new computations made.

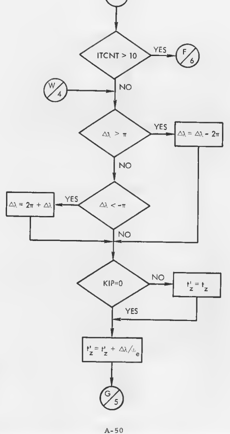
After the minimization is complete, the impact latitude is tested to see if it lies within the bounds of the alternate target line. If it is not within the bounds, linear predictions are used to iteratively change return inclination so that the impact latitude is near the end of the target line. After the return inclination is found which results in an acceptable landing latitude, landing time is adjusted so that impact is within 0.005 earth radius of the target line.

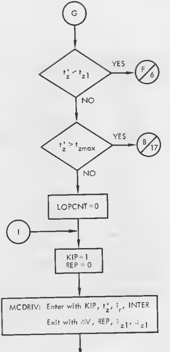
The landing time is now incremented by 24 hours, and the problem begun again for a new day of landing. This procedure is repeated until the maximum landing time is exceeded. If required by input, the other direction of motion is considered.

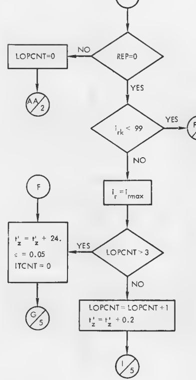

CLL
L = LFLAG
ISOL = 0
$q_m = 0$
$h_{mins} = h_{min}$
LFLAG = 1
$\Delta V_{max} = 100.$
KFLG = 0
$\Delta VS = 10^{10}$
$\Delta V_{maxs} = \Delta V_{max}$
$TZS = 10^{10}$
CRIT = 7
ISSFLG = 1
AA
KFLG = 1
= 0
CIRCUM
= 2
$q_m = 1$
KFLG = 0
= 1
KIP = 0
$\upsilon_r = \upsilon_{rmax}$
INTER = 1
$q_a = 0$
$i_r = 0.5235988$
REP = 0
$\epsilon = 0.05$
KOUNT = 0
ITCNT = 0
A
2

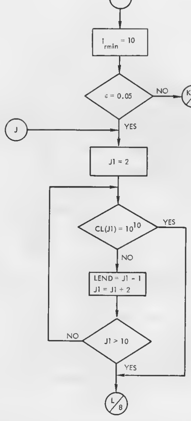

A
i_rk < 99
YES
i_r = |i_rk|
i_max = i_r
ε_x = 0.005
INTER = -i_rk / i_r
NO
KOUNT = KOUNT + 1
u_r = u_r - 0.002
t_zmin = t_zmin + 1
MCDRIV : Enter with KIP, i_r, INTER, u_r
Exit with REP, λ_z1, t_z
NO
YES
KOUNT = 5
REP = 0
YES
NO
B/17
t_z1 = t_z
KOUNT = 0
LOPCNT = 0
AA
q_d = 1
and
h_p < h_mins
YES
ΔV = 10^10
NO
TBLOOK: Enter with μ_z1, LINE
Exit with λ_t, LF
C/3

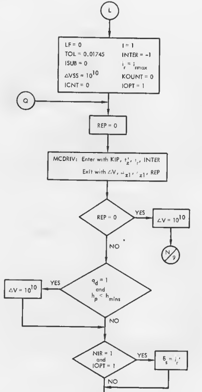

C
Δλ = λz1 - λt
ITCNT > 0
and
|Δλ| < ε
YES
D
7
NO
ITCNT > 7
NO
YES
q''a = qa
NO
ε = 0.01
YES
q''a = qa
ITCNT= ITCNT + 1
E
4

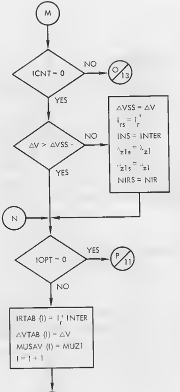

E
ITCNT > 10
YES
F/6
W/4
NO
Δλ > π
YES
Δλ = Δλ - 2π
NO
Δλ = 2π + Δλ
YES
Δλ < -π
NO
KIP=0
NO
t'z = tz
YES
t'z = t'z + Δλ/ωe
G/5

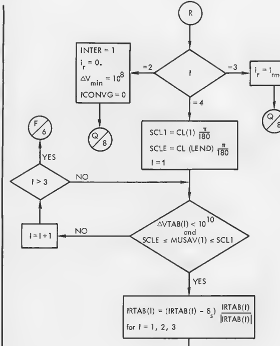

G
$t_z' < t_{z1}$
YES
F/6
NO
$t_z' > t_{zmax}$
YES
B/17
NO
LOPCNT = 0
I
KIP = 1
REP = 0
MCDRIV: Enter with KIP, t_z', i_r, INTER
Exit with ΔV, REP, λ_{z1}, α_{z1}
H/6

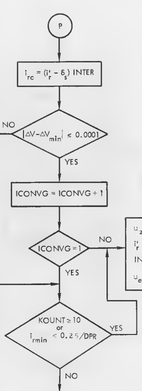

H
REP=0
NO
LOPCNT=0
AA 2
YES
i_rk < 99
YES
F 6
NO
F
i_r = i_rmax
t'_z = t'_z + 24.
ε = 0.05
ITCNT = 0
YES
LOPCNT > 3
NO
G 5
LOPCNT = LOPCNT + 1
t'_z = t'_z + 0.2
I 5

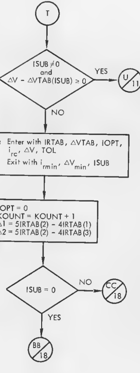

D
$t_{rmin} = 10$
$\varepsilon = 0.05$
NO
K/16
J
YES
J1 = 2
$CL(J1) = 10^{10}$
YES
NO
LEND = J1 - 1
J1 = J1 + 2
NO
J1 > 10
YES
L/8

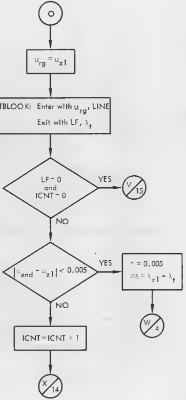

L
LF = 0
TOL = 0.01745
ISUB = 0
ΔVSS = 10^10
ICNT = 0
I = 1
INTER = -1
i_r = i_rmax
KOUNT = 0
IOPT = 1
Q
REP = 0
MCDRIV: Enter with KIP, t'_z, ı_r, INTER
Exit with ⊿V, ⊣_z1, ′_z1, REP
REP = 0
YES
ΔV = 10^10
N/9
NO
q_d = 1 and h_p < h_mins
YES
ΔV = 10^10
NO
NIR = 1 and IOPT = 1
YES
δ_s = i_r′
NO
M/9

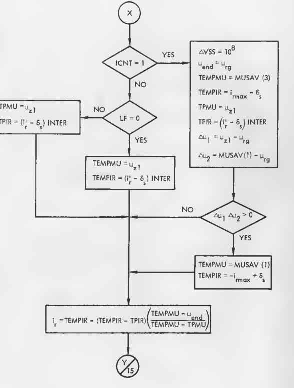

M
ICNT = 0
NO
O/13
YES
ΔVSS = ΔV
$i_{rs} = i_r'$
INS = INTER
$\lambda_{z1s} = \lambda_{z1}$
$\mu_{z1s} = \mu_{z1}$
NIRS = NIR
ΔV > ΔVSS
NO
YES
N
IOPT = 0
YES
P/11
NO
IRTAB (I) = i_r' INTER
ΔVTAB (I) = ΔV
MUSAV (I) = MUZ1
I = I + 1
R/10

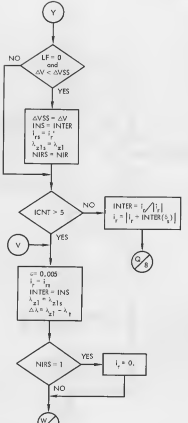

R
INTER = 1
$i_r = 0.$
$\Delta V_{min} = 10^8$
ICONVG = 0
= 2
I
= 3
$i_r = i_{rmax}$
Q/8
= 4
F/6
YES
Q/8
$SCL1 = CL(1) \frac{\pi}{180}$
$SCLE = CL(LEND) \frac{\pi}{180}$
I = 1
I > 3
NO
$\Delta VTAB(I) < 10^{10}$
and
$SCLE \le MUSAV(I) \le SCL1$
I = I + 1
NO
YES
$IRTAB(I) = (IRTAB(I) - \delta_s) \frac{IRTAB(I)}{|IRTAB(I)|}$
for I = 1, 2, 3
S/11

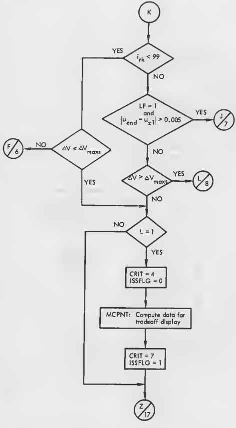

P
$i_{rc} = (i_r' - \delta_s)$ INTER
NO
$|\Delta V - \Delta V_{min}| \leq 0.0001$
YES
ICONVG = ICONVG + 1
U
$u_{z1} = u_{z1s}$
$i_r' = i_{rs}$
INTER = INS
$u_{end} = 10^{10}$
NO
ICONVG = 1
YES
S
KOUNT ≥ 10
or
$i_{rmin} < 0.25/DPR$
YES
O
13
NO
T
12

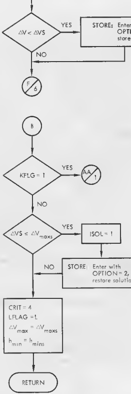

T
ISUB ≠ 0
and
ΔV - ΔVTAB(ISUB) ≥ 0
YES
U
11
NO
MINMIZ: Enter with IRTAB, ΔVTAB, IOPT,
i_rc, ΔV, TOL
Exit with i_rmin, ΔV_min, ISUB
IOPT = 0
KOUNT = KOUNT + 1
Δ1 = 5IRTAB(2) - 4IRTAB(1)
Δ2 = 5IRTAB(2) - 4IRTAB(3)
ISUB = 0
NO
CC
18
YES
BB
18

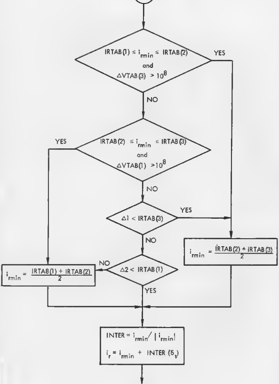

O
$u_{rg} = u_{z1}$
TBLOOK: Enter with u_{rg}, LINE
Exit with LF, λ_t
LF = 0 and ICNT = 0
YES
V/15
NO
$|u_{end} - u_{z1}| < 0.005$
YES
$\epsilon = 0.005$
$\Delta\lambda = \lambda_{z1} - \lambda_t$
W/4
NO
ICNT = ICNT + 1
X/14


X
ICNT = 1
YES
NO
ΔVSS = 10^8
$u_{end} = u_{rg}$
TEMPMU = MUSAV (3)
TEMPIR = $i_{rmax} - \delta_s$
TPMU = u_{z1}
TPIR = $(i_r' - \delta_s)$ INTER
$\Delta u_1 = u_{z1} - u_{rg}$
Δu_2 = MUSAV(1) $- u_{rg}$
TPMU = u_{z1}
TPIR = $(i_r' - \delta_s)$ INTER
NO
LF = 0
YES
TEMPMU = u_{z1}
TEMPIR = $(i_r' - \delta_s)$ INTER
NO
$\Delta u_1 \, \Delta u_2 > 0$
YES
TEMPMU = MUSAV (1)
TEMPIR = $-i_{rmax} + \delta_s$
i_r = TEMPIR - (TEMPIR - TPIR)$\left(\frac{\text{TEMPMU} - u_{end}}{\text{TEMPMU} - \text{TPMU}}\right)$
Y
15


Y
NO
LF = 0 and ΔV < ΔVSS
YES
ΔVSS = ΔV
INS = INTER
i_rs = i_r'
λ_z1s = λ_z1
NIRS = NIR
ICNT > 5
NO
INTER = i_r/|i_r|
i_r = |i_r + INTER(δ_s)|
Q/8
V
YES
ε = 0.005
i_r = i_rs
INTER = INS
λ_z1 = λ_z1s
Δλ = λ_z1 - λ_t
NIRS = 1
YES
i_r = 0.
NO
W/4


K
YES
$i_{rk} < 99$
NO
LF = 1
and
$|\mu_{end} - \mu_{z1}| > 0.005$
YES
J/7
NO
F/6
$\Delta V \leq \Delta V_{maxs}$
YES
$\Delta V > \Delta V_{maxs}$
YES
L/8
NO
NO
L = 1
YES
CRIT = 4
ISSFLG = 0
MCPNT: Compute data for tradeoff display
CRIT = 7
ISSFLG = 1
Z/17


Z
ΔV < ΔVS
YES
STORE: Enter with OPTION = 1; store solution
NO
F/6
B
KFLG = 1
YES
AA/1
NO
ΔVS ≤ ΔVmaxs
YES
ISOL = 1
STORE: Enter with OPTION = 2, restore solution
NO
CRIT = 4
LFLAG = L
ΔVmax = ΔVmaxs
hmin = hmins
RETURN


B8
IRTAB(1) ≤ i_rmin ≤ IRTAB(2) and ΔVTAB(3) > 10^8
YES
NO
IRTAB(2) ≤ i_rmin ≤ IRTAB(3) and ΔVTAB(1) > 10^8
YES
NO
Δ1 < IRTAB(3)
YES
NO
i_rmin = (IRTAB(2) + IRTAB(3))/2
Δ2 < IRTAB(1)
NO
YES
i_rmin = (IRTAB(1) + IRTAB(2))/2
INTER = i_rmin / |i_rmin|
i_r = i_rmin + INTER (δ_s)
Q/8

A-5 MCUA

Purpose

This routine contains the control logic for the unspecified area solution. The logic minimizes the ΔV required for return to earth as a function of landing time and return inclination. The landing site of the minimum ΔV solution is limited only by the reentry constraints.

Input

ΔV_{max}	maximum allowable ΔV
h_{min}	minimum altitude allowed at closest approach to the moon
r_m	radius of the moon, assumed spherical
u_{rmax}	maximum reentry velocity allowed
i_{rmax}	maximum allowable return inclination
t_{zmin}	minimum landing time allowed
t_{zmax}	maximum landing time allowed
CIRCUM	= 0, both circumlunar and noncircumlunar motion considered = 1, only noncircumlunar motion allowed = 2, only circumlunar motion allowed

Output

The minimum ΔV solution is output as well as the solution flag

ISOL	= 0, no solution found = 1, solution found

Discussion

The lower bound at t_z, the landing time, is determined by computing the landing time associated with returning to earth at the maximum allowable reentry velocity. If the input t_{zmin} is less than this computed t_z, reset the constraint so that

$$t_{zmin} = t_z$$

Set the upper and lower bounds of i_r so that

$$i_{rmax} = i_{rmax} - \delta_x$$

and

$$i_{rmin} = -(i_{rmax} - \delta_x)$$

where δ_x is the declination of the geocentric pseudostate. The constraints are now removed in order to insure finding a solution at each value of the independent variables. After the minimization is finished, the constraints will be checked.

At $t_z = t_{zmin}$, the minimum ΔV required for return is determined by assuming

$$\frac{\partial^2 \Delta V}{\partial i_r^2} \approx \text{constant}$$

in the region of the minimum. This is mechanized by constructing trajectories at

$$i_r = i_{rmin}$$

$$i_r = \frac{i_{rmin} + i_{rmax}}{2}$$

and

$$i_r = i_{rmax}$$

Then the value of the independent variable which results in minimum ΔV is approximated by MINMIZ to be

$$i_r = \frac{i_r(2) + i_r(3)}{2} - \left[\frac{\Delta V(3) - \Delta V(2)}{i_r(3) - i_r(2)}\right] \left[\frac{i_r(1) - i_r(3)}{2\left(\frac{\Delta V(2) - \Delta V(1)}{i_r(2) - i_r(1)} - \frac{\Delta V(3) - \Delta V(2)}{i_r(3) - i_r(2)}\right)}\right]$$

The value of ΔV associated with this i_r is also estimated by MINMIZ.

$$\Delta V_{estimated} = \Delta V(2) \frac{\left[i_r(2) - i_r\right]^2 \quad \left[\Delta V(1) - \Delta V(3)\right]}{\left[i_r(1) - i_r\right]^2 - \left[i_r(3) - i_r\right]^2}$$

Using the independent variables t_z and i_r, the return trajectory is constructed in MCDRIV and the actual ΔV required is determined. If

$$|\Delta V_{actual} - \Delta V_{estimated}| \leq 0.0001 \text{ earth radii}$$

then the assumption that

$$\frac{\partial^2 \Delta V}{\partial i_r^2} \approx \text{constant}$$

is sufficiently valid and no further iteration is required. Otherwise, additional i_r and ΔV predictions are made using an updated definition of the local minimum until the above convergence criterion is met.

At $t_z = (t_{zmin} + t_{zmax})/2.$ and $t_z = t_{zmax}$, the same procedures are followed to find a minimum ΔV and its related i_r. Now, it is assumed that ΔV is a function of t_z alone and

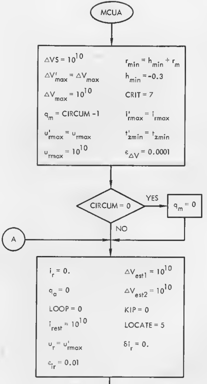
$$\frac{\partial^2 \Delta V}{\partial t_z^2} \approx \text{constant}$$

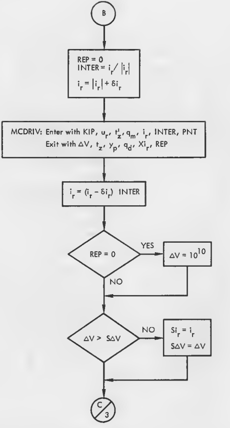
Routine MINMIZ is called to predict the t_z associated with minimum ΔV requirements and the related ΔV.

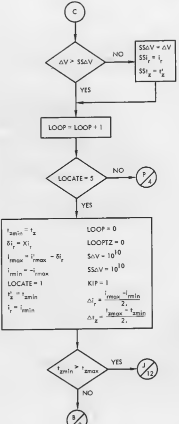
Then the i_r which minimizes ΔV at the predicted t_z is estimated by assuming i_r is linear with respect to t_z along the envelope of minimum ΔV solutions. The i_{rmin} and i_{rmax} boundaries are updated by using the error between the predicted and actual i_r in the last pass. A convergence tolerance ε_{ir} based on the rate of change of ΔV and i_r along the envelope of minimum ΔV is computed.

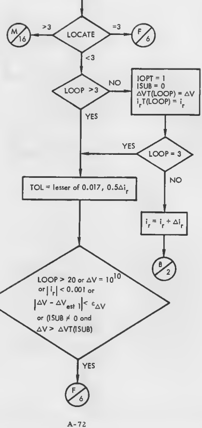
Then at the predicted t_z, the minimum ΔV required for return is found either as it was at $t_z = t_{zmin}$ or by specifying the predicted i_r if i_r has converged. If the actual minimum ΔV found differs from the predicted ΔV by less than 0.6 foot per second, the minimization is complete for the given direction of motion and the final constraint test is applied. Otherwise, the i_r of this solution is compared with the predicted value. If they differ by less than ε_{ir}, no further ΔV minimization is required with respect to i_r.

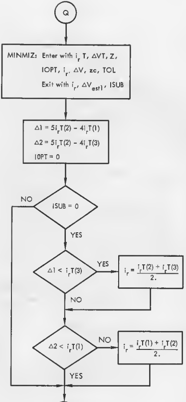
Iteration continues by computing new predicted values for t_z, ΔV, and i_r until the convergence criterion is met. When this occurs the constraints are tested, and, if necessary, linear partials are used to find the t_z associated with the pericynthion altitude constraint. The problem is then reinitialized for the other direction of motion.

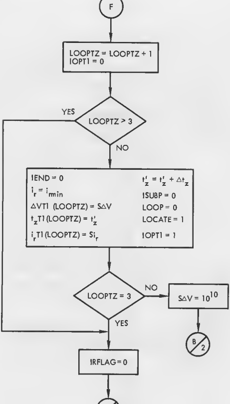

MCUA
$\Delta VS = 10^{10}$
$\Delta V'_{max} = \Delta V_{max}$
$\Delta V_{max} = 10^{10}$
q_m = CIRCUM -1
$\upsilon'_{rmax} = \upsilon_{rmax}$
$\upsilon_{rmax} = 10^{10}$
$r_{min} = h_{min} + r_m$
$h_{min} = -0.3$
CRIT = 7
$i'_{rmax} = i_{rmax}$
$t'_{zmin} = t_{zmin}$
$\varepsilon_{\Delta V} = 0.0001$
CIRCUM = 0
YES
$q_m = 0$
NO
A
$i_r = 0.$
$q_a = 0$
LOOP = 0
$i_{rest} = 10^{10}$
$\upsilon_r = \upsilon'_{rmax}$
$\varepsilon_{ir} = 0.01$
$\Delta V_{est1} = 10^{10}$
$\Delta V_{est2} = 10^{10}$
KIP = 0
LOCATE = 5
$\delta i_r = 0.$
B
2

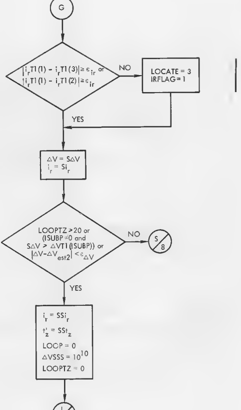

B
REP = 0
INTER = i_r / |i_r|
i_r = |i_r| + δi_r
MCDRIV: Enter with KIP, u_r, t'_z, q_m, i_r, INTER, PNT
Exit with ΔV, t_z, y_p, q_d, Xi_r, REP
i_r = (i_r − δi_r) INTER
REP = 0
YES
ΔV = 10^10
NO
ΔV > SΔV
NO
Si_r = i_r
SΔV = ΔV
C
3

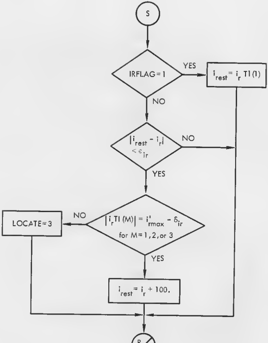

C
ΔV > SSΔV
NO
SSΔV = ΔV
SSi_r = i_r
SSt_z = t'_z
YES
LOOP = LOOP + 1
LOCATE = 5
NO
P/4
YES
t_zmin = t_z
δi_r = Xi_r
i_rmax = i'_rmax - δi_r
i_rmin = -i_rmax
LOCATE = 1
t'_z = t_zmin
i_r = i_rmin
LOOP = 0
LOOPTZ = 0
SΔV = 10^10
SSΔV = 10^10
KIP = 1
Δi_r = (i_rmax - i_rmin)/2.
Δt_z = (t_zmax - t_zmin)/2.
t_zmin > t_zmax
YES
J/12
NO
B/2

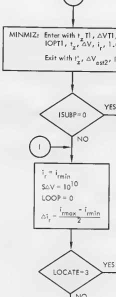

P
M 16
>3
LOCATE
=3
F 6
<3
LOOP >3
NO
IOPT = 1
ISUB = 0
ΔVT(LOOP) = ΔV
i_r T(LOOP) = i_r
YES
YES
LOOP = 3
NO
TOL = lesser of 0.017, 0.5Δi_r
i_r = i_r + Δi_r
B 2
Q 5
NO
LOOP > 20 or ΔV = 10^10
or |i_r| < 0.001 or
|ΔV - ΔV_est 1| < ε_ΔV
or (ISUB ≠ 0 ond
ΔV > ΔVT(ISUB)
YES
F 6

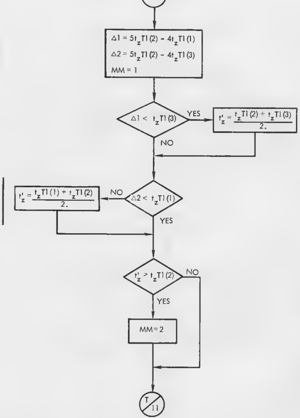

Q
MINMIZ: Enter with i_r T, ΔVT, Z,
IOPT, i_r, ΔV, zc, TOL
Exit with i_r, ΔV_est1, ISUB
Δ1 = 5i_rT(2) − 4i_rT(1)
Δ2 = 5i_rT(2) − 4i_rT(3)
IOPT = 0
ISUB = 0
NO
YES
Δ1 < i_rT(3)
YES
i_r = (i_rT(2) + i_rT(3)) / 2.
NO
Δ2 < i_rT(1)
NO
i_r = (i_rT(1) + i_rT(2)) / 2.
YES
B
2

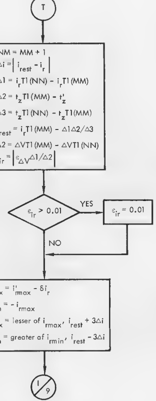

F
LOOPTZ = LOOPTZ + 1
IOPT1 = 0
LOOPTZ > 3
YES
NO
IEND = 0
$i_r = i_{rmin}$
ΔVT1 (LOOPTZ) = SΔV
t_zT1 (LOOPTZ) = t'_z
i_rT1 (LOOPTZ) = Si_r
$t'_z = t'_z + \Delta t_z$
ISUBP = 0
LOOP = 0
LOCATE = 1
IOPT1 = 1
LOOPTZ = 3
NO
$S\Delta V = 10^{10}$
B/2
YES
IRFLAG = 0
G/7

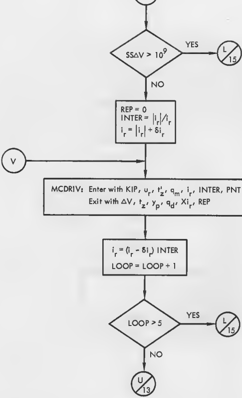

G
|i_r T1(1) - i_r T1(3)| ≥ ε_ir or
|i_r T1(1) - i_r T1(2)| ≥ ε_ir
NO
LOCATE = 3
IRFLAG = 1
YES
ΔV = SΔV
i_r = Si_r
LOOPTZ > 20 or
(ISUBP ≠ 0 and
SΔV > ΔVT1(ISUBP)) or
|ΔV - ΔV_est2| < ε_ΔV
NO
S
8
YES
i_r = SSi_r
t'_z = SSt_z
LOOP = 0
ΔVSSS = 10^10
LOOPTZ = 0
J
12

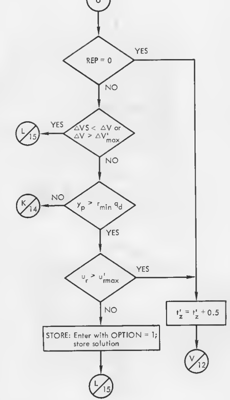

S
IRFLAG = 1
YES
$i_{rest} = i_r$ TI (1)
NO
$|i_{rest} - i_r| < \varepsilon_{ir}$
NO
YES
$|i_r$ TI (M)$| = i'_{rmax} - \delta_{ir}$
for M = 1, 2, or 3
NO
LOCATE = 3
YES
$i_{rest} = i_r + 100.$
R
9

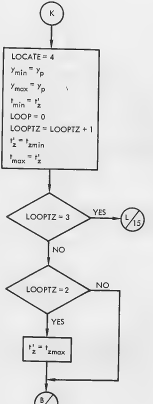

R
MINMIZ: Enter with t_z T1, ΔVT1, i_rT1, IOPT1, t_z, ΔV, i_r, 1.0
Exit with t_z', ΔV_{est2}, ISUBP
ISUBP = 0
YES
H/10
NO
I
$i_r = i_{rmin}$
$S\Delta V = 10^{10}$
LOOP = 0
$\Delta i_r = \frac{i_{rmax} - i_{rmin}}{2}$
LOCATE = 3
YES
$i_r = i_{rest}$
NO
$\varepsilon_{ir} < 10^{-4}$
YES
$i_{rest} = 10^{10}$
NO
B/2

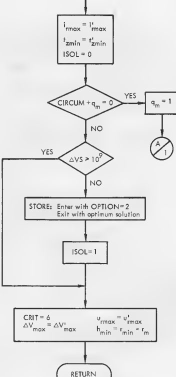

H
Δ1 = 5t_z T1 (2) - 4t_z T1 (1)
Δ2 = 5t_z T1 (2) - 4t_z T1 (3)
MM = 1
Δ1 < t_z T1 (3)
YES
t'_z = (t_z T1 (2) + t_z T1 (3)) / 2.
NO
Δ2 < t_z T1 (1)
NO
t'_z = (t_z T1 (1) + t_z T1 (2)) / 2.
YES
t'_z > t_z T1 (2)
NO
YES
MM = 2
T
11

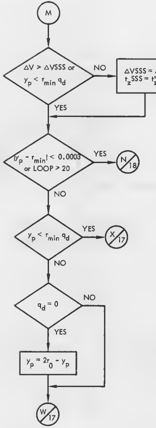

T
NM = MM + 1
Δi = | i_rest - i_r |
Δ1 = i_r T1 (NN) - i_r T1 (MM)
Δ2 = t_z T1 (MM) - t'_z
Δ3 = t_z T1 (NN) - t_z T1 (MM)
i_rest = i_r T1 (MM) - Δ1Δ2/Δ3
Δ2 = ΔVT1 (MM) - ΔVT1 (NN)
ε_ir = | ε_ΔV Δ1/Δ2 |
ε_ir > 0.01
YES
ε_ir = 0.01
NO
i_rmax = i'_rmax - δi_r
i_rmin = - i_rmax
i_rmax = lesser of i_rmax, i_rest + 3Δi
i_rmin = greater of i_rmin, i_rest - 3Δi
1/9

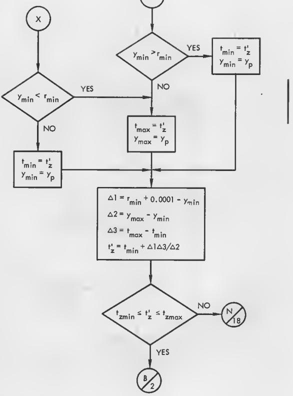

J
SSΔV > 10^9
YES
L/15
NO
REP = 0
INTER = |i_r|/i_r
i_r = |i_r| + δi_r
V
MCDRIV: Enter with KIP, υ_r, t'_z, q_m, i_r, INTER, PNT
Exit with ΔV, t_z, y_p, q_d, Xi_r, REP
i_r = (i_r - δi_r) INTER
LOOP = LOOP + 1
LOOP > 5
YES
L/15
NO
U/13

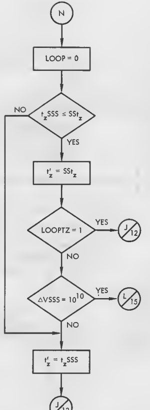

U
YES
REP = 0
NO
YES
L 15
△VS < △V or △V > △V'max
NO
NO
K 14
yp > rmin qd
YES
YES
ur > u'rmax
NO
t'z = t'z + 0.5
STORE: Enter with OPTION = 1; store solution
V 12
L 15


K
LOCATE = 4
$y_{min} = y_p$
$y_{max} = y_p$
$t_{min} = t'_z$
LOOP = 0
LOOPTZ = LOOPTZ + 1
$t'_z = t_{zmin}$
$t_{max} = t'_z$
LOOPTZ = 3
YES
L/15
NO
LOOPTZ = 2
NO
YES
$t'_z = t_{zmax}$
B/2


L
$i_{rmax} = i'_{rmax}$
$t_{zmin} = t'_{zmin}$
ISOL = 0
CIRCUM + q_m = 0
YES
q_m = 1
A
1
NO
YES
$\Delta VS > 10^9$
NO
STORE: Enter with OPTION = 2
Exit with optimum solution
ISOL = 1
CRIT = 6
$\Delta V_{max} = \Delta V'_{max}$
$u_{rmax} = u'_{rmax}$
$h_{min} = r_{min} - r_m$
RETURN


M
ΔV > ΔVSSS or
$y_p < r_{min} q_d$
NO
ΔVSSS = ΔV
t_zSSS = t'_z
YES
$|y_p - r_{min}| < 0.0003$
or LOOP > 20
YES
N/18
NO
$y_p < r_{min} q_d$
YES
X/17
NO
$q_d = 0$
NO
YES
$y_p = 2r_0 - y_p$
W/17


W
X
$y_{min} > r_{min}$
YES
$t_{min} = t'_z$
$y_{min} = y_p$
NO
$y_{min} < r_{min}$
YES
NO
$t_{max} = t'_z$
$y_{max} = y_p$
$t_{min} = t'_z$
$y_{min} = y_p$
$\Delta 1 = r_{min} + 0.0001 - y_{min}$
$\Delta 2 = y_{max} - y_{min}$
$\Delta 3 = t_{max} - t_{min}$
$t'_z = t_{min} + \Delta 1 \Delta 3 / \Delta 2$
$t_{zmin} \le t'_z \le t_{zmax}$
NO
N/18
YES
B/2


N
LOOP = 0
NO
$t_z SSS \le SSt_z$
YES
$t'_z = SSt_z$
LOOPTZ = 1
YES
J 12
NO
$\Delta VSSS = 10^{10}$
YES
L 15
NO
$t'_z = t_z SSS$
J 12

A-6 SEARCH

Purpose

This routine determines the time of maneuver associated with minimum ΔV requirements by searching over the solutions from a table of state vectors. If a local minimum is detected, additional state vectors are generated to further define the minimum.

Input and Output

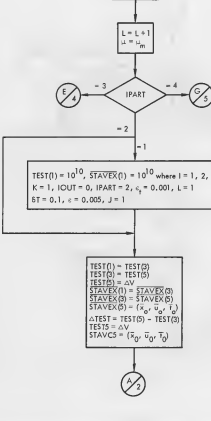
IPART	flag to control interval of search
$\overline{x}_o$	premaneuver position vector
x_o	premaneuver position magnitude
$\overline{u}_o$	premaneuver velocity vector
u_o	premaneuver velocity magnitude
ε	convergence tolerance on ΔV
ε_t	convergence tolerance on maneuver time
STAVEC(J)	table of state vectors and maneuver times
JUMP	number of state vectors in the table STAVEC
t_o	premaneuver time of day
ΔV	ΔV required for the maneuver

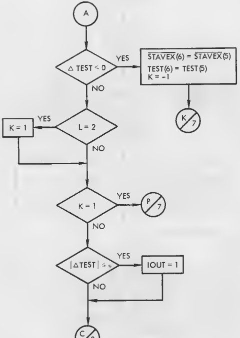
Discussion

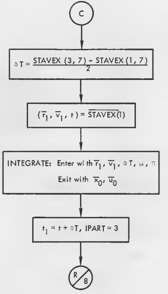
When the search option is required, the supervisory logic generates a table of premaneuver state vectors at various times along the trajectory. Using these states, analytic return-to-earth maneuvers are computed for the appropriate submode -- ATP, PTP, or FCUA. After the maneuver is determined for a given state vector and control is returned to MASTER, the ΔV required for the maneuver, the premaneuver state vector, and the ignition time are transferred to SEARCH.

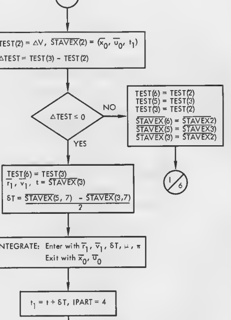
SEARCH saves ths best solution of those found and also retains the data for the last three table states investigated. When a local minimum is detected in the function $\Delta V = f(t_0)$, new state vectors are introduced on both sides of the minimum solution. These new state vectors are produced at the halfway time between the saved states. Return-to-earth data are produced for these states and the definition of the local minimum is updated. New state vectors are produced by the halving process until either ΔV on two successive passes is changing by less than $\varepsilon = 0.005$ earth radius per hour or the ignition time is changing by less than $\varepsilon_t = 0.001$ hour.

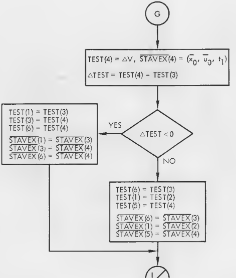
After all of the state vectors in the table have been examined and all local minima investigated, the premaneuver state vector and ignition time associated with minimum ΔV is output to MASTER. The analytic minimum ΔV trajectory is recalculated for use as target conditions and as a first guess for the impulsive precision solution.

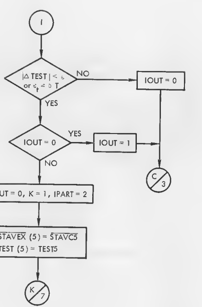

SEARCH
L = L + 1
μ = μm
= 3
IPART
= 4
E 4
G 5
= 2
= 1
TEST(I) = 10^10, STAVEX(I) = 10^10 where I = 1, 2, . . . ,7
K = 1, IOUT = 0, IPART = 2, εt = 0.001, L = 1
δT = 0.1, ε = 0.005, J = 1
TEST(1) = TEST(3)
TEST(3) = TEST(5)
TEST(5) = ΔV
STAVEX(1) = STAVEX(3)
STAVEX(3) = STAVEX(5)
STAVEX(5) = (x̄o, ūo, t̄o)
ΔTEST = TEST(5) - TEST(3)
TEST5 = ΔV
STAVC5 = (x̄0, ū0, t̄0)
A 2

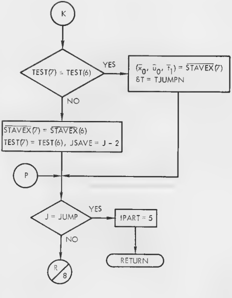

A
△ TEST < 0
YES
NO
STAVEX(6) = STAVEX(5)
TEST(6) = TEST(5)
K = -1
K/7
L = 2
YES
K = 1
NO
K = 1
YES
P/7
NO
|△TEST| ≤ ε
YES
IOUT = 1
NO
C/3

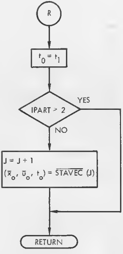

C
δT = STAVEX (3, 7) - STAVEX (1, 7) / 2
($\bar{r}_1$, $\bar{v}_1$, t) = $\overline{STAVEX(1)}$
INTEGRATE: Enter with $\bar{r}_1$, $\bar{v}_1$, δT, μ, π
Exit with $\bar{x}_0$, $\bar{u}_0$
t_1 = t + δT, IPART = 3
R/8


E
TEST(2) = ΔV, $\overline{STAVEX}(2) = (\overline{x}_0, \overline{\upsilon}_0, t_1)$
ΔTEST = TEST(3) − TEST(2)
ΔTEST ≤ 0
NO
YES
TEST(6) = TEST(2)
TEST(5) = TEST(3)
TEST(3) = TEST(2)
$\overline{STAVEX}(6) = \overline{STAVEX2})$
$\overline{STAVEX}(5) = \overline{STAVEX3})$
$\overline{STAVEX}(3) = \overline{STAVEX2})$
I/6
TEST(6) = TEST(3)
$\overline{r}_1, \overline{v}_1, t = \overline{STAVEX}(3)$
$\delta T = \frac{\overline{STAVEX}(5,7) - \overline{STAVEX}(3,7)}{2}$
INTEGRATE: Enter with $\overline{r}_1, \overline{v}_1, \delta T, \mu, \pi$
Exit with $\overline{x}_0, \overline{\upsilon}_0$
$t_1 = t + \delta T$, IPART = 4
R/8


G
TEST(4) = ΔV, $\overline{STAVEX}(4) = (\bar{x}_0, \bar{v}_0, t_1)$
ΔTEST = TEST(4) - TEST(3)
ΔTEST < 0
YES
NO
TEST(1) = TEST(3)
TEST(3) = TEST(4)
TEST(6) = TEST(4)
$\overline{STAVEX}(1) = \overline{STAVEX}(3)$
$\overline{STAVEX}(3) = \overline{STAVEX}(4)$
$\overline{STAVEX}(6) = \overline{STAVEX}(4)$
TEST(6) = TEST(3)
TEST(1) = TEST(2)
TEST(5) = TEST(4)
$\overline{STAVEX}(6) = \overline{STAVEX}(3)$
$\overline{STAVEX}(1) = \overline{STAVEX}(2)$
$\overline{STAVEX}(5) = \overline{STAVEX}(4)$
I/6


I
|Δ TEST| < ε or ε_t < δ T
NO
IOUT = 0
YES
IOUT = 0
YES
IOUT = 1
C/3
NO
IOUT = 0, K = 1, IPART = 2
STAVEX (5) = STAVC5
TEST (5) = TEST5
K/7


K
TEST(7) ≤ TEST(6)
YES
$(\bar{x}_0, \bar{u}_0, \bar{\tau}_1) = \overline{STAVEX}(7)$
δT = TJUMPN
NO
$\overline{STAVEX}(7) = \overline{STAVEX}(6)$
TEST(7) = TEST(6), JSAVE = J - 2
P
J = JUMP
YES
IPART = 5
RETURN
NO
R
8


R
$t_0 = t_1$
IPART > 2
YES
NO
J = J + 1
$(\bar{x}_o, \bar{v}_o, t_o) = \overline{STAVEC}$ (J)
RETURN

APPENDIX B

CONIC TRAJECTORY ROUTINES

CONTENTS

B-1 MCDRIV

Purpose

This routine constructs an overlapped conics trajectory for all moon-centered modes and ensures that the optimum case is stored for future reference. MCDRIV also contains the logic for converging on the PTP landing site.

Input

ϵ_t	time tolerance for pseudostate convergence
ϵ_x	distance tolerance for pseudostate convergence
μ_m	moon gravitational constant
r_m	lunar radius
R_s	radius of the PTS
h_{min}	minimum pericynthion altitude allowed
u_r	reentry velocity
u_{rmax}	maximum reentry velocity allowed
LFLAG	= 1, output all solutions = 2, output only optimum solution
t_{zmin}	minimum landing time allowed
ΔV_{max}	maximum available ΔV
t_0	preabort time
$\overline{v}_0$	preabort velocity vector
$\overline{y}_0$	preabort position vector
i_r	return inclination
t_z	time of landing desired
KIP	= 0, reentry velocity is the independent variable = 1, landing time is the independent variable

λ_z''	longitude of the PTP site
μ_z''	latitude of the PTP site
PNT	if PNT = 1, the stored solution is output
SRFLG	= 0, do not consider miss distance solutions in PTP optimization = 1, consider miss distance solutions in PTP optimization
CRIT	= 2, PTP mode = 4, ATP mode = 6, UA mode = 7, solution not to be considered in optimization
q_m	= 0, noncircumlunar motion postabort = 1, circumlunar motion postabort
INTER	= -1, A_{zr} < reentry azimuth for $i_r = \delta_x$ = +1, A_{zr} > reentry azimuth for $i_r = \delta_x$

Output

REP	= 0, no MCDRIV solution has been found ≠ 0, a solution has been found in MCDRIV
NIR	= 0, input i_r is greater than the declination of the pseudostate = 1, input i_r is less than the declination of the pseudostate
p_h	semilatus rectum of selenocentric conic
$\overline{\Delta V}_{tot}$	$\overline{\Delta V}$ vector
ΔV_{tot}	ΔV magnitude
T_{ar}	time from abort to reentry
h_p	pericynthion altitude
k_x	iteration counter for pseudostate convergence
a_h	semimajor axis for selenocentric conic

e_h	eccentricity of the selenocentric conic
T_{xr}	time from postpericynthion pseudostate to reentry
t_p	time of pericynthion passage
t_x	time of PTS exit
u_r	reentry velocity
$\overline{x}_x$	geocentric transearth pseudostate position vector
x_x	magnitude of $\overline{x}_x$
$\overline{y}_x$	selenocentric position vector at PTS exit
$\overline{v}_x$	selenocentric velocity vector at PTS exit
$\overline{x}''_x$	geocentric position vector of pseudostate from the previous MCDRIV iteration
$\Delta V''$	stored value of ΔV
q_d	$\begin{cases} = 0, \text{ no pericynthion passage} \\ = 1, \text{ pericynthion passage} \end{cases}$
q_a	$\begin{cases} = 0, \text{ no apogee passage} \\ = 1, \text{ apogee passage} \end{cases}$
$\check{r}_1$	unit vector to the reentry point (For PTP mode, this vector points to the landing site.)
$\overline{u}_x$	geocentric transearth pseudostate velocity vector
$\overline{v}_a$	postabort velocity vector
t_z	time of landing

Discussion

The iteration to find an overlapped conics solution is initiated by assigning first guess values to three parameters: postpericynthion pseudostate time (t_x), geocentric postpericynthion pseudostate position vector

($\bar{x}_x$), and the fictitious selenocentric abort position vector ($\bar{y}'_a$). If there are no suitable values available from a prior patch, the equations

$$t_x = t_o$$

$$\bar{x}_x = \bar{x}_{mo}$$

$$\bar{y}'_a = \bar{y}_o$$

are used to initialize these quantities, where t_o is the time of abort, $\bar{x}_{mo}$ is the geocentric position vector of the moon at time t_o, and $\bar{y}_o$ is the selenocentric preabort position vector. In addition, the position vector $\bar{x}_x$ is transformed to selenocentric reference

$$\bar{y}_x = \bar{x}_x - \bar{x}_{mx}$$

Routines AESR and FINDUX are used to compute the geocentric velocity vector $\bar{u}_x$ which, when used in conjunction with t_x and $\bar{x}_x$ to define a geocentric conic, will satisfy the target line, $\beta_r = f(u_r)$ as well as the reentry or landing conditions associated with the abort mode.

The velocity vector $\bar{u}_x$ is transformed to selenocentric reference

$$\bar{v}_x = \bar{u}_x - \bar{u}_{mx}$$

and, along with the position vector $\bar{y}'_a$, is input to routine INRFV. This routine computes the elements of a selenocentric conic which passes through $\bar{y}'_a$ and exits the pseudostate transformation sphere (PTS) with velocity $\bar{v}_x$. The elements of the conic define a selenocentric exit position vector ($\bar{y}'_x$) and a postmaneuver velocity vector ($\bar{v}_a$).

Routine PSTATE is called to compute a new time (t_x''') and selenocentric pseudostate position vector ($\bar{y}'_x$) for the next pass through AESR and FINDUX. Also the fictitious abort position vector ($\bar{y}'_a$) supplied to INRFV is updated and the INRFV postabort velocity vector ($\bar{v}_a$) is corrected for earth perturbations.

If the following inequalities

$$|t_x''' - t_x| < \varepsilon_t$$

$$|\bar{y}_x' - \bar{y}_x| \leq \varepsilon_x$$

and

$$\frac{\Delta y_o}{y_o} < \frac{\varepsilon_x}{10.}$$

are satisfied at the end of an iteration, the geocentric and selenocentric conics are sufficiently matched and iterations are ceased. The time and position components, t_x and $\bar{y}_x$, are now redefined to represent the time and position vector of the final INRFV conic at PTS exit. This is done for output and storage purposes.

If the conics have not sufficiently matched, they are tested to see if they are diverging. If so, the exit velocity vector $\bar{v}_x$ is averaged using the equations

$$v_m = \frac{v_x + v_x'}{2}$$

$$\bar{v}_x = \frac{\bar{v}_x + \bar{v}_x'}{2}$$

$$\bar{v}_x = \frac{\bar{v}_x v_m}{|\bar{v}_x|}$$

where $\bar{v}_x'$ is the selenocentric PTS exit velocity vector from the previous iteration. Then the next iteration begins by constructing a new selenocentric conic in INRFV.

If the conics are converging, the initialization parameters are updated by the PSTATE outputs

$$t_x = t_x'''$$

$$\bar{y}_x = \bar{y}'_x$$

and

$$\bar{x}_x = \bar{y}_x + \bar{x}_{mx}$$

and the computations repeated beginning the generation of a new geocentric conic in AESR and FINDUX.

This routine also contains the logic for converging on the PTP landing site. After an overlapped conics solution has converged, the following steps are taken if a PTP solution is desired.

First, routine LNDING is called to determine the transfer angle from reentry to impact (η_{rz1}) as well to update the coordinates (λ_{zo}, μ_{zo}) of the offset site. When FINDUX constructs a conic to a PTP, it does so by assuming that all motion from reentry to landing is planar. However, the reentry and landing routines, RENTRY and LNDING, output a crossrange angle. So, an offset site is used as a FINDUX target in order to account for this nonplanar motion.

Then the transfer angle from reentry to the old offset site (η_{rz} of FINDUX) is compared with η_{rz1} of LNDING. If the relationship

$$\eta_{rz1} - \varepsilon_{ss} \leq \eta_{rz} \leq \eta_{rz1} + \varepsilon_{ss}$$

is satisfied and if the distance between the actual landing site and the desired PTP site is sufficiently small, the PTP convergence is complete for the given day. Control is returned to MCSS with $\Delta T = 24$.

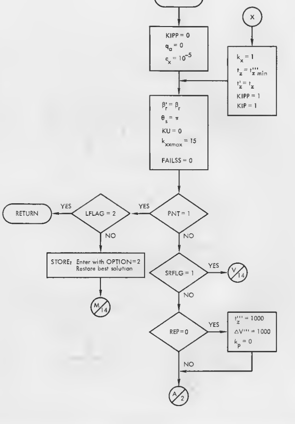
If the impact point has not converged to the PTP site, a linear prediction of the landing time error with respect to the difference in transfer angle is made by computing

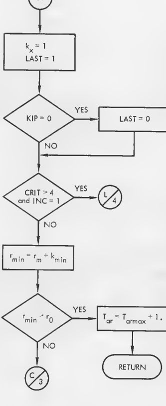
$$\Delta T = \frac{t'_z - t'_{zsav}}{\eta_{rz} - \eta_{sav}} (\eta_{rz1} - \eta_{rz})$$

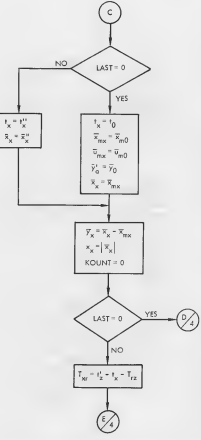
using the landing time (t'_{zsav}) and transfer angle (η_{sav}) from a previous pass as well as the current values of landing time (t'_z), transfer angle

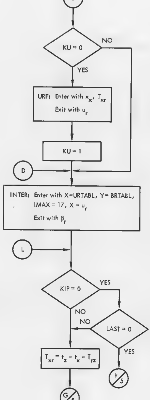
(η_{rz}) to the offset site, and the transfer angle (η_{rzi}) to the actual impact point. This ΔT is returned to MCSS to be used in the next iteration on the site.

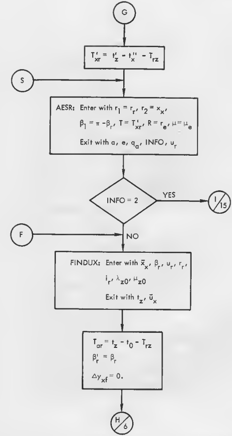
If more than 10 iterations are required for converging on the site, the out-ot-plane component of motion is set to zero and the transfer angle η_{rzi} is frozen in order to expedite convergence.

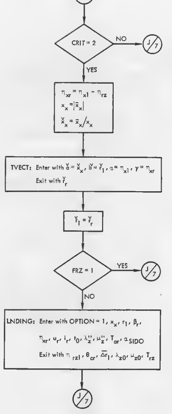

MCDRIV
KIPP = 0
$q_a = 0$
$\epsilon_x = 10^{-5}$
X
$k_x = 1$
$t_z = t'''_{z\,min}$
$t'_z = t_z$
KIPP = 1
KIP = 1
$\beta'_r = \beta_r$
$\theta_s = \pi$
KU = 0
$k_{xxmax} = 15$
FAILSS = 0
PNT = 1
YES
LFLAG = 2
YES
RETURN
NO
NO
STORE: Enter with OPTION=2
Restore best solution
M/14
SRFLG = 1
YES
V/14
NO
REP=0
YES
$t'''_z = 1000$
$\Delta V''' = 1000$
$k_p = 0$
NO
A/2

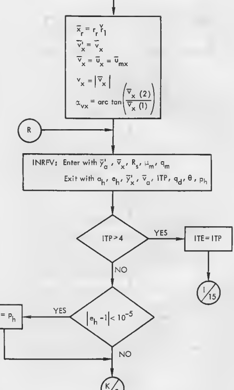

A
$k_x = 1$
LAST = 1
KIP = 0
YES
LAST = 0
NO
CRIT > 4
and INC = 1
YES
L/4
NO
$r_{min} = r_m + k_{min}$
$r_{min} > r_0$
YES
$T_{ar} = T_{armax} + 1.$
RETURN
NO
C/3

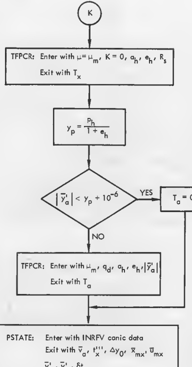

C
LAST = 0
NO
YES
$t_x = t''_x$
$\bar{x}_x = \bar{x}''_x$
$t_x = t_0$
$\bar{x}_{mx} = \bar{x}_{m0}$
$\bar{u}_{mx} = \bar{u}_{m0}$
$\bar{y}'_a = \bar{y}_0$
$\bar{x}_x = \bar{x}_{mx}$
$\bar{y}_x = \bar{x}_x - \bar{x}_{mx}$
$x_x = |\bar{x}_x|$
KOUNT = 0
LAST = 0
YES
D/4
NO
$T_{xr} = t'_z - t_x - T_{rz}$
E/4

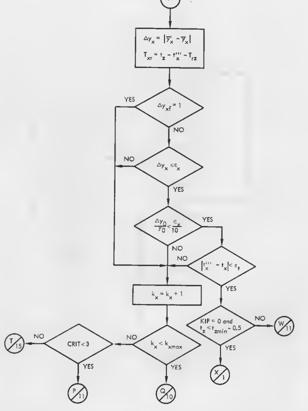

E
KU = 0
NO
YES
URF: Enter with x_x, T_{xr}
Exit with u_r
KU = 1
D
INTER: Enter with X=URTABL, Y= BRTABL,
IMAX = 17, X = u_r
Exit with β_r
L
KIP = 0
YES
NO
NO
LAST = 0
YES
$T_{xr} = t_z - t_x - T_{rz}$
F/5
G/5

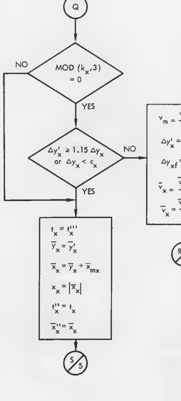

G
$T'_{xr} = t'_z - t''_x - T_{rz}$
S
AESR: Enter with $r_1 = r_r$, $r_2 = x_x$,
$\beta_1 = \pi - \beta_r$, $T = T'_{xr}$, $R = r_e$, $\mu = \mu_e$
Exit with a, e, q_a, INFO, u_r
INFO = 2
YES
I/15
F
NO
FINDUX: Enter with $\bar{x}_x$, β_r, u_r, r_r,
i_r, λ_{z0}, μ_{z0}
Exit with t_z, $\bar{u}_x$
$T_{ar} = t_z - t_0 - T_{rz}$
$\beta'_r = \beta_r$
$\Delta y_{xf} = 0.$
H/6

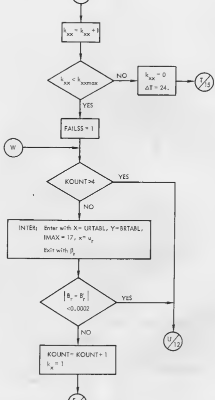

H
CRIT = 2
NO
J/7
YES
$\eta_{xr} = \eta_{x1} - \eta_{rz}$
$x_x = |\bar{x}_x|$
$\check{x}_x = \bar{x}_x / x_x$
TVECT: Enter with $\check{a} = \check{x}_x$, $\check{b} = \check{r}_1$, $\alpha = \eta_{x1}$, $\gamma = \eta_{xr}$
Exit with $\check{r}_r$
$\check{r}_1 = \check{r}_r$
FRZ = 1
YES
J/7
NO
LNDING: Enter with OPTION = 1, x_x, r_1, β_r,
η_{xr}, v_r, i_r, t_0, λ_z'', μ_z'', T_{ar}, α_{SIDO}
Exit with η_{rz1}, θ_{cr}, $\overline{\Delta r}_1$, λ_{z0}, μ_{z0}, T_{rz}
J/7

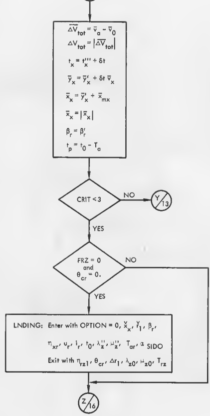

J
$\bar{x}_r = r_r \check{r}_1$
$\bar{v}'_x = \bar{v}_x$
$\bar{v}_x = \bar{\upsilon}_x = \bar{\upsilon}_{mx}$
$v_x = |\bar{v}_x|$
$\alpha_{vx} = \arctan\left(\frac{\bar{v}_x(2)}{\bar{v}_x(1)}\right)$
R
INRFV: Enter with $\bar{y}'_a$, $\bar{v}_x$, R_s, μ_m, q_m
Exit with a_h, e_h, $\bar{y}'_x$, $\bar{v}_a$, ITP, q_d, θ, p_h
ITP > 4
YES
ITE = ITP
NO
I/15
$|e_h - 1| < 10^{-5}$
YES
$a_h = p_h$
NO
K/8

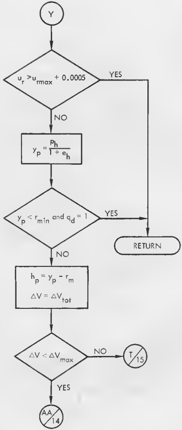

K
TFPCR: Enter with μ = μm, K = 0, ah, eh, Rs
Exit with Tx
yp = ph / (1 + eh)
|ȳ'a| < yp + 10^-6
YES
Ta = 0.
NO
TFPCR: Enter with μm, qd, ah, eh, |ȳ'a|
Exit with Ta
PSTATE: Enter with INRFV conic data
Exit with v̄a, t'''x, Δy0, x̄mx, ūmx
ȳ'a, ȳ'x, δt
N/9

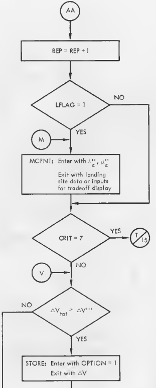

N
$\Delta y_x = |\bar{y}'_x - \bar{y}_x|$
$T_{xr} = t_z - t'''_x - T_{rz}$
YES
$\Delta y_{xf} = 1$
NO
NO
$\Delta y_x \leq \varepsilon_x$
YES
$\frac{\Delta y_0}{y_0} < \frac{\varepsilon_x}{10}$
YES
NO
NO
$|t'''_x - t_x| < \varepsilon_t$
YES
$k_x = k_x + 1$
KIP = 0 and
$t_z < t_{zmin} - 0.5$
NO
W 11
T 15
NO
CRIT<3
NO
$k_x < k_{xmax}$
YES
YES
YES
X 1
P 11
Q 10

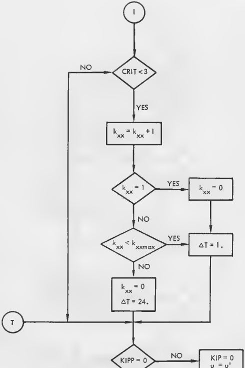

Q
MOD $(k_x, 3) = 0$
NO
YES
$\Delta y'_x \geq 1.15\ \Delta y_x$ or $\Delta y_x < \epsilon_x$
NO
YES
$v_m = \frac{v_x + v'_x}{2.}$
$\Delta y'_x = 200.$
$\Delta y_{xf} = 1.$
$\bar{v}_x = \frac{\bar{v}_x + \bar{v}_x'}{2.}$
$\bar{v}_x = \frac{\bar{v}_x v_m}{|\bar{v}_x|}$
R/7
$t_x = t'''_x$
$\bar{y}_x = \bar{y}'_x$
$\bar{x}_x = \bar{y}_x + \bar{x}_{mx}$
$x_x = |\bar{x}_x|$
$t''_x = t_x$
$\bar{x}''_x = \bar{x}_x$
S/5

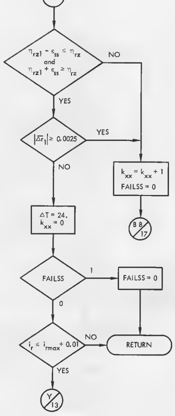

P
$k_{xx} = k_{xx} + 1$
$k_{xx} < k_{xxmax}$
NO
$k_{xx} = 0$
$\Delta T = 24.$
T/15
YES
FAILSS = 1
W
KOUNT >4
YES
NO
INTER: Enter with X = URTABL, Y = BRTABL,
IMAX = 17, x = υ_r
Exit with β_r
$|B_r - B'_r|$
<0.0002
YES
NO
U/12
KOUNT = KOUNT + 1
$k_x = 1$
S/5

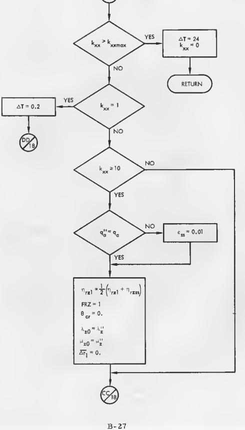

U
$\overline{\Delta V}_{tot} = \bar{v}_a - \bar{v}_0$
$\Delta V_{tot} = |\overline{\Delta V}_{tot}|$
$t_x = t_x''' + \delta t$
$\bar{y}_x = \bar{y}_x' + \delta t\, \bar{v}_x$
$\bar{x}_x = \bar{y}_x' + \bar{x}_{mx}$
$\bar{x}_x = |\bar{x}_x|$
$\beta_r = \beta_r'$
$t_p = t_0 - T_a$
CRIT < 3
NO
Y/13
YES
FRZ = 0 and $\theta_{cr} = 0.$
NO
YES
LNDING: Enter with OPTION = 0, $\check{x}_x$, $\check{r}_1$, β_r, η_{xr}, υ_r, i_r, t_0, λ_z'', μ_z'', T_{ar}, α_{SIDO}
Exit with η_{rz1}, θ_{cr}, Δr_1, λ_{z0}, μ_{z0}, T_{rz}
Z/16

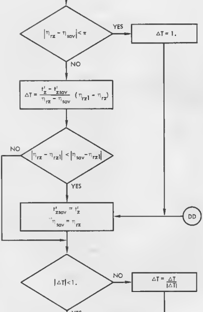

Y
$v_r > v_{rmax} + 0.0005$
YES
NO
$y_p = \frac{p_h}{1 + e_h}$
$y_p < r_{min}$ and $q_d = 1$
YES
RETURN
NO
$h_p = y_p - r_m$
$\Delta V = \Delta V_{tot}$
$\Delta V < \Delta V_{max}$
NO
T/15
YES
AA/14


AA
REP = REP + 1
LFLAG = 1
NO
YES
M
MCPNT: Enter with λz'', μz''
Exit with landing site data or inputs for tradeoff display
CRIT = 7
YES
T/15
NO
V
NO
ΔVtot > ΔV'''
YES
STORE: Enter with OPTION = 1
Exit with ΔV
T/15


I
NO
CRIT < 3
YES
$k_{xx} = k_{xx} + 1$
$k_{xx} = 1$
YES
$k_{xx} = 0$
NO
$k_{xx} < k_{xxmax}$
YES
$\Delta T = 1.$
NO
$k_{xx} = 0$
$\Delta T = 24.$
T
KIPP = 0
NO
KIP = 0
$\upsilon_r = \upsilon_r'$
YES
RETURN


Z
$\eta_{rz1} - \epsilon_{ss} \leq \eta_{rz}$ and $\eta_{rz1} + \epsilon_{ss} \geq \eta_{rz}$
NO
YES
$|\overline{\Delta r}_1| \geq 0.0025$
YES
NO
$k_{xx} = k_{xx} + 1$
FAILSS = 0
BB/17
$\Delta T = 24.$
$k_{xx} = 0$
FAILSS
1
FAILSS = 0
0
$i_r \leq i_{rmax} + 0.01$
NO
RETURN
YES
Y/13


BB
$k_{xx} > k_{xxmax}$
YES
$\Delta T = 24$
$k_{xx} = 0$
RETURN
NO
$k_{xx} = 1$
YES
$\Delta T = 0.2$
DD 18
NO
$k_{xx} \geq 10$
NO
YES
$q_a'' = q_a$
NO
$\epsilon_{ss} = 0.01$
YES
$\eta_{rz1} = \frac{1}{2}\left(\eta_{rz1} + \eta_{rzss}\right)$
FRZ = 1
$\theta_{cr} = 0.$
$\lambda_{z0} = \lambda_z''$
$\mu_{z0} = \mu_z''$
$\overline{\Delta r_1} = 0.$
CC 18


CC
$|\eta_{rz} - \eta_{sav}| < \pi$
YES
$\Delta T = 1.$
NO
$\Delta T = \frac{t'_z - t'_{zsav}}{\eta_{rz} - \eta_{sav}} (\eta_{rz1} - \eta_{rz})$
NO
$|\eta_{rz} - \eta_{rz1}| < |\eta_{sav} - \eta_{rz1}|$
YES
$t'_{zsav} = t'_z$
$\eta_{sav} = \eta_{rz}$
DD
$|\Delta T| < 1.$
NO
$\Delta T = \frac{\Delta T}{|\Delta T|}$
YES
$\eta_{rzss} = \eta_{rz1}$
$q''_a = q_a$
RETURN

B-2 FINDUX

Purpose

This routine computes the geocentric conic from the geocentric pseudostate to reentry for all moon-referenced modes.

Input

μ_e	earth gravitational constant
β_r	reentry flight-path angle
i_{rmax}	maximum return inclination allowed
r_r	reentry radius
T_{rz}	time from reentry to landing
α_{SIDO}	right ascension at time zero
i_r	return inclination
q_a	= 0, no apogee passage = 1, apogee passage
INTER	= -1, A_{zr} < reentry azimuth for $i_r = \delta_x$ = +1, A_{zr} > reentry azimuth for $i_r = \delta_x$
k_x	pseudostate iteration counter
u_r	reentry velocity
$\overline{x}_x$	geocentric pseudostate position vector
x_x	magnitude of $\overline{x}_x$
λ_z''	longitude of the PTP site
μ_z''	latitude of the PTP site
t_x	postpericynthion pseudostate time - either time of the maneuver or time of pericynthion passage

CRIT	= 2, PTP mode = 4, ATP mode = 6, UA mode = 7, solution not to be considered in MCDRIV optimization
ω_e	rotational rate of the earth

Output

a	semimajor axis of geocentric conic
e	eccentricity of geocentric conic
i_r	return inclination
t_z	landing time
η_{x1}	the transfer angle from the pseudostate to reentry; if CRIT = 2, the angle from the pseudostate to the landing site
$\bar{r}_1$	a unit vector pointing to the reentry point, if CRIT = 2, the vector points to the landing site
$\bar{u}_x$	geocentric pseudostate velocity vector
NIR	= 0, input i_r is greater than the declination of the pseudostate = 1, input i_r is less than the declination of the pseudostate
η_{rz}	transfer angle from reentry to landing
SOL	= 0, no solution found = 1, solution found

Discussion

The output time of landing (t_z) is found by determining in TFPCR the time from the pseudostate to perigee (T_x) and the time from perigee to reentry (T_r). From this, the time of landing can be computed as

$$t_z = t_x + T_x - T_r + T_{rz}$$

The true anomalies at the pseudostate and at reentry are computed using the equation

$$\eta = \arccos \left(\frac{\frac{p}{r} - 1.}{e} \right)$$

where

p = semilatus rectum of the conic

r = magnitude of the position vector

Subtracting the two true anomalies gives η_{xr}, the angle from the pseudostate to reentry.

The next computations depend on the input mode. If the PTP mode is used, then the unit vector toward the landing site is computed by

$$\check{r}_1 = (\cos \alpha_z \cos \mu_z, \sin \alpha_z \cos \mu_z, \sin \mu_z)$$

where α_z is the right ascension of the landing site at time t_z and is computed by

$$\alpha_z = \omega_e t_z + \lambda_z + \alpha_{SIDO}$$

The transfer angle from the pseudostate to landing can now also be computed using the equation

$$\eta_{xz} = \arccos (\sin \delta_x \sin \mu_z + \cos (\alpha_z - \alpha_x) \cos \delta_x \cos \mu_z)$$

where

δ_x = declination of the pseudostate

α_x = right ascension of the pseudostate

If $\sin (\alpha_z - \alpha_x)$ is negative, $\eta_{xz} = 2\pi - \eta_{xz}$. The reentry maneuver angle can be determined by subtracting η_{xr} from η_{xz}. Return inclination is computed using the equation

$$\cos i_r = \frac{\cos \delta_x \cos \mu_z \sin (\alpha_z - \alpha_x)}{\sin \eta_{xz}}$$

If the mode is not PTP, the unit vector toward reentry is computed as

$$\check{r}_1 = (\cos \alpha_r \cos \delta_r, \ \sin \alpha_r \cos \delta_r, \ \sin \alpha_r)$$

where

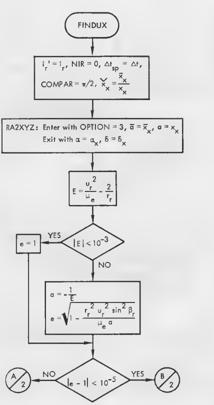
δ_r = right ascension at reentry

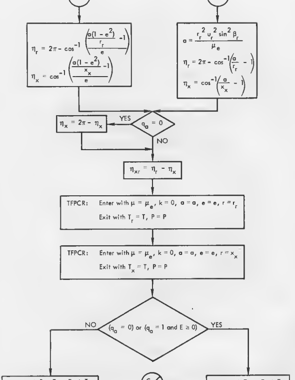
δ_r = reentry declination

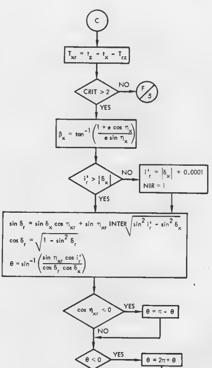
Finally, for either input mode, the geocentric pseudostate velocity vector is computed. The unit vector, $\check{u}_x$, can be found from the routine TVECT, and the magnitude is found using the formula

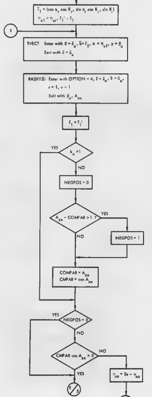
$$u_x = \sqrt{u_r^2 - 2\mu_e\left(\frac{1}{r_r} - \frac{1}{x_x}\right)}$$

which is derived from the vis viva integral.


FINDUX
$i_r' = i_r$, NIR = 0, $\Delta t_{sp} = \Delta t$,
COMPAR = $\pi/2$, $\check{x}_x = \frac{\bar{x}_x}{x_x}$
RA2XYZ: Enter with OPTION = 3, $\bar{a} = \bar{x}_x$, $a = x_x$
Exit with $\alpha = \alpha_x$, $\delta = \delta_x$
$E = \frac{v_r^2}{\mu_e} - \frac{2}{r_r}$
$|E| < 10^{-3}$
YES
$e = 1$
NO
$a = -\frac{1}{E}$
$e = \sqrt{1 - \frac{r_r^2 v_r^2 \sin^2 \beta_r}{\mu_e a}}$
$|e - 1| < 10^{-5}$
NO
A/2
YES
B/2


A
B
$\eta_r = 2\pi - \cos^{-1}\left(\frac{\frac{a(1-e^2)}{r_r} - 1}{e}\right)$
$\eta_x = \cos^{-1}\left(\frac{\frac{a(1-e^2)}{x_x} - 1}{e}\right)$
$a = \frac{r_r^2 v_r^2 \sin^2 \beta_r}{\mu_e}$
$\eta_r = 2\pi - \cos^{-1}\left(\frac{a}{r_r} - 1\right)$
$\eta_x = \cos^{-1}\left(\frac{a}{x_x} - 1\right)$
$q_a = 0$
YES
NO
$\eta_x = 2\pi - \eta_x$
$\eta_{xr} = \eta_r - \eta_x$
TFPCR: Enter with $\mu = \mu_e$, $k = 0$, $a = a$, $e = e$, $r = r_r$
Exit with $T_r = T$, $P = P$
TFPCR: Enter with $\mu = \mu_e$, $k = 0$, $a = a$, $e = e$, $r = x_x$
Exit with $T_x = T$, $P = P$
$(q_a = 0)$ or $(q_a = 1$ and $E \geq 0)$
NO
YES
$t_z = t_x + P - T_x - T_r + T_{rz}$
$t_z = t_x + T_x - T_r + T_{rz}$
C/3


C
$T_{xr} = t_z - t_x - T_{rz}$
CRIT > 2
NO
F
5
YES
$\beta_x = \tan^{-1}\left(\frac{1 + e \cos \eta_x}{e \sin \eta_x}\right)$
$i'_r > |\delta_x|$
NO
$i'_r = |\delta_x| + 0.0001$
NIR = 1
YES
$\sin \delta_r = \sin \delta_x \cos \eta_{xr} + \sin \eta_{xr}$ INTER $\sqrt{\sin^2 i'_r - \sin^2 \delta_x}$
$\cos \delta_r = \sqrt{1 - \sin^2 \delta_r}$
$\theta = \sin^{-1}\left(\frac{\sin \eta_{xr} \cos i'_r}{\cos \delta_r \cos \delta_x}\right)$
$\cos \eta_{xr} < 0$
YES
$\theta = \pi - \theta$
NO
$\theta < 0$
YES
$\theta = 2\pi + \theta$
NO
$\alpha_r = \alpha_x + \theta$
D
4


D
$\bar{r}_1 = (\cos \alpha_r \cos \delta_r, \sin \alpha_r \cos \delta_r, \sin \delta_r)$
$\eta_{x1} = \eta_{xr}, \bar{r}_1' - \bar{r}_1$
I
TVECT Enter with $\bar{a} = \bar{x}_x, \bar{b} = \bar{r}_1, \alpha = \eta_{x1}, \gamma = \beta_x$
Exit with $\bar{c} = \bar{u}_x$
RA2XYZ Enter with OPTION = 4, $\bar{r} = \bar{x}_x, \bar{v} = \bar{u}_x$,
$r = 1, v = 1$
Exit with β_x, A_{zx}
$\bar{r}_1 = \bar{r}_1'$
YES
$k_x > 1$
NO
NEGPOS = 0
A_{zx} - COMPAR > 1 ?
YES
NO
NEGPOS = 1
COMPAR = A_{zx}
CMPAR = cos A_{zx}
YES
NEGPOS = 0
NO
CMPAR cos A_{zx} > 0
NO
YES
$\eta_{xz} = 2\pi - \eta_{xz}$
E/5
H/6

(E)

$$\bar{\upsilon}_x = \check{\upsilon}_x \sqrt{\upsilon_r^2 - 2\mu_e\left(\frac{1}{r_r} - \frac{1}{x_x}\right)}$$

RETURN

(F)

$$\alpha_z = \omega_e t_z + \lambda_z'' + \alpha_{SIDO}$$

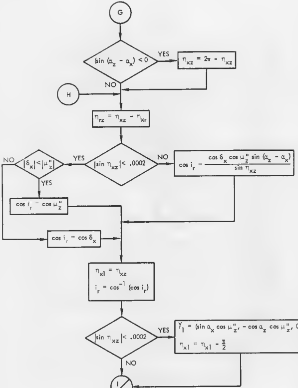
$$\check{r}_1 = (\cos\alpha_z \cos\mu_z'', \sin\alpha_z \cos\mu_z'', \sin\mu_z'')$$

$$\beta_x = \tan^{-1}\left(\frac{1 + e\cos\eta_x}{e\sin\eta_x}\right)$$

$$\eta_{xz} = \cos^{-1}(\sin\delta_x \sin\mu_z'' + \cos(\alpha_z - \alpha_x)\cos\delta_x \cos\mu_z'')$$

$$\check{R}_1' = \check{r}_1$$

(G/6)


G
$(\sin(\alpha_z - \alpha_x) < 0$
YES
$\eta_{xz} = 2\pi - \eta_{xz}$
NO
H
$\eta_{rz} = \eta_{xz} - \eta_{xr}$
$|\sin \eta_{xz}| < .0002$
YES
NO
$\cos i_r = \frac{\cos \delta_x \cos \mu''_z \sin(\alpha_z - \alpha_x)}{\sin \eta_{xz}}$
$|\delta_x| < |\mu''_z|$
NO
YES
$\cos i_r = \cos \mu''_z$
$\cos i_r = \cos \delta_x$
$\eta_{x1} = \eta_{xz}$
$i_r = \cos^{-1}(\cos i_r)$
$|\sin \eta_{xz}| < .0002$
YES
$\bar{Y}_1 = (\sin \alpha_x \cos \mu''_z, -\cos \alpha_z \cos \mu''_z, 0)$
$\eta_{x1} = \eta_{x1} - \frac{\pi}{2}$
NO
I/4

B-3 AESR

Purpose

This routine is used to compute the semimajor axis and eccentricity for a given flight-path angle, time of flight, and two radial distances. The routine requires a tolerance (ε) to be set by the calling program.

Input

r_1	first radial distance
r_2	second radial distance
β_1	flight-path angle for first radius
T	flight time
R	radius of the central body
μ	gravitational constant
ε	tolerance on time

Output

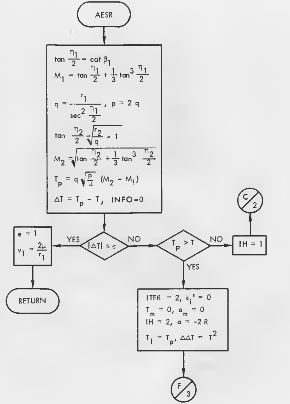
a	semimajor axis; or p, the semilatus rectum
e	eccentricity
k_2	= 0, no apogee passage ≠ 0, apogee passage
INFO	= 0, no solution found ≠ 0, solution found
V_1	reentry velocity

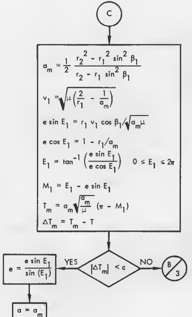
Discussion

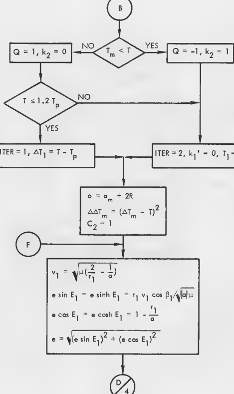
There is no closed-form solution to this problem, so an iterative technique is used. The first guess is made by assuming there is a parabolic trajectory between r_2 and r_1. The flight time of such a trajectory is computed and compared with the input flight time (T). If the input flight time is less than the parabolic flight time, the trajectory is hyperbolic. If the input flight time is greater than the parabolic flight time, the trajectory is elliptic.

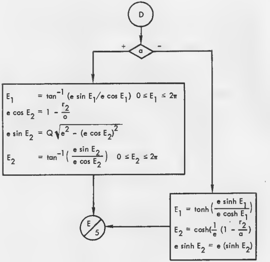
An elliptic trajectory may or may not require apogee passage. This is determined by comparing the flight time obtained by assuming r_2 is at apogee with the input flight time.

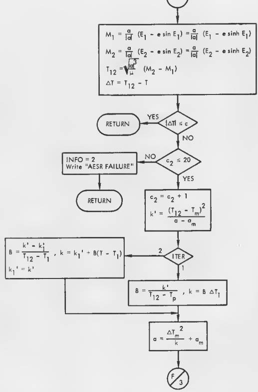
The flight time between r_2 and r_1 is computed by assuming a value for the semimajor axis. The computed time is then compared with the input flight time, and if the two are within the given tolerance, the iteration is complete. If not, the semimajor axis is adjusted, and a new flight time is computed.


AESR
tan η1/2 = cot β1
M1 = tan η1/2 + 1/3 tan^3 η1/2
q = r1 / sec^2 η1/2 , p = 2 q
tan η2/2 = √(r2/q - 1)
M2 = √(tan η2/2 + 1/3 tan^3 η2/2)
Tp = q √(p/μ) (M2 - M1)
ΔT = Tp - T, INFO=0
|ΔT| ≤ ε
YES
NO
e = 1
v1 = 2μ/r1
RETURN
Tp > T
NO
IH = 1
C/2
YES
ITER = 2, ki' = 0
Tm = 0, am = 0
IH = 2, a = -2 R
T1 = Tp, ΔΔT = T^2
F/3


C
$a_m = \frac{1}{2} \frac{r_2^2 - r_1^2 \sin^2 \beta_1}{r_2 - r_1 \sin^2 \beta_1}$
$v_1 = \sqrt{\mu \left(\frac{2}{r_1} - \frac{1}{a_m}\right)}$
$e \sin E_1 = r_1 v_1 \cos \beta_1 / \sqrt{a_m \mu}$
$e \cos E_1 = 1 - r_1 / a_m$
$E_1 = \tan^{-1} \left(\frac{e \sin E_1}{e \cos E_1}\right) \quad 0 \le E_1 \le 2\pi$
$M_1 = E_1 - e \sin E_1$
$T_m = a_m \sqrt{\frac{a_m}{\mu}} (\pi - M_1)$
$\Delta T_m = T_m - T$
$|\Delta T_m| < \epsilon$
YES
NO
B/3
$e = \frac{e \sin E_1}{\sin (E_1)}$
$a = a_m$
RETURN


B
$T_m < T$
NO
YES
$Q = 1, k_2 = 0$
$Q = -1, k_2 = 1$
$T \le 1.2\,T_p$
NO
YES
$ITER = 1, \Delta T_1 = T - T_p$
$ITER = 2, k_1' = 0, T_1 = T_p$
$a = a_m + 2R$
$\Delta\Delta T_m = (\Delta T_m - T)^2$
$C_2 = 1$
F
$v_1 = \sqrt{\mu(\frac{2}{r_1} - \frac{1}{a})}$
$e \sin E_1 = e \sinh E_1 = r_1 v_1 \cos \beta_1 / \sqrt{|a|\mu}$
$e \cos E_1 = e \cosh E_1 = 1 - \frac{r_1}{a}$
$e = \sqrt{(e \sin E_1)^2 + (e \cos E_1)^2}$
D
4


D
a
+
-
$E_1 = \tan^{-1} (e \sin E_1 / e \cos E_1) \quad 0 \le E_1 \le 2\pi$
$e \cos E_2 = 1 - \frac{r_2}{o}$
$e \sin E_2 = Q \sqrt{e^2 - (e \cos E_2)^2}$
$E_2 = \tan^{-1} \left(\frac{e \sin E_2}{e \cos E_2} \right) \quad 0 \le E_2 \le 2\pi$
$E_1 = \text{tonh} \left(\frac{e \sinh E_1}{e \cosh E_1} \right)$
$E_2 = \cosh\left(\frac{1}{e} \left(1 - \frac{r_2}{a}\right)\right)$
$e \sinh E_2 = e (\sinh E_2)$
E
5


E
M1 = a/|a| (E1 - e sin E1) = a/|a| (E1 - e sinh E1)
M2 = a/|a| (E2 - e sin E2) = a/|a| (E2 - e sinh E2)
T12 = √(|a|^3/μ) (M2 - M1)
ΔT = T12 - T
YES
RETURN
|ΔT| ≤ ε
NO
INFO = 2
Write "AESR FAILURE"
NO
c2 ≤ 20
YES
RETURN
c2 = c2 + 1
k' = (T12 - Tm)^2 / (a - am)
B = (k' - k1') / (T12 - T1) , k = k1' + B(T - T1)
k1' = k'
2
ITER
1
B = k' / (T12 - Tp) , k = B ΔT1
a = ΔTm^2 / k + am
F
3

B-4 INRFV

Purpose

This routine constructs a conic, given an initial position vector and a velocity vector at a distance from the occupied focus. The solution is found by solving a quartic equation for sin β_2.

Input

$\overline{r}_1$	initial position vector
$\overline{v}_2$	velocity vector at the second position
r_2	magnitude of the second position vector
μ	gravitational constant
k_3	= 0, noncircumlunar or retrograde motion as specified by QDFLG = 1, circumlunar motion
$\pi/2$	$\pi/2$
π	π
2π	2π
k_x	pseudostate iteration counter
QDFLG	= 0, use circumlunar or noncircumlunar motion = 1, use retrograde motion
A_0	premaneuver azimuth

Output

a	semimajor axis of the conic
e	eccentricity of the conic
θ	difference in true anomaly from $\overline{r}_1$ to $\overline{r}_2$
$\overline{r}_2$	position vector of second point
$\overline{v}_1$	velocity vector at first position

k_1	$\begin{cases} = 0, \text{ no pericenter passage} \\ = 1, \text{ pericenter passage} \end{cases}$
p	semilatus rectum of the conic
I	counts the roots from the quartic tested; error flag if $I > 4$
Ψ	angle from $\bar{r}_1$ to $\bar{v}_2$ measured in direction of motion
q_m	$\begin{cases} = 0, \text{ noncircumlunar motion} \\ = 1, \text{ circumlunar motion} \end{cases}$

Discussion

Given an initial position vector and a velocity vector at a second position, a closed-form solution for a conic is found by solving a quartic equation for $\sin \beta_2$. The polar equation is

$$\frac{p}{r_2} = \frac{1 - \cos \theta}{\frac{r_2}{r_1} - \cos \theta - \sin \theta \cot \beta_2} \qquad (1)$$

where

$$\beta_2 = \frac{\pi}{2} - \gamma_2$$

The angle Ψ is defined as

$$\Psi = \cos^{-1}\left(\frac{\bar{r}_1 \cdot \bar{v}_2}{r_1 \quad v_2}\right)$$

Then

$$\theta = \Psi - \beta_2$$

and from the angular momentum vector equations

$$h = \sqrt{p\mu} = r \; v \sin \beta$$

it can be written

$$\sin \beta_2 = \sqrt{\frac{\mu p}{r_2^2 v_2^2}}$$

$$\cos \beta_2 = \sqrt{1 - \frac{\mu p}{r_2^2 v_2^2}} \qquad (r_2 v_2 > 0).$$

Substituting into Equation (1) and rearranging gives

$$p = \frac{r_1 r_2 \sqrt{\frac{\mu p}{r_2^2 v_2^2}} - r_1 r_2 \sin \psi \left(\frac{\mu p}{r_2^2 v_2^2}\right) - r_1 r_2 \cos \psi \sqrt{1 - \frac{\mu p}{r_2^2 v_2^2}} \sqrt{\frac{\mu p}{r_2^2 v_2^2}}}{r_2 \sqrt{\frac{\mu p}{r_2^2 v_2^2}} - r_1 \sin \psi} \qquad (2)$$

It is possible to cast this equation into the quartic form. Let

$$X = \sqrt{\frac{\mu p}{r_2^2 v_2^2}} = \sin \beta_2$$

$$\beta' = \frac{r_2^2 v_2^2}{\mu}$$

Substituting into Equation (2), rearranging, and squaring yields

$$X^4 (K^2 r_2^2) + X^3 (2K r_1 r_2^2 \sin \psi - 2K^2 r_1 r_2 \sin \psi) + X^2 (K^2 r_1^2 \sin^2 \psi$$
$$+ r_1^2 r_2^2 \sin^2 \psi + r_1^2 r_2^2 \cos^2 \psi - 2K r_1 r_2 - 2K r_1^2 r_2 \sin^2 \psi)$$
$$+ X(2K r_1^2 r_2 \sin \psi - 2 r_1^2 r_2^2 \sin \psi) + r_1^2 r_2^2 - r_1^2 r_2^2 \cos^2 \psi = 0$$

The coefficients in this quartic may be written as

$$AX^4 + BX^3 + CX^2 + DX + E = 0$$

where

$$A = \beta'^2 r_2^2$$

$$B = 2\beta' \sin\psi\, r_1 r_2 (r_2 - \beta')$$

$$C = r_1^2 r_2^2 + \beta' \sin^2\psi\, r_1^2(\beta' - 2r_2) - 2r_1 r_2^2 \beta'$$

$$D = -2 \sin\psi\, r_1^2 r_2 (r_2 - \beta')$$

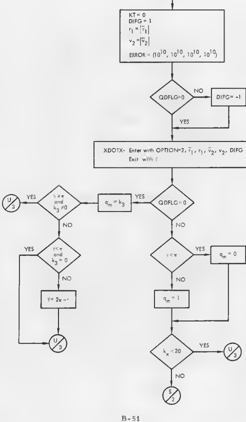
$$E = r_1^2 r_2^2 \sin^2\psi$$

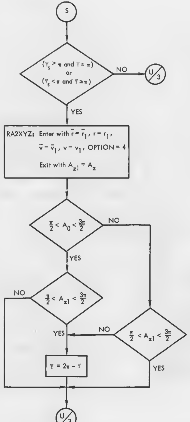
Since $X = \sin\beta_2$, all roots not between 0 and 1 may be discarded. The desired root may be chosen by

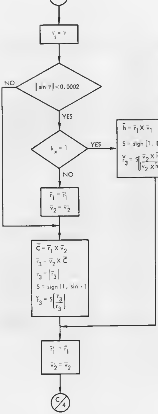
$$\theta = \psi - \sin^{-1} X$$

$$\frac{X^2 r_2 v_2^2}{\mu} - \frac{1 - \cos\theta}{\frac{r_2}{r_1} - \cos\theta - \sin\theta\sqrt{\frac{1 - X^2}{X}}} < \epsilon$$

When the proper root has been determined, the desired outputs may be computed using standard conic formulas.


INRFV
KT = 0
DIFG = 1
$r_1 = |\bar{r}_1|$
$v_2 = |\bar{v}_2|$
ERROR = (10^{10}, 10^{10}, 10^{10}, 10^{10})
QDFLG=0
NO
DIFG= -1
YES
XDOTX: Enter with OPTION=2, $\bar{r}_1$, r_1, $\bar{v}_2$, v_2, DIFG
Exit with ψ
QDFLG = 0
YES
$q_m = k_3$
$\psi \geq \pi$ and $k_3 \neq 0$
YES
U 3
NO
$\psi < \pi$ and $k_3 = 0$
YES
NO
$\psi = 2\pi - \psi$
U 3
NO
$\psi < \pi$
YES
$q_m = 0$
NO
$q_m = 1$
$k_x < 20$
YES
U 3
NO
S 2


S
($\Psi_s > \pi$ and $\Psi \leq \pi$)
or
($\Psi_s < \pi$ and $\Psi \geq \pi$)
NO
U/3
YES
RA2XYZ: Enter with $\bar{r} = \bar{r}_1$, $r = r_1$,
$\bar{v} = \bar{v}_1$, $v = v_1$, OPTION = 4
Exit with $A_{z1} = A_z$
$\frac{\pi}{2} < A_0 < \frac{3\pi}{2}$
NO
YES
NO
$\frac{\pi}{2} < A_{z1} < \frac{3\pi}{2}$
YES
NO
$\frac{\pi}{2} < A_{z1} < \frac{3\pi}{2}$
$\Psi = 2\pi - \Psi$
YES
U/3


U
$\gamma_s = \gamma$
NO
$|\sin \gamma| < 0.0002$
YES
$k_x = 1$
YES
NO
$\bar{h} = \bar{r}_1 \times \bar{v}_1$
S = sign [1, DIFG (h (3))]
$\check{r}_3 = S\left[\frac{\bar{v}_2 \times \bar{h}}{|\bar{v}_2 \times \bar{h}|}\right]$
$\bar{r}_1 = \bar{r}_1'$
$\bar{v}_2 = \bar{v}_2'$
$\bar{C} = \bar{r}_1 \times \bar{v}_2$
$\bar{r}_3 = \bar{v}_2 \times \bar{C}$
$r_3 = |\bar{r}_3|$
S = sign [1, sin ·]
$\check{r}_3 = S\left[\frac{\bar{r}_3}{r_3}\right]$
$\bar{r}_1' = \bar{r}_1$
$\bar{v}_2' = \bar{v}_2$
C/4

C

$$\beta' = \frac{r_2^2 v_2^2}{\mu}$$

$$A = \beta'^2 r_2^2$$

$$B = 2\beta' \sin \Psi r_1 r_2 (r_2 - \beta')$$

$$C = r_1^2 r_2^2 + \beta' \sin^2 \Psi r_1^2 (\beta' - 2r_2) - 2r_1 r_2^2 \beta'$$

$$D = -2 \sin \Psi r_1^2 r_2 (r_2 - \beta')$$

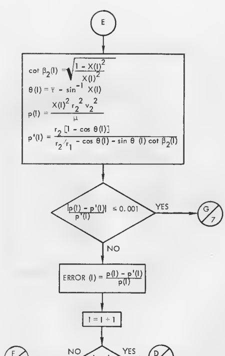
$$E = r_1^2 r_2^2 \sin^2 \Psi$$

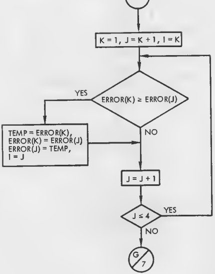
DROOTS: Enter with A, B, C, D, E, N = 4
Exit with $X(1) = x_1$, $X(2) = x_2$, $X(3) = x_3$, $X(4) = x_4$, M, I

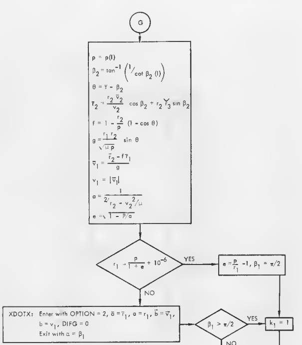
I = 4?

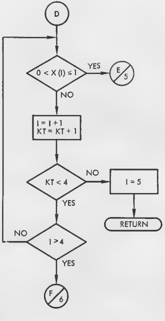
YES: I = 5 → RETURN

NO: I = 1 → D/8


E
cot β2(I) = √((1 - X(I)²) / X(I)²)
θ(I) = Ψ - sin⁻¹ X(I)
p(I) = X(I)² r2² v2² / μ
p'(I) = r2 [1 - cos θ(I)] / (r2/r1 - cos θ(I) - sin θ(I) cot β2(I))
|p(I) - p'(I)| / p'(I) ≤ 0.001
YES
G 7
NO
ERROR (I) = (p(I) - p'(I)) / p(I)
I = I + 1
NO
I ≤ 4
YES
F 6
D 8


F
K = 1, J = K + 1, I = K
YES
ERROR(K) ≥ ERROR(J)
TEMP = ERROR(K),
ERROR(K) = ERROR(J)
ERROR(J) = TEMP,
I = J
NO
J = J + 1
J ≤ 4
YES
NO
G
7


G
$p = p(I)$
$\beta_2 = \tan^{-1}\left(1/\cot\beta_2(I)\right)$
$\theta = \Psi - \beta_2$
$\bar{r}_2 = \frac{r_2 \bar{v}_2}{v_2} \cos\beta_2 + r_2 \check{r}_3 \sin\beta_2$
$f = 1 - \frac{r_2}{p}(1 - \cos\theta)$
$g = \frac{r_1 r_2}{\sqrt{\mu p}} \sin\theta$
$\bar{v}_1 = \frac{\bar{r}_2 - f\bar{r}_1}{g}$
$v_1 = |\bar{v}_1|$
$a = \frac{1}{2/r_2 - v_2^2/\mu}$
$e = \sqrt{1 - p/a}$
$r_1 < \frac{p}{1+e} + 10^{-6}$
YES
$e = \frac{p}{r_1} - 1$, $\beta_1 = \pi/2$
NO
XDOTX: Enter with OPTION = 2, $\bar{a} = \bar{r}_1$, $a = r_1$, $\bar{b} = \bar{v}_1$, $b = v_1$, DIFG = 0
Exit with $\alpha = \beta_1$
$\beta_1 > \pi/2$
YES
$k_1 = 1$
NO
$k_1 = 0$
RETURN


D
0 < X (I) ≤ 1
YES
E
5
NO
I = I + 1
KT = KT + 1
KT < 4
NO
I = 5
RETURN
YES
NO
I > 4
YES
F
6

B-5 PSTATE

Purpose

This routine updates the time and position components of a previously estimated postpericynthion pseudostate vector that defines the transearth geocentric conic. It also updates the fictitious abort position vector supplied to INRFV, and corrects the INRFV postabort velocity vector to account for earth perturbations.

Input

k_x	pseudostate iteration counter
a_h	semimajor axis of selenocentric conic
e_h	eccentricity of selenocentric conic
p_h	semilatus rectum of selenocentric conic
t_0	time of abort
T_x	flight time from pericynthion to PTS exit
$\bar{x}_{mo}$	geocentric position vector of the moon at time t_0
$\bar{u}_{mo}$	geocentric velocity vector of the moon at time t_0
$\bar{y}_0$	selenocentric preabort position vector
$\bar{y}_a'$	fictitious abort position vector used in last call to INRFV
$\bar{v}_a$	selenocentric postabort velocity vector
$\bar{v}_x$	selenocentric PTS exit velocity vector
v_x	magnitude of $\bar{v}_x$
$\bar{y}_x'$	selenocentric PTS exit position vector
θ	transfer angle from $\bar{y}_a'$ to $\bar{y}'_x$
R_s	radius of the PTS
μ_e	earth gravitational parameter

μ_m	moon gravitational parameter
β_a	postabort flight-path angle
β_x	PTS exit flight-path angle of selenocentric conic
T_a	flight time from pericynthion to the maneuver position

Output

$\bar{v}_a$	corrected postabort velocity vector
t_x'''	postpericynthion pseudostate transformation time for next iteration
$\bar{y}_x'$	selenocentric postpericynthion position vector for next iteration
$\bar{x}_{mx}$	geocentric position vector of the moon at the postpericynthion pseudostate transformation time
$\bar{u}_{mx}$	geocentric velocity vector of the moon at post-pericynthion pseudostate transformation time
q_b	flag which indicates the magnitude of R_s, and whether control passed through the prepericynthion or postpericynthion branch
Δy_0	magnitude of the abort position mismatch
$\bar{y}_a'$	fictitious abort position vector to be used in next pass through INRFV
δt	difference between PTS exit time and time of postpericynthion pseudostate transformation

Discussion

In the overlapped conic logic, the flight time interval T_a is given a minus sign if the abort position lies on the prepericynthion leg of the INRFV conic. The time interval T_x is always positive. Depending on the value of T_a, the logic flow within the PSTATE routine follows one or the other of two major branches that are described below.

1. Prepericynthion Branch

The prepericynthion branch ($T_a < -1.0$ hour) makes a new estimate of the time (t_x''') of the postpericynthion pseudostate to be used in defining the geocentric conic in the next iteration. This time is given by

$$t_x''' = t_0 - T_a$$

and the corresponding selenocentric position vector is given by

$$\bar{y}_x' = \bar{y}_x' - T_x \bar{v}_x$$

The pseudostate time t_x''', thus defined, is the pericynthion passage time of the INRFV conic. Definition of the postpericynthion pseudostate in this manner assures that, upon convergence of the iterations, the final INRFV conic will represent the true osculating selenocentric conic at the pericynthion passage time (at least within the accuracy of the pseudostate theory). It further assures that three-body effects will be properly accounted for on the postpericynthion leg of the postmaneuver trajectory.

Some additional computations are necessary to account for three-body effects on the prepericynthion leg. The first requirement is to compute the prepericynthion pseudostate for time t_x'''. This is accomplished by propagating the selenocentric state vector (t_0, $\bar{y}'_a$, $\bar{v}_a$) backward along the INRFV conic to the sphere entrance point,

$$t_n = t_0 - (T_x + T_a)$$

$\bar{y}_n$ = sphere entry position vector

$\bar{v}_n$ = sphere entry velocity vector

and then propagating forward linearly (i.e., with constant velocity) to the time t_x'''. The resulting selenocentric pseudostate is then converted to geocentric coordinates

$$\bar{u}_p^{*n} = \bar{v}_n + \bar{u}_{mx}$$

$$\overline{x}_p^{*n} = \overline{y}_n + T_x \overline{v}_n + \overline{x}_{mx}$$

and propagated backward, along a geocentric conic using KEPLER, to the maneuver time t_0. This defines the new pseudostate $(t_0, \overline{x}_a^{*n}, \overline{u}_a^{*n})$. A pseudostate-to-real-state transformation is made by converting to selenocentric coordinates

$$\overline{y} = \overline{x}_a^{*n} - \overline{x}_{m0}$$

$$\overline{v} = \overline{u}_a^{*n} - \overline{u}_{m0}$$

where $\overline{x}_{m0}$, $\overline{u}_{m0}$ are the position and velocity vectors of the moon at time t_0. Then the linear flight time to a new PTS entry point is computed by

$$\Delta t = \frac{-(\overline{y} \cdot \overline{v}) - \sqrt{(\overline{y} \cdot \overline{v})^2 + (\overline{v} \cdot \overline{v})(R_s^2 - \overline{y} \cdot \overline{y})}}{(\overline{v} \cdot \overline{v})}$$

and a new sphere entrance state vector is computed by

$$t_n = t_0 + \Delta t$$

$$\overline{v}_n = \overline{v}$$

$$\overline{y}_n = \overline{y} + \Delta t \, \overline{v}$$

The state vector $(t_n, \overline{y}_n, \overline{v}_n)$ is propagated by KEPLER along the selenocentric conic to the abort time t_0. This defines the new selenocentric post-abort state vector $(t_0, \overline{y}_a, \overline{v}_a)$. The abort position errors and INRFV target position for the next pass is computed by

$$\overline{\Delta y}_0 = \overline{y}_0 - \overline{y}_a$$

$$\overline{y}_a' = \overline{y}_a' + \overline{\Delta y}_0$$

$$\Delta y_0 = \left| \overline{\Delta y}_0 \right|$$

2. Postpericynthion Branch

If the maneuver position lies on the postpericynthion leg of the INRFV conic, or not more than one hour from pericynthion on the prepericynthion leg, logic flows through the PSTATE postpericynthion branch. In such cases, an adequate simulation of three-body effects over the entire trajectory is assured simply by defining the postpericynthion pseudostate components in an appropriate manner.

The postpericynthion equations

$$t_x''' = t_0$$

and

$$\bar{y}_x' = \bar{y}_x' - (T_x - T_a)\,\bar{v}_x$$

guarantee that, upon convergence of the iterations, the final INRFV conic will represent the true osculating selenocentric conic at the maneuver time t_0. This means that the computation of $\bar{y}_a$ is not required and that position vector continuity at time t_0 can be assured by setting

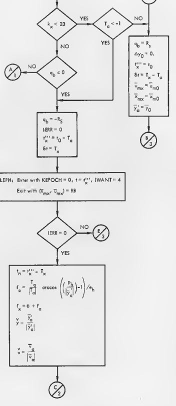
$$\bar{y}_a' = \bar{y}_0$$

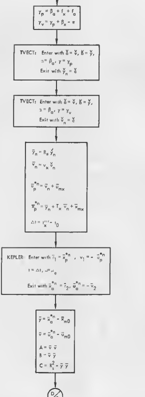
The equation for position discontinuity at maneuver time is

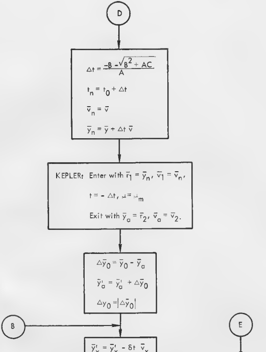
$$\Delta y_0 = 0$$

by definition.

When this branch of the PSTATE logic is followed, the postmaneuver velocity vector $\bar{v}_a$ retains the value previously assigned in the INRFV subroutine.


PSTATE
$k_x < 23$
YES
NO
$T_a < -1$
NO
YES
A
$q_b \le 0$
NO
A/1
YES
$q_b = R_s$
$\Delta y_0 = 0.$
$t_x''' = t_0$
$\delta t = T_x - T_a$
$\bar{u}_{mx} = \bar{u}_{m0}$
$\bar{x}_{mx} = \bar{x}_{m0}$
$\bar{y}_a' = \bar{y}_0$
B/3
$q_b = -R_S$
IERR = 0
$t_x''' = t_0 - T_a$
$\delta t = T_x$
JPLEPH: Enter with KEPOCH = 0, $t = t_x'''$, IWANT = 4
Exit with $(\bar{x}_{mx}, \bar{u}_{mx})$ = RB
IERR = 0
NO
E/3
YES
$t_n = t_x''' - T_x$
$f_a = \frac{T_a}{|T_a|} \arccos\left(\left(\frac{p_h}{|\bar{y}_a'|}\right) - 1\right) / e_h$
$f_x = \theta + f_a$
$\check{y} = \frac{\bar{y}_a'}{|\bar{y}_a'|}$
$\check{v} = \frac{\bar{v}_a}{|\bar{v}_a|}$
C/2


C
$\gamma_p = \beta_a + f_x + f_a$
$\gamma_v = \gamma_p + \beta_x - \pi$
TVECT: Enter with $\check{a} = \check{v}$, $\check{b} = \check{y}$,
$\alpha = \beta_a$, $\gamma = \gamma_p$
Exit with $\check{y}_n = \check{c}$
TVECT: Enter with $\check{a} = \check{v}$, $\check{b} = \check{y}$,
$\alpha = \beta_a$, $\gamma = \gamma_v$
Exit with $\check{v}_n = \check{c}$
$\bar{y}_n = R_s \check{y}_n$
$\bar{v}_n = v_x \check{v}_n$
$\bar{u}_p^{*n} = \bar{v}_n + \bar{u}_{mx}$
$\bar{x}_p^{*n} = \bar{y}_n + T_x \bar{v}_n + \bar{x}_{mx}$
$\Delta t = t_x''' - t_0$
KEPLER Enter with $\bar{r}_1 = \bar{x}_p^{*n}$, $v_1 = -\bar{u}_p^{*n}$
$t = \Delta t$, $\mu = \mu_e$
Exit with $\bar{x}_a^{*n} = \bar{r}_2$, $\bar{u}_a^{*n} = -\bar{v}_2$
$\bar{y} = \bar{x}_a^{*n} - \bar{x}_{m0}$
$\bar{v} = \bar{u}_a^{*n} - \bar{u}_{m0}$
$A = \bar{v}\ \bar{v}$
$B = \bar{v}\ \bar{y}$
$C = R_s^2 - \bar{y}\ \bar{y}$
D/3


D
$\Delta t = \frac{-B - \sqrt{B^2 + AC}}{A}$
$t_n = t_0 + \Delta t$
$\bar{v}_n = \bar{v}$
$\bar{y}_n = \bar{y} + \Delta t \, \bar{v}$
KEPLER: Enter with $\bar{r}_1 = \bar{y}_n$, $\bar{v}_1 = \bar{v}_n$,
$t = -\Delta t$, $\mu = \mu_m$
Exit with $\bar{y}_a = \bar{r}_2$, $\bar{v}_a = \bar{v}_2$.
$\Delta\bar{y}_0 = \bar{y}_0 - \bar{y}_a$
$\bar{y}'_a = \bar{y}'_a + \Delta\bar{y}_0$
$\Delta y_0 = |\Delta\bar{y}_0|$
B
$\bar{y}'_x = \bar{y}'_x - \delta t \; \bar{v}_x$
E
Write "Error in JPLEPH"
RETURN

APPENDIX C

INPUT/OUTPUT ROUTINES

CONTENTS

C-1 CONVTI

Purpose

This routine converts the input parameters to internal units.

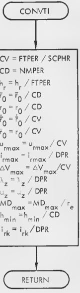
Input

h_r	reentry altitude
u_{rmax}	maximum reentry velocity
i_{rmax}	maximum return inclination
$\bar{r}_0$, $\dot{\bar{r}}_0$	position and velocity vectors premaneuver
ΔV_{max}	maximum available abort velocity
λ_z	PTP longitude
μ_z	PTP latitude
MD_{max}	maximum allowable miss distance for PTP
h_{min}	minimum pericynthion altitude allowed
FTPER	feet per earth radius
SCPHR	seconds per hour
NMPER	nautical miles per earth radius
i_{rk}	desired return inclination
r_0, $\dot{r}_0$	magnitudes of the position and velocity vectors premaneuver

Output

The input quantities expressed in internal units (earth radii, earth radii per hour, hours, radians) and the following conversion factors:

CV	conversion factor for velocities
CD	conversion factor for distances


CONVTI
CV = FTPER / SCPHR
CD = NMPER
$h_r = h_r$ / FTPER
$\bar{r}_0 = \bar{r}_0$ / CD
$r_0 = r_0$ / CD
$\dot{\bar{r}}_0 = \dot{\bar{r}}_0$ / CV
$\dot{r}_0 = r_0$ / CV
$u_{rmax} = u_{rmax}$ / CV
$i_{rmax} = i_{rmax}$ / DPR
$\Delta V_{max} = \Delta V_{max}$/CV
$\lambda_z = \lambda_z$ / DPR
$u_z = u_z$ / DPR
$MD_{max} = MD_{max} / r_e$
$h_{min} = h_{min}$ / CD
$i_{rk} = i_{rk}$/DPR
RETURN

C-2 INVAL

Purpose

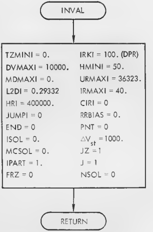
This routine initializes the input variables.

Input

None

Output

Initialized values of the input variables. These values will be used unless input otherwise.


INVAL
TZMINI = 0.
IRKI = 100. (DPR)
DVMAXI = 10000.
HMINI = 50.
MDMAXI = 0.
URMAXI = 36323.
L2DI = 0.29332
IRMAXI = 40.
HRI = 400000.
CIRI = 0
JUMPI = 0
RRBIAS = 0.
END = 0
PNT = 0
ISOL = 0.
ΔV_{st} = 1000.
MCSOL = 0.
JZ = 1
IPART = 1.
J = 1
FRZ = 0
NSOL = 0
RETURN

C-3 MCPNT

Purpose

This routine calculates the output trajectory data for all moon referenced modes. In the ATP and PTP modes, it is also used to calculate landing site data while converging on the target area.

Input

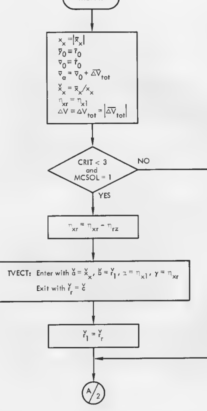
$\overline{\Delta V}_{tot}$	ΔV vector
η_{rz1}	angle from reentry to landing
t_z	time of landing
η_{x1}	angle from pseudostate position to $\check{r}_1$
u_r	reentry velocity
$\check{r}_1$	unit vector to the reentry position; if the PTP mode is used, the unit vector points to the landing site
$\overline{u}_x$	geocentric pseudostate velocity vector
$\overline{x}_x$	geocentric pseudostate position vector
ISSFLG	= 0, compute all output data = 1, calculate landing site data for the ATP or PTP iteration
λ_z''	desired longitude of landing for PTP mode
μ_z''	desired latitude of landing for PTP mode
CRIT	= 2, PTP mode = 4, ATP mode = 6, FCUA mode = 7, solution is not to be considered in optimization
$\overline{r}_0$, $\dot{\overline{r}}_0$	preabort selenocentric position and velocity vectors

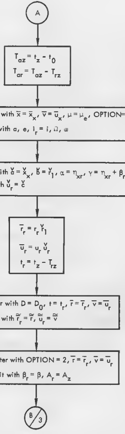
D_0	Gregorian date of EPOCH
EPOCH	reference time of day
t_0	preabort time measured from EPOCH
T_{rz}	time from reentry to landing
r_r	reentry radius
LETSGO	= 1, maneuver is performed at a discrete time = 2, generate tradeoff display data = 3, optimize ΔV as a function of maneuver time
MCSOL	= 0, miss distance solution found in PTP mode = 1, hit solution found in PTP mode

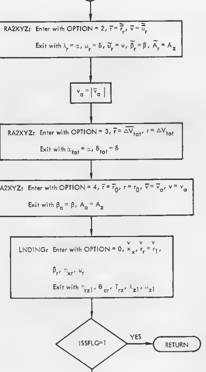
Output

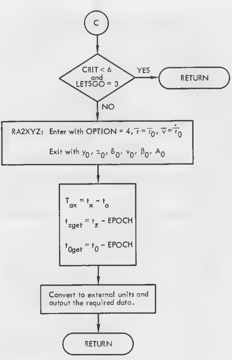
λ_{z1}	longitude of impact
μ_{z1}	latitude of impact
$\overline{v}_a$	postmaneuver velocity vector
ΔV	abort ΔV magnitude
t_z	time of landing
T_{ar}	time from maneuver to reentry
T_{ax}	time from maneuver to PTS exit
$\overline{x}_x$	geocentric pseudostate position vector
$\overline{u}_x$	geocentric pseudostate velocity vector
$\overline{y}_x$	selenocentric PTS exit position vector
$\overline{v}_x$	selenocentric PTS exit velocity vector
η_{rz}	transfer angle from reentry to landing
β_r	reentry flight-path angle

t_{zget}	ground elapsed time of landing
λ_r	longitude at reentry
μ_r	latitude at reentry
i_r	return inclination
u_r	reentry velocity
MD	miss distance
q_a	apogee passage flag
q_d	pericynthion passage flag
η_{xr}	transfer angle from $\overline{x}_x$ to reentry
ψ	angle from selenocentric maneuver position to the PTS exit velocity vector
r_p	radius of pericynthion
t_{0get}	time of the maneuver measured from epoch time of the reference day


MCPNT
$x_x = |\bar{x}_x|$
$\bar{r}_0 \equiv \bar{r}_0$
$\bar{v}_0 \equiv \dot{\bar{r}}_0$
$\bar{v}_a = \bar{v}_0 + \overline{\Delta V}_{tot}$
$\check{x}_x = \bar{x}_x / x_x$
$\eta_{xr} = \eta_{x1}$
$\Delta V \equiv \Delta V_{tot} = |\overline{\Delta V}_{tot}|$
CRIT < 3 and MCSOL = 1
NO
YES
$\eta_{xr} = \eta_{xr} - \eta_{rz}$
TVECT: Enter with $\check{a} = \check{x}_x$, $\check{b} = \check{r}_1$, $\alpha = \eta_{x1}$, $\gamma = \eta_{xr}$
Exit with $\check{r}_r = \check{c}$
$\check{r}_1 = \check{r}_r$
A/2


A
$T_{az} = t_z - t_0$
$T_{ar} = T_{az} - T_{rz}$
XYZ2AE: Enter with $\bar{x} = \bar{x}_x$, $\bar{v} = \bar{u}_x$, $\mu = \mu_e$, OPTION=1
Exit with a, e, $i_r = i$, Ω, ω
TVECT: Enter with $\check{a} = \check{x}_x$, $\check{b} = \check{r}_1$, $\alpha = \eta_{xr}$, $\gamma = \eta_{xr} + \beta_r$
Exit with $\check{u}_r = \check{c}$
$\bar{r}_r = r_r \check{r}_1$
$\bar{u}_r = u_r \check{u}_r$
$t_r = t_z - T_{rz}$
EGTR: Enter with $D = D_0$, $t = t_r$, $\bar{r} = \bar{r}_r$, $\bar{v} = \bar{u}_r$
Exit with $\tilde{\bar{r}}_r = \tilde{\bar{r}}$, $\tilde{\bar{u}}_r = \tilde{\bar{v}}$
RA2XYZ: Enter with OPTION = 2, $\bar{r} = \bar{r}_r$, $\bar{v} = \bar{u}_r$
Exit with $\beta_r = \beta$, $A_r = A_z$
B/3


B
RA2XYZ: Enter with OPTION = 2, $\bar{r} = \tilde{\bar{r}}_r$, $\bar{v} = \tilde{\bar{u}}_r$
Exit with $\lambda_r = \alpha$, $\mu_r = \delta$, $\tilde{u}_r = v$, $\tilde{\beta}_r = \beta$, $\tilde{A}_r = A_z$
$v_a = |\bar{v}_a|$
RA2XYZ: Enter with OPTION = 3, $\bar{r} = \overline{\Delta V}_{tot}$, $r = \Delta V_{tot}$
Exit with $\alpha_{tot} = \alpha$, $\delta_{tot} = \delta$
RA2XYZ: Enter with OPTION = 4, $\bar{r} = \bar{r}_0$, $r = r_0$, $\bar{v} = \bar{v}_a$, $v = v_a$
Exit with $\beta_a = \beta$, $A_a = A_z$
LNDING: Enter with OPTION = 0, $\check{x}_x$, $\check{r}_r = \check{r}_1$,
β_r, η_{xr}, u_r
Exit with η_{rz1}, θ_{cr}, T_{rz}, λ_{z1}, μ_{z1}
ISSFLG=1
YES
RETURN
NO
C/4


C
CRIT < 6 and LETSGO = 3
YES
RETURN
NO
RA2XYZ: Enter with OPTION = 4, $\bar{r} = \bar{r}_0$, $\bar{v} = \dot{\bar{r}}_0$
Exit with γ_0, α_0, δ_0, v_0, β_0, A_0
$T_{ax} = t_x - t_o$
$t_{zget} = t_z$ - EPOCH
$t_{0get} = t_0$ - EPOCH
Convert to external units and output the required data.
RETURN

C-4 READ

Purpose

This routine reads the program inputs and transfers them to internal program names. The u_r and β_r tables are converted to internal units.

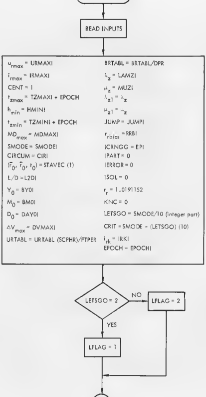
Input

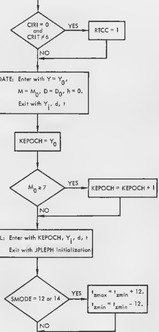
All program inputs

Output

Internal values of the program variables and the following internal flags:

ISOL	solution flag
IPART	flag used in SEARCH
IERROR	error flag
LETSGO	= 1, abort at a discrete time = 2, produce tradeoff display = 3, optimize ΔV as a function of maneuver time
CRIT	= 2, primary target point mode = 4, alternate target point mode = 6, fuel critical unspecified area mode
LFLAG	= 1, print flag for tradeoff display = 2, print flag for discrete and search cases


READ
READ INPUTS
u_rmax = URMAXI
i_rmax = IRMAXI
CENT = 1
t_zmax = TZMAXI + EPOCH
h_min = HMINI
t_zmin = TZMINI + EPOCH
MD_max = MDMAXI
SMODE = SMODEI
CIRCUM = CIRI
(r̄_0, ṙ_0, t_0) = STAVEC (1)
L/D = L2DI
Y_0 = BY0I
M_0 = BM0I
D_0 = DAY0I
ΔV_max = DVMAXI
URTABL = URTABL (SCPHR)/FTPER
BRTABL = BRTABL/DPR
λ_z = LAMZI
μ_z = MUZI
λ_z1 = λ_z
μ_z1 = μ_z
JUMP = JUMPI
r_rbias = RRBI
ICRNGG = EPI
IPART = 0
IERROR = 0
ISOL = 0
r_r = 1.0191152
KNC = 0
LETSGO = SMODE/10 (integer part)
CRIT = SMODE - (LETSGO) (10)
i_rk = IRKI
EPOCH = EPOCHI
LETSGO = 2
NO
LFLAG = 2
YES
LFLAG = 1
E
2


E
CIRI = 0 and CRIT ≠ 6
YES
RTCC = 1
NO
XDATE: Enter with Y = Y0, M = M0, D = D0, h = 0. Exit with Yi, d, t
KEPOCH = Y0
M0 ≥ 7
YES
KEPOCH = KEPOCH + 1
NO
INIJPL: Enter with KEPOCH, Yi, d, t Exit with JPLEPH initialization
SMODE = 12 or 14
YES
tzmax = tzmin + 12.
tzmin = tzmin - 12.
NO
RETURN

C-5 STORE

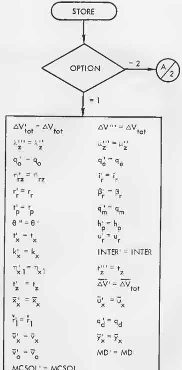
Purpose

This closed routine stores trajectory parameters for the optimum trajectory and restores them on request.

Input and Output

OPTION	$\begin{cases} = 1, \text{ store the optimum case} \\ = 2, \text{ output the optimum case} \end{cases}$
ΔV_{tot}	abort ΔV magnitude
λ_z''	longitude of the PTP
μ_z''	latitude of the PTP
q_a	apogee passage flag
q_e	reentry direction flag
η_{rz}	reentry downrange angle
i_r	return inclination
r_r	reentry radius
β_r	reentry flight-path angle
t_p	time of pericynthion passage
q_m	direction of motion about moon flag
θ'	transfer angle from the selenocentric maneuver position to PTS exit position
h_p	altitude of pericynthion passage
t_x	time of exit of PTS
u_r	reentry velocity
k_x	pseudostate iteration counter
INTER	return plane selector flag

η_{x1}	transfer angle from $\overline{x}_x$ position to $\check{r}_1$
t_z	time of landing
$\overline{\Delta V}_{tot}$	abort ΔV vector
$\overline{x}_x$	geocentric pseudostate position vector
$\overline{u}_x$	geocentric pseudostate velocity vector
$\check{r}_1$	unit vector to the reentry point; if a PTP solution is found, this vector points to the landing site
q_d	pericynthion passage flag
$\overline{v}_x$	selenocentric PTS exit position vector
$\overline{y}_x$	selenocentric PTS exit velocity vector
$\overline{v}_a$	selenocentric postmaneuver velocity vector
MD	miss distance
MCSOL	flag for miss distance cases


STORE
OPTION
= 2
A/2
= 1
$\Delta V'_{tot} = \Delta V_{tot}$
$\Delta V''' = \Delta V_{tot}$
$\lambda'''_z = \lambda''_z$
$\mu'''_z = \mu''_z$
$q'_o = q_o$
$q'_e = q_e$
$\eta'_{rz} = \eta_{rz}$
$i'_r = i_r$
$r'_r = r_r$
$\beta'_r = \beta_r$
$t'_p = t_p$
$q'_m = q_m$
$\theta'' = \theta'$
$h'_p = h_p$
$t'_x = t_x$
$u'_r = u_r$
$k'_x = k_x$
INTER' = INTER
$\eta'_{x1} = \eta_{x1}$
$t'''_z = t_z$
$t'_z = t_z$
$\overline{\Delta V}' = \overline{\Delta V}_{tot}$
$\bar{x}'_x = \bar{x}_x$
$\bar{u}'_x = \bar{u}_x$
$\check{r}'_1 = \check{r}_1$
$q'_d = q_d$
$\bar{v}'_x = \bar{v}_x$
$\bar{y}'_x = \bar{y}_x$
$\bar{v}'_o = \bar{v}_o$
MD' = MD
MCSOL' = MCSOL
RETURN

(A)

$\Delta V_{tot} = \Delta V'_{tot}$	$\lambda''_z = \lambda'''_z$
$\mu''_z = \mu'''_z$	$q_a = q'_a$
$q_e = q'_e$	$\eta_{rz} = \eta'_{rz}$
$i_r = i'_r$	$r_r = r'_r$
$\beta_r = \beta'_r$	$t_p = t'_p$
$q_m = q'_m$	$\theta' = \theta''$
$h_p = h'_p$	$t_x = t'_x$
$u_r = u'_r$	$k_x = k'_x$
INTER = INTER'	$\eta_{x1} = \eta'_{x1}$
$t_z = t'''_z$	$t_z = t'_z$
$\overline{\Delta V}_{tot} = \overline{\Delta V}'$	$\overline{x}_x = \overline{x}'_x$
$\overline{u}_x = \overline{u}'_x$	$\check{r}_1 = \check{r}'_1$
$q_d = q'_d$	$\overline{v}_x = \overline{v}'_x$
$\overline{y}_x = \overline{y}'_x$	$\overline{v}_a = \overline{v}'_a$
MD = MD'	MCSOL = MCSOL'

RETURN

APPENDIX D

UTILITY ROUTINES

CONTENTS

D-1 CNTS

Purpose

This data block generates a number of useful constants and stores them in the labelled common blocks/CONSTS/and/PIE/.

Input

None

Output

DPR	= deg/rad	= 57.2957795	/PIE/
CPI	= π	= 3.14159265	/PIE/
C2PI	= 2π	= 6.28318531	/PIE/
CPIO2	= $\pi/2$	= 1.57079633	/PIE/
FTPER	= ft/r_e	= 20925738.2	
NMPER	= $n\ mi/r_e$	= 3443.93359	
KMPER	= km/r_e	= 6378.16500	
SCPHR	= sec/hr	= 3600.	
C3PIO2	= $3\pi/2$	= 4.71238899	
RE	= r_e	= 1.0	earth radius
RM	= r_m	= 0.272506277	moon radius
MUE	= μ_e	= 19.9094165	earth gravitational constant
MUM	= μ_m	= 0.244883757	moon gravitational constant
OMGE	= ω_e	= 0.2625161436	earth rotation rate
EPSX	= ϵ_x	= 0.0005	distance matching tolerance for overlapped conics
KMAX	= k_{max}	= 50	maximum number of iterations

D-2 DROOTS

Purpose

This closed routine determines the real roots of equations of degree one, two, three, or four with real coefficients. Although no complex roots are obtained, their number is determined. The calculations are made in double precision.

Input

Coefficients A, B, C, D, E of the equations

$$Ax^4 + Bx^3 + Cx^2 + Dx + E = 0$$

$$Bx^3 + Cx^2 + Dx + E = 0$$

$$Cx^2 + Dx + E = 0$$

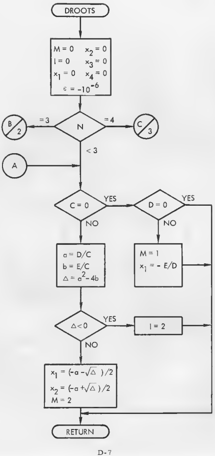
$$Dx + E = 0$$

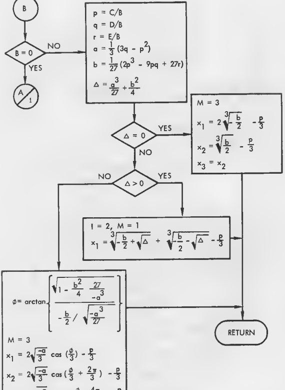
N degree of the polynomial ($1 \le N \le 4$)

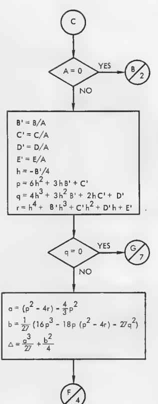
MPRT print flag if MPRT = $\binom{1}{2}$ then $\binom{\text{no print}}{\text{print}}$ is executed.

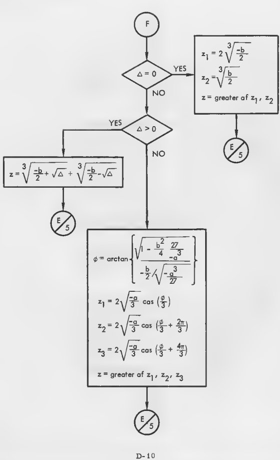
Output

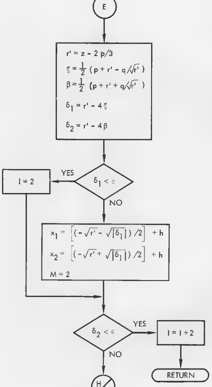
x_1, x_2, x_3, x_4 (X1, X2, X3, X4)	These are the real roots of the equation. If only two real solutions are found, x_3 and x_4 are set equal to zero. If no real solutions exist, all x_i are set to zero.
M	the number of real roots
I	the number imaginary roots (M + I = N)

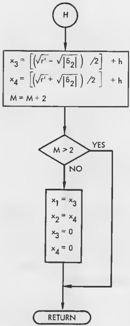

DROOTS
M = 0 $x_2 = 0$
I = 0 $x_3 = 0$
$x_1 = 0$ $x_4 = 0$
$\epsilon = -10^{-6}$
B/2
= 3
N
= 4
C/3
< 3
A
C = 0
YES
NO
D = 0
YES
NO
$a = D/C$
$b = E/C$
$\Delta = a^2 - 4b$
M = 1
$x_1 = -E/D$
$\Delta < 0$
YES
I = 2
NO
$x_1 = (-a - \sqrt{\Delta})/2$
$x_2 = (-a + \sqrt{\Delta})/2$
M = 2
RETURN

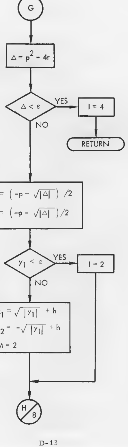

B
B = 0
NO
YES
A
1
$p = C/B$
$q = D/B$
$r = E/B$
$a = \frac{1}{3}(3q - p^2)$
$b = \frac{1}{27}(2p^3 - 9pq + 27r)$
$\Delta = \frac{a^3}{27} + \frac{b^2}{4}$
$\Delta = 0$
YES
NO
M = 3
$x_1 = 2\sqrt[3]{-\frac{b}{2}} - \frac{p}{3}$
$x_2 = \sqrt[3]{\frac{b}{2}} - \frac{p}{3}$
$x_3 = x_2$
NO
$\Delta > 0$
YES
I = 2, M = 1
$x_1 = \sqrt[3]{-\frac{b}{2} + \sqrt{\Delta}} + \sqrt[3]{-\frac{b}{2} - \sqrt{\Delta}} - \frac{p}{3}$
$\phi = \arctan\left[\frac{\sqrt{1 - \frac{b^2}{4}\ \frac{27}{-a^3}}}{-\frac{b}{2} / \sqrt{\frac{-a^3}{27}}}\right]$
M = 3
$x_1 = 2\sqrt{\frac{-a}{3}} \cos\left(\frac{\phi}{3}\right) - \frac{p}{3}$
$x_2 = 2\sqrt{\frac{-a}{3}} \cos\left(\frac{\phi}{3} + \frac{2\pi}{3}\right) - \frac{p}{3}$
$x_3 = 2\sqrt{\frac{-a}{3}} \cos\left(\frac{\phi}{3} + \frac{4\pi}{3}\right) - \frac{p}{3}$
RETURN

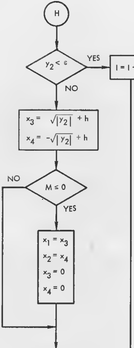

C
A = 0
YES
B/2
NO
B' = B/A
C' = C/A
D' = D/A
E' = E/A
h = -B'/4
$p = 6h^2 + 3hB' + C'$
$q = 4h^3 + 3h^2 B' + 2hC' + D'$
$r = h^4 + B'h^3 + C'h^2 + D'h + E'$
q = 0
YES
G/7
NO
$a = (p^2 - 4r) - \frac{4}{3}p^2$
$b = \frac{1}{27}(16p^3 - 18p(p^2 - 4r) - 27q^2)$
$\Delta = \frac{a^3}{27} + \frac{b^2}{4}$
F/4


F
$\Delta = 0$
YES
$z_1 = 2\sqrt[3]{\frac{-b}{2}}$
$z_2 = \sqrt[3]{\frac{b}{2}}$
z = greater of z_1, z_2
E/5
NO
$\Delta > 0$
YES
$z = \sqrt[3]{\frac{-b}{2} + \sqrt{\Delta}} + \sqrt[3]{\frac{-b}{2} - \sqrt{\Delta}}$
E/5
NO
$\phi = \arctan\left\{\frac{\sqrt{1 - \frac{b^2}{4}\,\frac{27}{-a^3}}}{-\frac{b}{2}\Big/\sqrt{-\frac{a^3}{27}}}\right\}$
$z_1 = 2\sqrt{\frac{-a}{3}}\cos\left(\frac{\phi}{3}\right)$
$z_2 = 2\sqrt{\frac{-a}{3}}\cos\left(\frac{\phi}{3} + \frac{2\pi}{3}\right)$
$z_3 = 2\sqrt{\frac{-a}{3}}\cos\left(\frac{\phi}{3} + \frac{4\pi}{3}\right)$
z = greater of z_1, z_2, z_3
E/5


E
r' = z - 2 p/3
ξ = 1/2 (p + r' - q/√r')
β = 1/2 (p + r' + q/√r')
δ1 = r' - 4ξ
δ2 = r' - 4β
YES
I = 2
δ1 < ε
NO
x1 = [(-√r' - √|δ1|) /2] + h
x2 = [(-√r' + √|δ1|) /2] + h
M = 2
δ2 < ε
YES
I = I + 2
NO
H/6
RETURN


H
$x_3 = \left[\left(\sqrt{r'} - \sqrt{|\delta_2|}\right) /2\right] + h$
$x_4 = \left[\left(\sqrt{r'} + \sqrt{|\delta_2|}\right) /2\right] + h$
M = M + 2
M > 2
YES
NO
$x_1 = x_3$
$x_2 = x_4$
$x_3 = 0$
$x_4 = 0$
RETURN


G
$\Delta = p^2 - 4r$
$\Delta < \epsilon$
YES
$I = 4$
RETURN
NO
$y_1 = \left(-p + \sqrt{|\Delta|}\right)/2$
$y_2 = \left(-p - \sqrt{|\Delta|}\right)/2$
$y_1 < \epsilon$
YES
$I = 2$
NO
$x_1 = \sqrt{|y_1|} + h$
$x_2 = -\sqrt{|y_1|} + h$
$M = 2$
H
8


H
y2 < ε
YES
I = I + 2
NO
x3 = √|y2| + h
x4 = -√|y2| + h
NO
M ≤ 0
YES
x1 = x3
x2 = x4
x3 = 0
x4 = 0
M = M + 2
RETURN

D-3 EGTR

Purpose

This routine is used to transform earth equatorial inertial position and velocity vectors to the geographic rotating system.

Input

D	Gregorian date of epoch
t	time in hours (0^h GMT)
$\bar{r}$	geocentric radius vector
$\bar{v}$	geocentric velocity vector
D_J	Julian date of epoch; this input must be available in the common block/DAYJ/. See TJUDAT

Output

$\tilde{\tilde{r}}$	geographic radius vector
$\tilde{\tilde{v}}$	geographic velocity vector

EGTR

SIDCOM: Enter with $JD_0 = JD$, $\Delta T = 0$, $N = 0$
Exit with $\alpha_{SID} = \alpha_{SID}$ $T = \text{Time}$

$$\alpha_{SID} = \alpha_{SID} + t\,\omega_e$$

$$\hat{G} = \begin{pmatrix} \cos\alpha_{SID} & \sin\alpha_{SID} & 0 \\ -\sin\alpha_{SID} & \cos\alpha_{SID} & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$\bar{v}_e = (-\omega_e\,y,\ \omega_e\,x,\ 0)$$
$$\tilde{\bar{r}} = \hat{G}\,\bar{r}$$
$$\tilde{\bar{v}} = \hat{G}(\bar{v} - \bar{v}_e)$$

RETURN

D-4 INIJPL

Purpose

This routine initializes the JPLEPH routine to the correct day and year.

Input

EPOCH	desired Besselian year for the reference epoch
IYEAR	the year Y_J from XDATE
IDAY	day of year from XDATE
IHOUR	time of day (0 hour GMT)

Output

Initialization data for JPLEPH

D-5 INTER

Purpose

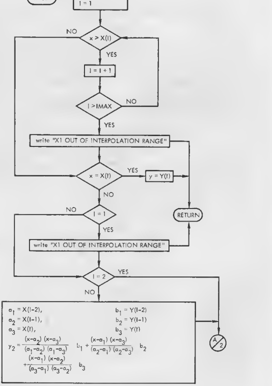
This utility routine performs interpolation within any specified table. The answer is the average of two LaGrange parabolic curve fits, the first parabola using the two points to the left of the interpolated point and the first point to the right, the second using one point to the left and two points to the right. Due allowance is made for interpolation at the extreme ends of the table. Used with URBRER in this program.

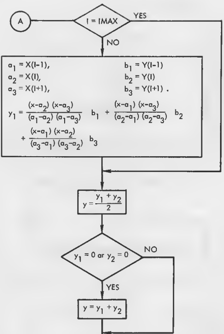
Input

X	table of independent variable
Y	table of dependent variable
IMAX	number of entries in the tables
x	value of the independent variable for which interpolation is to be performed

Output

y	interpolated value of the dependent variable


INTER
$y_1 = 0,\ y_2 = 0$
I = 1
NO
x > X(I)
YES
I = I + 1
I > IMAX
NO
YES
write "X1 OUT OF INTERPOLATION RANGE"
x = X(I)
YES
y = Y(I)
NO
RETURN
NO
I = 1
YES
write "X1 OUT OF INTERPOLATION RANGE"
I = 2
YES
NO
$a_1 = X(I-2),\quad b_1 = Y(I-2)$
$a_2 = X(I-1),\quad b_2 = Y(I-1)$
$a_3 = X(I),\quad b_3 = Y(I)$
$y_2 = \frac{(x-a_2)(x-a_3)}{(a_1-a_2)(a_1-a_3)} b_1 + \frac{(x-a_1)(x-a_3)}{(a_2-a_1)(a_2-a_3)} b_2 + \frac{(x-a_1)(x-a_2)}{(a_3-a_1)(a_3-a_2)} b_3$
A
2


A
I = IMAX
YES
NO
a1 = X(I-1),
b1 = Y(I-1)
a2 = X(I),
b2 = Y(I)
a3 = X(I+1),
b3 = Y(I+1) .
$y_1 = \frac{(x-a_2)(x-a_3)}{(a_1-a_2)(a_1-a_3)} b_1 + \frac{(x-a_1)(x-a_3)}{(a_2-a_1)(a_2-a_3)} b_2 + \frac{(x-a_1)(x-a_2)}{(a_3-a_1)(a_3-a_2)} b_3$
$y = \frac{y_1 + y_2}{2}$
$y_1 = 0$ or $y_2 = 0$
NO
YES
$y = y_1 + y_2$
RETURN

D-6 JPLEPH-NEWT

Purpose

This routine is designed to provide ephemeris data for the sun, moon, and earth, and a matrix of precession-nutation-libration for selenographic coordinate transformations. The coordinate system for all data is defined by the mean equator and the ecliptic at the beginning of the nearest Besselian year. The x-axis points towards the vernal equinox, the z-axis along the mean pole, and the y-axis designed to complete a right-hand coordinate system.

Input

KEPOCH	The first time JPLEPH is called, this argument should contain the desired Besselian year for the reference epoch (1951 - 1999). For this first call of JPLEPH, the routine XDATE must be called prior to the JPLEPH call in order to establish the required base time in the common block /DATE/. On subsequent entries, KEPOCH should be input as zero to bypass the initialization.
T	Time (in hours) relative to the base time described in /DATE/. A double precision input is required.
IWANT	Integer flag selecting ephemeris data as described in output.

Output

RS	six cells of sun ephemeris data
RB	nine cells of earth-moon ephemeris data
PNL	nine cells of libration matrix

If IWANT	RS	RB	PNL
= 1	position of sun with respect to earth	position and velocity of moon with respect to earth	not used
= 2	position of sun with respect to earth	position and velocity of earth with respect to moon	libration matrix

If IWANT	RS	RB	PNL
= 3	not used	position and velocity of moon with respect to earth	not used
= 4	not used	position and velocity of moon with respect to earth	not used
= 5	not used	not used	libration matrix

The data in RB are stored in the form

$\overline{x}_m(i) = RS(i)$ $\quad i = 1, 2, 3$

$\overline{u}_m(i) = RS(i + 6)$

IERROR Error flag. If IERROR = 0 no errors were detected.

Supplemental Routine

NEWT Fortran routine used for interpolation among the ephemeris data

D-7 KEPLER

Purpose

This routine is used to solve the two-body problem for the final position and velocity given the initial position and velocity vectors plus the transit time.

Input

$\bar{r}_1$	initial position vector
$\bar{v}_1$	initial velocity vector
t	transit time from $\bar{r}_1$ to $\bar{r}_2$
μ	gravitational constant

Output

$\bar{r}_2$	final position vector
$\bar{v}_2$	final velocity vector

Discussion

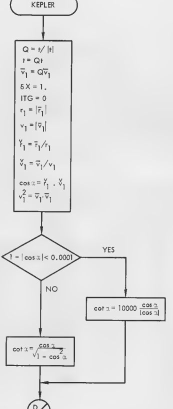
This routine uses Battin's modified transcendental functions, S(Arg) and C(Arg), in the Herrick universal formulation of Kepler's time of flight problem. The following non-dimensional variables are defined:

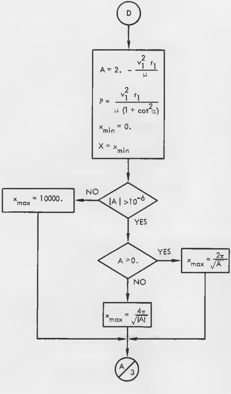
$P = p/r_1$ where p is the semilatus rectum

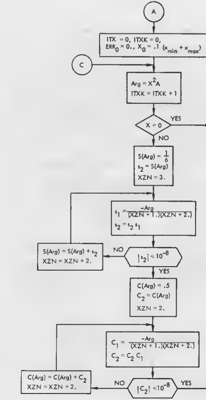
$A = r_1/a$ where a is semimajor axis

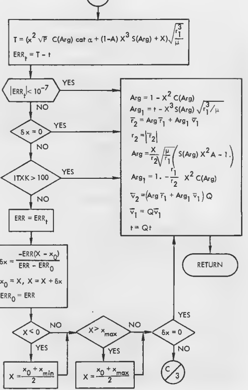
$X = r_1 x$ where x is Herrick's variable, $\Delta E\sqrt{a}$ for the ellipse, $\Delta G\sqrt{-a}$ for hyperbola

$$T = t\sqrt{\frac{\mu}{r_1^3}}$$


KEPLER
$Q = t/|t|$
$t = Qt$
$\bar{v}_1 = Q\bar{v}_1$
$\delta X = 1.$
$ITG = 0$
$r_1 = |\bar{r}_1|$
$v_1 = |\bar{v}_1|$
$\check{r}_1 = \bar{r}_1/r_1$
$\check{v}_1 = \bar{v}_1/v_1$
$\cos\alpha = \check{r}_1 \cdot \check{v}_1$
$v_1^2 = \bar{v}_1 \cdot \bar{v}_1$
$1 - |\cos\alpha| < 0.0001$
YES
NO
$\cot\alpha = 10000 \frac{\cos\alpha}{|\cos\alpha|}$
$\cot\alpha = \frac{\cos\alpha}{\sqrt{1 - \cos^2\alpha}}$
D
2


D
A = 2. − $\frac{v_1^2 r_1}{\mu}$
P = $\frac{v_1^2 r_1}{\mu (1 + \cot^2\alpha)}$
x_{min} = 0.
X = x_{min}
NO
x_{max} = 10000.
$|A| > 10^{-6}$
YES
A > 0.
YES
$x_{max} = \frac{2\pi}{\sqrt{A}}$
NO
$x_{max} = \frac{4\pi}{\sqrt{|A|}}$
A/3


A
ITX = 0, ITXK = 0,
ERR0 = 0., X0 = .1 (xmin + xmax)
C
Arg = X2A
ITXK = ITXK + 1
X = 0
YES
S(Arg) = 1/6
C(Arg) = .5
NO
S(Arg) = 1/6
s2 = S(Arg)
XZN = 3.
s1 = -Arg / (XZN + 1.)(XZN + 2.)
s2 = s2 s1
S(Arg) = S(Arg) + s2
XZN = XZN + 2.
NO
|s2| < 10-8
YES
C(Arg) = .5
C2 = C(Arg)
XZN = 2.
C1 = -Arg / (XZN + 1.)(XZN + 2.)
C2 = C2 C1
C(Arg) = C(Arg) + C2
XZN = XZN + 2.
NO
|C2| < 10-8
YES
B
4


B
$T = (x^2 \sqrt{P}\ C(Arg) \cot\alpha + (1-A) X^3 S(Arg) + X)\sqrt{\frac{r_1^3}{\mu}}$
$ERR_t = T - t$
$|ERR_t| < 10^{-7}$
YES
NO
$\delta x = 0$
YES
NO
ITXK > 100
YES
NO
$ERR = ERR_t$
$\delta x = \frac{-ERR(X - x_0)}{ERR - ERR_0}$
$x_0 = X,\ X = X + \delta x$
$ERR_0 = ERR$
X < 0
NO
YES
$X = \frac{x_0 + x_{min}}{2}$
$X > x_{max}$
NO
YES
$X = \frac{x_0 + x_{max}}{2}$
$\delta x = 0$
YES
NO
C/3
$Arg = 1 - X^2 C(Arg)$
$Arg_1 = t - X^3 S(Arg)\sqrt{r_1^3/\mu}$
$\bar{r}_2 = Arg\ \bar{r}_1 + Arg_1\ \bar{v}_1$
$r_2 = |\bar{r}_2|$
$Arg = \frac{X}{r_2}\sqrt{\frac{\mu}{r_1}}\left(S(Arg) X^2 A - 1.\right)$
$Arg_1 = 1. - \frac{r_1}{r_2} X^2 C(Arg)$
$\bar{v}_2 = \left(Arg\ \bar{r}_1 + Arg_1\ \bar{v}_1\right) Q$
$\bar{v}_1 = Q\bar{v}_1$
$t = Qt$
RETURN

D-8 LNDING

Purpose

This routine computes the landing longitude and latitude and time from reentry to landing using an empirical reentry range function. Also, a landing site "offset" is computed for the iteration loop used to acquire a desired landing location.

Input

$\check{x}_x$	unit vector directed toward the position of the vehicle at the geocentric pseudostate position vector
$\check{r}_r$	unit vector directed toward the reentry position
η_{xr}	the angle from the geocentric pseudostate position to reentry
β_r	reentry flight-path angle
u_r	reentry speed
i_r	reentry inclination
t_0	time of abort
T_{ar}	time from abort to reentry
λ_z''	longitude of the required landing site
μ_z''	latitude of the required landing site
ω_e	earth's rotational rate
α_{SID0}	right ascension at time zero
OPTION	= 0, compute the impact coordinates = 1, compute the "offset" site
k_x	iteration counter for pseudo state conics convergence

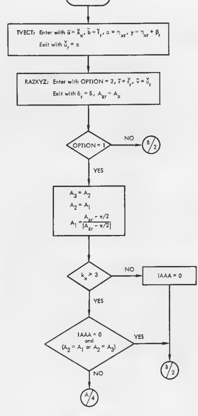
Output

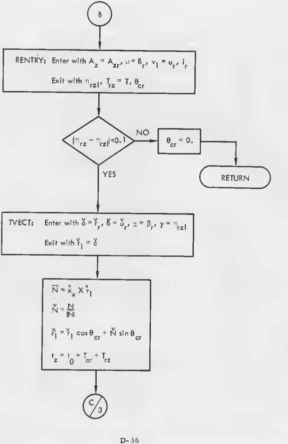
For OPTION = 0:

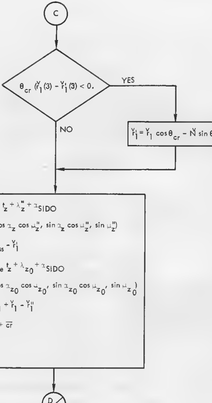
λ_{z0}	landing longitude
μ_{z0}	landing latitude
T_{rz}	time from reentry to landing

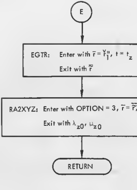
For OPTION = 1:

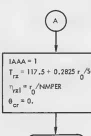
λ_{z0}	"offset" site longitude
μ_{z0}	"offset" site latitude
T_{rz}	time from reentry to landing
η_{rz1}	inplane component of the angle from reentry to landing
θ_{cr}	out-of-plane component of the angle from reentry to landing
$\overline{\Delta r}_1$	vector from the required landing site to the actual landing point


LNDING
TVECT: Enter with $\bar{a} = \check{X}_x$, $\bar{b} = \check{Y}_r$, $\alpha = \eta_{xr}$, $\gamma = \eta_{xr} + \beta_r$
Exit with $\check{u}_r = c$
RA2XYZ: Enter with OPTION = 2, $\bar{r} = \check{r}_r$, $\bar{v} = \check{u}_r$
Exit with $\delta_r = \delta$, $A_{zr} = A_z$
OPTION = 1
NO
B/2
YES
$A_3 = A_2$
$A_2 = A_1$
$A_1 = \frac{A_{zr} - \pi/2}{|A_{zr} - \pi/2|}$
$k_x > 3$
NO
IAAA = 0
YES
IAAA = 0 and ($A_2 = A_1$ or $A_2 = A_3$)
YES
B/2
NO
A/4


B
RENTRY: Enter with $A_z = A_{zr}$, $\mu = \delta_r$, $v_1 = u_r$, i_r
Exit with η_{rzl}, $T_{rz} = T$, θ_{cr}
$|\eta_{rz} - \eta_{rzl}| < 0.1$
NO
$\theta_{cr} = 0.$
RETURN
YES
TVECT: Enter with $\check{a} = \check{r}_r$, $\check{b} = \check{u}_r$, $\alpha = \beta_r$, $\gamma = \eta_{rzl}$
Exit with $\check{r}_1 = \check{c}$
$\overline{N} = \check{x}_x \times \check{r}_1$
$\check{N} = \frac{\overline{N}}{|\overline{N}|}$
$\check{r}_1' = \check{r}_1 \cos\theta_{cr} + \check{N} \sin\theta_{cr}$
$t_z = t_0 + T_{ar} + T_{rz}$
C/3


C
$\theta_{cr} (\check{r}_1(3) - \check{r}'_1(3)) < 0.$
YES
NO
$\check{r}'_1 = \check{r}_1 \cos\theta_{cr} - \check{N} \sin\theta_{cr}$
$\alpha_z = \omega_e t_z + \lambda''_z + \alpha_{SIDO}$
$\check{r}_{ss} = (\cos\alpha_z \cos\mu''_z, \sin\alpha_z \cos\mu''_z, \sin\mu''_z)$
$\overline{\Delta r}_1 = \check{r}_{ss} - \check{r}'_1$
$\alpha_{z0} = \omega_e t_z + \lambda_{z0} + \alpha_{SIDO}$
$\check{r}''_1 = (\cos\alpha_{z0} \cos\mu_{z0}, \sin\alpha_{z0} \cos\mu_{z0}, \sin\mu_{z0})$
$\overline{cr} = \overline{\Delta r}_1 + \check{r}_1 - \check{r}''_1$
$\overline{r}''_1 = \check{r}''_1 + \overline{cr}$
$\check{r}''_1 = \frac{1}{|r''_1|}$
D/4





NOTE: $\overset{Y}{r}_1$ is the unit vector to the inplane landing point

$\overset{Y}{r}_1'$ is the unit vector to the landing point considering cross range

$\overset{Y}{r}_1''$ is the unit vector to the "offset" site

$\overset{Y}{r}_{ss}$ is the vector to the required site

D-9 MINMIZ

Purpose

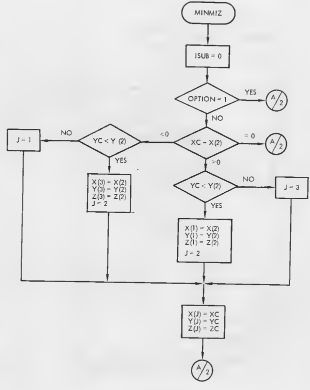
Given three values of X and Y, this routine predicts the X, Y coordinates which minimize Y. This is done by assuming that the first derivative of Y = f(X) is linear.

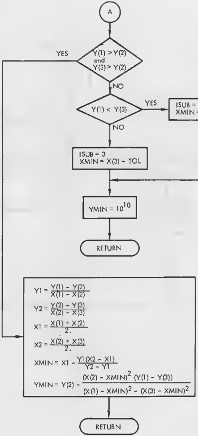
Input

X	table of values for the active independent variable
Y	table of values for the dependent variable
Z	table of values for the passive independent variable
OPTION	= 0, place XC, YC, ZC in the X, Y, Z tables, then predict the minimum using the updated tables = 1, predict minimum using the input X, Y, Z tables
XC, YC, ZC	current values of X, Y, Z to be placed in the tables
TOL	a step size in X to be used if no local minimum in Y is indicated within the range of X

Output

XMIN	predicted value of X which will minimize Y
YMIN	predicted value of the minimum Y
ISUB	= 0, a local minimum has been detected ≠ 0, no local minimum was found so the step size TOL will be used


MINMIZ
ISUB = 0
OPTION = 1
YES
A/2
NO
XC - X(2)
= 0
A/2
< 0
YC < Y (2)
NO
J = 1
YES
X(3) = X(2)
Y(3) = Y(2)
Z(3) = Z(2)
J = 2
> 0
YC < Y(2)
NO
J = 3
YES
X(1) = X(2)
Y(1) = Y(2)
Z(1) = Z(2)
J = 2
X(J) = XC
Y(J) = YC
Z(J) = ZC
A/2


A
Y(1) > Y(2) and Y(3) > Y(2)
YES
NO
Y(1) < Y(3)
YES
NO
ISUB = 1
XMIN = X(1) + TOL
ISUB = 3
XMIN = X(3) - TOL
YMIN = 10^10
RETURN
Y1 = (Y(1) - Y(2))/(X(1) - X(2))
Y2 = (Y(2) - Y(3))/(X(2) - X(3))
X1 = (X(1) + X(2))/2.
X2 = (X(2) + X(3))/2.
XMIN = X1 - Y1(X2 - X1)/(Y2 - Y1)
YMIN = Y(2) - (X(2) - XMIN)^2 (Y(1) - Y(3)) / ((X(1) - XMIN)^2 - (X(3) - XMIN)^2)
RETURN

D-10 RA2XYZ

Purpose

This closed routine has four options to calculate the following:

Option 1) $\bar{r}$ and $\bar{v}$ given r, α, δ, v, β, A_z

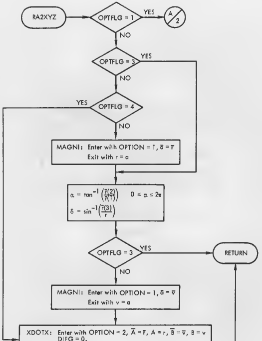
Option 2) r, α, δ, v, β, A_z given $\bar{r}$ and $\bar{v}$

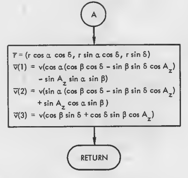
Option 3) α, δ given $\bar{r}$

Option 4) β, A_z given $\bar{v}$

Notation

$\bar{r}$	position vector
$\bar{v}$	velocity vector
r	magnitude of $\bar{r}$
v	magnitude of $\bar{v}$
α	right ascension of $\bar{r}$
δ	declination of $\bar{r}$
β	flight-path angle
A_z	azimuth


RA2XYZ
OPTFLG = 1
YES
A
2
NO
OPTFLG = 3
YES
NO
OPTFLG = 4
YES
NO
MAGNI: Enter with OPTION = 1, $\bar{a} = \bar{r}$
Exit with r = a
$\alpha = \tan^{-1}\left(\frac{\bar{r}(2)}{\bar{r}(1)}\right)$ $0 \le \alpha \le 2\pi$
$\delta = \sin^{-1}\left(\frac{\bar{r}(3)}{r}\right)$
OPTFLG = 3
YES
RETURN
NO
MAGNI: Enter with OPTION = 1, $\bar{a} = \bar{v}$
Exit with v = a
XDOTX: Enter with OPTION = 2, $\bar{A} = \bar{r}$, A = r, $\bar{B} = \bar{v}$, B = v
DIFG = 0.
Exit with $\alpha = \beta$
$c = \bar{r} \cdot \bar{v}/r$
$A_z = \tan^{-1}\left(\frac{\bar{r}(1)\,\bar{v}(2) - \bar{r}(2)\,\bar{v}(1)}{r\,\bar{v}(3) - c\,\bar{r}(3)}\right)$ $0 \le A_z \le 2\pi$


A
$\bar{r} = (r \cos\alpha \cos\delta,\ r \sin\alpha \cos\delta,\ r \sin\delta)$
$\bar{v}(1) = v(\cos\alpha(\cos\beta \cos\delta - \sin\beta \sin\delta \cos A_z) - \sin A_z \sin\alpha \sin\beta)$
$\bar{v}(2) = v(\sin\alpha(\cos\beta \cos\delta - \sin\beta \sin\delta \cos A_z) + \sin A_z \cos\alpha \sin\beta)$
$\bar{v}(3) = v(\cos\beta \sin\delta + \cos\delta \sin\beta \cos A_z)$
RETURN

D-11 RENTRY

Purpose

This subroutine computes the downrange, crossrange, and time from reentry to landing for both the guided and manual reentry modes. The computations are in the form of polynomial curve fits of empirical data.

Input

L/D	lift to drag ratio
ICRNGG	= 0, indicates a guided reentry = 1, indicates a manual reentry to the MSFN target line = 2, indicates a manual reentry to the contingency target line
u_r	inertial speed
i_r	inertial reentry speed
A_Z	inertial reentry azimuth
μ	reentry latitude
r_{rbias}	relative range override

Output

η_{rz1}	downrange angle from reentry to landing
θ_{cr}	crossrange angle from reentry to landing
T	time from reentry to landing

Reentry Coefficients

$b_1 = 0.61974454 \times 10^4$

$b_2 = -0.26277752$

$b_3 = -0.38675677 \times 10^{-5}$

$b_4 = 0.15781674 \times 10^{-9}$

$b_5 = 0.67856872 \times 10^{-4}$

$b_6 = -0.14887772 \times 10^{-18}$

$cc = 0.105$

$ee_1 = 0.18957317 \times 10^3$

$ee_2 = 0.17640466$

$ee_3 = 0.19321074 \times 10^{-2}$

$ff_1 = 0.64623407 \times 10^2$

$ff_2 = 0.57834928 \times 10^{-1}$

$ff_3 = -0.48255307 \times 10^{-3}$

$jj_{11} = 0.10718858 \times 10^7$

$jj_{12} = -0.16271240 \times 10^3$

$jj_{13} = 0.98775571 \times 10^{-2}$

$jj_{14} = -0.29943037 \times 10^{-6}$

$jj_{15} = 0.45325217 \times 10^{-11}$

$jj_{16} = -0.27404876 \times 10^{-16}$

$pp_{11} = 0.590 \times 10^2$

$pp_{12} = 0.3006$

$jj_{21} = 0.18262030 \times 10^6$

$jj_{22} = -0.27810612 \times 10^2$

$jj_{23} = 0.16998821 \times 10^{-2}$

$jj_{24} = -0.51884802 \times 10^{-7}$

$jj_{25} = 0.79087925 \times 10^{-12}$

$jj_{26} = -0.48128071 \times 10^{-17}$

$q_{11} = 0.0$

$q_{12} = 0.555 \times 10^{-4}$

$q_{13} = -0.1025 \times 10^1$

$q_{14} = 0.400 \times 10^3$

$q_{15} = 0.335 \times 10^3$

$q_{16} = -0.4215 \times 10^2$

$q_{21} = 0.0$

$q_{22} = 0.555 \times 10^{-4}$

$q_{23} = -0.1025 \times 10^1$

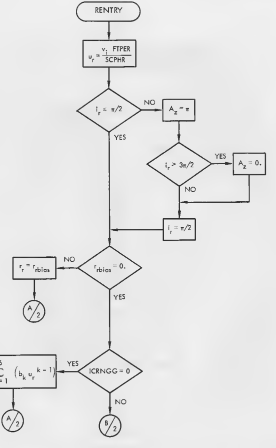
$q_{24} = 0.7000 \times 10^3$

$q_{25} = 0.3100 \times 10^3$

$q_{26} = -0.4500 \times 10^2$

$pp_{21} = 0.193 \times 10^3$

$pp_{22} = 0.1795$


RENTRY
u_r = v_i FTPER / SCPHR
i_r ≤ π/2
NO
A_z = π
YES
i_r > 3π/2
YES
A_z = 0.
NO
i_r = π/2
NO
r_r = r_rbias
r_rbias = 0.
YES
A/2
YES
r_r = Σ_{k=1}^{6} (b_k u_r^{k-1})
ICRNGG = 0
NO
A/2
B/2

(B)

$i = ICRNGG$

$r = (q_{i2}\, u_r + q_{i3})\,(q_{i4})\,(L/D - 0.3) + \sum_{k=1}^{6} ii_{ik}\, u_r^{\,k-1}$

$T = pp_{i1} + pp_{i2}\, r$

$CR = cc\ T \cos\mu \cos(A_z + .3\mu) + q_{i5}\, L/D + q_{i6}$

$\eta_{rz1} = |r|/NMPER$

$\theta_{cr} = CR/NMPER$

$u_r = v_i$

RETURN

(A)

$r = r_r + (ff_1 + ff_2\, r_r + ff_3\, u_r) \cos i_r$

$T = ee_1 + ee_2\, r + ee_3\, u_r$

$CR = cc\ T \cos\mu \cos(A_z + .3\mu)$

$\eta_{rz1} = |r|/NMPER$

$\theta_{cr} = CR/NMPER$

$u_r = v_i$

RETURN

D-12 SIDCOM

Purpose

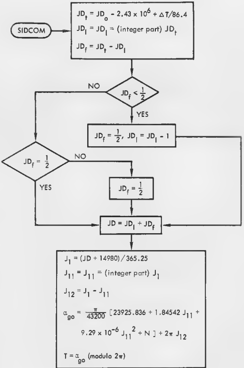
This closed routine computes the sidereal time, midnight, Greenwich, when the Julian date of an epoch and time from that Julian date of epoch are specified.

Input

JD_0	Julian date
ΔT	time from 0^h GMT (kiloseconds)
N	tabulated nutation in right ascension, often set equal to zero

Output

α_{g0}	sidereal time
T	α_{g0} (modulo 2π)


SIDCOM
$JD_t = JD_o - 2.43 \times 10^6 + \Delta T/86.4$
$JD_I = JD_I$ = (integer part) JD_t
$JD_f = JD_t - JD_I$
$JD_f < \frac{1}{2}$
NO
YES
$JD_f = \frac{1}{2}$, $JD_I = JD_I - 1$
$JD_f = \frac{1}{2}$
NO
YES
$JD_f = \frac{1}{2}$
$JD = JD_I + JD_f$
$J_1 = (JD + 14980)/365.25$
$J_{11} = J_{11}$ = (integer part) J_1
$J_{12} = J_1 - J_{11}$
$\alpha_{go} = \frac{\pi}{43200}$ [23925.836 + 1.84542 J_{11} + 9.29×10^{-6} J_{11}^2 + N] + $2\pi J_{12}$
$T = \alpha_{go}$ (modulo 2π)
RETURN

D-13 TBLOOK

Purpose

This routine returns the longitude of a point where the latitude of the midpoint of the reentry footprint axis of symmetry intersects a contingency line.

Inputs

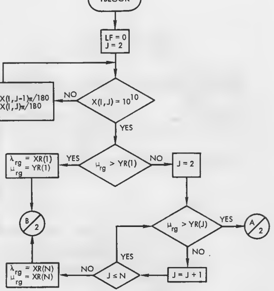
I:
- = 1, line 1
- = 2, line 2
- = 3, line 3
- = 4, line 4
- = 5, line 5
- = 6, line 6
- = 7, line 7
- = 8, line 8

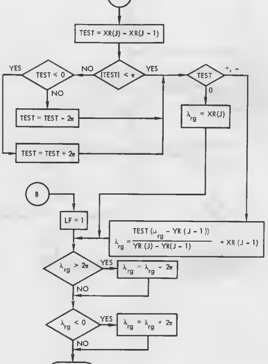
μ_{rg} latitude of axis midpoint

Outputs

λ_{rg} longitude of contingency line at specified latitude

The contingency lines are defined by the coordinates of the end-points of the straight line segments which compose the contingency lines. The latitudes and longitudes are ordered in tables. The desired longitude is calculated using the equation of a straight line through two points.


TBLOOK
LF = 0
J = 2
N = J/2
YR(N) = X(I, J-1)π/180
XR(N) = X(I, J)π/180
J = J + 2
NO
X(I, J) = 10^10
YES
λ_rg = XR(1)
μ_rg = YR(1)
YES
μ_rg > YR(1)
NO
J = 2
B/2
μ_rg > YR(J)
YES
A/2
NO
YES
λ_rg = XR(N)
μ_rg = XR(N)
NO
J ≤ N
J = J + 1


A
TEST = XR(J) - XR(J - 1)
YES
TEST < 0
NO
|TEST| < π
YES
TEST
+, -
NO
0
TEST = TEST - 2π
λrg = XR(J)
TEST = TEST + 2π
B
LF = 1
λrg = TEST (μrg - YR (J - 1)) / YR (J) - YR(J - 1) + XR (J - 1)
λrg > 2π
YES
λrg = λrg - 2π
NO
λrg < 0
YES
λrg = λrg + 2π
NO
RETURN

D-14 TFPCR

Purpose

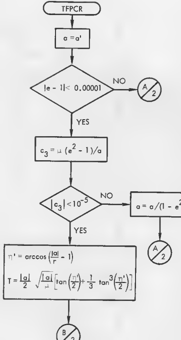
This closed routine calculates the time from the pericenter to a desired distance for a conic for which a (or p) and e are given. The time is measured positively if the motion is directed away from the focus and negatively otherwise.

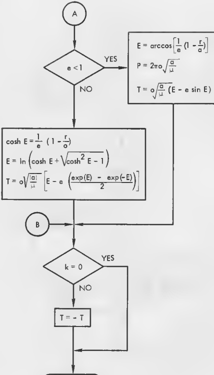
Input

μ	gravitational constant
k	outward leg (0.), and return leg (1.) flag. k is input as a floating point number
a'	semimajor axis or semilatus rectum
e	eccentricity
r	radial distance from focus

Output

T	time from pericenter to the radius r
P	period of the orbit. If $e \geq 1$, P is not computed.


TFPCR
a = a'
|e - 1|< 0.00001
NO
A/2
YES
c3 = μ (e2 - 1)/a
|c3| < 10-5
NO
a = a/(1 - e2)
A/2
YES
η' = arccos (|a|/r - 1)
T = |a|/2 √(|a|/μ) [tan(η'/2) + 1/3 tan3(η'/2)]
B/2


A
e < 1
YES
NO
E = arccos [1/e (1 - r/a)]
P = 2πa √(a/μ)
T = a √(a/μ) (E - e sin E)
cosh E = 1/e (1 - r/a)
E = ln (cosh E + √(cosh² E - 1))
T = a √(|a|/μ) [E - e ((exp(E) - exp(-E))/2)]
B
k = 0
YES
NO
T = - T
RETURN

D-15 TJUDAT

Purpose

This function computes the Julian date of an epoch given the Gregorian year, month, day, and portion of day of the epoch.

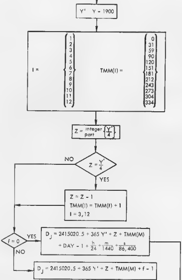
Input

Y	Gregorian year of epoch
M	month of epoch
D	day of epoch
h	hour of epoch
m	minute of epoch
s	second of epoch
f	fraction of a day of epoch

h, m and s are used to specify the fractional part of the epoch day if, and only if, f = 0.

Output

J_D	Julian date of epoch


TJUDAT
Y' Y - 1900
I = 1 2 3 4 5 6 7 8 9 10 11 12
TMM(I) = 0 31 59 90 120 151 181 212 243 273 304 334
Z = integer part (Y'/4)
NO
Z = Y'/4
YES
Z = Z - 1
TMM(I) = TMM(I) + 1
I = 3, 12
f = 0
YES
NO
D_J = 2415020.5 + 365 Y' + Z + TMM(M) + DAY - 1 + h/24 + m/1440 + s/86,400
D_J = 2415020.5 + 365 Y' + Z + TMM(M) + f - 1
YES
TMM(3) = 59
NO
RETURN
TMM(I) = TMM(I) - 1, I = 3, 12

D-16 TVECT

Purpose

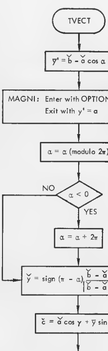
This closed routine will compute a third unit vector lying in the plane of two given unit vectors and displaced by a given angular distance from the first.

Input

- $\check{a}$ first unit vector
- $\check{b}$ second unit vector
- α angle between $\check{a}$ and $\check{b}$ measured in the desired direction of $\check{c}$
- γ the angular position of the third vector from $\check{a}$ and measured in the same sense as α

Output

$\check{c}$(CBAR) the desired unit vector


TVECT
$\overline{y}' = \check{b} - \check{a} \cos \alpha$
MAGNI: Enter with OPTION = 1, $\overline{a} = \overline{y}'$
Exit with $y' = a$
$\alpha = \alpha$ (modulo 2π)
$\alpha < 0$
NO
YES
$\alpha = \alpha + 2\pi$
$\check{y} = \text{sign}(\pi - \alpha) \dfrac{\check{b} - \check{a} \cos \alpha}{\left|\check{b} - \check{a} \cos \alpha\right|}$
$\check{c} = \check{a} \cos \gamma + \overline{y} \sin \gamma$
RETURN

D-17 URBRER

Purpose

This data block generates the u_r, and β_r tables for the variable reentry mode. The values are stored in the labeled common block/ TABLE/. The constants are generated without a specific CALL statement.

Input

None

Output

	BRTABL	
URTABL	CONTINGENCY	MSFN
29,500	95.28	94 39
30,000	95.41	94.59
30,500	95.55	94.78
31,000	95.66	94.96
31,500	95.77	95.12
32,000	95.86	95.27
32,500	95.96	95.41
33,000	96.05	95.54
33,500	96.14	95.67
34,000	96.21	95.80
34,500	96.29	95.92
35,000	96.36	96.03
35,500	96.43	96.14
36,000	96.48	96.24
36,500	96.55	96.35
37,000	96.61	96.44
38,000	96.74	96.58

D-18 URF

Purpose

This closed routine computes an earth reentry velocity, when given a radial distance and the flight time from the radial distance to reentry. A polynomial approximation is used with coefficients generated by a least-squares program. The data for the curve fit were generated for distances between 40 and 80 earth radii and times corresponding to reentry velocities slightly greater than parabolic velocity down to the minimum possible. The following constants were employed in the data generation.

$$r_r = 1.0193762 \text{ er}$$

$$\beta_r = 96 \text{ deg}$$

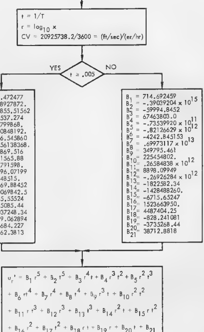
$$\mu_e = 19.909416 \frac{(er)^3}{(hr)^2}$$

Input

T	flight time (hr)
x	radial distance (er)

Output

u_r	reentry velocity (er/hr)


URF
t = 1/T
r = log10 x
CV = 20925738.2/3600 = (ft/sec)/(er/hr)
YES
t ≥ .005
NO
B1 = 468.472477
B2 = 7888927872.
B3 = -61855.51562
B4 = 110537.274
B5 = 171799868.
B6 = 4620848192.
B7 = -906.545860
B8 = -7956138368.
B9 = 245869.516
B10 = 9781565.88
B11 = 233791598.
B12 = -3296.07199
B13 = 68248515.
B14 = 17869.88452
B15 = -20069842.5
B16 = 8065.55524
B17 = 8235085.44
B18 = -1037248.34
B19 = -419.062894
B20 = 966684.227
B21 = 30762.3813
B1 = 714.692459
B2 = -.39039204 x 10^15
B3 = -59994.8452
B4 = 67463803.0
B5 = -.73539920 x 10^11
B6 = -.82126629 x 10^12
B7 = -4242.845153
B8 = .69973117 x 10^13
B9 = 349795.461
B10 = 225454802.
B11 = .26584838 x 10^12
B12 = 8898.09949
B13 = -.26926284 x 10^12
B14 = -1822582.34
B15 = -1428488260.
B16 = -6715.65247
B17 = 1523663950.
B18 = 4487404.25
B19 = -828.241081
B20 = -3735268.44
B21 = 38712.8818
u_r' = B1 r^5 + B2 t^5 + B3 r^4 t + B4 r^3 t^2 + B5 r^2 t^3 + B6 r t^4 + B7 r^4 + B8 t^4 + B9 r^3 t + B10 r^2 t^2 + B11 r t^3 + B12 r^3 + B13 t^3 + B14 r^2 t + B15 r t^2 + B16 r^2 + B17 t^2 + B18 r t + B19 r + B20 t + B21
u_r = u_r'/CV
RETURN

D-19 XDATE

Purpose

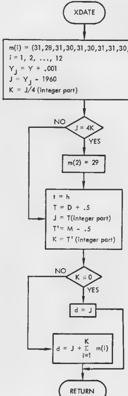
This routine is used only before the first calling of JPLEPH, and establishes the base time in the common block /DATE/.

Input

Y	nearest Besselian year from epoch
M	month of epoch
D	day of month
h	time of day (0^h GMT)

Output

Y_J	year
d	day of year
t	time of day (0^h GMT)


XDATE
m(i) = (31,28,31,30,31,30,31,31,30,31,30,31)
i = 1, 2, ..., 12
Y_J = Y + .001
J = Y_J - 1960
K = J/4 (integer part)
NO
J = 4K
YES
m(2) = 29
t = h
T = D + .5
J = T(integer part)
T' = M - .5
K = T' (integer part)
NO
K ≤ 0
YES
d = J
d = J + Σ_{i=1}^{K} m(i)
RETURN

D-20 XDOTX

Purpose

This closed routine has three options to calculate the following:

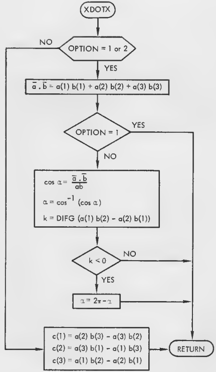
Option 1) $\overline{a} \cdot \overline{b}$ given $\overline{a}$ and $\overline{b}$

Option 2) $\overline{a} \cdot \overline{b}$, α, and $\cos \alpha$ given $\overline{a}$, $\overline{b}$, a, b, and DIFG

Option 3) $\overline{c}$ given $\overline{a}$ and $\overline{b}$

Notation

OPTFLG	option flag
$\overline{a}$	first vector
$\overline{b}$	second vector
a	magnitude of $\overline{a}$
b	magnitude of $\overline{b}$
DIFG	flag indicating the direction of rotation from $\overline{a}$ to $\overline{b}$
$\overline{a} \cdot \overline{b}$	dot product of $\overline{a}$ and $\overline{b}$
α	included angle between $\overline{a}$ and $\overline{b}$
$\cos \alpha$	cosine of α
c	cross product of $\overline{a}$ and $\overline{b}$


XDOTX
OPTION = 1 or 2
NO
YES
$\bar{a} \cdot \bar{b}$ = a(1) b(1) + a(2) b(2) + a(3) b(3)
OPTION = 1
YES
NO
$\cos \alpha = \frac{\bar{a} \cdot \bar{b}}{ab}$
$\alpha = \cos^{-1} (\cos \alpha)$
k = DIFG (a(1) b(2) - a(2) b(1))
k < 0
NO
YES
$\alpha = 2\pi - \alpha$
c(1) = a(2) b(3) - a(3) b(2)
c(2) = a(3) b(1) - a(1) b(3)
c(3) = a(1) b(2) - a(2) b(1)
RETURN

D-21 XYZ2AE

Purpose

This routine is used to convert Cartesian position and velocity vectors to conic elements. There are four options available.

Option 1) Calculate r, v, a, e, i, Ω, ω, η, β given $\bar{r}$ and $\bar{v}$

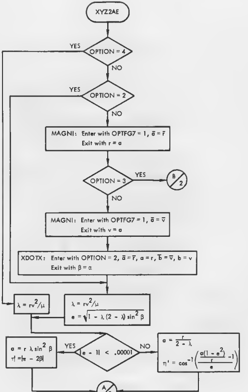
Option 2) Calculate a, e, η given r, v, β

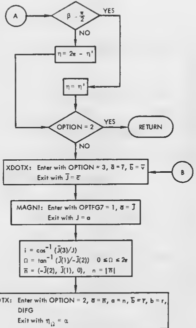
Option 3) Calculate i, Ω, η_Ω given $\bar{r}$, $\bar{v}$

Option 4) Calculate a, η, i, Ω, ω given r, v, β, e, $\bar{r}$, $\bar{v}$

Notation

OPTION	equals 1, 2, 3, or 4
$\bar{r}$	position vector
r	magnitude of $\bar{r}$
$\bar{v}$	velocity vector
v	magnitude of $\bar{v}$
μ	gravitational constant
β	flight-path angle
a	semimajor axis, for e = 1 a is actually p, the semilatus rectum
e	eccentricity
i	inclination
Ω	right ascension of the ascending node
ω	argument of perifocus
η	true anomaly
DIFG	flag describing the direction of travel
η_Ω	true anomaly of ascending node


XYZ2AE
OPTION = 4
YES
NO
OPTION = 2
YES
NO
MAGNI: Enter with OPTFG7 = 1, $\bar{a} = \bar{r}$
Exit with r = a
OPTION = 3
YES
B/2
NO
MAGNI: Enter with OPTFG7 = 1, $\bar{a} = \bar{v}$
Exit with v = a
XDOTX: Enter with OPTION = 2, $\bar{a} = \bar{r}$, a = r, $\bar{b} = \bar{v}$, b = v
Exit with $\beta = \alpha$
$\lambda = rv^2/\mu$
$\lambda = rv^2/\mu$
$e = \sqrt{1 - \lambda (2 - \lambda) \sin^2 \beta}$
$|e - 1| < .00001$
YES
NO
$a = r \lambda \sin^2 \beta$
$\eta' = |\pi - 2\beta|$
$a = \frac{r}{2 - \lambda}$
$\eta' = \cos^{-1}\left(\frac{\frac{a(1 - e^2)}{r} - 1}{e}\right)$
A/2


A
β - π/2
YES
NO
η = 2π - η'
η = η'
OPTION = 2
YES
RETURN
NO
XDOTX: Enter with OPTION = 3, ā = r̄, b̄ = v̄
Exit with J̄ = c̄
B
MAGNI: Enter with OPTFG7 = 1, ā = J̄
Exit with J = a
i = cos⁻¹ (J̄(3)/J)
Ω = tan⁻¹ (J̄(1)/-J̄(2))   0 ≤ Ω ≤ 2π
n̄ = (-J̄(2), J̄(1), 0),   n = |n̄|
XDOTX: Enter with OPTION = 2, ā = n̄, a = n, b̄ = r̄, b = r,
DIFG
Exit with η_Ω = α
η = η_Ω
YES
OPTION = 3
NO
ω = η_Ω - η
RETURN

REFERENCES

1. Garrison, T. P. and Davis, D. R.: RTCC Requirements for Apollo 14 (H-3) Mission: Earth-Centered Return-to-Earth Conic Subprocessor. MSC Internal Note No. 70-FM-23.

2. Davis, R. S.: RTCC Requirements for Apollo 14 (H-3) Mission: Return-to-Earth Processor Supervisory and Precision Computation Logic. unpublished MSC Internal Note.

3. Wilson, S. W. Jr.: A Pseudostate Theory for the Approximation of Three-Body Trajectories. TRW Note No 69-FMT-765 (11176-H304-R0-00), August 15, 1969.

4. Cafford, D. M., et al: RTCC Requirements for Mission C: Return-to-Earth Abort Conic Subprocessor for Moon-Centered Aborts MSC Internal Note No. 68-FM-27, February 1, 1968.

5. Northcutt, F. M.: Revised Unspecified Area Logic. TRW Letter No. 5524.8-101, October 1, 1969.

6. Cafford, D. M.: Results of the Application of the Offset Vector Method to the Return-to-Earth Processor. TRW Letter No. 5524.8-100, October 1, 1969.

7. Monroe, J. D.: Analytic Return-to-Earth Program - Moon Reference Volume II. TRW Note No. 69-FMT-737 (11176-H144-R0-00), May 1969.